International Journal of Scientific Research in Computer Science and Engineering

Research Paper

Volume-1, Issue-1, Jan- Feb-2013

Available online at www.isroset.org

ISSN 2320-7639

Frequent Navigation Pattern Mining from Web usage data

Vaibhav Jain

Institute of Engineering & Technology, Devi Ahilya Vishwavidyalaya Indore, India

Received: 06 Jan 2013 Revised: 17 Jan 2013 Accepted: 10 Feb 2013 Published: 28 Feb 2013

Abstract- Web usage mining provides the information about the user and their behavioural aspects of the web navigation. Traditional frequent sequence pattern mining algorithms are limited in analyzing information from big datasets. However, a graph based approach with the efficient version of apriori algorithm can generate frequent patterns from large datasets. In our work, we have implemented a web graph approach for generating user sessions and apriori all algorithm for generating frequent patterns.

Keywords: Frequent Pattern Mining, Apriori Algorithm, Web Usage Mining, User Session Generation.

I. INTRODUCTION

As an essential data mining task, frequent pattern mining has applications ranging from intrusion detection to credit card fraud prevention and drug discovery. The web usage mining has various applications like link prediction, site reorganization and web personalization. Result of all of these applications depends on the outcome of web usage mining process which includes session construction and frequent navigation pattern discovery phases.

Producing accurate user sessions and navigation patterns is not an easy task since http protocol is stateless and connectionless [2]. Also, in reactive session construction, where it is not possible to generate web log information to identify individual users (like cookies), all users behind a proxy server will have the same IP number and therefore, these users will be seen as a single client on the server side. These problems can be handled by proactive strategies.

The proposed work is motivated to work with the web usage pattern analysis of the end user therefore a number of techniques exist for analyzing the web usage patterns from the web logs. Among them, association pattern analysis, frequent pattern analysis and predictive data modelling are the key techniques. In all these techniques, the web user is targeted for recovering the web usage patterns. But the amount of data during the data analysis streamed is in huge quantity additionally due to increasing usages of web, these logs are growing more frequently. Therefore the traditional computing technique is not suitable for analyzing the growing data more accurately.

Traditional frequent-sequence-pattern-mining algorithms are hard to analyse information from big datasets. We propose an efficient way to predict navigation patterns of web users by improving frequent navigation pattern mining algorithms based on the apriori algorithm which can handle huge datasets efficiently [11]. In order to analyse the user behaviour and their accurate patterns, our work includes development of frequent navigation pattern mining from web usage data using an efficient version of apriori algorithm.

II. RELATED WORK

Bayir et al. [2] proposed a novel framework called Smart Miner for web usage mining problem which uses link information for producing accurate user sessions and frequent navigation patterns. Unlike the simple session concepts in the time and navigation based approaches, where sessions are sequences of web pages requested from the server or viewed in the browser, Smart Miner sessions are set of paths traversed in the web graph that corresponds to user's navigations among web pages [12]. They modeled session construction as a new graph problem and utilized a new algorithm. Smart-SRA to solve this problem efficiently. For the pattern discovery phase, they proposed an efficient version of the apriori-all technique which uses the structure of web graph to increase the performance. From the experiments, they confirmed that smart-miner produces at least 30% more accurate web usage patterns than other approaches including previous session construction methods.

Cooley et al. [9] proposed a work where they collected works of web access information for Web logs. Input data consists of the three server logs- access, referrer, and agent, the HTML files that make up the site, and any optional data such as registration data or remote agent logs. The first part of Web Usage Mining, called pre-processing, includes the domain dependent tasks of data cleaning, user identification, session identification, and path completion. Data cleaning is

the task of removing log entries that are not needed for the mining process. User identification is the process of associating page references, even those with the same IP address, with different users. Session identification takes all of the page references for a given user in a log and breaks them up into user sessions. As with user identification, the site topology is needed in addition to the server logs for this task. Path completion fills in page references that are missing due to browser and proxy server caching. This step differs from the others in that information is being added to the log. The other pre-processing tasks remove data or divide the data up according to users and sessions. Each of these tasks is performed in order to create a user session file which will be used as input to the knowledge discovery phase.

In another work by Cooley et al. [8], the World Wide Web (WWW) continues to grow at an astounding rate in both the sheer volume of traffic and the size and complexity of Web sites. The complexity of tasks such as Web site designs Web server design, and of simply navigating through a Web site has increased along with this growth. An important input to these design tasks is the analysis of how a Web site is being used. Usage analysis includes straightforward statistics, such as page access frequency, as well as more sophisticated forms of analysis, such as finding the common traversal paths through a Web site. *Web Usage Mining* is the application of data mining techniques to usage logs of large Web data repositories in order to produce results that can be used in the design tasks mentioned above [3]. However, there are several preprocessing tasks that must be performed prior to applying data mining algorithms to the data collected from server logs. This paper presents several data preparation techniques in order to identify unique users and user sessions. Also, a method to divide user sessions into semantically meaningful transactions is defined and successfully tested against two other methods. Transactions identified by the proposed methods are used to discover association rules from real world data [13].

III. METHODOLOGY USED

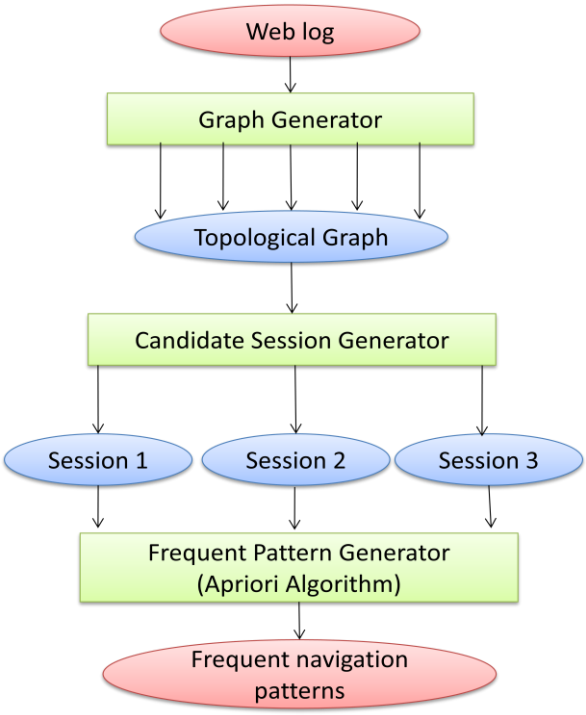

Fig. 1: Proposed Framework for generating frequent patterns

Fig. 1 showed our proposed framework to find frequent navigation pattern from large web log data. We propose a way to predict navigation patterns of web users by frequent navigation pattern mining algorithms based on the programming model and apriori algorithm which can handle huge datasets efficiently. In order to analyse the user behaviour and their accurate pattern the proposed work includes development of frequent navigation pattern mining from web usage. In addition of that how the data is stored, retrieved and processed the following tasks are included in the entire work.

Web Log:

IP Address, User id Time Method/ URL/ Protocol Status Size Referrer Agent
123.456.78.9 - [25/Apr/1998:03:04:41 -0500] "GET A.html HTTP/1.0" 200 3290 - Mozilla/3.04 (Win95, I)
123.456.78.9 - [25/Apr/1998:03:05:34 -0500] "GET B.html HTTP/1.0" 200 2050 A.html Mozilla/3.04 (Win95, I)
123.456.78.9 - [25/Apr/1998:03:05:39 -0500] "GET L.html HTTP/1.0" 200 4130 - Mozilla/3.04 (Win95, I)
……..

Fig. 2: Snapshot of web log used

As shown in Fig. 2, we have a log file contains thousands of log entry of proxy ip form web server [10] .we use this log file to find frequent navigation pattern .we store this log file in data base to show all entry in log file in relational data base form.

Graph generation:

Graph is set of path traversed by user. Our graph contain different path in which user access the pages [12]. These set of pages also known as sessions. Set of session will become our one transaction .After processing the log file we generate set of transaction consist of number of session. These transaction are generated corresponds to particular proxy.

Candidate Session Generation:
We use link information for producing accurate user sessions and frequent navigation patterns where sessions are sequences of web pages requested from the server or viewed in the browser**.** Sessions are set of paths traversed in the web graph that corresponds to users' navigations among web pages [11]. The access data stream of web user are partitioned into shorter page request sequences called candidate sessions, by using session duration time and page-stay time rules after processing the log file we generate set of transaction consist of number of session. These transaction are generated corresponds to particular proxy.

$S = \{S_1, S_2 \ldots\ldots S_n\}$ is a session constructed by algorithm having n paths, such that each $S_x = [P_1, \ldots, P_i, P_{i+1}, \ldots, P_n] \in S$, satisfies the following topological rule conditions:

Topology Rule:
– $\forall_I$: Link(P_i, P_{i+1}) = true

Maximal sub session generation:
Candidate sessions are divided into maximal sub-sessions such that for each consecutive page pair in the sequence there exists a link from previous one to latter one. At the same time, page stay time rule for consecutive pages is also satisfied which is obtained in previous generated candidate sessions. We also add Maximality Rule for generated maximal sessions.

Finding the frequent navigation patterns:
The Apriori algorithm is a classical approach of pattern mining therefore a number of frequent navigation pattern are generated using Apriori algorithm [11]. In this phase algorithm is used that are utilized with proposed pattern extraction technique.

Apriori All algorithm:

After concluding the effective algorithm and techniques of big data processing and association pattern learning a new concept of web log access pattern mining is implemented by Apriori All algorithm. In this algorithm, first task is to find all combinations of items that have transaction support above minimum support. Call those combinations frequent itemsets. Next, use the frequent itemsets to generate the desired rules. The general idea is that if, say, ABCD and AB are frequent itemsets, then we can determine if the rule ABCD holds by computing the ratio r = support(ABCD)/support(AB). The rule holds only if r >= minimum confidence. Note that the rule will have minimum support because ABCD is frequent. The pseudo code is shown below:

```
L1 = large 1-sequences; // Result of itemset phase
for ( k = 2; Lk-1 0; k++) do
begin
        Ck = New Candidates generated from Lk-1
        for each customer-sequence c in the
        database do
           Increment the count of all candidates in
        Ck that are contained in c.
        Lk = Candidates in Ck with minimum
        support.
end
Answer = Maximal Sequences in k Lk ;
```

Fig. 3 : Pseudo code for Apriori All algorithm

Generating Association Rule:
After generation of frequent pattern we generate the association rule from patterns generated for some users and calculate confidence , lift and conviction to check rule satisfy property association rule mining or not. Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction.

IV. EVALUATION PARAMETERS

In order to evaluate the developed system, we used confidence, lift and conviction parameters.

Item set – A collection of one or more items.

Support count (σ) – Frequency of occurrence of an item set –

Support – Fraction of transactions that contain an item set –

Frequent Item set – An item set whose support is greater than or equal to a minsup threshold

Association Rule – An implication expression of the form X → Y, where X and Y are item sets – Example: {Milk, Diaper} → {Beer}

Confidence:

Number of cases in which the rule is correct relative to the number of cases in which it is applicable.

{X} ⇒ {Y} Measures how often items in Y appear in transactions that Contain X.

Lift: The lift of a rule is defined as the ratio of the observed support to that expected if X and Y were independent. Lift is a value that gives us information about the increase in probability of the then (consequent) given the if (antecedent)

part. Lift is simply the ratio of Confidence to Expected Confidence.

Lift (X→ Y) = sup(X U Y) /sup(X) sup(Y)

Conviction:
A high value therefore means that the consequent Y depends strongly on the antecedent X in (X → Y). High Value of conviction indicates that there is more association between items. Lift measures how far from independence are

It ranges within [0 to +∞] Values close to 1 imply that A and C are independent and the rule is not interesting. Values far from 1 indicate that the evidence of X provides information about Y in (X → Y).

Conv(X → Y) = 1 − sup(Y)/ 1 − conf(X → Y)

The larger the lift ratio, the more significant the association so association decrease as confidence value increases. So in this case good association for less support value.

V. RESULTS AND DISCUSSION

Here, we provide experimental results over web server logs of ceng.metu.edu.tr for analyzing results. We measured the accuracies in terms of support, confidence and lift.

No. of items = 10, No. of transactions = 5, Min. confidence = 60%				
Support	**Confidence**	**Lift**	**Conviction**	**Remarks**
0.4	0.66	1.66	1.764	Largest Conviction, Largest Lift
0.6	0.75	1.25	1.6	
0.6	0.6	1.5	1.5	
0.8	0.8	1	1	Lowest Conviction, Lowest Lift

Table 1: Lift and Conviction values obtained for ip address 151.48.123.70

If some rule has a lift of 1 that means the probability of occurrence of the item set and the consequent of the rule are independent of each other. In a clear manner, two events are independent of each other; no rule can be derived to involve these two different events using same rule. The same is performed for the conviction parameters. On the other hand if the lift is > 1 means degree to which two occurrences or transactions' are dependent on one another, and can be combined using the same rules. Additionally the rules are potentially useful for predicting the events. As shown in Table 1, according to the measured values, first three observations produce effective rules as compared to last observation. The remaining middle observations are average.

No of item: 8**, No of Transaction:**7, **Min confidence**=				
Support	**Confidence**	**Lift**	**Conviction**	**Remarks**
0.28	0.66	2.35	2.11	Largest Lift
0.57	0.8	1.41	2.14	Largest
0.71	0.83	1.16	1.68	
0.85	0.857	1	1	Lowest

Table 2: Lift and Conviction values obtained for an ip address 82.83.107.47

As shown in Table 2, according to the obtained performance in terms of confidence all the rules are effective but according to the other two parameters over the four rules only three rules are helpful to provide more coverage of transactions but there is a rule in fourth position is not able to define the effective rule. Here support value 0.57 conviction is largest they are more dependent.

User/IP	**Process time**	**Session generation time**	**Pattern Gen. Time**	**Avg. Time**
151.48.123.70	6172	62	40	6272 ms
82.83.107.47	6172	44	9	6252 ms
200.88.101.168	6172	56	1	6229 ms
77.194.85.234	6170	39	1	6210 ms

Table 3: Average processing time required for generating navigation patterns

Table 3 shows the overall time taken in ms by all the modules for generating frequent patterns according to different users.

VI. CONCLUSION & FUTURE WORK

In this study, we have implemented the generation of session reconstruction heuristics according to a set of data and have applied it to some experimental data, the generated session are processes for session. Then we apply Apriori algorithm to find frequent pattern from maximal session. We get the different frequent pattern according to different Ip or user. The pattern is used further for generating association rule on basis of minimum support, minimum confidence, lift and conviction. If some rule has a lift of 1 that means the probability of occurrence of the item set and the consequent of the rule are independent of each other. In clearer manner two events are independent of each other; no rule can be derived to involve these two different events using same rule. The same is performed for the conviction parameters. On the other hand if the lift is > 1 means degree to which two occurrences or transactions' are dependent on one another, and can be combined using the same rules. Additionally the rules are potentially useful for predicting the events.

As a future work, we are planning to improve Web Analytics Service by intelligent applications that work on frequent navigation patterns .Specifically, we are planning to design a decision support system which will execute user defined filters over frequent access patterns for advertisement or fraud pattern analysis using Hadoop framework. In our implementation data is static and cached in the memory for the all the iterations. When this static data size increases then we can t say for their accuracy, It would be interesting to test the performance of the algorithm with respect to time.. The run-time system takes care of the details of partitioning the input data, scheduling the program's execution across a set of machines. So we implementation of Apriori algorithm on a large cluster of machines using big data and hadoop framework.

REFERENCES

[1] R. Agrawal, R. Srikant, "Mining sequential patterns", In ICDE, USA, pp.–14, 1995.

[2] M. A. Bayir, D. Guney, T. Can, "Integration of topological measures for eliminating non-specific interactions in protein interaction networks", Discrete Applied Mathematics, Vol.157, Issue10, pp.2416–2424, 2008.

[3] J. Borges, M. Levene, "Generating dynamic higher-order markov models in web usage mining", PKDD'05 Proceedings of the 9th European conference on Principles and Practice of Knowledge Discovery in Databases, UK, pp.34-45, 2005.

[4] S. Brohee, J.V. Helden, " Evaluation of clustering algorithms for protein-protein interaction networks", BMC bioinformatics, Vol.7, Issue.1, p.488-496, 2006.

[5] L. D. Cat ledge, J. E. Pitkow. Characterizing browsing strategies in the world-wide web. Computer Networks and ISDN Systems, Vol.27, Issue.6, pp.1065–1073, 1995.

[6] D. Chakrabart, C. Faloutsos, "Graph mining: Laws, generators, and algorithms", ACM Computer Survey, Vol.38, Issue.1, pp.204-212, 2006.

[7] R. Cooley, B. Mobasher, J. Srivastava, "Web mining: information and pattern discovery on the World Wide Web", Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence, Newport Beach, CA, pp. 558-567, 1997.

[8] R. Cooley, B. Mobasher, J. Srivastava, "Data preparation for mining world wide web browsing patterns", Knowl. Inf. Syst., Vol.1, Issue.1, pp.5–32, 1999.

[9] R. Cooley, P.-N. Tan,J. Srivastava, "Discovery of interesting usage patterns from web data", Proceeding Web-based Knowledge Discovery and Data Mining WEBKDD'99, London, pp. 163–182, 1999.

[10] J. Dean, S. Ghemawat. "Map reduce: Simplified data processing on large clusters", In OSDI, USA, pp.137–150, 2004.

[11]J. Han, J. Pei, M. Kamber, "Data mining concept and techniques ", Elsevier, Netherlands, pp.1-744, 2011.

[12]D. Donato, L. Laura, S. Leonardi, S. Millozzi, "The web as a graph: How far we are", ACM Transaction Internet Technology, Vol.7, Issue.1, pp. 128-136, 2007.

[13] B. Mobasher, N. Jain, E. Han, J. Srivastava, "Web Mining: Pattern discovery from World", University of Minnesota, USA, pp.1-12, 1996.

[14]Wide Web transactions. Technical Report TR 96-050, Department of Computer Science, University of Minnesota, Minneapolis, pp.1-25, 1996.

