Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

A generic Data Mart architecture to support Web mining

J. D. Velásquez[1], H. Yasuda[1], T. Aoki[1] & R. Weber[2]
[1]*Research Center for Advanced Science and Technology, University of Tokyo*
[2]*Department of Industrial Engineering, University of Chile*

Abstract

Visits in a Web site leave behind important information about the behavior of the visitors. This information is stored in log files, which can contain many registers but part of them do not contain relevant information. In such cases, user behavior analysis turns out to be a complex and time-consuming task.

In order to analyze Web site visits, the relevant information has to be filtered and studied in an efficient way. We introduce a generic Data Mart architecture to support advanced Web mining, which is based on a Star model and contains the relevant historical data from visits to the Web site. Its fact table contains various additive measures that support the intended data mining tasks, whereas the dimension tables store the parameters necessary for such analysis, e.g. period of analysis, range of pages within a session.

This generic repository allows one to store different kinds of information derived from visits to a Web site, such as e.g. time spent on each page in a session and sequences of pages in a session. Since the Data Mart has a flexible structure that allows one to add other interesting parameters describing visitors navigation, it provides a flexible research platform for various kinds of analysis.

Based on these sources, user behavior can be characterized and stored in user behavior vectors that serve as input for data mining. For example, similar visits can be grouped together and typical user behavior can be identified, which allows improvement of Web sites and an understanding of user behavior.

The application of the presented methodology to the Web site of a Chilean university shows its benefits. We analyzed visits to the respective Web site and could identify clusters of typical visitors. The analysis of these clusters is used for improved online marketing activities.

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

1 Introduction

To understand Web visitor browsing can be the key in order to improve both, the respective Web content and recognition of visitor behavior. This way, an organization can assure its success in Internet.

Web log files are important data sources about user behavior [3, 6]. Depending on the traffic on a Web site, these files may contain millions of registers with a lot of irrelevant information each, such that its analysis becomes a complex task.

In this work a methodology for Web mining is introduced, based on a generic Data Mart architecture for the efficient analysis of Web site visits.

From all the possibly available data, two variables are analyzed: the content of the visited pages and the time spent in each one of them.

The proposed methodology was applied to analyze of a certain Web site. The particular results show the method's effectiveness, suggesting changes in content and structure of the respective Web site.

This paper is organized as follows. In section 2, a method to compare user sessions is introduced. Section 3 shows the Self-organizing Feature Maps (SOFM) as tool in the Web mining process. The Data Mart technology used in this work is introduced in section 4 and section 5 shows a practical application in a particular Web site. Finally, section 6 concludes the present work and points at extensions.

2 Comparing user sessions

2.1 Web page processing

We represent a document (Web page) by a vector space model [1], in particular by vectors of words.

Let R be the number of different words in a Web site and Q be the number of Web pages, a vectorial representation of the Web site would be a matrix M of dimension RxQ that contains the vectors of words in its columns:

$$M = (m_{ij}) \quad i = 1, \ldots, R \quad and \quad j = 1, \ldots, Q \tag{1}$$

where m_{ij} is the weight of word i in document j.

In order to estimate these weights, we use a variation of the *tfxidf-weighting* [1], defined by equation 2.

$$m_{ij} = f_{ij} \left(1 + \frac{sw(i)}{TR}\right) * \log\left(\frac{R}{n_i}\right) \tag{2}$$

where f_{ij} is the number of occurrences of word i in document j and n_i is the total number of times that word i appears in the entire collection of documents. Additionally, we propose to augment word importance if a user searches for a specific word. This is done by sw (special words) which is an array with dimension R.

It contains in component i the number of times that a user asks for word i in the search process during a given period (e.g.: one month). TR is the total number

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

of times that a user searches words in the Web site during the same period. If TR is zero, $\frac{sw(i)}{TR}$ is defined as zero, i.e. if there has not been any word searching, the weight m_{ij} depends just on the number of occurrences of words.

2.2 User behavior vector

In order to analyze the behavior in the Web two variables are particularly important: the pages visited by a user and the time spent in each one of them.

Definition 1 (User Behavior Vector). $\mathbf{U} = \{u(1), \ldots, u(V)\}$
where $u(i) = (u_p(i), u_t(i))$, *and* $u_p(i)$ *is the Web page that the user visits in the event* i *of his/her session.* $u_t(i)$ *is the time that the user spent visiting the Web page.* V *is the number of pages visited in a certain session.*

Figure 1 shows a common structure of a Web site. If we have a user visiting the pages 1,3,6,11 and spending 3, 40, 5, 16 seconds respectively, the corresponding user vector is: U = ((1,3),(3,40),(6,5),(11,16))

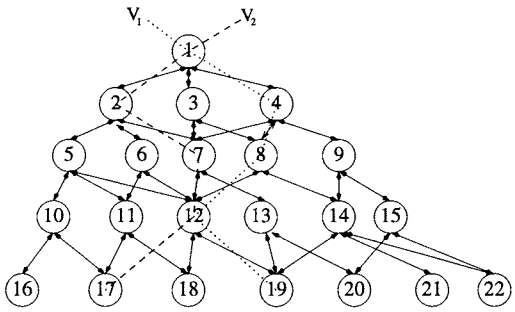

Figure 1: A common structure of a Web site and the representation of user behavior vectors.

2.3 Distance measure between two pages

With the above definitions we can use vectorial linear algebra, in order to define a distance measure between two Web pages.

Definition 2 (Word Vector Page).
$\mathbf{WP^k} = (wp_1^k, \ldots, wp_R^k) = (m_{1k}, \ldots, m_{Rk})$ *with*
$k = 1, \ldots, Q$

We used the angle's cosine as similarity measure between two page vectors [1].

$$dp_{ij} = dp(WP^i, WP^j) = \frac{\sum_{k=1}^{R} wp_k^i wp_k^j}{\sqrt{\sum_{k=1}^{R}(wp_k^i)^2}\sqrt{\sum_{k=1}^{R}(wp_k^j)^2}} \qquad (3)$$

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

Definition 3 (Page Distance Matrix). $\mathbf{DM} = (dp_{ij}) \quad i, j = 1, \ldots, Q$

Definition 4 (Page Distance Vector). $\mathbf{D_{AB}} = (dp(a_1, b_1), \ldots, dp(a_m, b_m))$ *where* $A = \{a_1, \ldots, a_m\}$ *and* $B = \{b_1, \ldots, b_m\}$ *are sets of word page vectors with the same cardinality.*

2.4 Defining a measure to compare user sessions

We define a measure that combines both characteristics of the user behavior vector (time and page content). We introduce it in equation 4. It measures the similarity between the behavior of two users i and j.

$$dub(U^i, U^j) = \frac{1}{L} \sum_{k=1}^{L} \min \left\{ \frac{u_t^i(k)}{u_t^j(k)}, \frac{u_t^j(k)}{u_t^i(k)} \right\} * dp(u_p^i(k), u_p^j(k)) \qquad (4)$$

with dp distance measure between the content of two pages.

The first element of the equation 4, $\min\{\frac{u_t^i(k)}{u_t^j(k)}, \frac{u_t^j(k)}{u_t^i(k)}\}$ is indicating the user's interest for the pages visited. We assume that the time spent on a page is proportional to the interest the user has in its content. This way, if the times spent are close to each other, the value of the expression will be near 1. In the opposite case, it will be near 0.

As second element, we use dp (equation 3) because it is possible that two users visit different Web pages in the Web site, but the content is similar, e.g., one page contains information about classic rock and another one about progressive rock. In both cases the users have interest in music, specifically in rock. This is a variation compared to the approach proposed in [6], where only the user's path was considered but not the content of each page.

Finally, we combine in equation 4 the content of the visited pages with the time spent on each of the pages. This combination is done by multiplication, such that we can distinguish between two users who had visited similar pages but spent different times on each of them. Similarly we can separate between users that spent the same time visiting pages with different content and position in the Web.

3 Mining the information from Web site visits

We used an artificial neural network of the Kohonen type (Self-organizing Feature Map; SOFM) in order to analyze visitor behavior on the Web. Schematically, it is presented as a two-dimensional array in whose positions the neurons are located [8]. Each neuron is constituted by an n-dimensional vector, whose components are the synaptic weights. By construction, all the neurons receive the same input at a given moment.

The notion of neighborhood among the neurons provides diverse topologies. In this case the thoroidal topology was used [9], which means that the neurons closest to the ones of the superior edge, are located in the inferior and lateral edges (see figure 2)

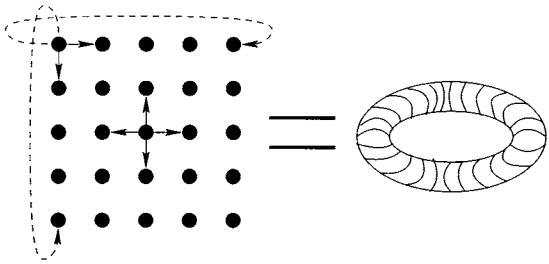

Figure 2: Proximity of the user behavior vector in a network of thoroidal Kohonen.

The U vectors have two components (time and content) for each Web page. Therefore it is necessary to modify both when the neural network changes the weights for the winner neuron and its neighbors.

The time component of the U vector is modified with a numerical adjustment, but the page component needs a different updating scheme. In the preprocessing step, we constructed a matrix (see Definition 3) with the pairwise distance among all pages in the Web site. Using this information we can adjust the respective weights. Let N be a neuron in the network and E the user behavior vector example presented to the network, using definition 4, the page distance vector is:

$$D_{NE} = (d_p(N_p(1), E_p(1)), \ldots, d_p(N_p(H), E_p(H))) \tag{5}$$

Now the adjustment is over the D_{NE} vector, i.e., we have $D'_{NE} = D_{NE} * f_p$, with f_p adjustment factor. Using D'_{NE}, it will be necessary to find a set of pages whose distances with N be near to D'_{NE}. Thus the final adjustment for the winner and its neighbor neurons is given by equation 6.

$$N^{n+1} = \left(N_t^n(i) * f_t, p \in P / D'_{NE}(i) \approx d_p(p, N_p^n(i))\right) \tag{6}$$

4 Applying Data Mart technology

A generic Data Mart architecture is proposed in order to store efficiently the data necessary for the above described Web mining process, i.e., static information describing the content of a page and dynamic information about the time spent on each page.

This section presents a star model [5] for Data Mart repository and explains the transformation process in information.

4.1 Data Mart model

The star model was selected in order to store the information generated by the transformation step. Figure 3 shows the tables composition. This model was implemented using a **Relational Database Management System**.

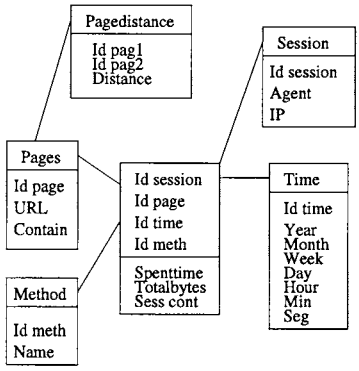

Figure 3: Data Mart model.

In the fact table, three kinds of measure were defined: the time spent in each page, the quantity of real sessions that visited a page and the total bytes transmitted by the user session. These are additive measures and using OLAP techniques [4] we can obtain statistics about user behavior. However, these analysis are only complementary to the use of data mining tools as proposed in this paper.

This star model is composed by four dimension tables. **Time** contains time stamp events, **Pages** the URL and the text in each page. Additionally, a snow flake scheme[5] is shown to store efficiently the distance among Web pages. **Session** table contains the IP and Agent that identify a real session and finally **Method** has the access method used.

4.2 Data transformation process

A Data Staging Area (DSA) [5, 7] was defined in order to process data. Logically, it is composed by two tables: **Weblogs** and **Logclean**. The first one contains data about the Web log registers. It is stored using a code in PHP language directly form the Web log archive. The second one contains the registers after the cleaning and transformation process (see figure 4).

Weblogs	
Ip	varchar2(18)
TimeStamp	date
Method	varchar2(20)
Status	numer(4)
Bytes	number(8)
Url	varchar2(20)
Agent	varchar2(20)

Logclean	
Ip	varchar2(18)
TimeStamp	date
Bytes	number(8)
Url	number(4)
Agent	number(2)
Session	number(4)
Timespent	number(4)

Figure 4: Data Staging Area model.

4.2.1 Web logs processing

The **weblogs** table is loaded with the data about a visitor's session from the respective Web logs, specifically: IP address and agent, Time stamp, Embedded session Ids, Method, Status, Software Agents, Bytes transmitted, Objects required (page, picture, movies, etc).

We only consider registers whose code is not error and the URL parameter link to Web page objects. Other objects, such as pictures, sound, links, etc. are not considered.

In the sessionization process, we used a time oriented heuristic [2]. It considers a maximum time duration given by a parameter, which is usually 30 minutes in the case of total session time, and identified the transactions that belong to a specific session. This process can be realized using tables and program filters. Figure 5 shows the transformation sequence for Web log registers.

IP	Agent
165.182.168.101	MSIE 5.01
165.182.168.101	MSIE 5.01
165.182.168.101	MSIE 5.01
165.182.168.101	MSIE 5.5
165.182.168.101	MSIE 5.5
165.182.168.101	MSIE 5.5
165.182.168.101	MSIE 5.5
204.231.180.195	MSIE 6.0
204.231.180.195	MSIE 6.0
204.231.180.195	MSIE 6.0
204.231.180.195	MSIE 6.0
204.231.180.195	MSIE 6.0

	Date
.....	16 Jun 02 16:39:02
.....	16 Jun 02 16:39:58
	16 Jun 02 16:42:03
	16 Jun 02 16:24:06
.....	16 Jun 02 16:26:05
.....	16 Jun 02 16:42:07
	16 Jun 02 16:58:03
.....	16 Jun 02 16:32:06
	16 Jun 02 16:34:10
	16 Jun 02 16:38:40
	16 Jun 02 17:34:20
.....	16 Jun 02 17:35:45

IP	Agent	Date	Sess
165.182.168.101	MSIE 5.01	16 Jun 02 16:39:02	1
165.182.168.101	MSIE 5.01	16 Jun 02 16:39:58	1
165.182.168.101	MSIE 5.01	16 Jun 02 16:42:03	1
165.182.168.101	MSIE 5.5	16 Jun 02 16:24:06	2
165.182.168.101	MSIE 5.5	16 Jun 02 16:26:05	2
165.182.168.101	MSIE 5.5	16 Jun 02 16:42:07	2
204.231.180.195	MSIE 6.0	16 Jun 02 16:32:06	3
204.231.180.195	MSIE 6.0	16 Jun 02 16:34:10	3
204.231.180.195	MSIE 6.0	16 Jun 02 16:38:40	3
204.231.180.195	MSIE 6.0	16 Jun 02 17:34:20	4
204.231.180.195	MSIE 6.0	16 Jun 02 17:35:45	4

Figure 5: Sessionization process.

Using a pseudo code as shown in figure 6, the registers are grouped by Ip and Agent, sorted by date and stored in a data structure known as "cursor". It is similar to stream load in RAM memory. This code only considers the sessions that have a minimum quantity of registers (Having SQL instruction), for instance if the session has less than three registers, it is not considered a real session.

```
declare cursor log as %
select ip, timestamp, method, agent,bytes, url
from weblogs
where  status not in error_list %clean errors
group by  ip,agent %is possible use more columns
having count(*) > 3
order by timestamp;

open log;
fetch log into old
fetch log into new
session:= last_session_in_data_mart;
totaltime:=0;
numinsert:=0;
```

```
while  log%found  loop
 difftime:=new.timestamp - old.timestap;
 if old.ip = new.ip or old.agent = new.agent
   or  diff_time < 30 then
  insert  old.ip,old.agent,  old.timestamp,
   difftime, session into Logclean;
     totaltime:=totaltime+difftime;
     numinsert:=numinsert+1;
 else %end session's log registers
   insert  old.ip,old.agent, old.timestamp,
   totaltime/numinsert, session into Logclean;
   session++; %count sessions
    totaltime:=0;
    numinsert:=0;
 end if;
  old:=new;
  fetch log into new
  end loop; %main loop
```

Figure 6: SQL pseudo code for sessionization.

Finally, the **Logclean** table contains the log registers grouped by IP and Agent. The url parameter is number, because it uses the same codification as dimension table pages in the Data Mart. If the url exists, the identifier number associated is

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

inserted in the Logclean table. Else a new register is created in the dimensional table and the insert process into Logclean is repeated.

5 Application

In order to proof the effectiveness of the methodology proposed in this work, a Web site was selected (http://www.dii.uchile.cl/~diplomas/). It contains information about education programs for professionals and belongs to the Department of Industrial Engineering of the University of Chile. Its focus is to show the community of professionals the variety of qualification programs, the method of payment, curriculum of the professors, etc.

The site is written in Spanish, has 142 static Web pages and approximately 42.000 Web log registers, corresponding to the period August to October, 2002.

5.1 Applying the proposed Data Mart architecture

The Data Mart model, introduced in figure 3, was implemented in Oracle 9i data base engine and uses the special capabilities of data warehousing projects.

The data staging area shown in the figure 4 was implemented in two tables inside the data base.

Next the transformation stage (word process) was executed by a PHP code and the matrix with the distances among page vectors was loaded. The dimensions of this matrix is 6234×122, i.e., $R = 6234$ and $Q = 122$. With this data, the distancepage table was calculated and stored in the Data Mart.

The transformation stage ends with the sessionization process. This task is implemented in engine's language and considers 30 minutes as the longest user session. Only 7% of the users have sessions with 7 or more pages visited and 11% visited less than 3 pages. Therefore, we supposed three and six as minimum and maximum number of components in a user behavior vector, respectively. Using these filters, we identified 4113 user behavior vectors.

Finally, the load step is executed by a code in engine's language that takes the data in the staging area and stores it in the Data Mart.

5.2 Applying self-organizing feature maps on the Data Mart

The SOFM used has 6 input neurons and 16 output neurons. The thoroidal topology maintains the continuity in clusters, which allows to study the transition among the preferences of the users from one cluster to another.

The data mining tool interacts with the Data Mart through a stream, that is created by the previous execution of a code in PHP, that receives as input the rank of dates to analyze. The training of the neural network was carried out on the same computer where we developed the Data Mart. The time necessary for training was 2,5 hours and the epoch parameter was 100.

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

5.3 Results

Using the period between August and October 2002, we found 24.000 registers in the Data Mart and identified finally 4113 user behavior vectors. These were mapped into the 256 neurons of the SOFM. With this, we can identify six main clusters as shown in table 1. Its second and third column contain the center neurons of each of the clusters, representing the visited pages and the time spent in each one of them.

Table 1: User behavior clusters.

Cluster	Pages Visited	Time spent in seconds
1	(2,15,60,42,70,62)	(3,5,113,67,87,43)
2	(5,43,65,75,112,1)	(4,53,40,63,107,10)
3	(6,47,67,7,48,112)	(4,61,35,5,65,97)
4	(10,51,118,87,105,1)	(5,80,121,108,30,5)
5	(11,55,37,87,114,12)	(3,75,31,43,76,8)
6	(13,57,41,98,120,107)	(4,105,84,63,107,30)

The pages in the Web site were labelled with a number to facilitate its analysis. Table 2 shows the main content of each page.

The cluster analysis shows that the users are interested in the profile of the students, the program and the faculty staff's curriculums (Cluster 1). Additionally, they have preferences for courses about environmental topics and visit the page where they can ask for information (Clusters 2 and 3). Here, program and schedule are very important for the user. Finally, it can be seen that there are users interested in new courses (Cluster 4), in the students profile and the course objectives (Clusters 5 and 6).

Reviewing the clusters found, it can be inferred that the users show interest in the profile of the students, the schedules and contents of the courses and the professors who are in charge of each subject. Based on our analysis we proposed to change the structure of the Web site, privileging the described information.

With these changes, the new Web site's structure unifies the content of all Web pages, showing in their upper part links to particular subjects. Additionally, the first page for each course contains specific information about the objectives, duration and student profile.

The new Web site is working from January 2003 and currently we have the Web logs corresponding to February and March 2003. Based on this data, we can derive only a preliminary analysis, because the market conditions have changed, i.e., the cyclical demand for courses is different. However, analyzing a variable like average time spent by page provides an approximation. For the main page about

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

Table 2: Pages and their content.

Pages	Content
1	Home page
2,...,14	Main page about a course
15,...,28	Presentation of the program
29,...,41	Objectives
42,...,58	Program: Course's modules
59,...,61	Student profile
62,...,68	Schedule and dedication
69,...,91	Instructors staff's curriculums
92,...,108	Menu to solicited information
108,...,121	Information:cost, schedule, etc.
122	News page

courses, the user stay increased from 3.5 to 25 seconds. For the complete session, the average time stay increased from 302 to 381 seconds.

6 Conclusions

A methodology to study the user behavior in a Web site has been introduced, applying as core the Data Mart technology. In the first part, we built the information repository using the star model.

Next we defined a new distance measure based on two characteristics derived from the user sessions: pages visited and time spent in each page. Using this distance in a self organizing map, we found clusters from user sessions.

Since the proposed Data Mart is a basic platform with historical information, it will be possible to analyze the variation in the user behavior using the suggested changes, verify its validity and integrate other tools like OLAP.

It will be necessary to continue applying our methodology to other Web sites in order to get new hints on future developments and develop criteria to evaluate the effectiveness of changes proposed in a site.

Acknowledgement

This work has been funded partially by the Millenium Scientific Nucleus on Complexes Engineering Systems.

Transactions on Information and Communications Technologies vol 29, © 2003 WIT Press, www.witpress.com, ISSN 1743-3517

References

[1] M. W. Berry, S. T. Dumais and G. W. O'Brien, Using linear algebra for intelligent information retrieval, *SIAM Review*, Vol. 37, pages 573-595, December 1995.

[2] R. Cooley, B. Mobasher and J. Srivastava, Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems* Vol. 1, pages 5-32, 1999.

[3] J. Han, Data Mining: An Overview from Database Perspective, *Tutorial in PAKDD Conference*, 1998.

[4] Z. Huang, J. Ng, D. W. Cheung, M. K. Ng and W. Ching, A Cube Model for Web Access Sessions and Cluster Analysis, *Proc. of WEBKDD*, San Francisco CA, August, pages 47-57, 2001.

[5] W. H. Inmon, Building the data warehouse (2nd ed.), *John Wiley & Sons, Inc.*, New York, NY, 1996.

[6] A. Joshi and R. Krishnapuram, On Mining Web Access Logs. *In Proceedings of the 2000 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pages 63-69, 2000.

[7] R. Kimball and R. Merx, The Data Webhouse Toolkit. *Wiley Computer Publisher*, 2000.

[8] T. Kohonen, Self-Organization and Associative Memory, *Springer-Verlag*, 2nd edition, 1987.

[9] J. Velásquez, H. Yasuda, T. Aoki and R. Weber, Voice Codification using Self Organizing Maps as Data Mining Tool. *Procs. of the Second International Conference on Hybrid Intelligent Systems*, Santiago, Chile, pages 480-489, December, 2002.