ISBN 978-83-60810-51-4

Web User Navigation Patterns Discovery from WWW Server Log Files

Paweł Weichbroth
PHD Student
Katowice University of Economics
Katowice, Poland
pawel1739@gmail.com

Mieczysław Owoc
Department of Artificial Intelligence Systems
Wroclaw University of Economics
Wroclaw, Poland
email: mieczyslaw.owoc@ue.wroc.pl

Michał Pleszkun
Institute of Computer Science
Wroclaw University of Technology
Wroclaw, Poland
email: michal.pleszkun@pwr.wroc.pl

Abstract—**Abstract. Continued growth of user number and size of shared content on Web sites cause the necessity of automatic adjusting content to users' needs. In the literature of Web Mining, such actions are referred to personalization and recommendation which led to improve the visibility of presented content. To perform adequacy actions which correspond to the expected users' needs we can utilize web server log files. Mining such data with accurate constraints can lead to the discovery of web user navigation patterns. Such knowledge is used by personalization and recommendation systems (PRS) due to performed actions against user behavior during a visit on the web portal. In these paper we present the system framework for mining web user navigation patterns in order to knowledge management. We focus on constraints which are critical factors to evaluate the effectiveness of the implemented algorithm. On the other hand, these constraints can be perceived as knowledge validation criteria due to its adequacy. Thus only adequate knowledge can be added to existing in PRS knowledge base.**

I. Introduction

EVOLUTION of designing and developing Web sites from static to dynamic approach has let them update easily. Furthermore, intensive development and proliferation of WWW network resulted in other new modeling methods. It's obvious—being recognized and visited in the Web means that the content is up-to-date and satisfies its visitors. Widespread scope of content topics shared and presented on the Web site affects on the size and depth level of its structures. This results in negative impression of presented content and weaker usability.

These problems can be solved by recommendation and personalization. Those two definitions penetrate themselves and often are function interchangeable. However recommendation is narrow image of personalization and defined as suggestion—pointing specific information objects to the user, demanding from him some interactions. Additionally, personalization is associated with unique user identification (for example using authentication mechanism) and due to tune and tweak structure and appearance Web site to his preferences. Generally, these two definitions can be seen as taking any actions adapting objects' presentation on the Web site to the user needs. The consequence of conducting such actions may be higher user retention calculated by the number of opened pages and total session duration. In this paper, we present a system framework for mining web user navigation patterns from web server log files. For the implemented in C# data mining algorithm, we defined nine constraints: minimum single duration access, minimum session time, minimum support, minimum confidence, minimum rule length, include item, exclude item, date and time.

The remaining of this paper is organized as follows: in section 2, we present related work which concern web mining algorithms and related software tools. In section 3 we discuss web usage mining process and its related tasks. Our developed framework and implemented algorithm are presented in section 4. In this section we reported results as well. The paper wraps up with conclusions.

II. Related Work

Generally, Web mining is the application of data mining algorithms and techniques to large Web data repositories [1]. Web usage mining refers to the automatic discovery and analysis of generalized patterns which describe user navigation paths (e.g. clickstreams), collected or generated as a result of user interactions with Web site [2]. Studies related to our work are concerned with two areas: constraint-based data mining algorithms applied in Web Usage Mining and developed software tools (systems). One of the most common algorithm applied in Web Usage Mining is the Apriori algorithm proposed in [3]. Web user navigation patterns were represented by association rules in [4-8]. Sequence mining can be also used to mine Web user navigation patterns. Compare to association rules, such knowledge holds additional information—it put forward the sequence of requested pages (e.g. if user visits page A, and then page C, it will visit page D). Based on this, users activity can be determined and predictions to the next page can be calculated. In [4] Chen et al. presented algorithm to mine maximal forward chain of web pages.

The sequence mining algorithms inherited much from association mining algorithms. Many of them are extensions of the firsts. The major difference is that in association mining the discovered patterns are intra-sequence, where in the sequence mining the inter-sequence patterns are discovered (e.g. if the gold market's price index increases more than the exchange rate of the EURO in the first week, the real estate market's price index will probably increase more than that of the oil market in the second week). In [9] we can see sequence

978-83-60810-51-4/$25.00 © 2012 IEEE

patterns mining over a data warehouse where an adoption of the SPADE algorithm can be observed. In [10] was proposed algorithm called EISP-Miner (Enhanced Inter-Sequence Pattern Miner) which is able to discover a complete set of frequent inter-sequence patterns efficiently. Instead of using Apriori-like level-wise generation and checking of frequent patterns, the search method is partition-based and divide-and-conquer approach. These features guarantee higher efficiency, evaluated on synthetic dataset and a real dataset, measured by the run time versus the number of transactions and maxspan (maxspan is user-specified maximum span threshold), memory usage vs. minimum support and number of transactions. A Web Utilization Miner (WUM) [11] is a mining system for a discovery of navigation patterns. It can assist in obtaining this kind of knowledge, defined as directed graph which summarizes the traversal movements of a group of users and satisfies certain human-centric criteria that make it interesting. There are more than a few commercially available Web server log analysis tools [12-14]. A detailed overview, which is out of the scope of this work, can be found e.g. [15, 16].

III. WEB USAGE MINING

The objective of the knowledge discovery from databases (KDD) process is to extract new, interesting and useful knowledge [17-20] using a variety of data mining methods and techniques such as [21]: description (e.g. clustering, automatic text summarization, cognitive linguistic and visualization) and prediction (e.g. regression, classication—for example: association rule mining and sequential pattern discovery). In the case that the data origin is the Web, the process is called Web mining instead of KDD.

Web mining concerns a varied range of applications that aims at discovering, evaluating and employing hidden knowledge from Web data sources. It can be roughly categorized into three different groups based on which part of the Web is to be explored [22-24]. These three categories are:

1) Web content mining (WCM), involves mining, extraction and integration of useful data in the content of web pages, e.g. structured text data (plain text content), semi-structured data (HTML code), pictures and downloadable files [25]. WCM is the process of extracting knowledge from the content of documents or their descriptions [26].
2) Web structure mining (WSM), focuses on the inner-document structure which means discovering the link structure of the hyperlinks at the inter-document level [25]. In other words, WSM is the process of inferring knowledge from the World-Wide Web organizations and links between references and referents in the Web [26, 27].
3) Web usage mining (WUM) (or Web log mining [27]), operates on the data from server access logs, information from users' registration application forms, users profiles (e.g. bookmarks or folders [25]) and transactions [23]. To put it simply, WUM (Figure 1) is the process is the process of extracting interesting patterns in Web access logs [26].

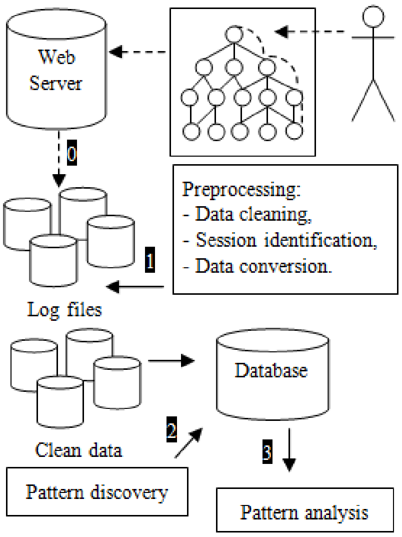


Fig. 1. A logical view on web usage mining

Web usage mining consists of three main tasks (Figure 1) [28, 29]:

1) Preprocessing, contains three separate phases: (a) cleaning which means that useless entries (e.g. graphic and multimedia objects) have to be removed, (b) session identification by assign all requests from one user to one unique session and (c) data conversion into the format specific for the software tool.
2) Pattern discovery, means applying the presented algorithm with defined constraints by the user to the data.
3) Pattern analysis is a human domain task and means understanding the results obtained by the algorithm and drawing conclusions.

It can be noticed on Figure 1, we add one prior task to the web usage process, similarly to [30], which concerns data collection denoted as (0). These data represents users navigation behavior, recorded in log files attached to Web server. We think that these four steps embrace all actions which take place. Moreover, we depict precisely the origin of the input data.

Nowadays, WUM has five main application areas [25]:

1) Personalization is the ability to tailor content and recommend objects. It implies that PRS system must be able to anticipate users' needs and provide them with objects which they might appreciate based on previous interactions of other users and interactions with current user. Therefore, the personalization task can be viewed as a prediction problem: the system attempts to predict the user's interest in specific content [26]. Recommendation and personalization techniques are classified into three different categories [31]: rule-based filtering, content-based filtering and collaborative

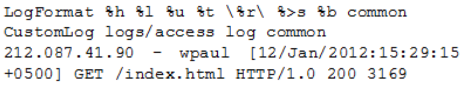
filtering. In this work, presented system delivers data only for the first technique.

2) System improvements concern of analyzing collected web data due to provide understanding web traffic behavior. Such improvements may bring in advanced load balancing, data distribution or polices for web caching and higher security [25].
3) Modification of web site—based on discovered web user navigation patterns—will be possible which means internal links rearrangement due to improve their visibility.
4) Business intelligence, noticeable in e-commerce activities like email marketing campaigns, cross- and upselling techniques developed and observed in e-markets (e.g. amazon.com, merlin.pl).
5) Characterization of use, web server log files combine with additional information (e.g. source IP address decoded to demographic location like country and city of the web site visitor) can deliver further knowledge of the users.

A. Data format and preprocessing

Clickstreams are hits collection from specific and unique user sessions. Assuming that user sessions in web logs are registered and constructed by some sort technology (e.g. Apache server or Information Internet Services), we first present format of the collected web log data. Then we focus on preprocessing task, necessary for web user navigation patterns mining.

A Web server log files usually record a full history of requested access to files by users, shared on WWW server. The majority of http servers [32, 33] uses Common Log Format (CLF) standard defined by CERN and NCSA as part of http protocol [34]. An ordinary log entry has the following format as shown in Figure 2.



```
LogFormat %h %l %u %t \%r\ %>s %b common
CustomLog logs/access log common
212.087.41.90  -  wpaul  [12/Jan/2012:15:29:15
+0500] GET /index.html HTTP/1.0 200 3169
```

Fig. 2. CLF format and imitation of sample entry

According to this standard, a log entry contains:

- %h (212.087.41.90), client IP address,
- %l (-), remote login name of the user,
- %u (wpaul), authenticated user name
- %t (12/Jan/2012:15:29:15 +0500), access time and time zone,
- \%r\(GET /index.html HTTP/1.0), request method, requested URL page name and the transmission protocol,
- %¿s (200), an error code,
- %b(3169), the number of bytes transmitted.

Data stored in databases can suffer from various kinds of errors due to manual data input by human. Web server log data is generated automatically and therefore we can eliminate these kinds of errors. However, there are other reasons for data preprocessing e.g. configuration and implementation errors, de-spidering or server down times. These sources of data aggravation must be investigated. If they occur, appropriate measures for data quality improvement have to be undertaken.

In order to conduct reliable research, we have to identify unique users and user sessions. Just then, such transformed data can be used as input for mining algorithm. Several sessionization heuristics have been proposed e.g. agent-based, time-based, access-log in conjunction with the referrer log and site topology [25, 35]. Our solution depends on the mechanism which generates a unique token to each new visitor. A token is generated by the server and is sent back during every page request which comes from the client (user's Web browser).

B. Pattern discovery

The objective of mining process is to discover sequential association rules. This knowledge will form the knowledge base which can be used in recommendation and personalization systems. Detailed description is given in section 4.

IV. THE SYSTEM FRAMEWORK

Developed framework (Figure 3) is divided into six main components:

- Database—stores preprocessed data from log files and results from mining process,
- Data Access Service (DAS)—a mapping of database structure into a program classes,
- Controller—implementation of framework main logic. It controls all data flow and main functionalities
- User Interface—interface for human-computer interactions along with data,
- Algorithm class—source code of Apriori-like algorithm,
- File Controller—it consists of two functions. First one is to parse large log files to database and the second is to save a output of algorithm results.

Framework has been implemented in .Net 4.0 and uses Microsoft SQL Server 2008. As a connection scheme for this kind of data mining problems we have choose a fully connection mode which means that whole datasets isn't kept in operational memory, but it's loaded on demand. It is necessary because of data size that usually are operated by this framework.

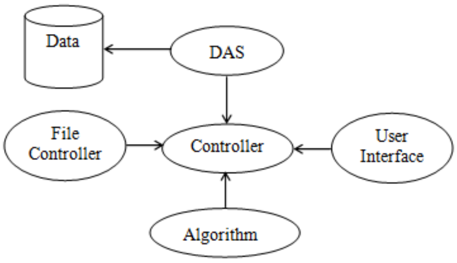


Fig. 3. System framework components

A. The algorithm

Designing an efficient algorithm is crucial for mining sequential association rules. Our approach is based on Apriori algorithm and has the following anti-monotone Apriori property: if any length k frequent itemset is not frequent in the database, then its length k+1 super-frequent itemset cannot be frequent [36].

Structures implemented in framework are based on a hierarchy of logs. The root of the hierarchy is collection of sessions. Each session is represented by a structure SessionStructure, which contains a collection of LogStructure which represent a one entry (html page request from the client) and its attributes.

Algorithm has been divided to four procedures where each represents one constraint due to data pruning before execution of the Apriori algorithm. The removal reason is optimization of time which is needed to discover strong association rules. In listing below is an entry point for data processing

```
function Main()
{
    read D for every SessionId;
    for each dItem in D do
    {
        sort dItem.d by dItem.d.time;
        if(checkBeforeApriori(dItem)) then
        {
            delete dItem from D;
        }
    }
    D = Apriori(D);
    checkPostApriori(ref D);
    Return D;
}
```

Listening above shows that main function has data preprocessing where we can have some data pruning before execution of Apriori algorithm as well as patterns purification after we have some results from the algorithm.

```
procedure checkBeforeApriori(SessionStructure S)
{
    if(checkMinimumPageDuration(S)) then
    {
        return true;
    }
    else if(checkMinSessionDuration(S)) then
        {
            return true;
        }
    else
    {
        return false;
    }
}
```

`CheckBeforeApriori` is procedure where all the constraints which should be checked before actual algorithm are called. If in future new constraints will arrive it has to be implemented in new procedure and this procedure should be called from `CheckBeforeApriori` procedure.

```
procedure checkMinPageDuration(SessionStructure S)
{
    for(i=0, i<S.d.Count-1, i++)
    {
        S.d[i].duration = S.d[i+1].timeStart - S.d
        [i].timeStart;
        if(S.d[i].duration < min_dur) then
        {
            return true;
        }
    }
    return false;
}
```

First constraint is checked in `checkMinPageDuration` procedure. It's minimum single duration access constraint. This procedure shows how the new constraints should be implemented. Constraints procedure have to return true, only if we want to delete checked session from data we are processing.

```
procedure checkMinSessionDuration(SessionStructure S)
{
    S.startTime = S.d[0].time;
    S.endTime =S.d[dItem.d.Count-1].time
    if((S.endTime - S.startTime) < min_sess) then
    {
        return true;
    }
    return false;
}
```

Next constraint (minimum session time) is checked in `checkMinSessionDuration` procedure.

```
procedure checkPostApriori(ref SessionStructure S)
{
    for each dItem in D do
    {
        if(dItem.d.Count < min_leng) then
        {
            delete dItem;
        }
    }
}
```

The last constraint is checked after the Apriori algorithm has ended finding rules. This constraint concerns minimum rule length, that's why it have to be check here.

B. Results

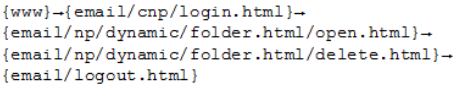
We conducted our research on onet.pl server log files which covers five hours of users' usage activity (from 2PM to 7PM, 29th December 2008). There are over 2,13 million sessions and log file size is 512 MB. After preprocessing, clean data was imported to the database and analyzed. The mining process aims to find frequent itemsets and discover web user navigation patterns. In the first survey we set up two constraints: minimum support on 0,01 and minimum confidence on 0,5. Below, we present items with the highest support (given in brackets):

- www (80.36
- email/cnp/login.html.php3 (27.53%),
- email/np/dynamic/folder.html (26.29%),
- email/np/dynamic/folder.html/open.html (15.43%),
- sport/volleyball/news.html (13.17%),
- info/world/item.html (12.70%),
- info/country/item.html (12.68%),
- email/np/dynamic/folder.html/delete.html (11.96%),
- sport/ski jumps/news.html (10.01%).

The results from the first study, in 95% demonstrated the relationship between the homepage and subpages available directly from it and the inner relationships within the autonomous services: email and sympathy. Therefore, most of discovered patterns, proved not to bring relevant knowledge about user navigation paths. These conclusions imply to define additional constraint. In this case, we conducted second survey in which three sets www, sympathy and mail were excluded and two primary constraints were left on the same level. This third constraint led to reduce the number of sessions to 1.4 million and file size to 105 MB. Below, we present items with the highest support:

- info]/world/item.html (29.03%),
- sport/football/first league/news.html (14.63%),
- sport/formula one/news.html (14.14%),
- info]/country/item.html (13.14%),
- business/stock market/news.html (11.24%),
- info/cnn/item cnn.html (9.02%),
- business/news.html (7.00%),
- business]/stock market.html (6.93%),
- business]/pap.html (5.58%),
- sport]/football/uefa championships/news.html (5.25%),
- info]/science/item.html (4.58%).

This operation allowed to eliminate temporal web user navigation paths that were classified as hot news on the home page of the portal. In addition, there have been detected dependencies within the email and sympathy service, which appear to be not very useful (e.g. *user opens home page then select email service, provides login name and password in authentication form, then checks inbox, deletes some unwanted emails and logs out)*. Figure 4 shows some of these obvious and useless patterns from the first study.



```
{www}→{email/cnp/login.html}→
{email/np/dynamic/folder.html/open.html}→
{email/np/dynamic/folder.html/delete.html}→
{email/logout.html}
```

Fig. 4. A spam web user navigation pattern

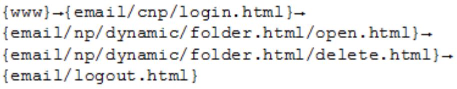
Comparing these two survey (Table 1), in the first 64 one-element frequent itemsets were found where 18 of them were excluded in the second study. The content of these pages is constantly updated and the file names remain unchanged. It is worth mentioning, that this conclusion concerns to permanent paths—in the manner that they are independent of uploading new subpages. For example, a static name-space of some pages applies to sets like info/domestic/item.html, business/pap.html or business]/ market.html, whose content is updated regularly. Defining additional constraint led to a significant decrease in the number of association rules. However, discovered knowledge seems to be more useful (Figure 5). It can be applied to predict web users navigation paths. It means that content of a web portal can be automatically adopted to potential user preferences by the PRS system (e.g. *2.4% of users open in UEFA Championship page, then select news from formula one and select news from extra football league*).

Nonetheless, results from the second survey are not in the opposite to previous results and ultimately confirm them. The valuable patterns for PRS systems laid in the bottom and just had to uncovered. Our approach assumes to identify irrelevant sets and remove them from the data. On the other hand, it is possible to lower the minimum support and confidence level. Although the run time of mining process will increase dramatically. This justifies our hybrid and iterative method of web usage mining.

TABLE I
NUMBER OF FREQUENT ITEMSETS AND ASSOCIATION RULES IN FIRST AND SECOND SURVEY (MIN. SUPPORT = 0,01/ MIN. CONFIDENCE = 0,5).

	Survey			
	1		2	
Items	◊	□	◊	□
1	64	-	98	-
2	167	142	74	24
3	177	430	29	34
4	418	196	1	4
5	40	407	0	0
6	14	146	0	0

◊ frequent itemsets □ association rules



```
{www}→{email/cnp/login.html}→
{email/np/dynamic/folder.html/open.html}→
{email/np/dynamic/folder.html/delete.html}→
{email/logout.html}
```

Fig. 5. A sample of useful web user navigation pattern

V. CONCLUSION

Discovering patterns from log files seems to be interesting and a promising way of generation new knowledge bases. As we stressed it can be valuable source for personalization and recommendation systems (PRS) but one of the crucial aspect is quality of generated knowledge. Six main components are essential in the elaborated system framework: database, data access service, controller, algorithm, file controller and user interface.

Main findings from the research can be formulated in the following manner:

- proposed framework of the system (in our opinion) is adequate for fulfill basic goals defined for the discussed systems,
- implemented A-priori algorithm allows for efficient data processing of log files including its hierarchical format,
- applied in the system procedures (parts of the elaborated algorithm) are responsible for the particular constraints included to the system and covering essential aspects of generated knowledge bases,
- results achieved in the research are promising and confirming usability of the implemented method (especially including expert roles in monitoring and refinement of generated knowledgebase).

Further investigations can be devoted to extensions of procedural aspects of the presented framework. For example specification of constraints can be reformulated and also measurement aspects can be elaborated in order to offer more holistic approach to the web user navigation patterns discovery.

References

[1] B. Mobasher, N. Jain, E.-H.S. Han and J. Srivastava, Web Mining: Pattern Discovery from World Wide Web Transactions, (1996).

[2] B. Mobasher and O. Nasraoui, Web Usage Mining, Web Data Mining. Exploring Hyperlinks, Contents, and Usage Data, B. Liu, (Ed.), Springer-Verlag, (2011).

[3] R. Agrawal, T. Imielinski and A. Swami, Mining Association Rules between Sets of Items in Large Databases Proc. SIGMOD 93: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, ACM, 207-216, (1993).

[4] M.-S. Chen, J.S. Park and P.S. Yu, Data mining for path traversal patterns in a web environment, Proc. Distributed Computing Systems, 1996., Proceedings of the 16th International Conference on Distributed Computing Systems, 385-392, (1996).

[5] P. Batista, J. Silva and C. Grande, Mining Web Access Logs of an On-line Newspaper, (2002).

[6] O.R. Zaiane, M. Xin and J. Han, Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs, IEEE Computer Society, 19, (1998).

[7] L. Shen, L. Cheng, J. Ford, F. Makedon, V. Megalooikonomou and T. Steinberg, Mining the Most Interesting Web Access Associations, Proc. WebNet 2000—World Conference on the WWW and Internet, AACE—Association for the Advancement of Computing in Education, 489-494, (1999).

[8] J.R. Punin, M.S. Krishnamoorthy and M.J. Zaki, Web Usage Mining—Languages and Algorithms, Proc. Studies in Classification, Data Analysis, and Knowledge Organization, Springer-Verlag, 88-112, (2001).

[9] A. Demiriz, WebSPADE: a parallel sequence mining algorithm to analyze web log data, Proc. ICDM 2003. IEEE International Conference on Data Mining, IEEE Computer Society, 755—758, (2002).

[10] C.-S. Wang and A.J.T. Lee, Mining inter-sequence patterns, Expert Systems with Applications: An International Journal, vol. 36, no. 4, 8649-8658, (2009).

[11] M. Spiliopoulou and L.C. Faulstich, WUM: A Tool for Web Utilization Analysis, The World Wide Web and Databases. Lecture Notes in Computer Science, vol. 1590/1999, 184-203, (1999).

[12] Webtrends, Products and Services, (2012); http://webtrends. com.

[13] Weblogexpert, Weblogexpert. Information, (2012); http://www. weblog-expert.com/.

[14] AlterWind Log, AlterWind Log Analyzer Professional, (2012); www.alterwind.com/loganalyzer/.

[15] D. Pierrakos, G. Paliouras, C. Papatheodorou and C.D. Spyropoulos, Web Usage Mining as a Tool for Personalization: A Survey, User Modeling and User-Adapted Interaction, vol. 13, no. 4, 311-372 (2003).

[16] A.H.F. Laender, B.A. Ribeiro-Neto, A.S.d. Silva and J.S. Teixeira, A brief survey of web data extraction tools, SIGMOD Record, vol. 31, no. 2, 84-93, (2002).

[17] U. Fayyad, G. Piatetsky-Shapiro, P. Smith and R. Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, (1996).

[18] U. Fayyad, G. Piatetsky-Shapiro and P. Smyth, From Data Mining to Knowledge Discovery in Databases, AI Magazine, vol. 17, no. 3, 37-54, (1996).

[19] W.J. Flawley, G. Piatetsky-Shapiro and C.J. Matheus, Knowledge Discovery in Databases: An Overview, Knowledge Discovery in Databases, G. Piatetsky-Shapiro and W. J. Flawley (Ed.), AAAI/MIT Press, 1-30 (1991).

[20] G. Piatetsky-Shapiro and W. Frawley, Knowledge Discovery in Databases, MIT Press, (1991).

[21] O. Maimon and L. Rokach, Introduction to Knowledge Discovery and Data Mining, Data Mining and Knowledge Discovery Handbook. Second Edition, O. Maimon and L. Rokach (Eds.), Springer, 1-15, (2010).

[22] R. Kosala and H. Blockel, Web mining research: A survey, Newsletter of the Special Interest Group (SIG) on Knowledge Discovery and Data Mininig SIGKDD: GKDD Explorations, vol. 1, (2000).

[23] S. Madria, S.S. Bhowmick, W.-k. Ng and E.P. Lim, Research Issues in Web Data Mining, Data Warehousing and Knowledge Discovery, 303-312, (1999).

[24] J. Borges and M. Levene, Data Mining of User Navigation Patterns, Book Data Mining of User Navigation Patterns, Series Data Mining of User Navigation Patterns, Springer-Verlag, 92-111, (2000).

[25] S. Araya, M. Silva and R. Weber, A methodology for web usage mining and its application to target group identification, Fuzzy Sets and Systems, vol. 148, no. 1, (2004), pp. 139—152.

[26] C. Romero, S. Ventura, A. Zafra and P.d. Bra, Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems, Computers & Education, vol. 53, no. 3, 828—840, (2009).

[27] J. Sivaramakrishnan and V. Balakrishnan, Web Mining Functions in an Academic Search Application, Informatica Economica, vol. 13, no. 3, 132-139, (2009).

[28] R. Ivancsy and I. Vajk, Frequent pattern mining in web log data, Acta Polytechnica Hungarica, vol. 3, no. 1, 77-90 (2006).

[29] J. Srivastava, R. Cooley, M. Deshpande and P.N. Tan, Web usage mining: discovery and applications of usage patterns from web data, ACM SIGKDD Explorations Newsletter, vol. 1, no. 2, (2000).

[30] V. Chitraa and A.S. Davamani, A Survey on Preprocessing Methods for Web Usage Data, International Journal of Computer Science and Information Security, vol. 7, no. 3, 78-83, (2010).

[31] M.J. Pazzani and D. Billsus, Content-Based Recommendation Systems, The Adaptive Web. Lecture Notes in Computer Science, vol. 4321/2007, 325-341, (2007).

[32] R. Pamnani and P. Chawan, Web Usage Mining: A Research Area in Web Mining, (2010).

[33] G.S. Prasad, N.V.S. Reddy and U.D. Acharya, Knowledge Discovery from Web Usage Data: A Survey of Web Usage Pre-processing Techniques, Information Processing and Management. Communications in Computer and Information Science, vol. 70, 505-507, (2010).

[34] W3C, Logging Control in W3C httpd, (1995); http://www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format.

[35] R. Cooley, B. Mobasher and J. Srivastava, Data Preparation for Mining World Wide Web Browsing Patterns, Knowledge and Information Systems, vol. 1, 5-32, (1999).

[36] R. Agrawal and R. Srikant, Fast algorithms for mining association rules, Proc. Proceedings of the Twentieth International Conference on Very Large Data Bases, Morgan Kaufmann