ISSN:0975-9646

Comparative Analysis of Apriori Algorithm and Frequent Pattern Algorithm for Frequent Pattern Mining in Web Log Data.

Rahul Mishra, Abha choubey

Department of Computer Science & Engineering.
Shri Shankaracharya College of Engineering and Technology, Bhilai C.G. India

Abstract- **The growth and popularity of the internet has increased the growth of web marketing. Extracting usage patterns deals with the weblog records to discover user access patterns of Web Pages. Weblog databases provide rich information about what kind of users will access which kind of web pages. Analyzing and exploring regularities in Weblog records can identify potential customers for electronic commerce and also enhance and improve the quality of Internet information services to end users.**

Keywords: Web usage mining, Association rules, Apriori, FP-tree.

1. INTRODUCTION

Being given a set of transactions of the clients, the purpose of the association rules is to find correlations between the sold articles. Knowing the associations between the offered products and services helps those who have to take decisions to implement successful marketing techniques. Based on the obtained results and comparative statistical interpretations, we issued hypotheses referring to performance, precision and accuracy of the two processes Apriori and Frequent pattern tree algorithm.

The World Wide Web contains an enormous amount of information in the form of a rather unstructured collection of hyperlinked documents, which increasingly makes finding relevant documents with useful information a challenge. At any point in time, each web site is visited by many users, with different goals, who are interested in different information content or types of presentations. Even the same user may visit the same web site for different purposes at different times.

Pattern mining is a promising approach in support of this goal. Assuming that past navigation behavior is an indicator of the users' interests, then, the records of this behavior, kept in the form of the web-server logs, can be mined to infer what the users are interested in. On that basis, recommendations can be dynamically generated, to help new web-site visitors find the information of i

Web-site designers want to increase the number of visitors and the time that these visitors spend on their web site. To accomplish that, they have to supply attractive content. And to make their content attractive, web-site designers and content providers need to know what their potential visitors want, in order to organize their content according to their visitors needs, and, if possible, according to individual preferences.

1.1 Web Usage Mining

Web usage mining refers to the automatic discovery and associated data collected or generated as a result of user interactions with Web resources on one or more Websites .The goal is to capture, model, and analyze the behavioral patterns and the profiles of users interacting with Web sites. The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed by groups of users with common needs or interests [1].

Web Usage Mining is the application of data mining techniques to discover interesting usage patterns from Web data, in order to understand and better serve the needs of Web-based applications. Usage data captures the identity or origin of Web users along with their browsing behavior at a Web site. 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.

The aim in web mining is to discover and retrieve useful and interesting patterns from a large dataset. In web mining, this dataset is the huge web data. Web data contains different kinds of information, including, web structure data, web log data, and user profiles data. Web mining is the application of data mining techniques to extract knowledge from web data, where at least one of structure or usage data is used in the mining process.

Web usage mining has various application areas such as web pre-fetching, link prediction, site reorganization and web personalization. Most important phases of web usage mining are the reconstruction of user sessions by using heuristics techniques and discovering useful patterns from these sessions by using pattern discovery techniques like association rule mining, Apriori, Fp algorithms [2].

Recent research has focused on deploying data-mining methods for understanding and predicting web-site visitor behavior. Research in Web mining spans three areas: Web-content mining, Web structure mining and Web-usage mining. Web-content mining refers to the mining of structured content from unstructured web pages. Applications include customer support, automated e-mail routing and reply, and knowledge management, such as document clustering, content categorization, and keyword extraction and associations.

Web-structure mining focuses on analyzing the link structure of the Web to identify interesting relationships and patterns describing the connectivity of documents in the Web. Such relationships are then used to retrieve relevant documents in response to user requests. Finally, Web usage mining focuses on analyzing the visitor's navigation of a web site to assess problems with its organization, such as long traversal paths for example, or to identify paths that lead to sales and cross-sales.

Web Usage Mining consists of three phases which are named data processing, pattern discovery and pattern analysis. This phase has two parts called data cleaning and filtering. Filtering is the most important task in web usage mining since the quality of mined patterns depends on this directly. In the pattern discovery phase, Special pattern discovery algorithms applied on raw data which is output of the data processing phase .In the pattern analysis phase interesting knowledge is extracted from frequent patterns and these results are used in various applications such as personalization, system improvement data processing, pattern discovery data cleaning and filtering phase . In the pattern analysis improvement, site modification.

Web server data correspond to the user logs that are collected at Web server. Some of the typical data collected at a Web server include IP addresses, page references, and access time of the users and is the main input to the present Research. This Research work concentrates on web usage mining and in particular focuses on discovering the web usage patterns of websites from the server log files. This Research work concentrates on web usage mining and in particular focuses on discovering the web usage patterns of websites from the server log files. The comparison of Apriori algorithm and Frequent Pattern Growth algorithm is done.

1.2 Association Rule

Association rule is used to find out the items which are frequently used together. The Presence of one set of items in a transaction implies other set of items.The terms used in these rule are

Support: The support of an association rule X implies Y is the percentage of transaction in the database that consists of
X U Y.

Confidence: The confidence for an association rule X implies Y is the ratio of the number of transaction that contains X U Y to the number of transaction that contains X.

Large Item Set: A large item set is an item set whose number of occurrences is above a threshold or support.

The task of association rule mining is to find correlation relationships among different data attributes in a large set of data items, and this has gained lot of attention since its introduction. Such relationships observed between data attributes are called association rules. A typical example of association rule mining is the market basket analysis [1].

1.3 Pattern mining from Web Transaction

A user session is all of the page references made by a user during a single visit to a site. A transaction differs from a user session in that the size of a transaction can range from a single page reference to the entire page references in a user session, depending on the criteria used to identify transactions [1]. Once user transactions or sessions have been, identified, there are several kinds of access pattern mining that can be performed depending on the needs of the analyst, such as path analysis, discovery of association rules and sequential patterns and clustering, and classification.

Association rule mining discovery techniques are generally applied to databases of transactions where each transaction consists of a set of items. In such a framework the problem is to discover all associations and correlations among data items where the presence of one set of items in a transaction implies (with a certain degree of confidence) the presence of other items. In context of web usage mining, this problem amounts to discovering the correlations among references to various files available on the server by a given client. Each transaction is comprised of a set of URLs accessed by a client in one visit to the server.

Association Rules are probably the most elementary data mining technique and, at the same time, the most used technique in Web Usage Mining. When applied to Web Usage Mining, association rules are used to find associations among web pages that frequently appear together in users' sessions. The typical result has the form "A.html, B.html) C.html" which states that if a user has visited page A.html and page B.html, it is very likely that in the same session, the same user has also visited page C.html. Most of the commercial applications of Web Usage Mining exploit consolidated statistical analysis techniques. In contrast, research in this area is mainly focused on the development of knowledge discovery techniques specifically designed for the analysis of web usage data.

2. APRIORI ALGORITHM

It searches for large itemsets during its initial database pass and uses its result as the seed for discovering other large datasets during subsequent passes. Rules having a support level above the minimum are called large or frequent itemsets and those below are called small itemsets. The algorithm is based on the large itemset property which states: Any subset of a large itemset is large and any subset of frequent item set must be frequent.

The first algorithm for mining all frequent itemsets and strong association rules was the AIS algorithm by [3]. Shortly after that, the algorithm was improved and renamed Apriori. Apriori algorithm is, the most classical and important algorithm for mining frequent itemsets.The Apriori algorithm performs a breadth-first search in the search space by generating candidate k+1-itemsets from frequent k itemsets. The frequency of an item set is computed by counting its occurrence in each transaction.

Apriori is an influential 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.

There are two steps for understanding that how Lk-1 is used to find Lk:-

2.1 The join step:-

To find Lk, a set of candidate k-itemsets is generated by joining Lk-1 with itself. This set of candidates is denoted Ck.

2.2 The prune step:-

Ck is a superset of Lk, that is, its members may or may not be frequent, but all of the frequent k-itemsets are included in Ck. A scan of the database to determine the count of each candidate in Ck would result in the determination of Lk.Ck, however, can be huge, and so this could involve heavy computation to reduce the size of Ck.

3. User behavior based on FP-tree

A frequent-pattern tree (or FP-tree in short) is a tree structure. It consists of one root labeled as "null", a set of item-prefix subtrees as the children of the root, and a frequent-item-header table. Each node in the item-prefix subtree consists of three fields: item-name, count, and node-link, where item-name registers which item this node represents, count registers the number of transactions represented by the portion of the path reaching this node, and node-link links to the next node in the FP-tree carrying the same item-name, or null if there is none[2].

Input: A transaction database DB and a minimum support threshold

Output: FP-tree, the frequent-pattern tree of DB.

FP-growth is a well-known algorithm that uses the FP tree data structure to achieve a condensed representation of the database transactions and employs a divide and-conquer approach to decompose the mining problem into a set of the above problem by Reducing passes, Shrinking number of candidates and facilitating support counting of candidates. An FP-tree-based pattern-fragment growth mining method is developed, which starts from a frequent length-1 pattern (as an initial suffix pattern), examines only its conditional-pattern base (a "sub-database" which consists of the set of frequent items co-occurring with the suffix pattern), constructs its (conditional) FP-tree, and performs mining recursively with such a tree.

The FP-growth algorithm is one of the fastest approaches for frequent item set mining. The FP-growth algorithm uses the FP-tree data structure to achieve a condensed representation of the database transaction and employees a divide-and conquer approach to decompose the mining problem into a set of smaller problems. In essence, it mines all the frequent itemsets by recursively finding all frequent itemsets in the conditional pattern base which is efficiently constructed with the help of a node link structure.

A prefix tree is a data structure that provides a compact representation of transaction data set. Each node of the tree stores an item label and a count, with the count representing the number of transactions, which contain all the items in the path from the root node to the current node.

The frequent items are computed as in the Apriori algorithm and represented in a table called header table. Each record in the header table will contain the frequent item and a link to a node in the FP-Tree that has the same item name. Following this link from the header table, one can reach all nodes in the tree having the same item name. Each node in the FP-Tree, other than the root node, will contain the item name, support count, and a pointer to link to a node in the tree that has the same item name.

3.1 The main components of FP tree

- It consists of one root labeled as "root", a set of item prefix sub-trees as the children of the root, and a frequent-item header table.
- Each node in the item prefix sub-tree consists of three fields: item-name, count, and node-link, where item-name registers which item this node represents, count registers the number of transactions represented by the portion of the path reaching this node, and node-link links to the next node in the FP tree carrying the same item-name, or null if there is none.
- Each entry in the frequent-item header table consists of two fields, (1) item-name and (2) head of nodelink, which points to the first node in the FP-tree carrying the item-name.

4. Time Comparison

As a result of the experimental study, revealed the performance of Apriori and FP-tree algorithm. The run time is the time to mine the frequent itemsets. The experimental result of time is shown in Fig.1 reveals that the FP-tree outperforms the Apriori approach.

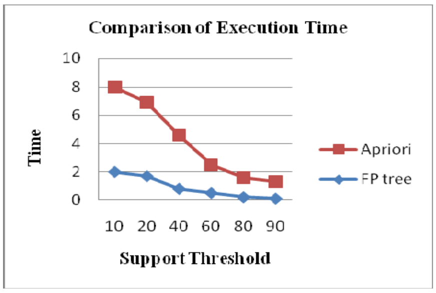

Fig. 1 Execution Time Comparison

5. Conclusion

The association rules play a major role in many data mining applications, trying to find interesting patterns in data bases. Apriori is the simplest algorithm which is used for mining of frequent patterns from the transaction database. 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.Apriori algorithm uses large item set property, easy to implement, but it repeatedly scan the database. Apriori takes more time to scan the large Frequents patterns**.** The frequent pattern tree (FP-tree) is used for storing compressed, crucial information about frequent patterns, and developed a pattern growth method, FP-growth, for efficient mining of frequent patterns in large databases. Fp algorithm uses divide and conquer approach and it is more efficient than apriori algorithm and also takes lesser time and gives better performance.

References

[1]Huiping peng “Discovery of Interesting Association Rules on Web Usage Mini ng” 2010 International Conference.

[2] Han J., Pei J., Yin Y,”Mining frequent patterns without candidate generation: A frequent-pattern tree approach “Data Mining and Knowledge, 2003.

[3] Agrawal R, Srikant R., "Fast Algorithms for Mining Association Rules", VLDB. Sep 12-15 1994, Chile, 487-99, pdf, ISBN 1-55860-153-8.

[4] B.Santhosh Kumar and K.V.Rukmani “Implementation of Web Usage Mining Using APRIORI and FP Growth Algorithms “in 2010.

[5]J. Srivastava, R. Cooley, M. Deshpande, PN. Tan, “Web usage mining: discovery and applications of usage patterns from web data”. Vol. 1, No. 2, 2000, pp.12–23.

[6]S.Veeramalai, N.Jaisankar and A.Kannan “Efficient Web Log Mining Using Enhanced Apriori Algorithm with Hash Tree and Fuzzy” in 2010.

[7]Jaideep Srivastava, Robert Cooleyz, Mukund Deshpande, Pang-Ning Tan proposed “Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data” in 2000.

[8]JIAWEI HAN, JIAN PEI, RUNYING MAO “Mining Frequent Patterns without Candidate Generation” in 2004.

[9]Hao Yan, Bo Zhang, Yibo Zhang, Fang Liu, Zhenming Lei “Web usage mining based on WAN users’ behaviors “in 2010.

[10] Sanjay Kumar Malik, Nupur Prakash, S.A.M. Rizvi” Ontology and Web Usage Mining towards an Intelligent Web focusing web logs” 2010 IEEE.