Indian Journal of Science and Technology, *Vol 7(9), 1350–1354, September 2014*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Efficiency Calculation of Mined Web Navigational Patterns

L. K. Joshila Grace* **and V. Maheswari**

Sathyabama University, Jeppiaar Nagar, Chennai, India; joshilagracejebin@gmail.com

Abstract

The proposed work does a web utility mining process for identifying the useful web navigation patterns. An optimal prefix tree is generated from the log file details and the mining is performed. From the set of mined web navigational patterns the efficiency is found by considering certain parameters. The parameters like frequency, utility, downloads, book mark, selection are considered for each web page and efficiency for the web navigation pattern is found. The work is done by using real data set extracted from an e-commerce web site and with synthetic data set. This provides a better analysis of the web site. The result provided by this proposed work can be used or various application in developing the web site contents.

Keywords: Non Sequential Pattern, Pattern Weight, Sequential Pattern, Utility, Web Path Traversal

1. Introduction

The World Wide Web consists of a collection of interlinked links of hyper linked documents. These hyper linked links are a set of web sites that provide various type of information. These web sites consist of wide range of interlinked web pages. The expansion of the contents in the World Wide Web increases tremendously each day. The necessity of a suitable web mining is needed for improvising various web pages of a web site. There are various types of web site like Personal Websites, Photo Sharing Websites, Social Websites, Authors Websites, Community Websites, Blogs, Informative Websites, Online Business Brochure/Catalog, Educational Websites, Gaming Website, Government Web site, E-commerce Websites etc., All these web sites have, competitive web sites in their own area for attracting the web site visitors. Therefore various mining techniques are needed to define their efficiency. The various types of web mining are web content mining, web structure mining and web usage mining. Each mining technique involves in the mining of the web site in various ways.

1.1 Web Content Mining

The mining depends on the contents of the web site. The contents will be of any type like image, text, video etc.

1.2 Web Structure Mining

This type of mining depends on the URL structure of the web site links. The hyper links are used for this type of web mining.

1.3 Web usage Mining

The mining done of this type depends on the value of the usage data of various web site users. These usage details are collected from the web log files of the web site. These log files resides in the web server of the particular web site. The proposed work deals only with this type of web mining process.

The log files are set of raw data residing in the web site memory. They contain values like IP address of the web site visitor, time utilized by the web server, browser details, type of request etc. These log file can be extracted application level or server level. These types of user details extracted from the log file are converted into web navigational patterns.

There are broad ranges of research carried out on utility mining for applying in various applications. Identifying user behavior, predicting the future navigation of the user, web site traffic analysis, web personalization etc. are some of the research areas in which analysis of the web

*Author for correspondence

navigational patterns are done. The proposed work aims in mining the frequent patterns and also calculates the efficiency of the web site.

The navigational pattern of the web site shows the users surfing activities throughout the web site. This type of surfing activities can be analyzed for the improvisation of the web site. These patterns analysis involves pattern recognition, extracting the patterns, calculating the quality value for the extracted patterns. These navigational patterns are classified into two types' sequential patterns and non sequential patterns[5]. Sequential patterns are one which follows a particular sequence. (a, b, c) (c, d, e) where a,b,c,d,e are web pages these represents sequential patterns. (a, *, b) (c, *, d) where a, b, c, d are web pages * represents any page in the web site. This type of representation is done for non sequential patterns.

The proposed work concentrates only on both sequential patterns. The factors that would be helpful for the analysis of the web navigational patterns are frequency, utility, downloads, book mark, selection, etc., the proposed work considers only selective parameters out of the whole set of available parameters in the web log file.

1.4 Frequency

It gives the number of user visited the web page while traversing through the web site. The count gets incremented as the number of users visiting the web path increase.

1.5 Utility

It provides the amount of time spent by the user in the same web page. There are various factors that will affect the value of utility indirectly. Considering the case where the user will have a very slow down load, due to high network traffic or the service provider has only limited speed allotted for the user. There are possibilities of user, opening the web site and not using it for a long period. These may increase the utility value which is not the actual value expected for finding the optimal pattern.

1.6 Down Loads

The value would provide the information of the number of users downloaded the content in the web page. The value is modified when the user would perform a down load. The down load may be content on the web page or the web page itself. As the number of user down loading increases the value gets incremented.

1.7 Selection

This type of parameter can be collected for web sites which have items for selection by the user. For example, the e-buying web site gives a number of options for the user to select. In such type of web pages analyzing this parameter is useful for finding the optimal pattern.

The proposed work extracts the web log file details, generate a prefix tree, mining is performed on the prefix tree and efficiency calculation is done. The prefix tree is an organized representation of the web navigational patterns. The prefix tree is considerably small even for large amount of data. Therefore the height of the tree is less which would need only less time for scanning the prefix tree.

2. Related work

This section discuss about the existing work which uses the prefix tree data structure of various application.

2.1 Extrapolation Prefix Tree for Data Stream Mining using a Landmark Model

In order to store the arriving transactions the prefix tree data structure is used[1]. The existing work compares this with CP- tree that is compact tree and shows efficiency. The prefix tree makes easier in updating process and execution time is very less. It involves in mining the frequent pattern.

2.2 Mining Frequent Itemsets using Node-Sets of a Prefix-tree

A novel node-set-based algorithm, Node Set (NS), is developed for mining frequent itemsets[2]. During a mining process, all the node-sets derive from a prefix-tree storing the complete frequent itemset information about the mined database. This requires a large memory to store the mined frequent Item set

2.3 Prefix-tree-projected Pattern Growth

This gives a novel induced subtree mining algorithm, called prefix tree span (i.e. Prefix-Tree-projected Induced-Subtree pattern), this extracts the induced subtree patterns by growing the frequent prefix-trees[3]. By using divide and conquer, the mining of local length-1 frequent subtree

patterns in Prefix- Tree-Projected database recursively will lead to the complete set of frequent patterns. This approach is restricted to only one pattern.

2.4 BPA: a Bitmap-prefix-tree Array

This is applied for discovering frequent closed itemset in large transaction database. An efficient BPA data structure is used to enhance not only computation-time and memory-space in the complete preprocessing data but also in those in the frequent searching[4].

These are the various frequent pattern recognition algorithm which uses the prefix tree data structure. The existing work deals with the patterns for finding the frequent patterns only. They are also restricted to a finite length of pattern only. Either the frequency or the utilization parameter is used in the existing work. This may not provide the efficient result to the user.

Modifications are done to the existing prefix tree and an optimal prefix tree is generated. This optimal prefix tree considers parameters like frequency, utility, down loads and selection. By using optimal prefix tree, optimal pattern are discovered and the efficiency can be calculated. This efficiency calculation is restricted to only the highly optimal pattern and reasonable optimal pattern found and not to all the patterns considered for experimentation.

3. Proposed Work

This section gives the detailed description of the step wise procedure of the web navigational pattern mining an efficiency calculation.

3.1 Extracting the Data

The log file of a particular e-commerce website was considered for which the pattern efficiency is found. The raw data of the log file is converted to the format useful for optimal prefix tree generation. Excluding the unnecessary parameters only the useful information are retrieved from the log file. The parameters considered for this work was frequency, utility, down loads and selection.

The extractions of web navigational patterns are not user specific. They are not saved according to the IP address. Each pattern is considered as individual pattern of some web site visitor.

The two type of input data considered are real data set and synthetic data set. The real data set is considered for a particular e-commerce website by using their log file details. Only the necessary parameters like frequency, utility, down loads and selection are extracted from the log file.

Another set of input considered are synthetic data set, they are generated for further processing. These data sets are generated with parameters like frequency, utility, down loads and selection. These are two data considered for mining the interesting patterns.

3.2 Generating Optimal Prefix Tree

The normal prefix tree considered has only frequency as its parameter for each node. The modifications done to the prefix tree enables to generate optimal prefix tree. The parameters like frequency, utility, down loads and selection are considered in each node of this optimal prefix tree.

In order to have a precise structure of the web log file the optimal prefix trees are used. The Figure 1 represents the individual node of the optimal prefix tree where "A" is the web page, the value 11 gives number of times the web page has been visited frequently, 53 is the time utilization value which is given in seconds, 2 is the count of number of user has downloaded the web page. The last cell containing 1 represents the item present in the web page is selected by one user.

For each web page considered in a web navigational pattern the time will always be greater than one, downloads will always be equal to one or zero, selection will also be always one or zero. So according the values extracted from the web log file the optimal prefix tree values gets modified. For all the surfing done by the user within the web site the entry is made in the web log file. The analysis is done only on a single web site. The log file details of one particular web site are only considered (Table 1).

3.3 Discovering the Matching Patterns

Matching patterns are mined only from a set of test data obtained (Table 2). Patterns of different length are

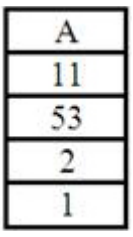

Figure 1. Optimal prefix tree node.

Table 1. Algorithm : Optimal prefix tree generation

1. ***Input***. Time duration, downloads, selection details of the web page extracted from web log file
2. Each continuous user pattern is extracted.
3. Construct the optimal prefix tree by making the first user of the extracted log file as the root node
4. **For** each consecutive user patterns verify
5. **If** there exist nodes with the same web navigational sequence of web pages in the optimal prefix tree
6. Modify all the necessary parameters present in the nodes that cover the path of the user.
7. **Else**
8. **If** partial web navigational sequence in the beginning of the web pattern is matching
9. Modify the details until the web navigational sequence and construct a new branch from the point the web navigational pattern does not match
10. **Else**
11. Generate a new node from the root.
12. **End if**
13. **End for**
14. ***Output***. Optimal Prefix tree based on the web navigational sequence of web pages visited by the users

Table 2. Algorithm : Discovery of matching sequential pattern

1. ***Input:*** Generated Optimal Prefix tree
2. Extract a set of web path traversal patterns with different length as test data
3. **For** each web path traversal pattern check with the generated Optimal prefix tree
4. **If** sequence is same
5. Consider the sequence
6. Efficiency calculation is done
7. **Else**
8. Ignore the sequence
9. **End if**
10. **End for**
11. ***Output:*** Matching web path traversal pattern that satisfies the sequence in the Optimal prefix tree

considered as test patterns. This work considers 5 length, 4 length, 3 length, 2 length and 1 length patterns. These test patterns are compared with the generated optimal prefix tree and the set of matching patterns are discovered. This mining was done for only sequential web path traversal pattern.

3.4 Efficiency Calculation

Equation 2 to Equation 4, was derived from the previous work[6]. The proposed work finds the efficiency of the pattern and it also considers some additional parameters.

$$P_{eff} = P_u + P_{imp} + P_{freq} + P_{sl} \quad (1)$$

$$P_u = \frac{P_{td}}{TN \times N_{td}^{\max}} \quad (2)$$

$$P_{td} = \sum_{i=1}^{n} td_i \quad (3)$$

$$P_{imp} = \frac{P_{bc}}{TN} \quad (4)$$

$$P_{freq} = TP_{count} \quad (5)$$

$$P_{sl} = Max(P_{sl}) \quad (6)$$

where,

P_{eff} → Efficiency value of a pattern
P_u → Utility of a pattern
P_{imp} → Importance of the pattern
P_{freq} → Frequency of the pattern
P_{sl} → Selection of the pattern
P_{td} → Time duration of total web pages in a pattern
TN → Total number of nodes in the prefix tree
$N_{td}^{\max}$ → Maximum time duration of a node in prefix tree
td_i → Time duration of i^{th} web page in a pattern
n → Total number (i.e. length) of web pages in a pattern
P_{bc} → The total number of web pages down loaded in a pattern
TP_{count} → Total number of similar pattern in the prefix tree

4. Experimental Results

There are two types of data considered for implementation of optimal pattern recognition, synthetic data set and real data set. The real data set are values that are obtained from a particular web sites log file. These log file details are collected for about six month. The web site considered containes seventy five interlinked web pages.

The synthetic data set is generated manually by considering twenty six interlinked links and all the parameters are considered for each web path.

The execution is implemented by considering different number of input dataset Figure 2 shows the comparison of the tree height of the variable number of dataset that are used in the work. The work is confined to only the static data received from the web log file and the synthetic data generated. The height of the tree is showing a very small variation even in case of large data set. The optimal prefix tree thus enables to generate a precise tree even for a large data set. Therefore traversing through the tree is also easier for large data set.

Figure 3 shows the comparison done between the two variable numbers of data set that is considered for experimentation. The time taken for execution is more for large number of dataset

From the entire set of optimal prefix tree a set of test data are extracted randomly. Out of the optimal prefix tree generated a fixed percentage of data are extracted as test data. Figure 4 shows the number of sequential patterns mined for a fixed amount of test data. The comparison is done between the real data set and synthetic data set.

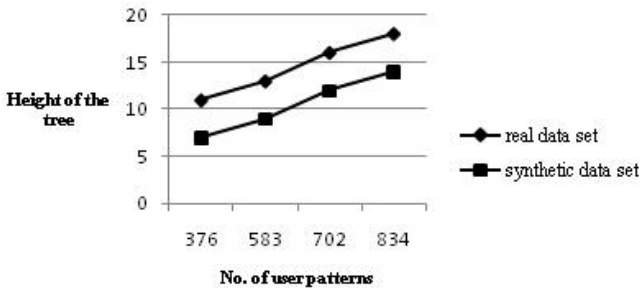

Figure 2. Comparison of height of optimal tree.

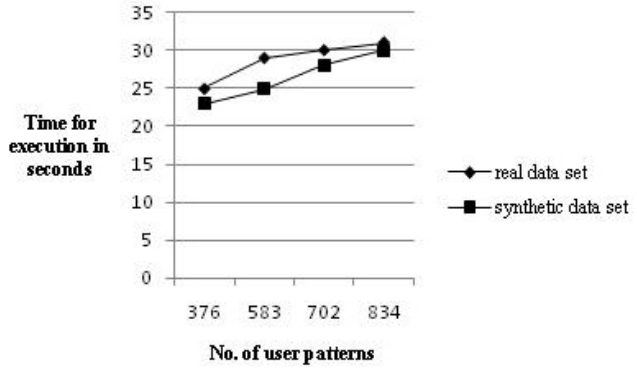

Figure 3. Comparison of the time taken for execution.

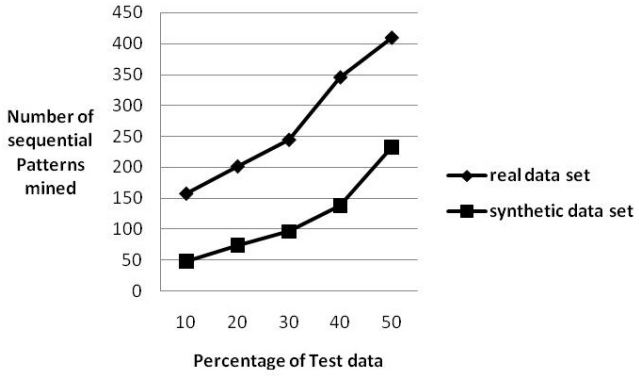

Figure 4. Comparison of number of sequential patterns mined.

5. Conclusion

The work provides the most optimal pattern out of the large number of web traversal patterns that are being generated. It helps the web developer to improvise the web site accordingly.

6. Future Enhancement

This work is limited to sequential data set and static web log data. The dynamic retrieval of log files would enable the system to work dynamically. The extension of this, to consider non sequential data set is under process.

7. References

1. Koh YS, Pears R, Dobbie G. Extrapolation prefix tree for data stream mining using a landmark model, data warehousing and knowledge discovery. Lecture Notes in Computer Science. 2012; 7448:340–51.
2. Qu J-F, Liu M. Mining frequent item sets using node-sets of a prefix-tree, database and expert systems applications. Lecture Notes in Computer Science. 2012; 7446:453–67.
3. Zhong H, Lu Y, Zhang H, Hu R, Zhou C, Zou L. Mining Frequent Induced Subtrees by Prefix-Tree-Projected Pattern Growth. Seventh International Conference on Web-Age Information Management Workshops, WAIM '06; 2006 Jun; Hong Kong, China. IEEE; 2006. p. 18.
4. Wachiramethin J, Werapun J. BPA: A Bitmap-Prefix-tree Array data structure for frequent closed pattern mining. Machine Learning and Cybernetics. 2009; 1:154–60.
5. Chena Y-L, Chena S-S, Ping-Yu Hsub. Mining hybrid sequential patterns and sequential rules. Information Systems. 2002; 27:345–62.
6. Grace JLK, Maheswari V. Utility, importance and frequency dependent algorithm for web path traversal using prefix tree data structure. J Theor Appl Inform Tech. 2014; 60.