International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

A Comparative Study of Rule Mining Based Web Usage Mining Algorithms

B. Uma Maheswari[1], Dr. P.Sumathi[2]

[1]Doctoral Student in Bharathiyar University, Coimbatore, Tamil Nadu, India

[2]Assistant Professor, Govt. Arts College, Coimbatore, Tamil Nadu, India

Abstract: ***Web usage mining attempts to discover useful knowledge from the secondary data obtained from the interactions of the users with the web. It also becomes very critical for effective web site management and for creating adaptive web sites, business and support services etc. The web mining field encompasses a wide array of issues, primarily aimed at deriving actionable knowledge from the web, and includes researchers from information retrieval, database technologies, and artificial intelligence. Most data used for mining is collected from web servers, clients, proxy servers, or server databases, all of which generate noisy data. Because web mining is sensitive to noise, data cleaning methods are necessary. Web usage mining itself can be classified further depending on the kind of usage data considered. They are web server data, application server data and application level data. Web server data correspond to the user logs that are collected at web server. This work compares the two standard web usage mining algorithms namely Apriori algorithm and Frequent Pattern algorithm. Particularly, this work focused on discovering the web usage patterns of websites from the server log files.***

Keywords: Web usage mining, Pattern discovery, Apriori algorithm, Frequent pattern mining.

1. Introduction

One of the applications of data mining techniques is the web mining it is used to extract knowledge from web data including web documents and hyperlinks between documents, us-age logs of web sites etc. Web content mining is the process of extracting useful information from the contents of web documents. The content data is the collection of facts a web page is designed to contain. It may consist of text and images, audio and video or structured records such as lists and tables. Application of text mining to web content has been the most widely researched.

Data mining is a set of techniques and tools used to the no trivial process of extracting and present implicit knowledge and no knowledge before, this information is useful and human reliable this is processing from a great set of data; with the object of describing in automatic way models, no knowledge before and to detect tendencies and patterns. The web usage mining is the process of applying techniques to detect patterns of usage to web page. The web usage mining use the data storage in the log files of web server as first resource; in this file the web server register the access at each resource in the server by the users.

The web service providers want to find the way to predict the users' behaviors and personalize information to reduce the traffic load and design the web site suited for the different group of users. The business analysts want to have tools to learn the user/consumers' needs. All of them are expecting tools or techniques to help them satisfy their demands and/or solve the problems encountered on the Web. Therefore, web mining becomes a popular active area and is taken as the research topic for this investigation. This research focused on Apriori and Frequent Pattern algorithm by comparing it for web usage mining.

The Apriori Algorithm is an influential algorithm for mining frequent itemsets for Boolean association rules The Apriori algorithm is used in data mining process for mining frequent patterns from the given data set. This algorithm uses an iterative approach called level-wise search, in which n-item sets are used to explore n+1 item sets. Some of the issues of Apriori algorithm are: Database scanning of the whole dataset for each iteration, the computational efficiency is very less because the whole database scans is needed every time, the cost of generating large number of candidate sets and scanning the database repeatedly. The repeated scan of the database is very costly. To overcome this, Frequent Pattern mining algorithm is introduced. The FP-Tree algorithmic rule, instructed by dynasty in, is another thanks to notice frequent piece teams while not utilizing applier generations, so advancing performance. For thus a lot of it values a divide-and-conquer strategy. The central a part of this methodology is that the usage of a particular arrangement entitled frequent-pattern tree (FP-tree) that keeps the piece set association information.

The paper can be organized as follows. Section II describes the related works, Section III describes the methodology used, section IV describes evaluation and section V discusses the conclusion of the proposed work.

2. Literature Survey

Srivastava et al (2000) provides a detailed taxonomy of the work in this area, including research efforts as well as commercial offerings. An up-to-date survey of the existing work is also provided. Zaïane and Luo (2001) discuss some data mining and machine learning techniques that could be used to enhance web-based learning environments for the educator to better evaluate the leaning process, as well as for the learners to help them in their learning Endeavour. Cho et al (2002) suggests a personalized recommendation methodology by which we are able to get further effectiveness and quality of recommendations when applied to an Internet shopping mall. The suggested methodology is based on a variety of data mining techniques such as web

Licensed Under Creative Commons Attribution CC BY

usage mining, decision tree induction, association rule mining and the product taxonomy. Büchner et al (1998) describes a novel way of combining data mining techniques on Internet data in order to discover actionable marketing intelligence in electronic commerce scenarios. Cho et al (2004) proposes a recommendation methodology based on web usage mining, and product taxonomy to enhance the recommendation quality and the system performance of current CF-based recommender systems.

Cooley et al (1997) defined web mining and present an overview of the various research issues, techniques, and development efforts. We briefly describe WEBMINER, a system for Web usage mining, and conclude the paper by listing research issues. Abraham and Vitorino (2003) proposed an ant clustering algorithm to discover web usage patterns (data clusters) and a linear genetic programming approach to analyze the visitor trends. SpeedTracer, a World Wide Web usage mining and analysis tool, was developed to understand user surfing behavior by exploring the Web server log files with data mining techniques. Wu et al (1998) describe the design of SpeedTracer and demonstrate some of its features with a few sample reports. Mobasher et al (2000) presented such a framework, distinguishing between the offline tasks of data preparation and mining, and the online process of customizing Web pages based on a user's active session. They describe effective techniques based on clustering to obtain a uniform representation for both site usage and site content profiles, and shows how these profiles can be used to perform real-time personalization. Pierrakos et al (2003) views Web personalization through the prism of personalization policies adopted by Web sites and implementing a variety of functions.

Cooley et al (1999) presents several data preparation techniques in order to identify unique users and user sessions. Transactions identified by the proposed methods are used to discover association rules from real world data using the WEBMINER system. Pei et al (2001) proposed a simple and novel hyper-linked data structure, H-struct and a new mining algorithm, H-mine, which takes advantage of this data structure and dynamically adjusts links in the mining process. Ezeife et al (2005) proposed a more efficient approach for using the WAP-tree to mine frequent sequences, which totally eliminates the need to engage in numerous re-constructions of intermediate WAP-trees during mining. Zaki (2002) formulate the problem of mining (embedded) sub trees in a forest of rooted, labeled, and ordered trees. He present TREEMINER, a novel algorithm to discover all frequent sub trees in a forest, using a new data structure called scope-list. Zheng et al (2001) compares five well-known association rule algorithms using three real-world datasets and an artificial dataset.

3. Research Methodology

This section describes the Apriori and Frequent Pattern algorithm for web usage mining and their advantages and limitations.

3.1 Apriori algorithm

Apriori algorithm captures large data sets during its initial database passes and uses this result as the base for discovering other large datasets during subsequent passes. Item sets having a support level above the minimum are called large or frequent item sets and those below are called small item sets. The algorithm is based on the large item set property which states: Any subset of a large item set is large and any subset of frequent item set must be frequent. Apriori algorithm is, the most supervised and important algorithm for mining frequent itemsets. Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. It generates candidate item sets of length k from item sets of length k-1. Then it prunes the candidates which have an infrequent sub pattern.

According to the downward closure lemma, the candidate set contains all frequent k-length item sets. After that, it scans the transaction database to determine frequent item sets among the candidates. Apriori is a supervised algorithm for mining frequent itemsets for Boolean association rules. Since the Algorithm uses prior knowledge of frequent item set it has been given the name Apriori. It is an iterative level wise search Algorithm, where k itemsets are used to explore (k+1)-itemsets. First, the set of frequents 1- itemsets is found. This set is denoted by L1. L1 is used to find L2, the set of frequent 2-itemsets, which is used to find L3 and so on, until no more frequent k-itemsets can be found. The finding of each Lk requires one full scan of database.

Apriori, which is significant, suffers from a number of inefficiencies or trade-offs, which have spawned other algorithms. Candidate generation generates large numbers of subsets (the algorithm attempts to load up the candidate set with as many as possible before each scan). Bottom-up subset exploration (essentially a breadth-first traversal of the subset lattice) finds any maximal subset S only after all 2^S-1 of its proper subsets.

3.1.1 Limitations of Apriori Algorithm

Apriori algorithm, in spite of being simple, has some limitation. They are,

It is costly to handle a huge number of candidate sets. For example, if there are 10^4 frequent 1-item sets, the Apriori algorithm will need to generate more than 10^7 length-2 candidates and accumulate and test their occurrence frequencies. Moreover, to discover a frequent pattern of size 100, such as $\{a_1, \ldots, a_{100}\}$, it must generate $2^{100} - 2 \sim 10^{30}$ candidates in total. This is the inherent cost of candidate generation, no matter what implementation technique is applied.

It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns.

In order to overcome the drawback inherited in Apriori, an efficient FP-tree based mining method, FP-growth, which contains two phases, where the first phase constructs an FP tree, and the second phase recursively Researches the FP tree and outputs all frequent patterns.

Licensed Under Creative Commons Attribution CC BY

3.2 FP Growth Algorithm

FP growth is employed to construct FP tree that is that the mining of frequent pattern. FP tree provides compressed dataset. It conjointly avoids repeatedly information scanning. The operating is as follows: Firstly it scans information and finds the support for every item. Then things area unit removed that don't seem to be frequent. Type alternative things in drizzling order supported counter worth. Next it reads one dealings at a time and plots it on tree. This algorithmic rule works as follows

- It compresses the input information conceiving Associate in Nursing FP tree instance to represent common things.
- It divides the compressed information into a group of conditional databases, every one attached with one common pattern
- Eventually, every information is extract one by one

Using this theme, the FP-Tree decrease the enquire charges yearning for tiny patterns recursively so concatenating then within the long common patterns, proposing higher property. In giant databases, it isn't probably to carry the FP-tree within the major memory. Associate in Nursing approach to deal with this problem is to foremost separate the information into a gaggle of lesser databases (called projected databases), so construct a common Pattern-tree from every of those smaller databases.

3.2.1 FP tree structure

FP tree may be a solid information design that preserved vital, fully important and quantitative information considering common patterns.

The main attributes of Frequent Pattern tree are:

a) It contains of 1 root marked as "root", a group of piece prefix sub-trees because the kid of the foundation, and a frequent item header chart.
b) One-by-one node within the piece prefix sub-tree contains of 3 areas:
   - *Item-name:* It lists that item this node represents.
   - *Count:* It registers the quantity of transactions depicted by the portion of the trail returning to the present node
   - *Node-link:* It connects to succeeding node within the FP¬ tree bearing the identical item-name, or null if there's none.
c) One-by-one application within the frequent-item header journal contains of 2 area: item-name

Header of node- link, that points to the primary node within the FP-tree carrying the item-name

3.2.2 Advantages of FP growth algorithm

The major advantages of FP-Growth algorithm is,

- Uses compact data structure
- Eliminates repeated database scan

FP-growth is an order of magnitude faster than other association mining algorithms and is also faster than tree Researching. The algorithm reduces the total number of candidate item sets by producing a compressed version of the database in terms of an FP-tree. The FP-tree stores relevant information and allows for the efficient discovery of frequent item sets.

The algorithm consists of two steps:

- Compress a large database into a compact, Frequent Pattern tree (FP-tree) structure – highly condensed, but complete for frequent pattern mining and avoid costly database scans
- Develop an efficient, FP-tree-based frequent pattern mining method (FP-growth) – A divide-and-conquer methodology: decompose mining tasks into smaller ones and avoid candidate generation: sub-database test only

3.2.3 Advantage of FP-tree structure

The most significant advantage of the FP-tree is that the algorithm scans the tree only twice. Apart from this major advantage, the others include,

- Completeness: The FP-tree contains all the information related to mining frequent patterns (given the minimum support threshold)
- Compactness: – The size of the tree is bounded by the occurrences of frequent items – The height of the tree is bounded by the maximum number of items in a transaction

4. Experimental Results

This section provides the experimental evaluation of Apriori and FP algorithm. The evaluation is carried out using accuracy and execution time of the algorithm. On standard dataset FP Growth performs the best and Apriori takes the maximum time. As a result of the experimental study, revealed the performance of Apriori and FP-tree algorithm. Table 1 shows the accuracy and execution time of the algorithm where FP algorithm outperforms.

Table 1: Accuracy and execution time for the algorithm

Algorithm	Accuracy (%)	Execution time (seconds)
Apriori	75	32
FP tree	91	15

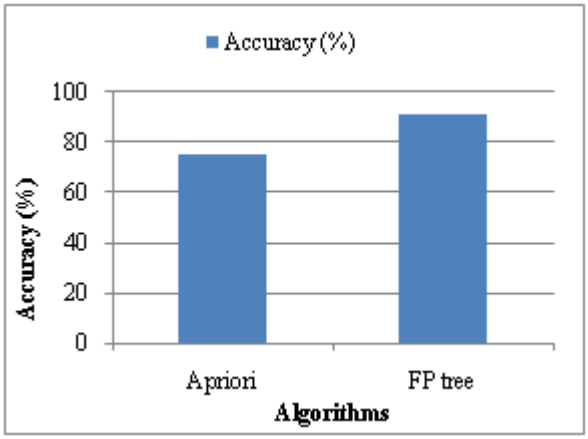

Figure 1: Accuracy

Fig. 1 shows that accuracy for Apriori and FP tree algorithm. The Apriori and FP-Growth algorithm achieves accuracy of 75% and 91% respectively. These results show that FP Algorithm attains better accuracy than Apriori Algorithm.

Licensed Under Creative Commons Attribution CC BY

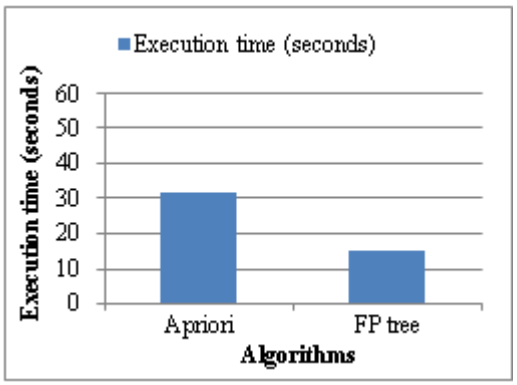

Figure 2: Execution time

Fig. 2 shows that FP-Growth algorithm takes less execution time than Apriori algorithm. So FP-Growth outperforms Apriori algorithm.

5. Conclusion

Web usage mining is the application of data mining techniques to discover usage patterns from Web data, in order to understand and better serve the needs of Web based applications. Web usage mining consists of three phases, namely preprocessing, pattern discovery, and pattern analysis. This Research work implements each of these phases. One of the algorithms which is very simple to use and easy to implement is the Apriori algorithm. This algorithm is used in the present Research work to generate association rules that associates the usage pattern of the clients for a particular website. The main drawback of Apriori algorithm is that the candidate set generation is costly, especially if a large number of patterns and/or long patterns exist which is rectified by FP growth algorithm. In future the algorithm can be extended to web content mining, web structure mining, etc.

References

[1] Srivastava, Jaideep, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. "Web usage mining: Discovery and applications of usage patterns from web data." *ACM SIGKDD Explorations Newsletter* 1, no. 2 (2000): 12-23.

[2] Zaïane, Osmar R., and J. Luo. "Web usage mining for a better web-based learning environment." In *Proceedings of conference on advanced technology for education*, pp. 60-64. 2001.

[3] Cho, Yoon Ho, Jae Kyeong Kim, and Soung Hie Kim. "A personalized recommender system based on web usage mining and decision tree induction."*Expert systems with Applications* 23, no. 3 (2002): 329-342.

[4] Büchner, Alex G., and Maurice D. Mulvenna. "Discovering internet marketing intelligence through online analytical web usage mining." *ACM Sigmod Record*27, no. 4 (1998): 54-61.

[5] Cho, Yoon Ho, and Jae Kyeong Kim. "Application of Web usage mining and product taxonomy to collaborative recommendations in e-commerce." *Expert systems with Applications* 26, no. 2 (2004): 233-246.

[6] Cooley, Robert, Bamshad Mobasher, and Jaideep Srivastava. "Web mining: Information and pattern discovery on the world wide web." In *Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on*, pp. 558-567. IEEE, 1997.

[7] Abraham, Ajith, and Vitorino Ramos. "Web usage mining using artificial ant colony clustering and linear genetic programming." In *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, vol. 2, pp. 1384-1391. IEEE, 2003.

[8] Wu, K-L., Philip S. Yu, and Allen Ballman. "Speedtracer: A web usage mining and analysis tool." *IBM Systems Journal* 37, no. 1 (1998): 89-105.

[9] Mobasher, Bamshad, Honghua Dai, Tao Luo, Yuqing Sun, and Jiang Zhu. "Integrating web usage and content mining for more effective personalization." In*Electronic commerce and web technologies*, pp. 165-176. Springer Berlin Heidelberg, 2000.

[10] Pierrakos, Dimitrios, Georgios Paliouras, Christos Papatheodorou, and Constantine D. Spyropoulos. "Web usage mining as a tool for personalization: A survey." *User modeling and user-adapted interaction* 13, no. 4 (2003): 311-372.

[11] Cooley, Robert, Bamshad Mobasher, and Jaideep Srivastava. "Data preparation for mining world wide web browsing patterns." *Knowledge and information systems* 1, no. 1 (1999): 5-32.

[12] Pei, Jian, Jiawei Han, Hongjun Lu, Shojiro Nishio, Shiwei Tang, and Dongqing Yang. "H-mine: Hyper-structure mining of frequent patterns in large databases." In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pp. 441-448. IEEE, 2001.

[13] Ezeife, Christie I., and Yi Lu. "Mining web log sequential patterns with position coded pre-order linked wap-tree." *Data Mining and Knowledge Discovery* 10, no. 1 (2005): 5-38.

[14] Zaki, Mohammed J. "Efficiently mining frequent trees in a forest." In*Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 71-80. ACM, 2002.

[15] Zheng, Zijian, Ron Kohavi, and Llew Mason. "Real world performance of association rule algorithms." In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 401-406. ACM, 2001.

Licensed Under Creative Commons Attribution CC BY