An application for clickstream analysis

C. E. Dinucă

Abstract**—** In the Internet age there are stored enormous amounts of data daily. Nowadays, using data mining techniques to extract knowledge from web log files has became a necessity. The behavior of Internet users can be found in the log files stored on Internet servers. Web log analysis can improve business firms that are based on a Web site through learning user behavior and applying this knowledge to target them for example to pages that other users with similar behavior have visited. The extraction of useful information from these data has proved to be very useful for optimizing Web sites and promotional campaigns for marketing, etc. In this paper I will focus on finding associations as a data mining technique to extract potentially useful knowledge from web usage data. I implemented in Java programming language, using NetBeans IDE, a program for identification of pages' association from sessions. For exemplification, I used the log files from a commercial web site.

Keywords**—**Apriori algorithm, Association rules, Clickstream analysis, Sessions' identification, Web server logs, Web usage mining.

I. Introduction

WEB mining is an area that lately has gained a lot of interested. This is due essentially to the exponential growth of the World Wide Web and its anarchic architecture and also due to the increase of its importance over the people's life. A Web site is a lot of interconnected web pages that are developed and maintained by a person or organization. Web mining studies analyzes and reveals useful information from the Web [11]. Web mining deals with the data related to the Web, they may be the data actually present in Web pages or the data concerning the Web activities. The Web can be viewed as the largest unstructured data source available, although the data on the Web sites, which composed them, is structured. This presents a challenging task for effective design of and access to Web pages. Web mining is a term used for applying data mining techniques to Web access logs [12]. Data mining is a non-trivial process of extracting previously unknown and potentially useful knowledge from large databases [5].

Web mining is an area that lately has gained a lot of interested. This is due essentially to the exponential growth of the World Wide Web and its anarchic architecture and also due to the increase of its importance over the people's life. Scientists and engineers want to extract information from it, in order to better understand and to improve its features. They applied data mining techniques on the web. Therefore, Web mining can be defined as the application of Data Mining techniques to the web related data.

Web mining can be divided into three categories: Web content mining, Web structure mining and Web usage mining [10]. Web content mining is the process of extracting knowledge from documents and content description. Web structure mining is the process of obtaining knowledge from the organization of the Web and the links between Web pages.

Web usage mining analyzes information about website pages that were visited which are saved in the log files of Internet servers to discover the previously unknown and potentially interesting patterns useful in the future. Web usage mining is described as applying data mining techniques on Web access logs to optimize web site for users.

Click-stream means a sequence of Web pages viewed by a user; pages are displayed one by one on a row at a time. Analysis of clicks is the process of extracting knowledge from web logs. This analysis involves first the step of data preprocessing and then applying data mining techniques. Data preprocessing involves data extraction, cleaning and filtration followed by identification of their sessions.

Due to the immense volume of Internet usage and web browsing in recent years, log files generated by web servers contain enormous amounts of web usage data that is potentially valuable for understanding the behavior of website visitors.

This knowledge can be applied in various ways, such as enhancing the way that the web pages are interconnected or for increasing the sales of the commercial web sites.

II. Data Preprocessing

Log files are created by web servers and filled with information about user requests on a particular Web site. They may contain information about: domains, sub domains and host names; resources requested by the user, time of request, protocol used, errors returned by the server, the page size for successful requests.

Because a successful analysis is based on accurate information and quality data, preprocessing plays an important role. Preparation of data requires between 60 and 90% of the time from data analysis and contributes to the success rate of 75-90% to the entire process of extracting knowledge [3].

For each IP or DNS we determine user sessions. The log files have entries like these:

95.175.194.33 - - [27/Jul/2011:07:23:04 -0500] "GET /css/preview_style.css HTTP/1.1" 200 2553 "http://www.nice-layouts.com/preview.php?p=34062" "Mozilla/5.0 (Windows;

Manuscript received November 28, 2011: Revised version received November 28, 2011.

C. E. Dinuca is PhD Student at the University of Craiova, Craiova, 200585, Romania (e-mail: clauely4u@yahoo.com).

U; Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"
95.175.194.33 - - [27/Jul/2011:07:23:04 -0500] "GET /css/tabright.gif HTTP/1.1" 200 2095 "http://www.nice-layouts.com/css/preview_style.css" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"
95.175.194.33 - - [27/Jul/2011:07:23:04 -0500] "GET /css/tableft.gif HTTP/1.1" 200 377 "http://www.nice-layouts.com/css/preview_style.css" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"
95.175.194.33 - - [27/Jul/2011:07:23:05 -0500] "GET /secure/none.gif HTTP/1.1" 200 827 "https://www.nice-layouts.com/secure/custom_css.css" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"

As can be noticed above, each record in the file contain the IP, date and time, protocol, page views, error code, number of bytes transferred. The steps needed for data preprocessing were presented in detail in [1]. For sessions' identification in the first case was considered that a user can not be stationed on a page more than 30 minutes. This value is used in several previous studies, as can be seen in the work [2]. The current study intends to add an improvement in sessions' identification by determining an average time of page visiting the sites for the visit duration determined by analysis of web site visit duration, data which can be found in the log files of the site. Thus, for each visited page, is calculated the visit duration, which is determined by the difference between two consecutive timestamps for the same user, which is identified by IP. For records of pages with the highest timestamp among those visited by a user is assigned a predefined value of our choice to 20,000 seconds. I calculate the average visit time for a page by the average of all the times spent on that page. When calculating the average visiting time we don't take into consideration the pages with the time less than 2 seconds and largest than 20,000 seconds. Thus for our analysis I selected only those log records that contained a web page, eliminating the required load images and other files adjacent to it, this information being considered not important for analysis. I kept only pages that have status code of class 200, a successfully loaded page. Thus, we calculated how long a user stayed on a page as the difference between consecutive timestamps of visited pages for the same person, same IP. I calculated the average visiting time for a page as the media of time spent for different users on that page and used this mean to better identify sessions. I have removed pages of double sessions and I just kept for review sessions with more than 1 page views.

After preprocessing stage we obtained a file containing the user sessions. I implemeted in Java the Apriori algoritm in order to obtain the association rules between the pages from the sessions. I applied this algorithm on the sessions obtained.

III. Associations Mining

Items that occur often together can be associated to each other. These together occuring items form a **frequent itemset**. Conclusions based on the frequent itemsets form **association rules.** For ex. {milk, cocoa powder} can bring a rule *cocoa powder* ➔ *milk.*

Consider we have database D consists of events T_1, T_2,... T_m, that is $D = \{T_1, T_2, ..., T_m\}$. Let there be an itemset X that is a subregion of event T_k, that is $X \subseteq T_k$.

The support can be defined as :

$$\sup(X) = \frac{\left|\{T_k \in D \mid X \subseteq T_k\}\right|}{\left|D\right|}$$

this relation compares number of events containing itemset X to number of all events in database.

Any frequent item set (support is higher than the minimal support): I frequent , $\sup(I) \geq \sup_{\min}$.

Properties of the Support of an Item Set are:

- No superset of an infrequent item set can be frequent, the well known Apriori property.
- All subsets of a frequent item set are frequent.

Algoritmul Apriori

Apriori algorithm defined in 1994 by Agrawal and Srikant is the benchmark among unsupervised learning system based on association rules. Apriori algorithm is the first and most important efficient algorithm for discovering association rules.

The general scheme of the Apriori algorithm after Borgelt[8]:

- Determine the support of the one element item sets and discard the infrequent items.
- Form candidate item sets with two items (both items must be frequent), determine their support, and discard the infrequent item sets.
- Form candidate item sets with three items (all pairs must be frequent), determine their support, and discard the infrequent item sets.
- Continue by forming candidate item sets with four, five etc. items until no candidate item set is frequent.

It is based on two main steps: candidate generation and pruning. All frequent item set mining algorithms are based on these steps in some form.

Apriori uses a scroll in depth strategy to compute the support sets of elements and uses a function to generate candidates that uses circumscribed lower of support property.

If we consider the time for the selected elements, then we have a sequential association.

In the case of clickstream analysis we can appply both of them. So, we can determine association with pages, sequential association, associations rules and sequential rules in order to determine navigation paths models from the log files.

I implemented an application for applying data mining algorithms on log files in order to extract intersting knowleadges from data.

IV. Case Study

To implement the algorithm presented earlier and the entire models extraction application I used the Java programming language, the code is written using NetBeans IDE.

For implementation I created the NetBeans project ClickstreamAnalysis and it's componenets can be seen in the following image :

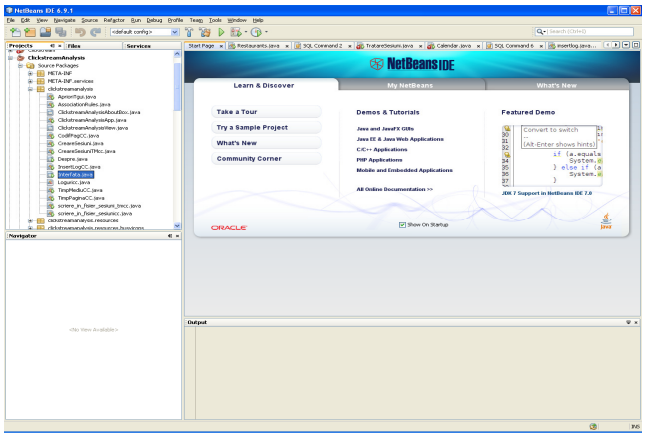


Fig. 1. The ClickstreamAnalysis project into NetBeans IDE

The application contains all the preprocessing steps needed to obtain the data in a form necessary for using it as input to algorithms. Using this developed application we can do the following operations. First we can read the data from web log file, clean the data and insert them in a database table in order to be able to perform the next preprocessing steps. Then we execute the following operations: the pages are codified, we calculate the average time for every page, we calculate the sessions with the method of in which we consider a user can't stay longer than 30 minutes on a page and then we also create the sessions using the method proposed by us with the average time for each sessions.

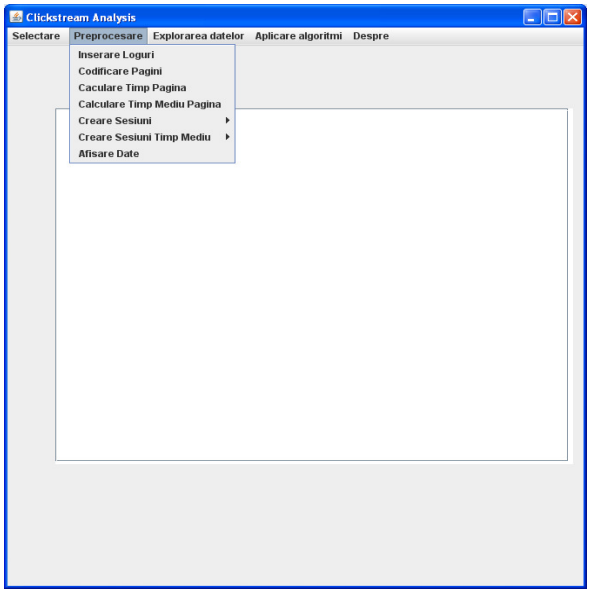


Fig. 2. The main window with the preprocess menu

After each preprocessing step the data can be seen in a window, as the one from figure 3, so the analyst is able to monitor and modify the data at each moment.

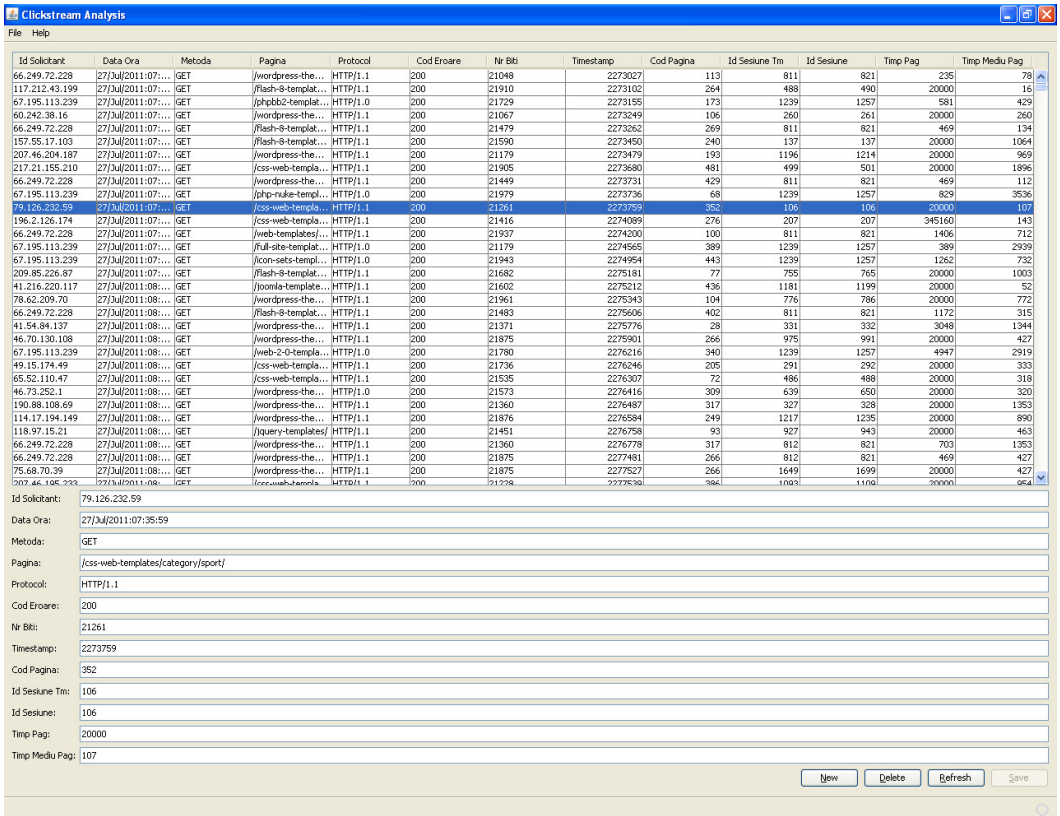


Fig. 3. The window for monitoring all the preprocessing steps.

The analyst can delete records of data that are considered irrelevant or outliers from the data. The form has the ability of refreshing the data.

So, in the interface I gave the opportunity for the user to choose the file with sessions. The content of the file must be in the following format :

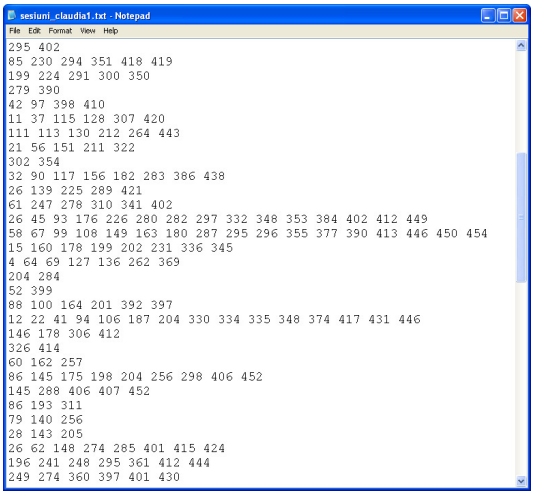


Fig.4.

As, it can be seen in the figure before, in the preprocessing stage, we codified the pages from the log files.

The user can also choose the minimum support threshold. After all these being set, we can run the algorithm.

After applying the Apriori algorithm we obtained some important associations between pages. For example, we obtained the result shown in Fig. 5.

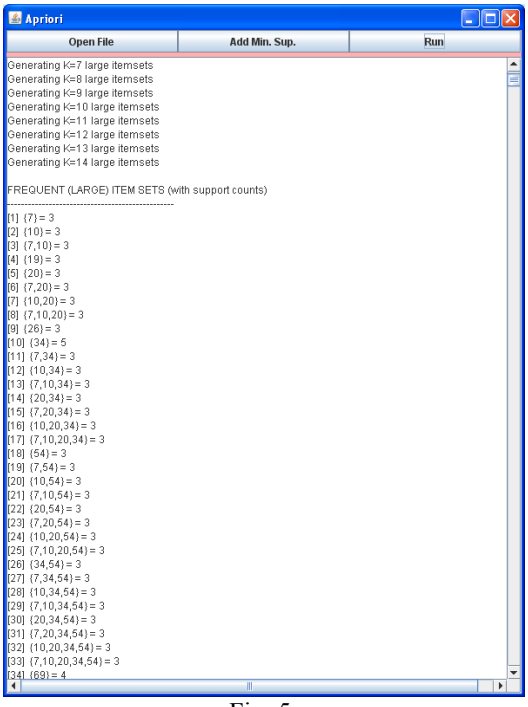


Fig. 5.

When we decrease the support, we obtain more association between pages.

Another way of obtaining association rules on pages from a web-site is by transforming this file with sessions that we obtained before in a matrix containing 0 and 1 and create an .arff file from all these pages, having as attributes the pages that can take values 0 or 1 and the relation defined between being the sessions, the data from the .arff are the values of the sessions.

The .arff file can be defined in a sparse or dense manner. After obtaining the .arff file, it can be applied to the Apriori algorithm from Weka, or any open source data mining tool that accepts this format.

Generating sets of pages frequently visited together is determined by going to the main window and choosing the application algorithms option AplicareAlgoritmi →FPGrowth. There is an option to generate steps of frequent pages accesed together with coded pages and with the exact name of the pages. FPGrowth algorithm is used to generate associations of pages frequently visited together [15].

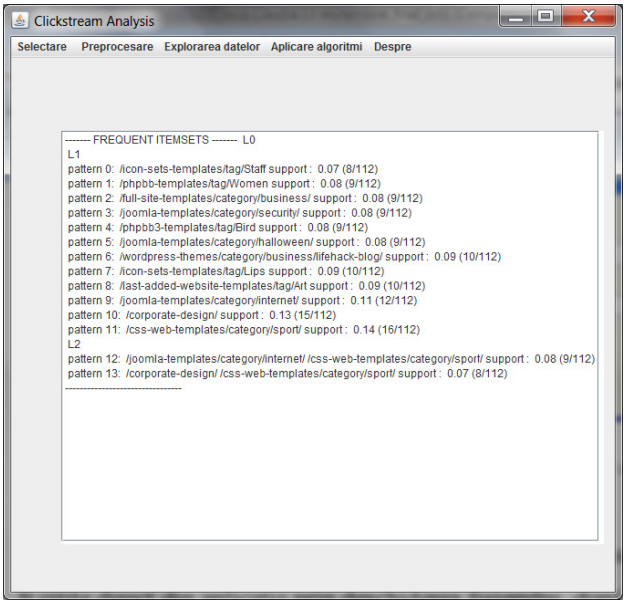
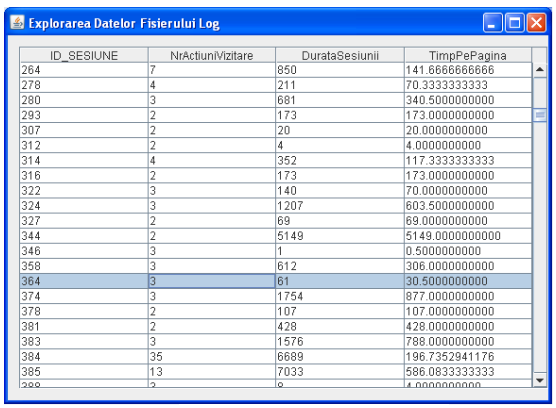


Fig. 6. Frequent itemsets presented in the main window of the application.



ID_SESIUNE	NrActiuniVizitare	DurataSesiunii	TimpPePagina
264	7	850	141.6666666666
278	4	211	70.3333333333
280	3	681	340.5000000000
293	2	173	173.0000000000
307	2	20	20.0000000000
312	2	4	4.0000000000
314	4	352	117.3333333333
316	2	173	173.0000000000
322	3	140	70.0000000000
324	3	1207	603.5000000000
327	2	69	69.0000000000
344	2	5149	5149.0000000000
346	3	1	0.5000000000
358	3	612	306.0000000000
364	3	61	30.5000000000
374	3	1754	877.0000000000
378	2	107	107.0000000000
381	2	428	428.0000000000
383	3	1576	788.0000000000
384	35	6689	196.7352941176
385	13	7033	586.0833333333

Fig 7. Exploratory data analysis

The menu **Data Exploration** contains Sessions' Information submenu which provides information about

resulted sessions such as session ID, number of visits in that session, session length, the average time per page in a session as it is shown in the figure 4.

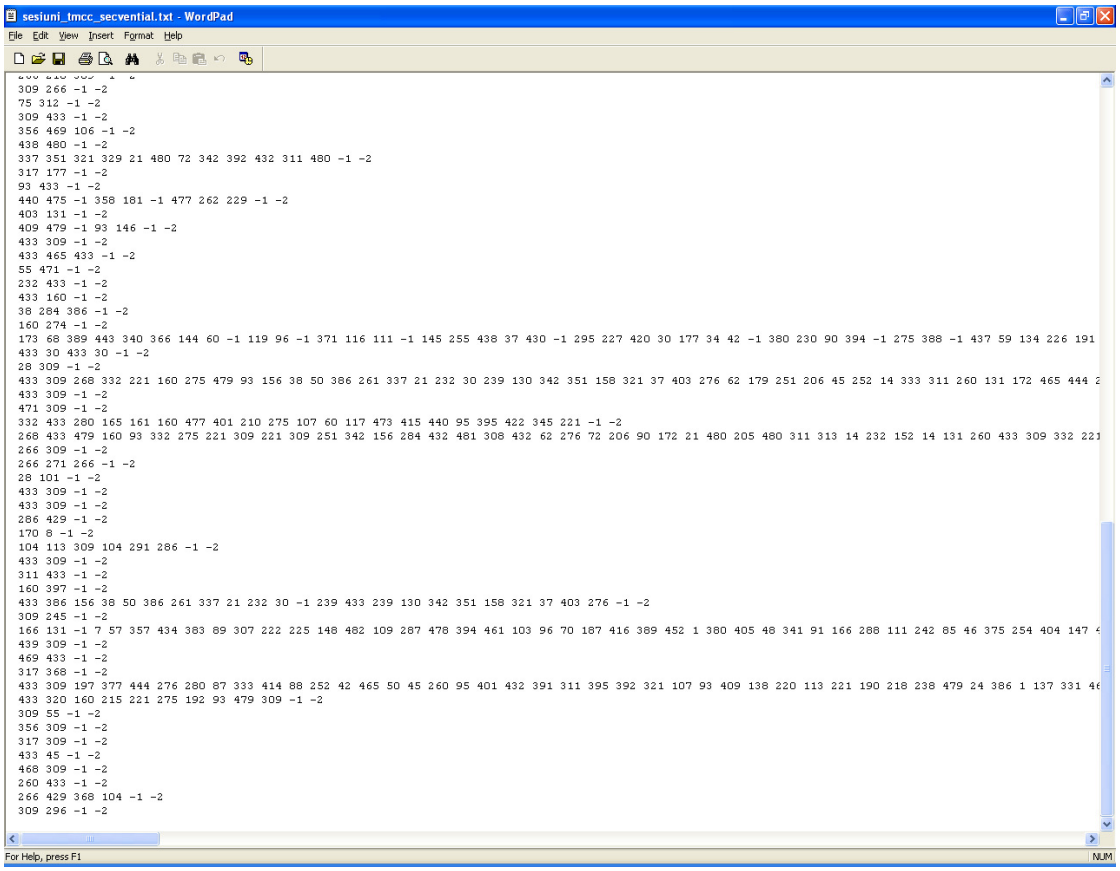


Fig. 8. The file with sequential sessions.

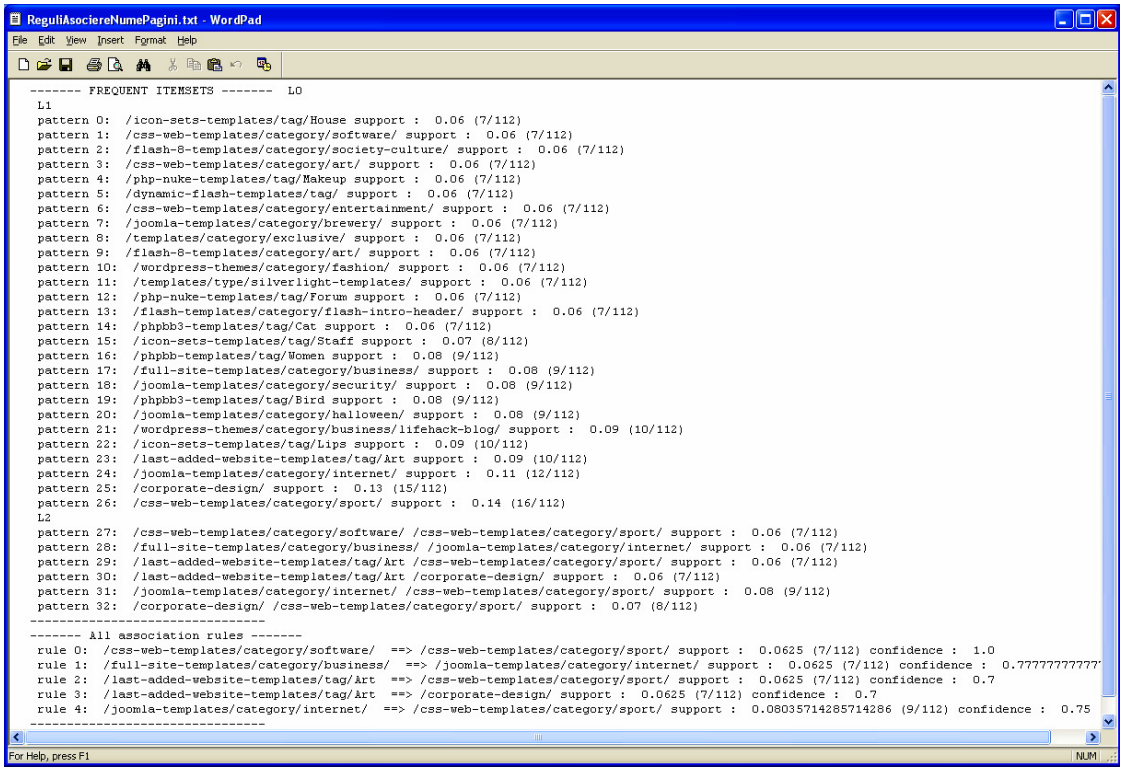


Fig. 9. Frequent itemsets and association rules obtained from web pages

In order to determine frequent web pages accessed together I use FP-Growth algorithm, and then the algorithm for generation of association rules created by Agraval and Srikant,1994 [9] from these frequent pages.

In order to apply data mining methods and algorithms for data mining the sessions are saved in a file like the one from figure 1, the sessions containing the coded pages. For algorithms which determine frequent sequential pages we use for input a text file which contains the user sessions in a temporal order as in Fig. 8. Thus, the value -1 is used to separate sessions from the same IP and the value -2 is used to separate sessions from different IPs.

Sequence mining is the task of finding temporal patterns over a database of sequences, in this case a data base of click streams. Sequence mining is considered to be an extension of associations mining that only finds nontemporal patterns.

This technique can have a very important role in knowledge discovery in web log data, due to the (temporally) ordered nature of click-streams.

The type of patterns that results from the application of this technique, can have an example like this:

”If user visits page X, and then page Y, it will visit page Z with c% of chance”. The algorithms for sequence mining inherited much from the association mining algorithms, and many of them are extensions of the firsts, where the main difference is that in sequence mining inter-sequence patterns are searched, where in the association mining the patterns searched are intra-sequence patterns.

For the determination of successive sets of frequently accessed together pages using PrefixSpan algorithm developed by Pei and others, 2004. Running the algorithm for generating frequent sets of sequential pages with minimum support 0.09 we obtain the results that can be seen below.

```
========== Algorithm - STATISTICS ==========
Total time ~ 55 ms
Frequent sequences count : 20
---------FREQUENT SEQUENTIAL PATTERNS -------
 L0
 L1
 pattern 1: (93 ) Sequence ID: 35 87 2 101 98 42 58 24 124 123 105 90   support : 0.09 (12/133)
 pattern 2: (104 ) Sequence ID: 101 39 37 42 77 131 10 73 14 21 20 113 23 58 28   support : 0.11
(15/133)
 pattern 3: (160 ) Sequence ID: 68 35 2 70 101 98 65 97 42 10 13 104 105 119 116 58 95 124
support : 0.14 (18/133)
 pattern 4: (221 ) Sequence ID: 35 2 70 71 101 98 65 42 10 104 105 24 58 124 123   support : 0.11
(15/133)
 pattern 5: (210 ) Sequence ID: 68 35 101 98 53 42 58 79 104 105 15   support : 0.08 (11/133)
 pattern 6: (249 ) Sequence ID: 101 8 42 10 46 44 119 16 48 20 23 58 56   support : 0.1 (13/133)
 pattern 7: (275 ) Sequence ID: 35 2 70 101 4 98 65 42 10 11 41 47 104 105 24 58 124   support :
0.13 (17/133)
 pattern 8: (266 ) Sequence ID: 1 101 98 67 42 77 131 10 79 106 107 14 54 80 23 58   support :
0.12 (16/133)
 pattern 9: (309 ) Sequence ID: 2 4 129 8 9 10 132 25 24 28 35 33 39 36 37 42 41 46 44 50 48 54 58
56 63 70 66 67 79 73 80 82 91 102 103 100 101 98 110 109 106 105 118 114 113 127 126 125 124
```

Data are implemented in the developed program. We run the algorithm for generating sequential rule by using the following path AplicareAlgoritmi → GenerareReguliSecventialeRuleGen.

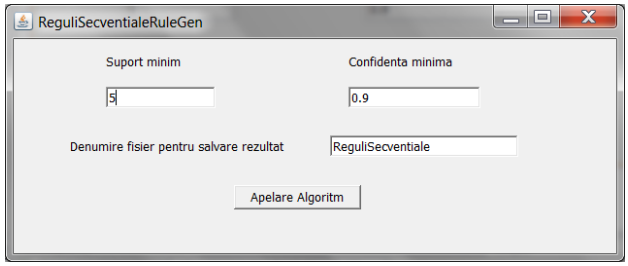


Fig. 10. Window used for algorithm call for generating sequential rules

After setting the minimum support, minimum confidence and the file name where you want to save the result file with the obtained rules we click on the button „Apelare Algoritm”. For convenience these rules may be obtained with coded pages. Having a minimum support threshold and a minimum confidence, we determine sequential rules by using RuleGen algorithm [14]. First, this algorithm applies another algorithm to determine frequent sequential pages, in this case we use PrefixSpan, and then frequent pairs of models are combined to determine sequential rules from pages. In the following images it can be seen the sequential rules obtained from web pages with the coded pages in Fig. 11. and exact name of the pages in Fig. 12.

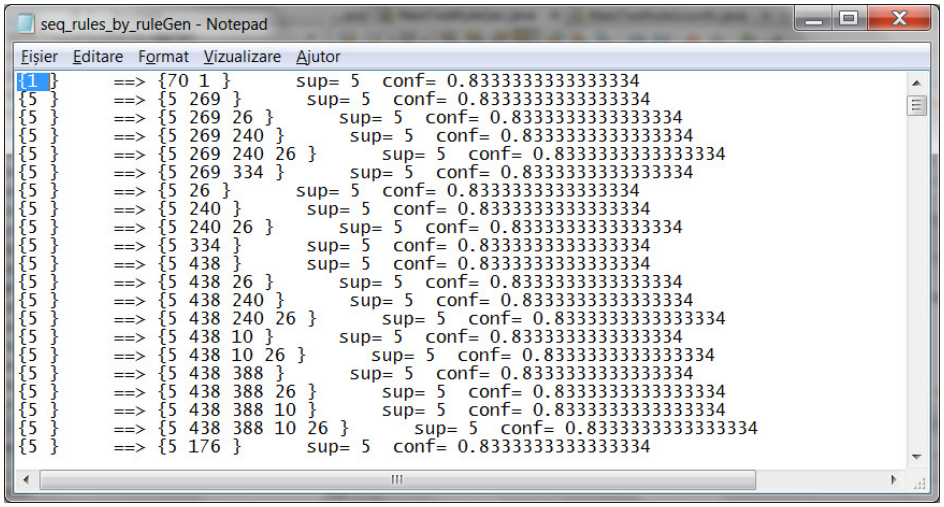


Fig. 11. Sequential rules obtained with the coded pages.

Fig. 12. Sequential rules from web pages with the exact name of pages.

V. Conclusions

Nowadays, the web is an important part of human life. The web is a very good place to do businesses. Today, large companies rethink their business using the Internet to improve business. Business carried on the Web offers the opportunity to potential customers or partners where our products and specific company can be found. To differentiate through the Internet economy, winning companies have realized that e-commerce transactions is more than just buying / selling, so the appropriate strategies are the key to improve competitive power. One effective technique used for this purpose is data mining. Data mining is the process of extracting interesting knowledge from data. Web mining is the use of data mining techniques to extract information from web data.

Web mining can be divided as was stated above in three categories: Web content minig, Web structure mining and Web usage mining. Data mining as applied to e-commerce is a breakthrough technology that can gather information in an automated fashion and build models used to predict customer purchasing decisions and navigation models with remarkable accuracy.

At the beginning I present the data preprocessing steps which has been performed on the log files from this commercial web site. Data preparation phase starts with data collection. Usually, the analyst does not participate in the process of data collection, so his goal is to select from existing data those that best fit the analyse it wishes to perform. Variables and records used depend on what it is desired to obtain. The primary source used for web usage mining are logs files of the server. The data used to analyze web usage mining may come from two sources: the period of testing and web logs. The log files from the testing periods are rarely used because of the large time required and high cost. Web log files consist of information which track web users work in their interaction with web servers. Logs files can be located as follows: on the web server, a proxy server or client computers. Logs available on Web servers are most often used because they contain accurate and complete data on site usage.

Here I presented the method that I proposed for session identification by adding the medium time that a user can spend on a specific page as a threshold for session identification. Having the data preprocessing step done, we can then go to another important step in web mining, the one of effectively extracting useful information from all this data. Mining the associations from web site pages is an important task as it helps web site designers to improve the design of the site. It gives better satisfaction for the final user. By mining associations of web pages from web logs the web site designer can discover the bad web page association and can change the design.

This article presents different ways of solving this problem. I apply different algorithms for discovering navigation patterns from data log files.

The novelty brought by this work is represented by the Java application with a friendly graphical user interface, use the mean time to identify sessions and application of different data mining algorithms on Web logs for navigation patterns extraction.

Analyzes aim is improving the site design and so leading to customer satisfaction and increasing the number of visits, visitors and therefore sales.

By it's architecture, the application provides a highly flexible environment and can be easily modified by its content, distributed and improved.

For the future I consider adding new modules to the applications developed in order to execute various data mining analysis.

REFERENCES

[1] C. E. Dinucă, **"**The process of data preprocessing for Web Usage Data Mining through a complete example", *Annals of the "Ovidius" University, Economic Sciences Series* Volume XI, Issue 1, 2011

[2] Z. Markov, D. T. Larose, *DATA MINING THE WEB, Uncovering Patterns in Web Content, Structure and Usage,* USA: John Wiley & Sons, 2007.

[3] Y. Nong, The handbook of Data Mining, Lawrence Erlbaum Associates, Publishers Mahwah, New Jersey, 2003.

[4] R. Cooley, B. Mobasher and J. Srivastava. Web Mining: Information and Pattern Discovery on the World Wide Web. A survey paper. In *Proc. ICTAI-97*.

[5] B. Liu, Web Data Mining: Exploring Hyperlinks, Contents and Usage Data, Springer Berlin Heidelberg New York, 2006.

[6] L. Clark, I. Ting, C. Kimble, P. Wrigth, D. Kudenko, Combining Ethnographic and Clickstream Data to Identify Strategies Information Research 11(2), 2006.

[7] R. Kohavi, R. Parekh, Ten supplementary analysis to improve e-commerce web sites, Proceedings of the Fifth WEBKDD workshop, 2003.

[8] C. Borgelt, *Frequent Pattern Mining*, Intelligent Data Analysis and Graphical Models Research Unit European Center for Soft Computing, 33600 Mieres, Spain, 2004.

[9] R. Agrawal, R. Srikant, Fast Algorithms for Mining Association Rules, IBM Almaden Research Center 650 Harry Road, San Jose, CA 95120,1994.

[10] R. Cooley, B. Mobasher, J. Srivastava, Web mining: Information and Pattern Discovery on the World Wide Web. A survey paper. In: Proc. ICTAI-97, 1997.

[11] O. Zaiane, Conference Tutorial Notes: Web Mining: Concepts, Practices and Research. In: Proc. SDBD-2000, 2000, pp. 410-474.

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

[13] A. M. Yahya, MD. B. S. Nasir, M. Norwati, I. U. Nur, M. Zaiton, ARS: Web Page Recommendation System for Anonymous Users Based On Web Usage Mining, Proceedings of the WSEAS European conference of systems, and European conference of circuits technology and devices, and European conference of communications, and European conference on Computer science, pp. 115-120, 2010.

[14] M. J. Zaki, SPADE: An Efficient Algorithm for Mining Frequent Sequences, Machine Learning, vol. 42, no.1-2, 2001, pp. 31-60.

[15] J. Han, J. Pei, Y. Yin, R. Mao, Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach, Data Mining and Knowledge Discovery,Volume 8, Issue 1, 53–87, 2004.

[16] G. Castellano, A. M. Fanelli, M. A. Torsello, Understanding Visitor Behaviors from Web Log Data, WSEAS Transactions on Computer Research, Vol. 2, No. 2, pp. 277-284, 2007.

[17] G. Castellano , A. M. Fanelli , M. A. Torsello, LODAP: a log data preprocessor for mining web browsing patterns, Proceedings of the 6th Conference on 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, pp.12-17, 2007.

[18] G. Castellano, A. M. Fanelli, M. A. Torsello, Mining usage profiles from access data using fuzzy clustering, Proceedings of the 6th WSEAS International Conference on SIMULATION, MODELLING AND OPTIMIZATION (SMO '06), pp. 157-160, 2006.