Discovery the User Access Pattern from Web Log Based on Association Rule Mining

Hsu Mon Thet Wai, Thandar Win
University of Computer Studies, Yangon
hsumonthetwai@gmail.com

Abstract

Nowadays, most of the people who may be in educational field, IT field, or business field or others widely use the internet. The e-learning, e-mailing, eCommerce and other online techniques are used. The usage data are recorded by the Web server as the Web Log Data. Web usage mining mines Web Log Data to discover user access patterns of Web page. Association Rules are used to discover the user access patterns. So, in this paper, the Support-ordered Trie Itemset (SOTrieIT) algorithm to analyze which is the most frequent visited website and Apriori algorithm, which is candidate generation algorithm to know the pairs of websites, are used. From the result of this paper, user can analyze the websites' popularity, usefulness and effectiveness. Web administration can use this system for finding the appropriate actions.

1. Introduction

The use of the World Wide Web as a medium for business, communication, education and government has increased at an amazing rate over the past few years. The goal of a website is to meet the needs of users and provide the valuable information. Data mining efforts associated with the Web is called Web mining. Web mining is divided into three classes: Web Usage Mining, Web Content Mining and Web Structure Mining. 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 Content Mining is the process of extracting knowledge from the contents of documents or their descriptions. Web Structure Mining is the process of inferring knowledge from the World Wide Web organizations and links between references and referents in the Web. [1]

Most of finding information is usually generated automatically by Web Servers and collected in server access log. A Log entry may contain the client IP address, user id, request method, and the URL of the page accessed, the protocol used for data transmission, the error code and the number of bytes transmitted. Web Usage Mining uses the Web Log Data to discover the user access pattern. There are three main tasks for performing Web Usage mining. These are Data preprocessing, pattern discovery and pattern analysis.

The main objective of this system is:

- To provide the implementation of web usage mining
- To know how to apply Association Rule Mining algorithm
- To know the fast frequent itemset discovery
- To analyze which site is popular and frequently use
- To improve the online business
- To know what the needs of their customers are

In this paper, there are six sections. Firstly, section 1 describes the introduction of this paper.

In section 2, related works for their system are presented.

Section 3 describes the Background Theory used by this system.

And then, section 4 is the description of proposed system architecture

In section 5, the implementation of this system is described.

Section 6 is the conclusion of this paper.

2. Related Work

Web usage mining is used in many different ways. To find the usage pattern, association rules, clustering, classification, etc are used. The WEBSHIFT system is designed to perform Web Usage Mining from server logs in the extended NSCA format (includes referrer and agent fields). [2] Yongjian Fu builds adaptive Websites by evolving site structure to facilitate user access. This approach consists of three steps: preprocessing, page classification and site reorganization. [3] And then the association rules are also used to find the access pattern. Yew-Kwong Woon critically examine existing preprocessing data structures in association rule mining for enhancing performance in an attempt to understand their strengths and weakness, and analyses culminate in a practical structure called the Support-Ordered Trie Itemset (SOTrieIT) and two synergistic association rule mining algorithms to accompany it. [4] In this paper, we implement the discovering of usage pattern using association rule. The Web Log data are used and to know the usage patterns, Support-Ordered Trie Itemset (SOTrieIT) algorithm and Apriori algorithm are used.

3. Background Theory

3.1 Web Usage Mining

Web usage mining is the type of Web mining activity that involves the automatic discovery of user access patterns from one or more Web servers. As more organizations rely on the Internet and the World Wide Web to conduct business, the traditional strategies and techniques for market analysis need to be revisited in this context. Organizations often generate and collect large volumes of data in their daily operations. Most of this information is usually generated automatically by Web servers and collected in server access logs.

Web usage mining looks at logs of Web access. General access pattern tracking is a type of usage mining that looks at Web pages visit history. Web Servers record access information as a click-stream-data into log files. Whenever a Web page is clicked, corresponding data will be generated and recorded. There is valuable information in the profile, such as the access patterns of users.

Analyzing such data can help these organizations to determine the life time value of customers, cross marketing strategies across products, and effectiveness of promotional campaigns, among other things. Analysis of server access logs and user registration data can also provide valuable information on how to better structure a Web site in order to create a more effective presence for the organization. For organizations that sell advertising on the World Wide Web, analyzing user access patterns helps in targeting ads to specific groups of users. [5]

A Web usage mining system is basically a data mining system for the particular domain of Web usage data. A data gathering and preprocessing module collects the Web usage data and cleans and transforms the log entries. A module of user behavior pattern discovery applies a variety of data mining algorithms, such as association rule mining, sequential pattern analysis, clustering, and classification on the formatted usage data, in order to discover useful patterns. [1]

3.1.1 Data Preprocessing

Preprocessing consists of converting the usage, content, and structure information contained in the various available data sources into the data abstractions necessary for pattern discovery. [2]

3.1.2 Pattern Discovery

Pattern Discovery is the key component of the Web mining, which converges the algorithms and techniques from data mining, machine learning, statistics and pattern recognition etc research categories.

It may separate into: statistical analysis, association rules, clustering, classification, sequential pattern, dependency Modelling.

Statistical Analysis: The analysts may perform different kinds of descriptive statistical analyses based on different variables when analyzing the session file; the statistical techniques are the most powerful tools in extracting knowledge about visitors to a Web site.

Association Rules: It refers to sets of pages that are accessed together with a support value exceeding some specified threshold. This technique can be used to discover unordered correlation between items found in a database of transactions.

Clustering: It is a technique to group together users or data items (pages) with the similar characteristics. It can facilitate the development and execution of future marketing strategies.

Classification: It is the technique to map a data item into one of several predefined classes, which help to establish a profile of users belonging to a particular class or category.

Sequential Pattern: This technique intends to find the inter-session pattern, such that a set of the items follows the presence of another in a time ordered set of sessions of episodes. It helps web marketer to predict the future trend.

Dependency Modelling: This technique provides a theoretical framework for analyzing the behaviour of users, and is potentially useful for predicting future we resource consumption.

3.1.3 Pattern Analysis

Pattern Analysis is the final stage of the Web usage mining. The goal of this process is to eliminate the irrelative rules or patterns and to extract the interesting rules or patterns from the output of the pattern discovery process. [6]

3.2 Association Rule Mining

Association rule mining, one of the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. [8]

The existing preprocessing data structures in association rule mining is examined for enhancing performance. A practical structure is Support-Ordered Trie Itemset (SOTrieIT) algorithm. The most well known algorithm for producing association rules is Apriori algorithm.

3.2.1 Support-Ordered Trie Itemsets (SOTrieIT) Algorithm

The SOTrieIT is constructed by extracting 1-itemsets and 2-itemsets from all transactions and using them to update the SOTrieIT. The SOTrieIT is ordered by support count. The bracket number beside a node's label denotes the support count. Its nodes are support-ordered and have two levels of nodes (excluding the special ROOT node). The main weakness of the SOTrieIT is that it can only discover L1 and L2, while its main strength lies in its speed in discovering L1 and L2. L1 and L2 can be found promptly because there is no need to scan the database. In addition, the search (depth-first)can be stopped at a particular level the moment a node representing a nonfrequent itemset is found because the nodes are all support-ordered. Another advantage of the SOTrieIT is that it can be constructed online, meaning that each time a new transaction arrives, and the SOTrieIT can be incrementally updated. It requires far less storage space than a trie because it is only two level deep and can be easily stored in both memory and files. [4]

3.2.2 Apriori Algorithm

Although several algorithms have been proposed for generating association rules, classic algorithm is the Apriori algorithm of Agrawal and Srikant. The key idea of the algorithm is to begin by generating frequent itemsets with just one item (1-itemsets)and to recursively generate frequent itemsets with 2 items, then frequent 3-itensets and so on until we have generated frequent itemsets of all sizes. Without loss of generality we will denote items by unique, consecutive (positive) integers and that the items in each itemset are in increasing order of this item number. When we refer to an item in a computation we actually mean this item number.

It is easy to generate frequent 1-itemsets. All we need to do is to count, for each item, how many transactions in the database include the item. These transaction counts are the supports for the 1-itemsets. We drop 1-itemsets that have support below the desired cut-off value to create a list of the frequent 1-itemsets.

The general procedure to obtain k-itemsets from (k-1)-itemsets for k=2, 3, is as follows. Create a candidate list of k-itemsets by performing a join operation on pairs of (k-1)-itemsets in the list. The join is over the first (k-2) items, I.e. a pair is combined if the first (k-2) items are the same in both members of the pair. If this condition is met the join of pair is a k-itemset that contains the common first (k-20 items and the two items that are not in common, one from each member of the pair. All frequent k-itemsets must be in this candidate list since every subset of size (k-1) of a frequent k-itemset must be a frequent (k-1) itemset. However, some k-itemsets in the candidate list may not be frequent k-itemsets. We need to delete these to create the list of frequent k-itemsets. To identify the k-itemsets that are not frequent we examine all subsets of size (k-1) of each candidate k-itemset. Notice that we need examine only (k-1)-itemsets that contain the last two items of the candidate k-itemset. If any one of these subsets of size (k-1) is not present in the frequent (k-1) itemset list, we know that the candidate k-itemsets cannot be a frequent itemset. We delete such k-itemsets from the candidate list. Proceeding in this manner with every itemset in the candidate list we are assured that at the end of our scan the k-itemset candidate list will have been pruned to become the list of frequent k-itemsets. We repeat the procedure recursively by incrementing k. We stop only when the candidate list is empty. [7]

Algorithm: Apriori.Find frequent itemsets using an iterative level-wise approach based on candidate generation
Input: Database, D, of transaction; minimum support threshold, min_sup.
Output: L, frequent itemsets in D.
Method:

(1) for (k=2; $L_{k-1} \neq \emptyset$; k++) {
(2) L1 = find_frequent_1-itemsets(D);
(3) C_k = **apriori_gen** (L_{k-1}, min_sup);
(4) for each transaction t $\in$ D { // scan D for counts
(5) C1 = subset (C_k, t); //get the subsets of t that are candidates
(6) for each candidate c $\in C_1$
(7) c.count++;
(8) }
(9) $L_k = \{c \in C_k | c.count \geq min\_sup\}$
(10) }
(11) Return L = $U_k L_k$;

procedure apriori_gen (L_{k-1} : frequent (k-1) - items: min-sup: minimum support threshold)

(1) for each itemset $l_1 \in l_{k-1}$
(2) for each itemset $l_2 \in l_{k-1}$
(3) if (l_1 [1] = l_2[1]) ^ (l1[2] = l2 [2]) ^…^ (l1 [k-2] = l2 [k-2] ^ (l1 [k-1] = l2 [k-2]) then {
(4) c = $l_1 \propto l_2$; //join step: generate candidates
(5) If **has_infrequent_subset (c, L_{k-1}) then**
(6) delete c; // prune step: remove unfruitful candidate
(7) else add c to C_k;
(8) }
(9) return c_k;

Procedure has_infrequent_subset (c: candidate k-itemset; L_{k-1}: frequent (k-1) –itemsets);
// use prior knowledge

(1) for each (k-1)-subset s of c
(2) if s L_{k-1} then
(3) Return TRUE;
(4) return FALSE;

4. Proposed System Architecture

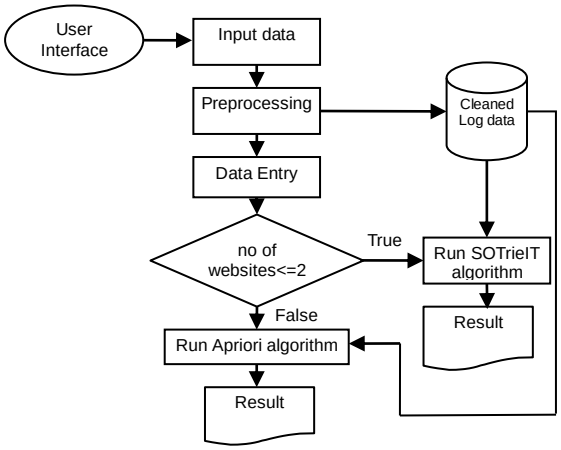

Figure.1. System Architecture

Web Log Data is used to discover the user access pattern. Firstly, user choose the path of the access log file from browse button and when the user click the parse button, the system will parse the raw log data into Cookie, Session, IP address, Protocol, Status, Size, Method. URL Request, User Agent, File type and also show the number of record. These parsed data have many unnecessary data, so user need to clean these data. To clean them, user clicks the clean button. The system will clean data which its protocol is not "TCP_MISS", status is not "200", method is not "GET" and file type is not "text/html" and also move advertisements. Then, the clean data are sectioned with same IP. Requests from the same IP address are grouped into a session. A session represents a single visit of a user. Each session contains the User_ID which is only for counting the number of user, IP address and visited URL requests. The preprocessed log data is an input for the algorithm used in this system. The User session is to provide the individual data of user. For each user, this section show the User's IP address and visited websites and also number of record count visited. SOTrieIT algorithm use the log data as input and find frequent websites which may be most visited websites (L1) and pairs of visited websites (L2). This algorithm is useful for the fast discovery the access pattern means that L1 and L2. Apriori algorithm runs when the number of websites is greater than 2. So, user needs to type the minimum support (min_sup) and minimum confidence (min_conf). And then, the result, the access pattern is shown.

The SOTrieIT allows L1 and L2 to be discovered without much computation and without database scans. On the other hand, regardless of the support threshold. The other algorithm needs more than one database scans; one to discover L1 and another to sort and prune transaction according to L1. So, the SoTrieIT algorithm can fastly discover the access pattern.

5. Implementation of the System

The system is implemented for discovering and analyzing web users' behaviors. It is developed by using Microsoft C#.Net and MySQL Server 2005 for data storage.

The main frame of the system which contains the menus such as File menu, Process menu and Help menu. File menu contains Home and Exit (to exit from the system). Process menu contains Preprocess, Show session and Run.

When the preprocess menu is clicked, Data preprocessing section is shown in **Figure.2**. In this section, we click the browse button to browse the path of the text log file and then we click parse button to parse the raw text file.

Figure.2. Parsed Log Data

After parsing step, log data are cleaned to get the relevant attributes. To do this, we will click the clean button. The result is as follow.

Figure.3. Cleaned Log Data

When we want to know the individual information, we can choose the Show Session menu. In this section, we can get User-ID (number of user), IP address and the visited URLs as shown in **Figure.4**.

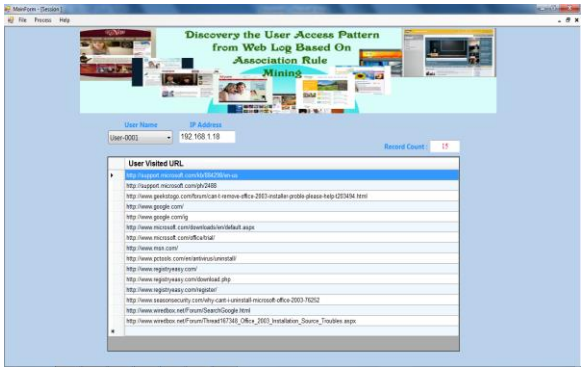

Figure.4. User Session

When the Run menu is clicked, the message box which asks for the number of websites appears as shown in **Figure.5**. We need to type the number of websites, so we type "1" for number of websites.

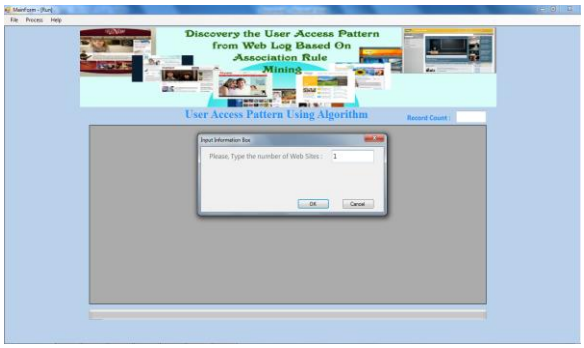

Figure.5. Message box for algorithm running

When the user click the "OK" button, the result of SOTrieIT algorithm is shown in **Figure.6.**

Figure.6. SOTrieIT result for most visited websites

When we type "2" for number of website in the message box, the SoTrieIT algorithm run and the output of pairs of website is shown as follow.

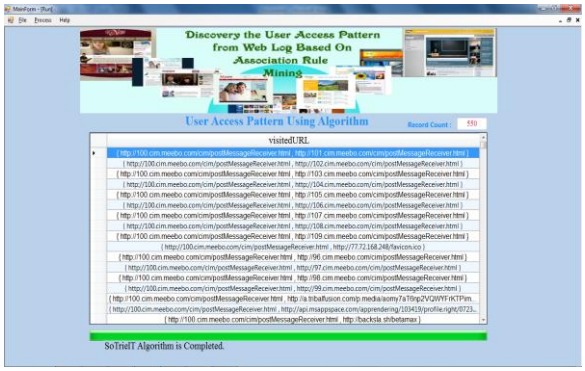

Figure.7. SOTrieIT result for pairs of visited website

If the number of website is greater than 2, two input textboxes for minimum support (min-sup) and minimum confidence (min-conf) appear to run the Apriori algorithm as shown in **Figure.8**.

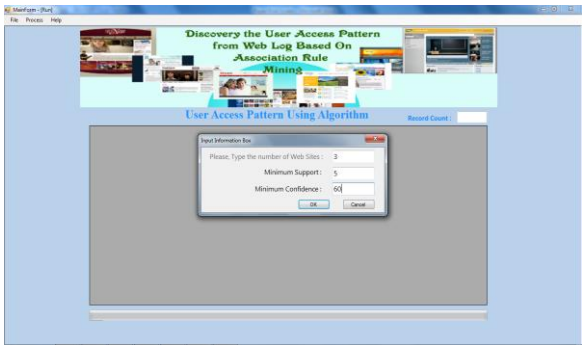

Figure.8. Message box for algorithm running

When the user click the "OK" button, the result of Apriority algorithm is shown in **Figure.9**.

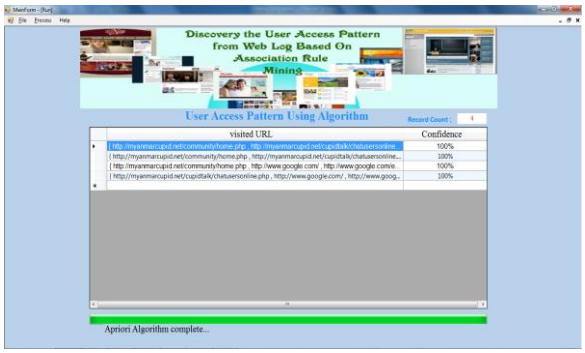

Figure.9. Apriori result

The existing system in reference [1] is analyzing and discovering the Web users' behaviors. The user can know the least usage patterns, the most usage patterns or usage pattern for desired percentage. The user can see the occurrences and top ten lists of usage patterns. By studying the resulted information of this, Web user can know the sequence of pages viewed and the on-line business can

advertise their product in the popular Web sites.

This system is also analyzing and discovering the Web users' behaviors. The user can know the individual information, means that which user uses which sites. And the user can know the most frequent websites of input Log data. Unlike the existing system, the user can know the pairs of visited websites. From these results, the user can analyze the websites' popularity, usefulness and effectiveness. Also, this system is used by Web administration for finding the appropriate actions.

6. Conclusion

This system provides the implementation of web usage mining and association rule mining algorithm, SOTrieIT and Apriori algorithm. It can give the fast discovery of frequent websites and also the pairs of websites. The web administrators only need to input Squid proxy Log file, the system can provides the result of user's behavior. So, they can analyze the users' behavior and their needs. This system is easy to use and know the required information and popular websites.

7. References

[1] Yi Yi Mar
"Analysis of Web Users' Behaviors Using Agglomerative Hierarchical Clustering Algorithm"
University of Computer Studies, Yangon
December, 2008

[2] Srivastava J.,Cooley R.,Deshpande M., Tan P.
"Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data"
University of Minnesota: January 2000

[3] Yongjian Fu, Ming-Yi Shih, Mario Creado, Chunhua Ju
"Reorganization Web sites Based on User Access Patterns"
Department of Computer Science
University of Missouri-Rulla

[4] Yew-Kwong Woon,
"A Support-Ordered Trie for Fast Frequent Itemset Discovery"
Wee-Keong Ng, Member, IEEE Computer Society, and Ee-Peng Lim, Senior Member, IEEE

[5] http://maya.cs.depaul.edu/~mobasher/webminer/survey/node6.html

[6] J.Han, J.Pei, and Y.Yin,
"Mining Frequent Patterns without Candidate Generation,"
Proc. ACM SIGMOD Conf, pp.1-12, 2000

[7] Jiawei Han, Micheline Kamber
"Data Mining: Concepts and Techniques"

[8] Sotiris Kotsiantis, Dimitris Kanellopoulos
"Association Rules Mining: A Recent Overview"
Educational Software Development Laboratory
Department of Mathematics,
University of Patras, Greece