AN EFFICIENT APPROACH FOR WEB-SITE ADAPTATION

Seema Jani, Sam Makki
Department of Electrical Engineering and Computer Science
University of Toledo, Toledo, Ohio, USA

Xiaohua Jia
Department of Computer Science
City University of Hong Kong, Kowloon, Hong Kong

Keywords: Web personalization, Data mining, Graph, Probability, Clustering

Abstract: This paper implements a novel approach defined as the Preference-function Algorithm (PFA) for web-site adaptation. The algorithm extracts future preferences from the users' past web navigational activities. Server web logs are used to identify users' navigation behaviours by examining the traverses of various web pages. In this approach, the sessions are modeled as a finite state graph, where each visited web page is defined as a state. Then, traversing among various states provides the framework for determining the interest of the users.

1 INTRODUCTION

The continuous growth of the World Wide Web poses many challenging issues for a person using it. Most Web structures are large and complicated and users often miss the goal of their inquiry, or receive ambiguous results when they try to navigate through the web pages (Eirinaki et. al., 2003). The business environment is such that in order to do well it is important to meet the needs of the customers and also create revenue by doing so. Therefore, the need to retain the attention of the users of a web-site and understand the needs of those users leads to the importance of analyzing the users' behavior. Once the behavior of the users is known, a company can use this information for a variety of objectives and actions such as, to personalize the web-site, make recommendations, enhance the web-site and target advertise.

The objective of a Web personalization system is to "provide users with information they want or need, without expecting them to ask for it explicitly" (Mulvenna et. al., 2000). Personalizing a web-site can be done in various ways. One way is to have the information provided by the users via questionnaires, surveys or registration forms. Another is to use demographic, geographic, or psychographic profiles or other information to divide or segment large populations into smaller groups. The last is to seek to understand the behavioral preferences of a specific, individual user and then deliver web-site content specifically targeted at that person. This is known as web usage mining (Eirinaki et. al., 2003) and is the one that this paper targets. This method is more dynamic in nature and specifically takes into account the navigational tendencies of the user within a specific web-site.

Web usage mining can be regarded as a three-phase process, consisting of the data preparation, pattern discovery, and pattern analysis phases (Srivastava et. al., 2000). In the first phase, Web server log data is processed in order to identify users, sessions, pages viewed and so on. The first phase, or data preparation phase, is one of the most time-consuming, but technically straightforward tasks. In the second phase, methods such as those in the area of Statistics or Artificial Intelligence are applied to detect interesting patterns. The second phase is the most diverse and expanding of all three phases and has been the subject of continuing research and advancement. In the third phase of the Web usage mining process, these patterns are stored so that they can be further analyzed and applied.

Jani S., Makki S. and Jia X. (2005).
AN EFFICIENT APPROACH FOR WEB-SITE ADAPTATION.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems - DISI*, pages 108-114

One main area of application is web-site adaptation, where information obtained from phase two is used in conjunction with specific web page relationships and actions. The relationships and actions are determined and provided by the owners of the web-site.

2 RELATED RESEARCH

Pattern analysis in the context of Web usage mining has been the subject of numerous research projects. Two distinct directions are, in general, considered in Web usage mining research: Statistics and Artificial Intelligence techniques. The first approach using Statistics consists of a range of applications from overall analysis to an adapted version of statistical data mining techniques (Srivastava et. al., 2000). These data mining techniques are those that mine for rules using the pre-defined support and confidence values. The second approach uses Artificial Intelligence techniques which draw upon methods and algorithms developed from machine learning and pattern recognition (Srivastava et. al., 2000).

2.1 Statistical Mining Techniques

The research and application of statistical techniques in the analysis of web usage data is a wide area. Three specific research areas are covered in this section. These are the overall statistics of the web usage data, probability analysis and standard data mining techniques.

Statistical techniques are the most common method to extract knowledge about the users of a web-site. There are different levels of analysis which have been used from the area of Statistics in web mining. One general area is the use of overall statistical information, which is given by some software programs. Several commercial software packages such as Analog (Analog, 2004) and OLAP (The OLAP Report, 2005) are available for web log analysis.

One more statistical approach which has been taken in the past research is the use of probability in the form of Markov chain modeling (Borges, 2004).

Another well known area of Statistics which has been used in the area of Web usage mining is the use of data mining techniques. These techniques involve finding association rules within a set of data and mining by determining the rules based on the rules' confidence and support (Agrawal et. al., 1994). One example of work done in this area is with the application of the "a priori" algorithm, in which the association rule mining searches for relationships between the items in the data.

2.2 Artificial Intelligence Techniques

The second type of technique involves the use of Artificial Intelligence techniques such as clustering and classification. Clustering and classification are machine learning techniques that are used to group together a set of items having similar characteristics.

In the Web Usage domain, there are two kinds of interesting clusters to be discovered: page clusters and usage clusters. Clustering of pages discovers groups of pages having related content. Clustering of users, however, tends to establish groups of users exhibiting similar browsing patterns.

Classification is the task of mapping a data item into one of several predefined classes. Classification is done by using supervised inductive learning algorithms such as decision tree classifiers, naïve Bayesian classifiers, k-nearest neighbor classifiers, Support Vector Machines, etc.

Mobasher (Mobasher et.al., 1999) modeled the profile based on the clustering approach and named it PACT, which stands for Profile Aggregations based on Clustering Transactions. The goal is to effectively capture common usage patterns from potentially anonymous click-stream data. The data is preprocessed and then used to create the profile. Preprocessing of the data is done in two steps: identifying users and determining pageviews. First, unique users are identified from the anonymous usage data. Erroneous or redundant references are removed from the data. Second, pageviews are identified. Pageview identification is the task of determining which page file accesses contribute to a single browse display. Relevant pageviews are included in transaction files and weights are assigned to reflect the significance of the pageview. The clustering approach however requires that the structure of the web-site to be known.

Classification is also used to determine the user preferences (Baglioni et. al., 2003). Classification algorithms require training data as input. In this example, the input is a set of cases whereby each case specifies values for a collection of attributes and for a class. The output of the classification algorithm is a model that describes or predicts the class value of a case on the basis of the values of the attributes of the case. The predictive accuracy of the extracted model is evaluated on a test set for which the actual class is known.

This method requires registered users to provide information about themselves. The registered users' information is broken up whereby 67% become the training set and 33% become the test set. The attributes of a class consist of the site pages or sections visited by the user and the class consists of the user's sex. In this case the goal is to accurately

determine the user's sex based on the web pages that a user visited. The classification accuracy of this approach is 54.8%. The major drawback is the information about the users which has to be obtained ahead of time.

3 OVERALL PROPOSAL

To extract the related information from the users' log, we use a new approach called the **Preference-function Algorithm.** The Preference-function consists of two components defined as the Likeness-factor and the Time-factor. The Likeness-factor provides an overall idea as to whether or not a particular web page has been visited frequently by the user. The Time-factor provides the overall interest of that web page. These two components are further elaborated upon in the next few sections. Also in this approach, knowledge of the structure of the web-site is not required because this knowledge will be gained from the user's specific actions. The overall process is divided into two steps as shown in Figures 1 and 2.

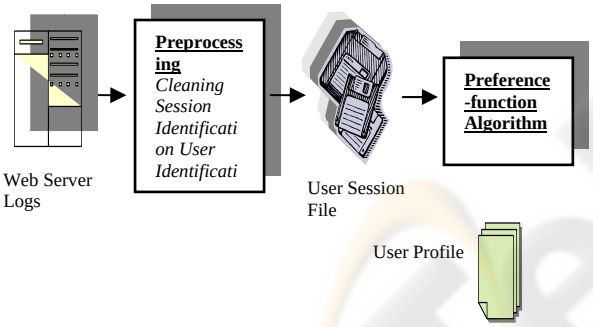


Figure 1: Step 1 in Preference-function Web Mining

In step 1, the user profile will be determined by first preprocessing the data, which is also known as data preparation, and then the specific usage mining will be performed using the Preference-function. The data preparation consists of creating useful information from the log files. Nevertheless should mention that, log files may contain many entries that are irrelevant or redundant for the mining task. Initially, the raw data is cleaned which includes removing all redundant or unnecessary information. All information relating to image files and map files, which exist in the log files, is irrelevant to identifying the user's behavior. Then, the sessions and transactions are identified based on the available data. For our experiment we consider a session length of a 6 hour period and will start when the first page is requested from a designated web-site. The end of the session will be determined when the user leaves the web-site, or when the time on one web page has exceeded 30 minutes. The 30 minute timeout assumption is based on the results from Catledge and Pitkow (Catledge et. al., 1995). Various transactions can occur during each session. Individual entries for page accesses are grouped into meaningful transactions. The transactions are first defined as being unique and are grouped together based on the IP addresses, since any access from different IP addresses is identified as a different transaction.

In step 2, as shown in Figure 2, involves making recommendations based on a company-provided scenario. In the case of this experiment, the web-site owner will include suggested actions in the scenario.

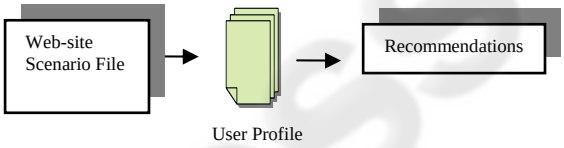


Figure 2: Step 2 in Preference-function Web Mining

3.1 Preference-function (PF)

Because the topology of the web-site is not known, information on the server log will provide a way to reconstruct a partial topology. The following assumptions will provide a framework for analyzing the server logs:

- A session is to start when the first page of a designated web-site is requested from any IP address.
- The beginning of a user session is determined to be the first time a unique IP address is observed.
- The end of the session will be determined when the user has left the web-site or when a timeout of 30 minutes has been exceeded on one web page.
- Each unique IP address will be compared with those that share the same operating system.
- Previously visited web pages are allowed.

Relevant information will be extracted from the server logs and combined to form a Preference-function (PF). The Preference-function will be composed of the multiplication of two variables: Likeness-factor and Time-factor.

3.2 Likeness-factor (LF)

In this paper, the sessions are modeled as a finite state machine with each web page visited within a web-site defined as a state. Two additional states will be added, S and F, to denote the start and final states. The transition from each state to the next will be denoted as an edge on a weighted directed graph. The Likeness-factor is determined by summing the path weights (pw) of the individual paths that were used to reach a particular state. The path weights are defined as follows: each unique path toward a particular state by definition will be given a partial path weight of 1; any repetition that takes place, such as leaving the current page and then returning, will be given a partial path index of ½; and the total path weight for the path is determined by summing the values for the partial path weight of the unique path plus each of the various detours. The Likeness-factor (LF) is determined by adding various path weights to a particular state. The path weight of a state (s) is determined in equation (1) where "n" is the number of detours once a specified state is reached:

$$pw_s = 1 + \sum_{i=1}^{n} pw_i \qquad (1)$$

The formula for the Likeness-factor is given by summing up all the path weights of all the paths (m) to a particular state (s):

$$LF_s = \sum_{j=1}^{m} pw_{sj} \qquad (2)$$

3.3 Time-factor (TF)

The reference length approach (Cooley et. al., 1997) is based on the assumption that the amount of time a user spends examining an object is related to the interest of the user for the object's contents. On this basis, a model for user sessions is obtained by distinguishing the navigational objects (i.e., containing only links interesting to the user) from the content objects (i.e., containing the information the user was looking for). The distinction between navigational and content accesses is related to the distance (in time) between one request and the next. This Time-factor is added to the evaluation function which was developed above in (1) and (2).

The Time-factor (TF) considers that, generally, the more time a user spends on a web page, the higher interest the user has for information on that page. Therefore, the TF is determined by dividing the time (sec) at a single web page (determined from the web server logs) by the total time for that session, where "l" is the number of sessions. The formula for TF is shown in equation (3) where "l" is the number of total states and "p" is the number of times that the state "s" appears:

$$TF_s = \frac{\sum_{k=1}^{p} t_{sk}}{\sum_{k=1}^{l} t_k} \qquad (3)$$

3.4 Overall Equation

From equations (2) and (3) we can compute the Preference-function as follows:

Preference-function (PFw) = Likeness-factor (LFw) * Time-factor (TFw) (4)

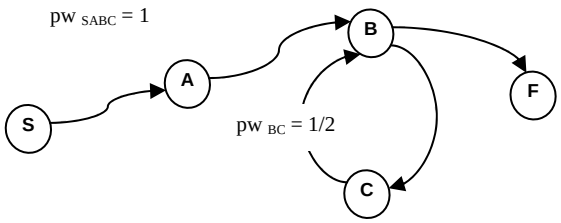
3.5 Example

Suppose we have a web-site which consists of web pages A, B and C, and based on the server log it is known that a unique user used the following paths:

$$\{A \to B \to C \to B\}$$
$$\{A \to C \to B \to C \to C\}$$
$$\{B \to C \to B\}$$
$$\{C \to A \to B\}$$

- Based on the first path, the path weights (pw) would be calculated as follows:

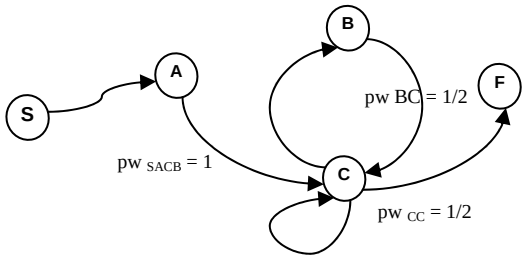
$$pw_{s \to A \to B \to C} = 1$$
$$pw_{C \to B} = 1/2$$



The pwB1 for this path would be:

pwB1 = 1 + ½ = 1 ½

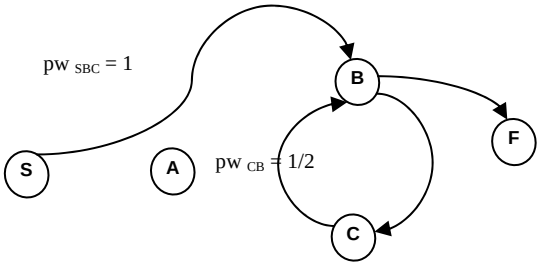
- For the second path the pw session



- The pwB2 for the second path would be:

pwB2 = 1

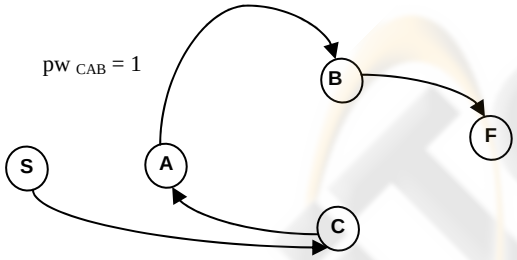
For the third path, pwB would be as follows:



The pwB3 for the third path would be:

pwB3 = 1 + ½ = 1 ½

- For the fourth path, pwB would be as follows:



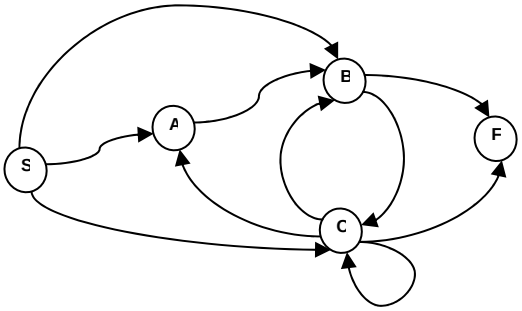
- The pwB4 for the fourth path would be:

pwB4 = 1

- Therefore the Likeness-factor for site B would be equal to

  - LFB = pwB1 + pwB2 + pwB3 + pwB4 = 1 ½ + 1 + 1 ½ + 1 = 5

- The overall graph is



pw SABC = 1 pw SACB = 1
pw SBC = 1 pw SCAB = 1
pw BC = ½ pw CB = ½

Also based on the server log it is known that the user spent 10 secs in A, 240 secs in B and 600 secs in C. Then the TF for each of the web pages becomes

TF(A) = 10/850 = 0.012
TF(B) = 240/850 = 0.282
TF(C) = 600/850 = 0.706

Therefore the Preference Factor for webpage B would be:

PFB = LFB * TFB
= 5 * 0.282
= 1.41

4 PROGRAMMING DESIGN AND IMPLEMENTATION

4.1 Programming Language Choice

The programming language Java was used in creating the Preference-function Web Mining. Several other programming languages such as C++, Visual Basic, Perl or Python could have been used for this implementation.

The Preference-factor Algorithm program can be shown as the following pseudo-code:

Begin {
Create GUI layout
Show GUI
Instantiate class variables
Wait for action from user

If *"Mine Logs" button pressed* ***{***
Display dialog box to choose file
Open Server Log File
Read Log File

Break up each line into individual components
Ip address, Date & Time, Command, Web page, Status, Bytes, Name, Zone, Type, Reference, Agent
Create new record with components Eliminate all the unnecessary and null items(such as .gif, .jpg, .css., .png, .ico, search type, .db
Create a session based on unique ip addresses and times
Store within each session the web pages (states) and clock time
Create a linked list of each session
Within each session create various paths taken by the user
Determine the time in seconds for each state
Determine the Likeness-factor by following the paths and determining if any states are visited again
Determine the total time for the session
Determine the Time-factor for each web page by dividing the time by the total time
Determine the Preference-function for each web page by multiplying the Likeness-factor and the Time-factor

Determine the highest and second-highest preference factor and its web page

Append to text area in GUI ***}***
end if
If *"Web Site File" button pressed* ***{***
Display dialog box to choose file
Open web-site file ***}***
end if

If *"Suggested Action" button pressed* ***{***
Read web-site file
Check to see if each line is equal to the web page with the highest or second-highest Preference-function
If equal then append text area with s uggestions Close web-site file ***}***
end if

If *"Exit" button pressed*
Exit program
end if
End

5 EXPERIMENTAL PROCEDURE

5.1 Step 1: Server Log Analysis

Data preparation is one of the central tasks in web mining (Baglioni et. al., 2003). In fact, it usually takes up most of the time of the total analysis process. First, unwanted requests, which are in the form of rows, need to be filtered out. These are usually image files, javascript files and style sheets, to name a few. Next, any unwanted information about the request needs to be filtered, such as the user identification, request method and query strings. The following are several lines from the server log file used for experimentation:

- localhost - - [04/Feb/2004:15:09:36 -0500] "GET /phpdev/ HTTP/1.0" 200 383
- localhost - - [04/Feb/2004:15:09:36 -0500] "GET /phpdev/yak1.htm HTTP/1.0" 200 1048
- localhost - - [04/Feb/2004:15:09:36 -0500] "GET /phpdev/yak2.htm HTTP/1.0" 200 6798

5.2 Step 2: A Recommendation Scenario

Ultimately the owner of the web-site has to give guidelines regarding the correlations and importance of web pages. This information is provided by the owner in the action file. Below is a sample action file for the sample web-site created for this project. For example, in the first line the web page name is '\phpdev\' and the action to be taken is 'php books'.

/phpdev/ → php books
/phpdev/yak1.htm → php books
/phpmyadmin/ → php admin books
/AnalogX/ → other web mining sites
/site/ → Apache info
/private/ → Apache info
/start_here.htm → Apache directory information
/phpinfo.php → php books

5.3 An Example of Preference-function Algorithm

A GUI was designed to provide for an easy interaction between program and the user. There are four actions that a user can take shown in Figure 3.

Figure 3: Four Actions

When the "Mine Logs" button is pressed, a dialog box appears for the user to choose which server log

file to open. Once the file has been opened, the results of the mining are shown.

Next, a web-site action file has to be selected. This is done by pressing the "Web Site File" button. When this button is pressed, a dialog box appears to select the file to use for web-site actions.

In order to get the suggested actions for the user preferred web-pages, the "Suggested Actions" button is pressed.

6 CONCLUSIONS

When planning web applications, information about the users' preferences makes designing web pages more relevant and useful. Being able to adapt web pages provides the flexibility needed in the ever-changing world of likes and dislikes. This paper implements a novel approach defined as the Preference-function Algorithm (PFA) for web-site adaptation. The algorithm extracts future preferences from users' past web navigational data. Server web logs are used as navigational information to formulate the Preference-function. Using states to model each web page, all sessions are modeled as a finite state graph and the traverses among the various states are determined to be the path of a particular user.

The viability of the Preference-function Algorithm is shown with a user-created web-site and the automatic creation of the server logs. The Preference-function is determined using the Likeness and Time-factors. The highest and second highest pages are determined to be the most preferred web pages by the unique users.

REFERENCES

Eirinaki M. and Vazirgiannis M. 2003. Web mining for web personalization, ACM Transactions on Internet Technology, Vol. 3, No. 1, pp. 1-27.

Mulvenna M. D., Anand S. S., and Buchner A. G. 2000. Personalization on the net using web mining, Communications ACM, 43, pp. 123-125.

Srivastava Jaideep, Cooley Robert, Despande Mukund, Tan Pang-Ning 2000. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, SIGKDD Explorations Volume I, Issue 2, pp. 12-23.

Stephen Turner, 2004. Analog, the most popular logfile analyser in the world, http://www.analog.cx/

The OLAP report, http://www.olapreport.com/

Borges Jose, 2004. "An Average Linear Time Algorithm for Web Usage Mining." Department of Computer Science, University College London, March 1998. To appear in International Journal of Information Technology and Decision Making.

Agrawal R., and Srikant R., 1994. "Fast Algorithms for Mining Association Rules." In Proceedings of the 20th VLDB Conference, Santiago, Chile, pp. 487-499.

Baglioni, M., Ferrara, U., Romei, A., Ruggieri, S., Turini, F., 2003. Preprocessing and Mining Web Log Data for Web Personalization. 8th Italian Conf. on Artificial Intelligence, Vol. 2829 of LNCS, pp. 237-249.

Catledge, L., Pitkow, J., 1995. Characterizing browsing behaviors on the world wide web. Computer Networks and ISDN Systems, 27(6), pp. 1065-1073.

Cooley, R., Mobasher, B., Srivastava, J., 1997, Grouping Web Page References into Transactions for Mining World Wide Web Browsing Patterns. Proceedings of KDEX '97, pp. 50-59.