Annals of "Dunarea de Jos" University of Galati
Fascicle I. Economics and Applied Informatics
Years XVII – nº2/2011 ISSN 1584-0409
www.ann.ugal.ro/eco

Association and Sequence Mining in Web Usage

Claudia Elena DINUCĂ[*]

ARTICLE INFO

Article history:
Accepted 1 June 2011
Available online 30 June 2011

JEL Classification
A12, G14, L21, M15, M21

Keywords:
Clickstream analysis, Web server logs, Association rules, Sessions identification, Apriori algorithm, Web usage mining, Sequence rule mining

ABSTRACT

Web servers worldwide generate a vast amount of information on web users' browsing activities. Several researchers have studied these so-called clickstream or web access log data to better understand and characterize web users. Clickstream data can be enriched with information about the content of visited pages and the origin (e.g., geographic, organizational) of the requests. The goal of this project is to analyse user behaviour by mining enriched web access log data. With the continued growth and proliferation of e-commerce, Web services, and Web-based information systems, the volumes of click stream and user data collected by Web-based organizations in their daily operations has reached astronomical proportions. This information can be exploited in various ways, such as enhancing the effectiveness of websites or developing directed web marketing campaigns. The discovered patterns are usually represented as collections of pages, objects, or re-sources that are frequently accessed by groups of users with common needs or interests. The focus of this paper is to provide an overview how to use frequent pattern techniques for discovering different types of patterns in a Web log database. In this paper we will focus on finding association as a data mining technique to extract potentially useful knowledge from web usage data. I implemented in Java, using NetBeans IDE, a program for identification of pages' association from sessions. For exemplification, we used the log files from a commercial web site.

1. Introduction

The Web is the universal information space that can be accessed by companies, governments, universities, students, teachers, businessmen and some users. In this universal space trading and advertising activities are held. A Web site is a lot of interconnected web pages that are developed and maintained by a person or organization. Web mining and analyzing studies reveal useful information on the web. 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 [13].

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. 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 [19].

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.

[*] Faculty of Economics and Business Administration, University of Craiova. Romania, Email addresses: clauely4u@yahoo.com (Claudia Elena Dinuca)

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 behaviour 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.

2. 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, subdomains 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 data analysis and contribute 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; 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 is identified by 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 implemented in Java the Apriori algorithm in order to obtain the association rules between the pages from the sessions. I applied this algorithm on the sessions obtained.

3. Mining frequent itemsets

One of the most well known and popular data mining techniques is the association rules (AR) or frequent itemsets mining algorithm. The algorithm was originally proposed by Agrawal et al. [14] for market basket analysis. Because of its significant applicability, many revised algorithms have been introduced since then, and AR mining is still a widely researched area.

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 $T_1, T_2, \ldots T_m$, that is D = $\{T_1, T_2, \ldots, T_m\}$. 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 :

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

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.

"Apriori" was the first association rules mining algorithm. Lots of improved algorithms (most of them are "apriori"-based) have been introduced since it was published.

Apriori algorithm

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. Apriori algorithm uses the same approach as the traditional association rules algorithm, except that it doesn't look back to the input you it makes iterations to increase the set of items with maximum support. Agrawal and Srikant notes that a lot of support items Minin, each item-subset of it must have minimum support. Given this achievement, Apriori algorithm was introduced to deal with lots of articles of dimension to k-1 (from previous iteration) with minimum support when they generate lots of items of size k, don't seeking to the all data entry again. This reduces the performance due to reduced complexity of the algorithm, without reducing the quality and completeness of external algorithm results.

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.

The pseudo-code for Apriori algorithm after Christian Borgelt [8] is specified in the following section :

Function apriori (A,T, $s_{\min}$)

```
Begin  //Apriori algorithm
        $k := 1;$  //initialize the item set size
        $E_k := \cup_{a \in A} \{\{a\}\}$ ;//start with single element sets
    $F_k := prune(E_K, T, s_{\min});$ // snd determine the frequent ones
        while $F_k \neq \Phi$ do begin  //while there are frequent items sets
            $E_{k+1} := candidates(F_K);$ //create item sets with one item more
            $F_{k+1} := prune(E_{k+1}, T, s_{\min});$ //and determine the frequent ones
            $k := k + 1;$
        end;
        return  $\cup_{j=1}^{k} F_j$ ; //return the frequent item sets
      end;
```

Function candidates(F_k)

```
Begin          //generate candidates with k+1 items
  E:= Ø;       //initialize the set of candidates
  forall f1, f2 ∈ $F_k$ //traverse all pairs of frequent item sets
```

```
    with f1={ $a_1,\ldots,a_{k-1},a_k$ } //that differ only in one item
    and f2={ $a_1,\ldots,a_{k-1},a_k^{'}$ } //and are in lexicographic order
    and $a_k < a_k^{'}$ do begin    //the order is arbitrary, but fixed
        f:=f1 ∪ f2={ $a_1,\ldots,a_{k-1},a_k,a_k^{'}$ } //the union has k+1 elements
        if ∀ a ∈ f : f-{a} ∈ $F_k$ //only if all subsets are frequent
        then E:=E ∪ {f};    //add the new item set to the candidates
      end;                  //otherwise it can not be frequent
  return E;                 //return the generated candidates
End.
```

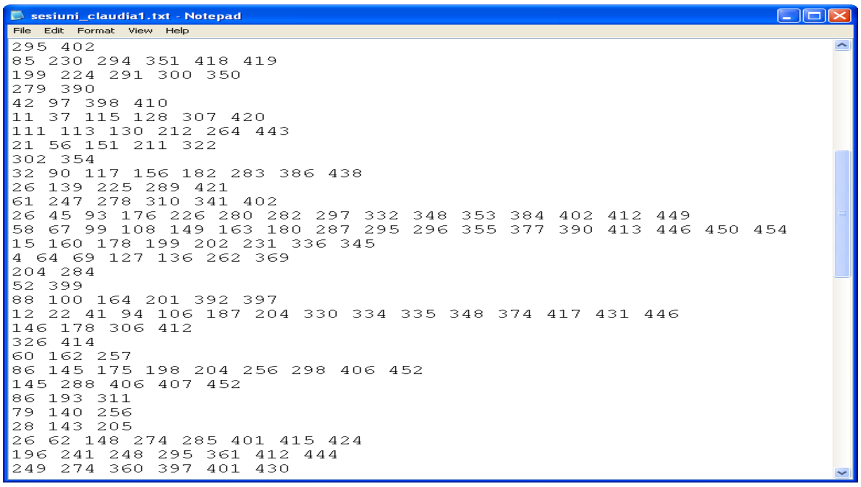
Function prune (E,T, $s_{\min}$)

```
Begin                   //prune infrequent candidates
  forall e ∈ E do  //initialize the support counters
      $s_T(e) := 0;$     //of all candidates to be checked
  forall t ∈ T do       //traverse the transactions
    forall e ∈ E do //traverse the candidates
      if $e \subseteq t$  //if transaction contains the candidates
      then $s_T(e) := s_T(e)+1;$ //increment the support counter
    F:= Ø    ; //initialize the set of frequent candidates
  forall  $e \in E$ do //traverse the candidates
    if $s_T(e) \geq s_{\min}$ //if a candidate is frequent
    then $F := F \cup \{e\};$ //add it to the set of frequent candidates
 return F; // return the pruned set of candidates
End.
```

4. Case Study

To implement the algorithms presented earlier I used the Java programming language, the code is written using NetBeans IDE.

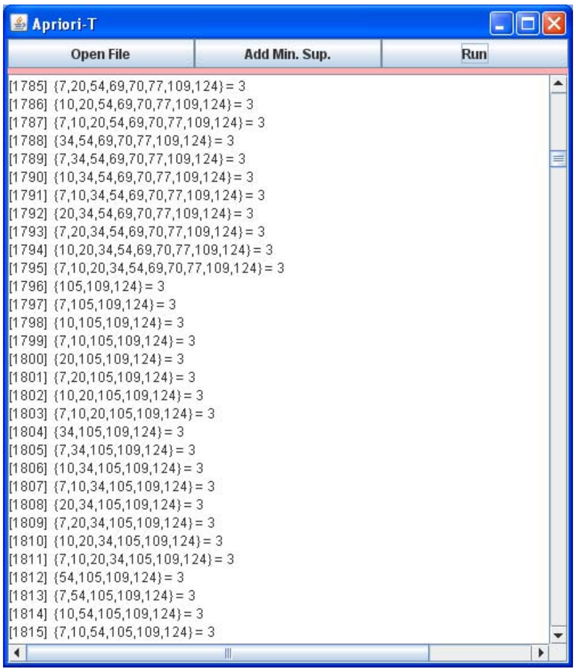
So, in the interface I gave the opportunity for the user to choose the file with sessions. The content of the file must be in the following format :



```
295 402
85 230 294 351 418 419
199 224 291 300 350
279 390
42 97 398 410
11 37 115 128 307 420
111 113 130 212 264 443
21 56 151 211 322
302 354
32 90 117 156 182 283 386 438
26 139 225 289 421
61 247 278 310 341 402
26 45 93 176 226 280 282 297 332 348 353 384 402 412 449
58 67 99 108 149 163 180 287 295 296 355 377 390 413 446 450 454
15 160 178 199 202 231 336 345
4 64 69 127 136 262 369
204 284
52 399
88 100 164 201 392 397
12 22 41 94 106 187 204 330 334 335 348 374 417 431 446
146 178 306 412
326 414
60 162 257
86 145 175 198 204 256 298 406 452
145 288 406 407 452
86 193 311
79 140 256
28 143 205
26 62 148 274 285 401 415 424
196 241 248 295 361 412 444
249 274 360 397 401 430
```

Fig. 1

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, I can run the algorithm. After applying the Apriori algorithm I obtained some important associations between pages. For example, I obtained :



```
[1785] {7,20,54,69,70,77,109,124}= 3
[1786] {10,20,54,69,70,77,109,124}= 3
[1787] {7,10,20,54,69,70,77,109,124}= 3
[1788] {34,54,69,70,77,109,124}= 3
[1789] {7,34,54,69,70,77,109,124}= 3
[1790] {10,34,54,69,70,77,109,124}= 3
[1791] {7,10,34,54,69,70,77,109,124}= 3
[1792] {20,34,54,69,70,77,109,124}= 3
[1793] {7,20,34,54,69,70,77,109,124}= 3
[1794] {10,20,34,54,69,70,77,109,124}= 3
[1795] {7,10,20,34,54,69,70,77,109,124}= 3
[1796] {105,109,124}= 3
[1797] {7,105,109,124}= 3
[1798] {10,105,109,124}= 3
[1799] {7,10,105,109,124}= 3
[1800] {20,105,109,124}= 3
[1801] {7,20,105,109,124}= 3
[1802] {10,20,105,109,124}= 3
[1803] {7,10,20,105,109,124}= 3
[1804] {34,105,109,124}= 3
[1805] {7,34,105,109,124}= 3
[1806] {10,34,105,109,124}= 3
[1807] {7,10,34,105,109,124}= 3
[1808] {20,34,105,109,124}= 3
[1809] {7,20,34,105,109,124}= 3
[1810] {10,20,34,105,109,124}= 3
[1811] {7,10,20,34,105,109,124}= 3
[1812] {54,105,109,124}= 3
[1813] {7,54,105,109,124}= 3
[1814] {10,54,105,109,124}= 3
[1815] {7,10,54,105,109,124}= 3
```

Fig. 2.

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 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. 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.

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 association 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.

5. Conclusions

At the beginning I present on short the data preprocessing which has been performed on the log files from this commercial web site. 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. 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 these 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. 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 Apriori algorithm on Web logs. In this study we have seen how the techniques of association and sequence mining can be naturally applied to the web usage mining task, extracting association rules or frequent patterns in the web log data. This kind of knowledge cannot be extracted with classical statistical analysis and gives much more insight to business analysts, consisting in a powerful tool to the business area. So by mining associations between web pages we can discover the pages frequent accessed together. This information can be helpful for site developer in order to arrange the pages and so bring customer satisfaction and increase sells. In the future we consider adding new modules to the applications developed in order to execute various data mining analysis. Future research direction consist in implementing another algorithms like FP-Growth[15] for generating association between web pages and then implementing also some algorithm for associations rule mining like the one of Agraval şi Srikant,1994 [16] also some algorithms for sequential rule mining like the one of RuleGen algorithm [17] and PrefixSpan[18].

All these algorithms will be applied for determining the frequent users' navigation models of web site. This will help web site designers and developers to improve the initial design created and so attract more visitors by the user friendly interface developed. So, the web site designers can determine the web pages that are not correct located and bring them to the right position and also in time, having these navigation models, it can be developed a system for next page prediction.

References

[1]. Claudia Elena 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]. Zdravko Markov, Daniel T.Larose, DATA MINING THE WEB, Uncovering Patterns in Web Content, Structure and Usage, USA: John Wiley & Sons, 2007

[3]. Nong, Y.: 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]. Liu B. (2006), Web Data Mining: Exploring Hyperlinks, Contents and Usage Data, Springer Berlin Heidelberg New York.

[6]. Clark L., Ting I., Kimble C., Wrigth P., Kudenko D. (2006), Combining Ethnographic and Clickstream Data to Identify Strategies Information Research 11(2), paper 249

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

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

[9]. Srivastava J., Cooley R., Deshpande M., Tan P.-N., Web usage mining: discovery and applications of usage patterns from web data, SIGKDD Explorations, 1(2), 2000, 12-23.

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

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

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

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

[14]. Agrawal R., Srikant R.: Mining sequential patterns. In Proceedings of the 11th International Conference on Data Engineering, 3-14, September 1995.

[15]. JIAWEI HAN, JIAN PEI, YIWEN YIN, RUNYING 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]. Rakesh Agrawal, Ramakrishnan Srikant, Fast Algorithms for Mining Association Rules, IBM Almaden Research Center 650 Harry Road, San Jose, CA 95120,1994

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

[18]. Jian Pei, Jiawei Ha şi alţii, Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 16, NO. 10, OCTOBER 2004.

[19].Mazilescu V.: Fuzzy Dynamic Discrimation Algorithms for Distributed Knowledge Management Systems, Annals of Dunarea de Jos University of Galati. Fascicile I. Economics and Applied Informatics, 2010, Years XVI, no. 2, p. 15-26.