Relational fuzzy approach for mining user profiles

G. CASTELLANO, A. M. FANELLI, M. A. TORSELLO
Department of Informatics
University of Bari
Via Orabona, 4 – 70126 Bari
ITALY

Abstract: - Capturing the characteristics and preferences of Web users into user profiles is a fundamental task to perform in order to implement forms of personalization on a Web site. In this paper, we present a relational fuzzy clustering approach to extract significant user profiles from session data derived from log files. In particular, a modified version of the CARD clustering algorithm is proposed in order to produce well distinct clusters corresponding to profiles reflecting the actual user preferences embedded in the available session data. Experimental results on session data extracted from log files of a sample Web site are reported.

Key-Words: - Web Personalization, user profiling, fuzzy clustering, relational clustering, session similarity measure.

1 Introduction

The rapid development of the World Wide Web as a medium for information dissemination has generated a growing interest in the domain of Web personalization [1]. In this context, the knowledge acquired from the analysis of the user's navigational behavior (usage data) can be conveniently exploited in order to customize the Web information space to the necessities of users. As a consequence, tools capable to automatically identify user profiles modeling the preferences of different user categories are becoming a fundamental component of Web personalization systems. Once user preferences are understood by analyzing the derived user profiles, personalized services can be provided to each user (such as, sending targeted advertisement to the connected users, adapting the content/structure of the Web site to the user needs, providing a guide to the user navigation, etc.). Different techniques have been investigated in literature for the identification of Web user profiles. In particular, among all the proposed methods in the area of Web Usage Mining (WUM) [2], unsupervised clustering techniques have been widely applied to categorize user sessions derived from log data into groups of users exhibiting similar preferences, e.g., users accessing to similar pages. Grouping users into categories according to usage data is not an easy task. Web usage data are characterized by vagueness, imprecision and uncertainty. Moreover, user categories are rarely well separated, since a user can exhibit interests characterizing different user categories. Hence, it is reasonable to allow access patterns to belong to several categories with different degrees instead of belonging to only one group. On the basis of these considerations, traditional clustering techniques result inadequate to extract user profiles expressing the real user navigational behavior patterns. Conversely, fuzzy clustering techniques seem to be particularly suited in this context because they can partition data into overlapping clusters. Moreover, since the actual number of user categories visiting a Web site is not known in advance, clustering algorithms capable of automatically determining the number of clusters are especially required to perform user profiling.

In this paper we apply a relational fuzzy clustering technique to discover user profiles from web usage data. These profiles are intended to be exploited in a Web personalization system that dynamically suggests links to Web pages retained interesting for the user. To this aim, we firstly preprocess Web log files in order to collect usage data and derive user sessions. Then a modified version of the Competitive Agglomeration Relational Data (CARD) algorithm [3] is applied to group similar user sessions into clusters, where each cluster represents a user profile, i.e. a group of users with similar navigational behavior. The CARD algorithm works on relational data representing the similarity values of all pairwise sessions, where the similarity measures is not necessarily based on the Euclidean distance. In the proposed modified CARD, the similarity between two sessions is defined in terms of the access time on pages common to both sessions. One main feature of the original CARD is the ability to automatically determine the number of clusters by removing low-cardinality clusters. However, in our experience, the CARD algorithm may fail in finding the actual

number of clusters underlying the data since it produces a redundant partition of data that includes clusters with a high overlapping degree (very low inter-cluster distance). The proposed modified CARD overcomes this limitation, by adding a post-clustering process that fuses very overlapping clusters into a single cluster. This enables the creation of well distinct clusters that correspond to actual user profiles reflecting the preferences of distinct categories of users.

The rest of paper is organized as follows. In section 2 we describe how user sessions are derived from log data. Section 3 concerns the creation of user profiles by the modified CARD algorithm. Section 4 shows experimental results on session data extracted from log files of a sample Web site. In Section 5, we close the paper giving conclusive remarks.

2 User session identification by log data preprocessing

User session identification is the process of analyzing usage data in order to extract useful information concerning user navigational behavior by structuring the requests contained into the Web log files. Access log files of a Web site consist in text files where the server stores all the accesses made by the users in chronological order. According to the Common Log Format, each log entry includes: the user's IP address, the request's date and time, the request method, the URL of the accessed page, the data transmission protocol, the return code indicating the status of the request, the size of the visited page in terms of number of bytes transmitted. Based on such information, we can determine the user sessions, i.e. the sequence of URLs that each user has accessed during his/her visit. To identify user sessions, we used LODAP (Log Data Preprocessor), a software tool that we previously presented in [4]. LODAP performs this task into three principal steps: data cleaning, data structuration and data filtering. Precisely, Web log data are cleaned from useless and irrelevant information in order to retain only the entries corresponding to the effective user requests (which can be effectively exploited to recognize user navigational behavior). Specifically, requests with an access method different from "GET", failed and corrupt requests, requests for multimedia objects (such as images, videos, sounds, etc.), visits made by Web robots are removed from log files. Next, log entries are structured into user sessions. Here, a user session is defined as the finite set of URLs accessed by a user within a predefined time period (in our work, 25 minutes). Since the information about the user login is not available, user sessions are identified by grouping the requests originating from the same IP address during the established time period. Finally, data are filtered in order to retain only the most relevant pages and user sessions. At the end of preprocessing, we obtain collection of n_S sessions denoted by the set $S = \langle s_1, s_2, ..., s_{n_S} \rangle$. Each session contains information about accesses to pages during the session time. Precisely, a user session is formally described as a triple $\mathbf{s}_i = \langle u_i, t_i, \mathbf{p}_i \rangle$ where u_i represents the user identifier, t_i is the access time of the whole session, $\mathbf{p}_i$ is the set of all pages (with corresponding access information) requested during the i-th session. Namely:

$$\mathbf{p}_i = \langle (p_{i1}, t_{i1}, N_{i1}), (p_{i2}, t_{i2}, N_{i2}), ..., (p_{i,n_i}, t_{i,n_i}, N_{i,n_i}) \rangle$$

with $p_{ij} \in P$, where N_{ij} is the number of accesses to page p_{ij} during the i-th session and t_{ij} is the total time spent by the user on that page during the i-th session.

3 User profiling by relational fuzzy clustering

A natural way to categorize user sessions into user profiles is to apply a clustering approach. There are two major classes of clustering techniques: those that work with object data and those that work with relational data (i.e. a set of similarity or dissimilarity values between data) [5]. Since data describing sessions are not numeric in nature, we adopt a fuzzy relational clustering approach to cluster sessions into categories. Particularly, in this work, we propose a modified version of the CARD (Competitive Agglomeration Relational Data) clustering algorithm. CARD was proposed in [3] as an extension of the Competitive Agglomeration [6] algorithm to work on relational data.

Before to describe the details underlying the adopted clustering approach, we proceed with the definition of the relation matrix containing the dissimilarity values between all session pairs which is required in the common relational clustering approaches [7].

To evaluate the (dis)similarity between two generic sessions, we define a measure that takes into account the average time that the user spent on viewing each site page during his/her visit. According to the definition of user session, the number of Web pages accessed by different users may vary. As a consequence, any two different session vectors s_i and

s_l may have different dimension. In order to create a homogeneous model for all sessions, we need to create vectors with the same number of components. Being n_P the number of different pages required in all the user sessions, we model the navigational behavior of a user during session s_i through a vector $\mathbf{b}_i = \left(b_{i1},...,b_{in_P}\right)$ where:

$$b_{ij} = \begin{cases} t_{ij} / N_{ij} & \text{if page } p_j \text{ is accessed in session } s_i \\ 0 & \text{otherwise} \end{cases}$$

Summarizing, we represent the session data as a $n_s \times n_p$ matrix $\mathbf{B} = \left[b_{ij}\right]$ where each entry represents the average time spent on the j-th page during the i-th session. Based on this matrix, we define the similarity between any two sessions s_i and s_l as follows:

$$Sim\left(s_i, s_l\right) = \frac{\sum_{j=1}^{n_P} b_{ij} \cdot b_{lj}}{\sqrt{\sum_{j=1}^{n_P}\left(b_{ij}\right)^2 \cdot \sum_{j=1}^{n_P}\left(b_{lj}\right)^2}} \qquad (1)$$

where $\sum_{j=1}^{n_P}\left(b_{ij}\right)^2$ is the square sum of the time spent by the user during session s_i and $\sum_{j=1}^{n_P} b_{ij} \cdot b_{lj}$ is the inner-product over the time spent on common pages visited in s_i and s_l. In order to compute the relation matrix, we simply derive the dissimilarity measure between two sessions as:

$$Diss\left(s_i, s_l\right) = 1 - Sim\left(s_i, s_l\right) \qquad (2)$$

Finally, the estimated dissimilarity values between all session pairs are mapped in a $n_S \times n_S$ matrix $\mathbf{R} = \left[Diss\left(s_i, s_l\right)\right]_{i.l=1...n_s}$ representing the relation matrix. Based on this matrix, sessions of users with similar preferences can be clustered together into user profiles by means of a relational clustering approach. To accomplish this, we implemented a modified version of the CARD algorithm. As common relational clustering approaches, CARD obtains an implicit partition of the object data by deriving the distances from the implicit objects (relational data) to a set of C implicit prototypes that summarize the data objects belonging to each cluster in the partition. Specifically, starting from the relation matrix $\mathbf{R}$, the following implicit distances are computed at each iteration step of the algorithm:

$$d_{ci} = \left(\mathbf{R}\mathbf{v}_c\right)_i - \mathbf{v}_c\mathbf{R}\mathbf{v}_c / 2 \qquad (3)$$

for all sessions $i = 1,...,n_s$ and for all implicit clusters $c = 1,...,C$, where $\mathbf{v}_c$ is the membership vector for the c-th cluster, defined as on the basis of the fuzzy membership values u_{ci} that describe the degree of belongingness of the i-th session in the c-th cluster. Once the implicit distance values d_{ci} have been computed, the fuzzy membership values u_{ci} are updated to optimize the clustering criterion, resulting in a new fuzzy partition of sessions. The process is iterated until the membership values stabilize.

Finally, a crisp assignment of sessions to the clusters is performed in order to derive a profile vector for each cluster. Precisely, each session is crisply assigned to the closest cluster, creating C clusters:

$$\chi_c = \left\{s_i \in \mathbf{S} \mid d_{ci} < d_{ki} \forall c \neq k\right\}$$
$$1 \le c \le C$$

Then, for each cluster χ_c a profile vector $\mathbf{x}_c = \left(x_{c1}, x_{c2}, ..., x_{cn_p}\right)$ is derived, where

$$x_{cj} = \frac{\sum_{s_i \in \chi_c} b_{ij}}{|\chi_c|}, j = 1,...,n_P \qquad (4)$$

The values x_{cj} represent the significance (in term of access time) of a given page p_j to the c-th profile. The main feature of CARD is the ability to automatically determine the number of clusters. The algorithm starts with a high number $C_{\max}$ of clusters and progressively reduces this number by removing, in each iteration step, low-cardinality clusters, i.e. all clusters having cardinality lower than a threshold. In [3] the authors state that the CARD algorithm is insensitive to the initial number of clusters $C_{\max}$ and converge always to the same final partition. However, in our experience, the CARD algorithm often provides different final partitions for different values of $C_{\max}$, thus failing in finding the actual number of clusters buried in data. Indeed, we have observed that CARD produces redundant partitions, with clusters having a high overlapping degree (very low inter-cluster distance). To overcome this limitation, we add a post-clustering process to CARD in order to remove redundant clusters. Precisely, for each pair of clusters, we compute the inter-cluster distance, defined as:

$$D_{ck} = \sum_{s_i \in \chi_c} \sum_{s_l \in \chi_k, i \neq l} \frac{\left\|\mathbf{b}_i - \mathbf{b}_l\right\|^2}{|\chi_c||\chi_k|} \qquad (5)$$

A high value of the inter-cluster distance means that the two clusters are separated, while a very low value means that clusters are very overlapping. To avoid redundant clusters, we apply the following heuristic: if the inter-cluster distance between two clusters drops below a small threshold $\varepsilon \in [0,1]$, the two clusters are fused together into a single cluster that

embraces sessions of both clusters. The addition of this heuristic leads to a modified CARD algorithm, that is able to produce always the same partition of session data, independently on the initial number of clusters $C_{\max}$. The derived clusters are sufficiently separate and correspond to actual user profiles reflecting the preferences of distinct users embedded in the available session data.

4 Experimental results

The proposed modified CARD was applied to derive user profiles from the access log data of a sample Web site. After the preliminary activity of log data preprocessing, a number of 434 user sessions were identified out of the initial 1025 sessions and 10 distinct pages were retained among the 25 pages composing the Web site. We indicate the selected pages by $p_1, p_2, \ldots, p_{10}$. Starting from these session data, we derived the matrix B containing the access time for each page in each session. Next, using the dissimilarity measure defined in section 3, we created the relational matrix $\mathbf{R}$.

Firstly, we applied the original CARD algorithm to such relational data. We performed several runs by setting different initial number of clusters (Cmax=30, 25, 20, 15, 10, 6). To evaluate the validity of the clustering results, we considered the Dunn's index [8] which is widely used in literature to evaluate how much clusters are compact and well separated. The Dunn's index D is defined as:

$$D = \min_{1\le c\le C}\left\{\min_{1\le k\le C}\left\{\frac{D_{ck}}{\max_{1\le k\le C}\{\Delta(\chi_k)\}}\right\}\right\} \qquad (6)$$

where C is the number of clusters, D_{ck} represents the inter-cluster distance between clusters χ_c and χ_k, while $\Delta(\chi_k)$ represents the intra-cluster distance of cluster χ_k. The goal is to maximize inter-cluster distances whilst minimizing intra-cluster distances. Hence, large values of Dunn's index correspond to a good cluster partition. Figure 1 shows the Dunn's index for the partitions obtained by CARD, together with the final number of clusters obtained in each trial. It can be seen that the algorithm produces different partitions for different values of Cmax. In some cases (Cmax=15, 10, 6) the CARD algorithm does not reduces at all the number of clusters. Moreover, all the derived partitions, except the one obtained with Cmax=6, contain cluster that overlap too much, as demonstrated by low values of the Dunn's index.

Next, we applied the CARD algorithm modified with the proposed post-clustering process. Namely, once the clustering was complete, all couples of clusters having inter-cluster distance lower than $\varepsilon = 1$ were joined into a single cluster. For each trial, independently on the initial number Cmax of clusters, the modified CARD algorithm provided a final partition of session data into 6 clusters, with inter-cluster distance values around 4.5. The details concerning one of the derived partitions are summarized in table 1, where, for each cluster, the cardinality and the intra-cluster distance are displayed. It can be seen that the derived clusters are very compact, as demonstrated by the low values of intra-cluster distance. For each cluster, the corresponding profile is summarized in table 1 in terms of most visited pages (pages with highest access time). A qualitative analysis of these profiles made by designer of the considered Web site confirmed that they correspond to actual user categories reflecting distinct user interests.

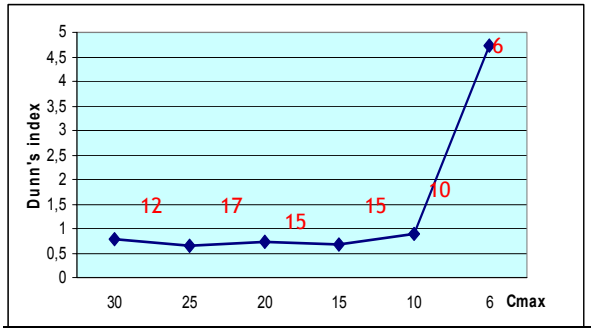

Figure 1. Results of the CARD algorithm

Table 1. Clusters and user profiles derived by modified CARD

	Relevant pages (and access time)	$\lvert\chi_c\rvert$	$\Delta(\chi_c)$
1	p_3 (0.83), p_6 (0.83), p_9 (0.76)	98	0.15
2	p_1 (0.87), p_8 (0.83)}	70	0.11
3	p_4 (0.88), p_7 (0.87), p_{10} (0.83)	98	0.09
4	p_5 (0.89), p_{10} (0.84)	91	0.11
5	p_1 (0.84), p_9 (0.74)	35	0.23
6	p_2 (0.88), p_9 (0.82)	42	0.10

4 Conclusion

We presented a fuzzy relational clustering approach to derive user profiles from session data. The approach uses a modification of the CARD algorithm in order to obtain a partition of the data that well represent the interest of distinct users of a Web site. Experimental results have shown that CARD is very

sensitive to the choice of the initial number of clusters and produces a redundant partition of data. Conversely, the proposed version of CARD is much more robust since it succeeds in finding always the same partition of session data, independently on the initial number of clusters. The derived clusters are sufficiently separate and correspond to actual user profiles that capture the preferences of distinct users embedded in the available session data.

The derived user profiles can be used for many web personalization applications. In particular, we are currently working to develop a recommendation system that dynamically suggests interesting links to a user according to his profile.

References:

[1] O. Nasraoui,"World Wide Web Personalization," in J. Wang (ed), Encyclopedia of Data Mining and Data Warehousing, Idea Group, 2005.

[2] D.G. Pierrakos, G. Paliouras, C. Papatheodorou, and C.D. Spyropoulos, "Web usage mining as a tool for personalization: A survey," User Modeling and User-Adapted Interaction, 13:311–372, 2003.

[3] O. Nasraoui and H. Frigui, "Extracting Web user profiles using relational competitive fuzzy clustering," Int. Journal on Artificial Intelligence Tools, 9(4):509-526, 2000.

[4] G. Castellano, A.M. Fanelli, A. M., Torsello, "LODAP: a LOg DAta Preprocessor for mining Web browsing patterns," Proc. of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Base (AIKED 2007), pp. 12-17, Corfu, Greece, February 16-19, 2007.

[5] O. Nasraoui, R. Krishnapuram, A. Joshi, and T. Kamdar, "Automatic Web User Profiling and Personalization using Robust Fuzzy Relational Clustering," in "E-Commerce and Intelligent Methods" in the series "Studies in Fuzziness and Soft Computing", J. Segovia, P. Szczepaniak, and M. Niedzwiedzinski, Ed, Springer-Verlag, 2002.

[6] H. Frigui and R. Krishnapuram, "Clustering by competitive agglomeration", Pattern Recognition, 30(7):1109-1119, 1997.

[7] W. Pedrycz, "Fuzzy relational clustering", Knowledge-based clustering, Ed. Pedrycz, 2005.

[8] M. Halkidi, Y. Batistakis, M. Vazirgiannis, "Cluster Validity Methods:Part II", in SIGMOD Record, September 2002.