Testing online navigation recommendations in a web site

Juan D. Velásquez[1] and Vasile Palade[2]

[1] Department of Industrial Engineering,
University of Chile, Chile, jvelasqu@dii.uchile.cl
[2] Computing Laboratory, University of Oxford,UK
Vasile.Palade@comlab.ox.ac.uk

Abstract. An online navigation recommendation system provides the prospective web site visitor with a set of pages that could be of his/her interest. Because the recommendations are given during the user session in the web site, it could be very damaging for the overall business of the company owning the web site, if the recommendations are erroneous. In this paper, we introduce an a priori method to estimate the success of an online navigation recommendation. The methodology was tested in a recommendation system that works with the data generated in a real web site, which proved the effectiveness of our approach.

1 Introduction

An efficient way to improve the relation between the web site and its users is by the personalization of the site structure and content, i.e., "*actions that tailor the web experience to a particular user, or a set of users* " [6, 10].

The web site personalization can be implemented by recommendations done directly to the web site users, which can be web masters, web designers, anonymous web visitors; in short, any users of the web site [5, 13].

Depending on the web user, the recommendations can be:

– Online - Provide the web site visitor, during his/her session in the site, with recommendations about interesting topics, for instance a web page, text contents, etc.
– Offline - Done mainly to the web master, web designers and, in general, to any person in charge of maintaining the structure and the content of the web site.

This paper is concerned with how to do online recommendations, especially those that are relative to the user navigation in the web site.

When a user visits a web site, the browsing behavior is related to what he/she is looking for. Identifying the user's behavior usually becomes a true challenge, because the site doesn't consider the particular needs of each user, e.g. if the user is an amateur in the web and needs assistance to find the desired information.

An online navigation recommendation system helps the user in searching for the desired information, through suggestions about web pages whose content could be of interest to that particular user [13].

Because these kind of systems give the recommendations during the user session in the web site, in some cases, it may happen that, if the user does not like the recommendations done by the system, he/she may decide to leave the site. This is a big problem, and our task of providing online recommendations could be a risky one for the business of the company owning the web site, if the recommendations are not done correctly.

In this paper, we introduce an a priori method to estimate the success of an online navigation recommendation. The paper is organized as follows. In Section 2, a short review about related research work is done. Section 3 shows the main characteristics of the online recommendation system that was used for testing the proposed methodology. This methodology is introduced in Section 4, and in Section 5 its application to a real-world case is presented. Finally, Section 6 presents the main conclusions and future work.

2 Background and Related work

Web personalization aproaches aim to personalize the user interaction with a web-based system, such as a web site, by understanding the user's explicit or implicit interests and desires.

2.1 Classifying the personalization approaches

A good classification of the web personalization approaches, considering functionality reasons, is presented in [8] and outlined below:

Memorization. This is the simplest expression of personalization, and it implies the storing of basic information about the user, such as name and visited pages. When the user visits again the web site, the system recognizes the user and shows the user's name and part of the last visit(s).

Guidance. It represents approaches that assist the user for finding what the user is looking for. In this sense, the personalization systems can recommend links to other pages and related contents.

Task performance support. It involves actions on behalf of the user, like sending e-mails, complete queries, or, in advanced systems, represent the user's interest, for example in a negotiation.

The common requirement in any web personalization system is to be provided with some mechanics to understand the user behavior in a web site. The web usage mining algorithms [3] are the current techniques used for this purpose.

2.2 Web site changes and recommendations

In creating a web site, a complex and meticulous process is followed, in order to get the best look and feel. Consequently, off-line changes in its structure and content are not frequent, even in web sites prone to changes, such as newspapers web sites. Although every day the newspaper's pages change, the structure and the main themes follow an editorial line.

If off-line changes could be a bit risky, the on-line changes are even more, because the visitors may loose the notion about "where they are" in the web site. In fact, some changes can violate the original web site links structure, i.e., there may be no physical link between the actual and the imposed page. When the visitor revisits the web site and wants to review the imposed page again, it may be hard for him/her to get it directly.

In order to avoid the above described problem, some authors [2] have proposed on-line recommendation systems that maintain the web site structure, i.e., there must be a physical link between the actual page and the recommended page. Those recommendations that breake the structure should be considered for off-line changes.

2.3 Creating the online navigation recommendations

Facing a high competition for catching or retaining customers in the digital market, an important way for companies to generate value is to understand, within the web site users sessions, what the users are looking for. In other words, "*users need to feel they have a unique personal relationship with the business*" [4].

A navigation personalization system assists the user by making recommendations about contents that could be of interest for him/her.

Independently of the technology used in the creation of the system, a common element is the representation of the knowledge about the user browsing behavior in the web site [1]. This knowledge can be extracted from usage data, like web log files [5, 14, 13], and from the experience of the web site users, for instance trough questionnaires about the usability of the web site [7].

The online navigation recommendation is created by making a comparison between the current user session and the patterns and rules that represent the knowledge about the user browsing behavior in the system. Then, a set of web page links are showed to the user, who can choose to follow the recommendation(s) or continue with his/her own navigation.

Here, the question is: what happens if the recommendation is erroneous and the user is lost in the hyperspace? Naturally, a bad recommendation would get the relationship between the web site and its users worse, and it represents a high risk for the business.

2.4 Considerations for testing an online recommendation

Usually, the offline structure and content changes in a web site can be reviewed through an "usability test" [7], where a group of selected users review the web site structure and fill a questionnaire with their impressions.

The same methodology could be used in the case of online recommendations using a group of simulated visitors, but it will not be a real situation, as the group of simulated visitors may usually have no incentive to search for some topics and show a real visitor behavior.

In [16], an association rule algorithm is applied for predicting the web log access. The effectiveness test is realized by comparing the prediction with the real navigation of a group of visitors, whose web log access is previously known. The problem with this method is that it needs the visitors' reaction for measuring the effectiveness of the prediction. A similar approach is applied in [17], where the test group is selected based on similar educational background and common interests. The web log registers are generated by giving to the users a search task about a specific topic. Then, the path prediction aims to help the users to find information related to that topic. The problem in this method is that a real test needs real users, i.e., persons with a real intention of searching a topic, without an external influence.

Independent of the navigation recommendation prediction method, the real test will always be when real users visit the web site, which could be very risky for the business. Therefore, an a priori effectiveness test will help improve the recommendation.

3 The online navigation recommendation system

In order to test the proposed methodology, the online recommendation system introduced in a previous paper [12] is used. The main characteristics of this system are shortly detailed below.

3.1 Modeling the user browsing behavior in a web site

Our user behavior model uses three variables: the sequence of visited pages, their text contents and the time spent on each page. The model is based on a n-dimensional visitor behavior vector which is defined as follows.

Definition 1 (User Behavior Vector). *It is defined as a vector $\upsilon = [(p_1, t_1) \ldots (p_n, t_n)]$, where the pair (p_i, t_i) represents the i^{th} page visited (p_i) and the percentage of time spent on it within a session (t_i), respectively.*

3.2 Comparing user sessions

Let α and β be two user behavior vectors of dimension C^{α} and C^{β}, respectively. Let $\Gamma(\cdot)$ be a function that returns the navigation sequence corresponding to a visitor vector. A similarity measure has been proposed elsewhere to compare visitor sessions, as follows [15]:

$$sm(\alpha, \beta) = dG(\Gamma(\alpha), \Gamma(\beta)) \frac{1}{\eta} \sum_{k=1}^{\eta} \tau_k * dp(p_{\alpha,k}, p_{\beta,k}) \tag{1}$$

where $\eta = \min\{C^{\alpha}, C^{\beta}\}$, and $dp(p_{\alpha,k}, p_{\beta,k})$ is a similarity measure for comparing the free text inside two web pages [11], in this case between the k^{th}page of vector α and the k^{th}page of vector β. The term $\tau_k = \min\{\frac{t_{\alpha,k}}{t_{\beta,k}}, \frac{t_{\beta,k}}{t_{\alpha,k}}\}$ is an indicator of the visitor's interest in the visited pages. The term dG is the similarity between the sequences of pages visited by two visitors [9].

3.3 Extracting knowledge from web logs

The knowledge extracted can be represented as patterns, and rules about how to use the patterns. Because the users' behavior can be grouped based on similar preferences, it looks convenient to apply a clustering technique in order to extract the navigation patterns.

For clustering the user sessions, a Self-Organizing Feature Map (SOFM) [14] was applied by using the similarity measure given in Eq. 1. The SOFM requires vectors of the same size. Let H be the dimension of the visitor behavior vector. If a visitor session has less than H elements, the missing components up to H are filled with zeroes. Otherwise, if the number of elements is greater than H, only the first H components are considered.

3.4 Creating the online navigation recommendation

Let $\alpha = [(p_1, t_1), \ldots, (p_m, t_m)]$ be the user behavior vector that corresponds to the current user session and $C_\alpha = [(p_1^\alpha, t_1^\alpha), \ldots, (p_H^\alpha, t_H^\alpha)]$ the closest centroid, such as $\max\{sm(\alpha, C_i)\}$, with C_i the set of centroids discovered. The recommendations are created as a set of pages whose text content is related to p_{m+1}^α.

Let $R_{m+1}(\alpha)$ be the online navigation recommendation for the $(m+1)^{th}$ page to be visited by the user α, where $\delta < m < H$ and δ is the minimum number of visited pages necessary to prepare a suggestion. Then, we can write $R_{m+1}(\alpha) = \{l_{m+1,0}^\alpha, \ldots, l_{m+1,j}^\alpha, \ldots, l_{m+1,k}^\alpha\}$, with $l_{m+1,j}^\alpha$ the j^{th} page link suggested for the $(m+1)^{th}$ page to be visited by visitor α, and k the maximum number of pages that can be suggested. In this notation, $l_{i+1,0}^\alpha$ represents the "no suggestion" state.

4 The proposed methodology for testing online recommendations

Using the discovered clusters as outlined above, we can classify the user browsing behavior into one of them, by comparing the cluster centroid with the current user session, using the similarity measure introduced in Eq. (1).

Here, we propose a method to test the effectiveness of the recommendations, based on the same kind of web data used in the pattern discovery stage [14]. More exactly, a percentage of all available web data is used to extract significant patterns about user behavior, and for these we define a set of rules. Then, we test the effectiveness of the recommendations using the remaining web data.

Let $ws = \{p_1, \ldots, p_n\}$ be the web site and pages that compound it. We can define some equivalence classes of pages, where the pages belonging to the same class contain similar information. The classes partition the web site in disjoint subsets of pages. Let Cl_x be the x^{th} equivalence class for the web site ws. $\bigcup_{x=1}^{w} Cl_x = ws$, where w is the number of equivalence classes.

Let $\alpha = [(p_1, t_1), \ldots, (p_H, t_H)]$ be a user behavior vector from the test set. Based on the first m pages actually visited, the proposed system recommends for the $(m+1)$ page several possibilities, i.e., pages to be potentially visited.

We test the effectiveness of the suggestions made for the $(m+1)^{th}$ page to be visited by the visitor α following this procedure. Let Cl_q be the equivalence class for p_{m+1}; if $\exists l^{\alpha}_{m+1,j} \in R_{m+1}(\alpha)$ and $l^{\alpha}_{m+1,j} \in Cl_q,\ j > 0$, then we assume the suggestion was successful.

The number of recommended pages obtained during the construction of the recommendation could be high, and the user may get confused on which page to follow next. We set in k the maximum number of pages per recommendation. By using the page similarity measure (see Eq. (1)), we can extract from Cl_q the k closest pages to p_{m+1} in the recommendation, as follows:

$$E^k_{m+1}(\alpha) = \{l^{\alpha}_{m+1,j} \in sort_k(dp(p_{m+1}, l^{\alpha}_{m+1,j}))\}. \qquad (2)$$

The "$sort_k$" function sorts the result of similarity measure dp in descending order and extracts the "k" link pages closest to the p_{m+1} page. A particular case is when $E_{m+1}(\alpha) = \{l^{\alpha}_{m+1,0}\}$, i.e., no suggestion is proposed.

The methodology proposed here allows to work with real user sessions and estimate the effectiveness of the online recommendations. It represents an alternative to testing the user answer in front of a navigation recommendation.

5 A real world application

We worked with a recommendation system that used data originated in the web site of the first Chilean virtual bank (www.tbanc.cl). The web data we used were collected between January and March 2003. Approximately eight millions of raw web log registers were collected.

After applying a cleaning and session reconstruction process, approximately 100,000 user behavior vectors were selected.

5.1 Applying clustering techniques

The SOFM used for clustering had 6 input neurons (that is, $H = 6$ in the user behavior vector) and 32 X 32 output neurons with a thoroidal topology in the feature map.

From the extracted user behavior vectors, we only considered those that contained six or more visited pages. With this restriction, around 65000 vectors were used in the experiment. A 75% of these vectors were applied to the SOFM to extract navigation patterns, and the remaining 25% formed the testing set.

The cluster identification is performed by using a visualization tool supported by a density cluster matrix, called winner matrix. It contains the number of times the output neurons win, during the training of the SOFM.

Table 1 contains the 4 discovered cluster centroids, represented by the sequence of visited pages and the the time spent on each centroid. The pages were previously labelled with a correlative number.

Table 1. Visitor behavior clusters

Cluster	Pages Visited	Time spent in seconds
A	(1,5,7,10,135,191)	(30,61,160,110,175,31)
B	(162,157,172,114,105,2)	(3,71,112,110,32,3)
C	(72,87,154,188,140,85)	(8,57,31,3,71,91)
D	(110,104,128,126,31,60)	(25,73,42,65,98,15)

Analyzing the cluster "A", we can see that the users are interested in the content of pages number 10 and 135. These pages contain, respectively, a general description of bank's soft credit products (low interest rate) and information about a specific credit card. Then, we can infer that the users are looking for information about soft credits, i.e., small amount of money to borrow, short period of repayment and low interest rate.

5.2 Representing the knowledge as patterns and rules

Table 2 shows an example about how the recommendation is created. Under the assumption that the cluster more similar to the current user session is C_A, the recommendation for the fourth page to be visited is prepared.

In this example, "S" is a stack implemented using a simple linked list that contains the pages to be suggested for the fourth page to be visited (navigation component), and the usage statistics associated with each of them (statistics component).

If a page that we are recommending does not exit in the current web site anymore (because of previous changes), the function **compare_page** will compare the suggested page with all pages in the entire web site, in order to recommend a page with similar content. The function **Pop(S)** extracts the next element in "S". The final set of pages is the list "L". **Extracted_Three_Links** represents the expression in Eq. 2 and extracts a subset with a maximum of three links, based on the associated statistics. The default is the no suggestion state. **Send** is an instruction for sending the recommendation to the system who will prepare the final web object containing the recommendation for the user.

Table 2. Extracting patterns and rules

C_A $\rightarrow$ [(1,30),(5,61),(7,160),(10,110),(135,175),(191,31)]
α $\rightarrow$ $[(2,10),(11,50),(25,120)]$ % current visitor
ws $\rightarrow$ $\{p_1,\ldots,p_{217}\}$ % current web site pages
S.navigation $\rightarrow$ $\{p_{33},p_{38},p_{41},p_{118},p_{157},p_{201}\}$
S.statistic $\rightarrow$ $\{1.2, 2.1, 1.8, 0.9, 0.8, 0.1\}$
Case C_A and SuggestionPage=4 :
Prepare_suggestion(p_0,0,L); % default "no suggestion"
% L: link page suggestion, 0: statistic associated
while S not null loop
if (S.navigation not in ws) then
S.navigation = compare_page(ws, S.navigation);
else if ((S.navigation <> $\alpha_{p_{1\ldots3}}$) and (S.statistic $> \gamma$)) then
Prepare_suggestion(S.navigation, S.statistic, L);
Pop(S); % Next element in S
end if;
end loop;
send(Extracted_Three_Links(L)); % $L \rightarrow \{p_{38},p_{41},p_{33}\}$

5.3 Results

In Fig. 1, the histogram shows the percentage of the accepted recommendations, using the proposed validation method.

If, by using the proposed methodology, just one page is suggested, slightly more than 50% of the users would accept it. This could be considered a very successful suggestion by the business expert, since we are dealing with a complex web site with many pages, many links between pages, and a high rate of visitors that leave the site after few clicks.

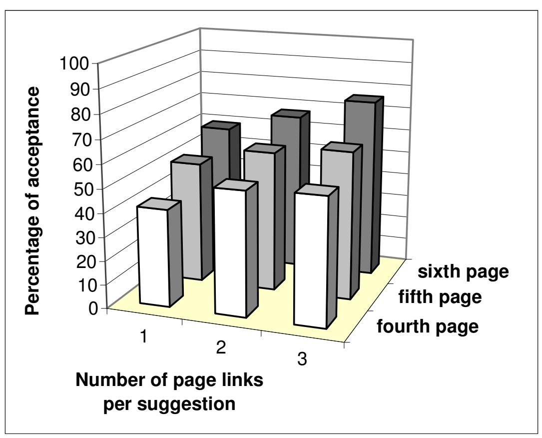

Fig. 1. Percentage of acceptance of online navigation recommendations

Furthermore, it should be mentioned that the percentage of acceptance would probably have been even higher if we actually had suggested the respective page during the real session. Since we compared past visits stored in log files, we could only analyze the behavior of visitors that did not actually receive any suggestion we proposed.

6 Conclusions

We introduced a methodology to a priori test the effectiveness of a system that provides online navigation recommendations for the prospective visitors of a web site.

The methodology used a percentage of all available web data to extract the navigation patterns, and the remaining data to test the effectiveness of the recommendation. With this procedure, an estimation on the success of the recommendation can be calculated.

This methodology was tested in a recommendation system that used data originated in a real web site. There is good evidence to suggest that, if this approach is used, the users will follow the recommendation in a high percentage of them.

As future work, it is interesting to compare our estimation with the real situation, i.e., to apply the recommendation on real users and to analyze its acceptance or rejection, in order to see if the estimation was correct or not.

References

1. P. Brusilovsky. Adaptive web-based system: Technologies and examples. Tutorial, IEEE Web Intelligence Int. Conference, Halifax, Canada, October 2003.
2. F. Coenen, G. Swinnen, K. Vanhoof, and G. Wets. A framework for self adaptive websites: tactical versus strategic changes. In *Procs. in 4th PAKDD Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 1–6, April 2000.
3. M. Eirinaki and M. Vazirgiannis. Web mining for web personalization. *ACM Trans. Inter. Tech.*, 3(1):1–27, 2003.
4. A. Kobsa, J. Koenemann, and W. Pohl. Personalised hypermedia presentation techniques for improving online customer relationships. *Knowledge Engineering Review*, 16(2):111–155, 2001.
5. B. Mobasher, R. Cooley, and J. Srivastava. Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8):142–151, 2000.
6. B. Mobasher, T. Luo, Y. Sung, and J. Zhu. Integrating web usage and content mining for more effective personalization. In *Procs. of the Int. Conf. on E-Commerce and Web Technologies*, pages 165–176, September, Greenwich, UK, 2000.
7. J. Nielsen. User interface directions for the web. *Communications of ACM*, 42(1):65–72, 1999.
8. D. Pierrakos, G. Paliouras, C. Papatheodorou, and C. D. Spyropoulos. Web usage mining as a tool for personalization: A survey. *User Modeling and User-Adapted*, 13:311–372, 2003.
9. T. A. Runkler and J. Bezdek. Web mining with relational clustering. *International Journal of Approximate Reasoning*, 32(2-3):217–236, Feb 2003.

10. S. Sae-Tang and V. Esichaikul. Web personalization techniques for e-commerce. *Lecture Notes in Computer Science*, 2252(1):36–44, 2001.
11. G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM archive*, 18(11):613–620, November 1975.
12. J. D. Velásquez, P. Estévez, H. Yasuda, T. Aoki, and E. Vera. Intelligent web site: Understanding the visitor behavior. *Lecture Notes in Artificial Intelligence*, 3213(1):993–1003, 2005.
13. J. D. Velásquez, R. Weber, H. Yasuda, and T. Aoki. Acquisition and maintenance of knowledge for web site online navigation suggestions. *IEICE Transactions on Information and Systems*, E88-D(5):993–1003, May 2005.
14. J. D. Velásquez, H. Yasuda, T. Aoki, and R. Weber. Acquiring knowledge about users's preferences in a web site. In *Procs.* 1^{th} *IEEE Int. Conf. on Information Technology: Research and Education*, pages 375–379, Newark, New Jersey, USA, August 2003.
15. J. D. Velásquez, H. Yasuda, T. Aoki, and R. Weber. A new similarity measure to understand visitor behavior in a web site. *IEICE Transactions on Information and Systems, Special Issues in Information Processing Technology for web utilization*, E87-D(2):389–396, February 2004.
16. H. Yang, S. Parthasarathy, and S. Reddy. On the use of constrained association rules for web mining. In *Procs. Int. Conf. WebKDD workshop on Knowledge Discovery in the Web*, pages 77–90, Edmonton, Alberta, Canada, 2002.
17. T. Zheng and R. Goebel. An ncm-based framework for navigation recommender systems. In *Procs.* 19^{th} *Int. Conf. Artificial Intelligence workshop Multi-Agent Information Retrieval and Recommender Systems*, pages 80–86, Edinburgh, Scotland, UK, July 2005.