Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 33 (2007) 743–753
Expert Systems with Applications
www.elsevier.com/locate/eswa

An approach based on the Adaptive Resonance Theory for analysing the viability of recommender systems in a citizen Web portal

José D. Martín-Guerrero [a,*], Paulo J.G. Lisboa [b], Emilio Soria-Olivas [a], Alberto Palomares [c], Emili Balaguer [c]

[a] *Digital Signal Processing Group, Electronics Engineering Department, University of Valencia, Spain*
[b] *The Statistics and Neural Computation Research Group, School of Computing and Mathematical Sciences, Liverpool John Moores University, United Kingdom*
[c] *Tissat, S.A., R&D Department, Spain*

Abstract

This paper proposes a methodology to optimise the future accuracy of a collaborative recommender application in a citizen Web portal. There are four stages namely, user modelling, benchmarking of clustering algorithms, prediction analysis and recommendation. The first stage is to develop analytical models of common characteristics of Web-user data. These artificial data sets are then used to evaluate the performance of clustering algorithms, in particular benchmarking the ART2 neural network with K-means clustering. Afterwards, it is evaluated the predictive accuracy of the clusters applied to a real-world data set derived from access logs to the citizen Web portal *Infoville XXI* (http://www.infoville.es). The results favour ART2 algorithms for cluster-based collaborative filtering on this Web portal. Finally, a recommender based on ART2 is developed. The follow-up of real recommendations will allow to improve recommendations by including new behaviours that are observed when users interact with the recommender system.

Keywords: Adaptive Resonance Theory; User profiling; Web citizen portal; Recommender systems

1. Introduction

Web mining has become an important research area from the 90s. This is because the huge popularity of the Web and the wide range of possibilities that it offers. One of the most important research efforts within Web mining is that related with finding interesting characteristics and patterns of the Web users and their usage of the Web. The importance of this kind of Web mining is that if users are correctly profiled, then it is possible to understand their behaviour in the portal, and in turn, to provide suitable services for them (Fu, Shandu, & Shih, 1999), especially where this can successfully anticipate demand by individual users. In fact, the study and development of personalized recommender systems is a very active field of research (Carberry, 2001), and some recommender systems become an important part of some Web sites providing e-commerce services, for instance, Amazon.com (http//www.amazon.com) and its subsidiary "CDNow" (http//www.cdnow.com). There are two main automatic approaches for recommendations which have been extensively tested and are scale-up to large amount of data, namely collaborative filtering and content-based (Zukerman & Albrecht, 2001) recommender systems.

Collaborative filtering is among the most widely used technologies today. These recommender systems aggregate ratings or other indicators of interest for web objects, such as frequency of access, to find user similarities based on indicator profiles and thus finally offer recommendations

* Corresponding author. Tel.: +34 963160198; fax: +34 963160466.
E-mail address: jose.d.martin@uv.es (J.D. Martín-Guerrero).

0957-4174/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2006.06.013

for new pages, services or products. Well-known recommender systems include GroupLens/NetPerceptions (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994), Ringo/Firefly (Shardanand & Maes, 1995), and Recommender (Hill, Stead, Rosenstein, & Furnas, 1995). The greatest strength of collaborative techniques is that they are independent from any machine-readable representation of the objects being recommended and they work appropriately for complex objects (for instance, music and movies) where variations in taste are responsible for much of the variation in preferences, sometimes called "people-to-people correlation" (Schafer, Konstan, & Riedl, 1999).

Content-based learning is used when a user's past behaviour is a reliable indicator of his/her future behaviour. Content-based models are particularly suitable for situations in which users tend to exhibit idiosyncratic behaviour. However, this approach requires a system to collect relatively large amounts of data from each user in order to enable the formulation of a statistical model. Typical examples of systems of this kind are text recommendation systems like the newsgroup filtering system, NewsWeeder (Lang, 1995) which uses words from its texts as features. This kind of learning, where the recommender learns a profile of the user's interests based on the features present in objects that the user has rated, is called "item-to-item correlation".

In this paper, we focus on people-to-people collaborative recommendation since it seems to be a more appropriate technique for citizen Web portals since our aim is to find inter-user similarities rather than idiosyncratic behaviours of individual users. In particular, our approach consists of profiling users' behaviour by using clustering algorithms, thus finding groups of similar users, and afterwards, recommending those objects in which the users will likely be interested in; this knowledge about users' tastes is extracted from the analysis of the services that are usually accessed by the users of the same group. The approach of user modelling, – by means of clustering algorithms or other techniques, – as a first stage of a collaborative recommender system is not unusual; an example of this kind of systems is Moonranker, a free access recommender of music, movies and books (Zhou, Weston, Gretton, & Schölkopf, 2003).

In particular, we propose a four-stage methodology to develop and evaluate a clustering recommender (Martin et al., 2006): user model, clustering algorithms' comparison, prediction analysis and recommendation (Fig. 1). Other recent works, such as (Geyer-Schulz & Hashler, 2002) also propose similar steps for evaluating a recommender. The main difference between our approach and that presented in Geyer-Schulz and Hashler (2002) comes from the third stage of our methodology, which is novel. The proposed methodology starts with a user model which produces artificial data sets which, in the second stage of the methodology, serves to evaluate clustering algorithms' performance, in order to benchmark the predictive accuracy of the algorithms for the different data sets.

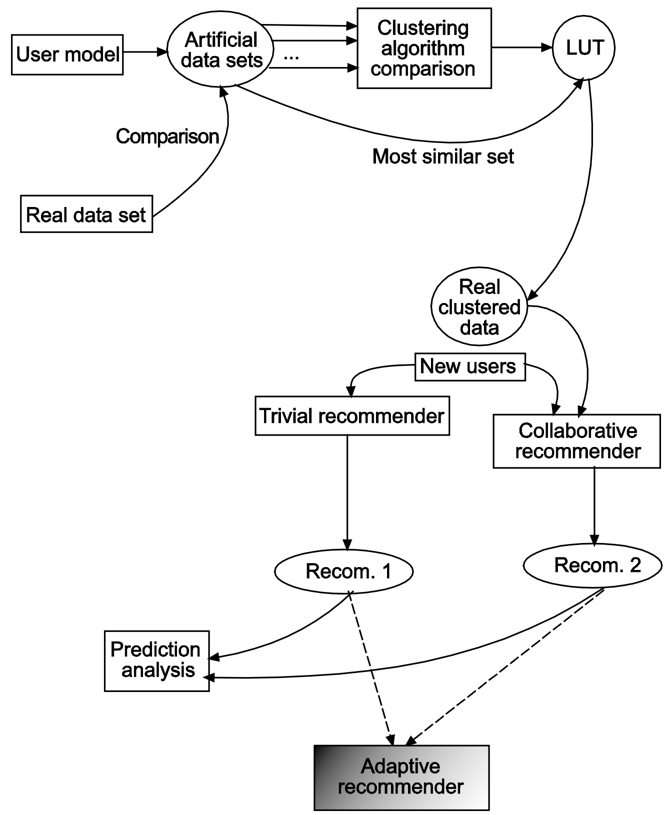

Fig. 1. General schematic of the proposed methodology for the design of cluster-based collaborative filters for web portals. Ellipses indicate the results of a previous step of the user model, and the dashed arrows and the gray-shaded square show a planned extension of the methodology (LUT means look-up table).

The benchmarking of clustering algorithms, focuses on the use of neural networks based on the Adaptive Resonance Theory network, ART2 (Carpenter & Grossberg, 1987, 1991) which applies to continuous data. This network can circumvent some of the usual drawbacks of classical algorithms, having been designed to solve the *stability–plasticity* dilemma, namely, the ability to adapt clusters to new data patterns, without disrupting the already established clusters. This feature can support online tracking user profiles, although this is not tested in the current study. In practice, this works by identifying the most appropriate cluster for a given user pattern, then testing whether the cluster prototype is a good-enough representation of the user pattern, which generates clusters with similar distance distributions but of different sizes, whereas the K-means algorithm (Duda, Hart, & Stork, 2001) tends to create clusters of similar sizes but potentially with large differences in distance distributions from one cluster to another. The ART2 algorithm is tuned with the desired degree of similarity, or maximum separation among patterns from the same group, rather than by prespecifying the overall number of clusters in the data. A novel component of the proposed methodology is the use of artificial data created from a sample of user accesses in order to refine and tune the clustering algorithms which, in this study, demonstrated accuracy improvements gained using the neural network model.

The third and crucial stage of our methodology is related to the evaluation of the viability of recommendations with a real data set. Once the Web users have been clustered, we compare the suitability, or more precisely, the prediction accuracy of a collaborative recommender that utilizes ART2 clustering with another collaborative recommender based on K-means clustering and with another recommender that only recommends the most likely object of the Web site that has not yet been accessed. If there is a considerable improvement when using a clustering recommender, then we can assume that it is actually useful taking into account inter-user similarities for recommendations. These prediction capabilities are called "implicit votes" in Breese, Keckerman, and Kadie (1998). It is important to point out that we do not measure the influence of the recommendations on the users, which is a phenomenon studied in many recent works (Baudisch & Brueckner, 2002; Kim, Ok, & Woo, 2002; Lee, Choi, & Woo, 2002; McNee, Lam, Konstan, & Riedl, 2003). Instead, we study the capability of the clustering algorithm for profiling user behaviour. In fact, our methodology predicts those objects which are accessed by the user without receiving any recommendations. Therefore, our methodology also enables the influence of the user interface of the recommendation to be separated from the effects of the knowledge extracted by our approach. Yet, once the recommendation system is implemented, it is important to follow up on the success of real recommendations, which will in general be different. In fact, it is logical to think that the success of real recommendations will be better than the success of our prediction analysis. This is because the presentation of attractive items should affect user behaviour positively (Cosley, Shyong, Albert, Konstan, & Riedl, 2003; McNee et al., 2003).

Most approaches usually skip the third stage, but we think that it is absolutely necessary as a preliminary step in the development of a recommender system. It enables us to measure how good the clustering is in terms of profiling user behaviour. It can be particularly interesting in certain Web portals, in which it is risky to develop a recommender without analysing its possible effectiveness, because of the expense involved in such development. The analysis of the effects of real recommendations is the fourth and last stage of the development of a recommender system.

The remainder of the paper is outlined as follows. Section 2 presents the data sets used in this study. Section 3 analyses our proposal for clustering recommendation. Section 4 shows the results achieved in this study, analysing the clustering achieved with the different data sets as well as our study to evaluate the feasibility of a future recommendation system. A discussion about the work is carried out in Section 5, and we present some conclusions and discuss some proposals for further work in Section 6.

2. Data sets

2.1. Artificial data sets

2.1.1. A user model of Web accesses

Web mining tools must be applicable to real data sets. However, the use of artificial data sets also becomes very important because of the following reasons:

- *Artificial data sets enable us to choose the most appropriate clustering method with real data.* This is a major reason for the use of artificial data sets. Before the real application of an algorithm, a rigorous analysis of its performance should be carried out. When dealing with real data, the desired clusters are not usually available a priori; hence it is difficult to determine whether the clusters found by the algorithm are right or wrong. However, when an artificial data set is created in a controlled situation, the clusters that must be found by the algorithms are defined in advance, thus allowing an analysis of the algorithms' performance.
- *Generalization to Web sites with different characteristics.* Web mining tools should be capable of working properly on different Web sites, covering heterogeneous user behaviours. Few real data sets that record user accesses are available because there are more and more restrictive data protection laws and also because of the confidentiality of the Web user data kept by the majority of companies. Still, a set might be available, but it would correspond to a particular site, so that if a clustering analysis is carried out on this set, it would only be valid for this site and those sites that have a very similar structure. However, artificial data sets can be used to carry out experiments with different site characteristics.

In this work, artificial data sets are generated by a Web user model which is capable of providing a wide range of scenarios. This user model takes into account some of the characteristics and constraints that can be observed in real *log* files (Andersen et al., 2000; Balaguer & Palomares, 2003; Breslau, Cao, Fan, Phillips, & Shenker, 1999; Su, Ye-Lu, & Zhang, 2000), namely:

- The number of users who log in a new session, i.e., those who access the site, decreases as the number of previously logged-in sessions increases.
- In each session, fewer users access a service (a service is any one of the possible objects that can be clicked on from a Web portal) when the number of previously consulted services increases.

These two characteristics are similar to modelling according Zipf's Law (Breslau et al., 1999). Assuming an exponential decrease (Fig. 2), the quantity of users N that access a certain number of services x in the yth session can be obtained from the expression:

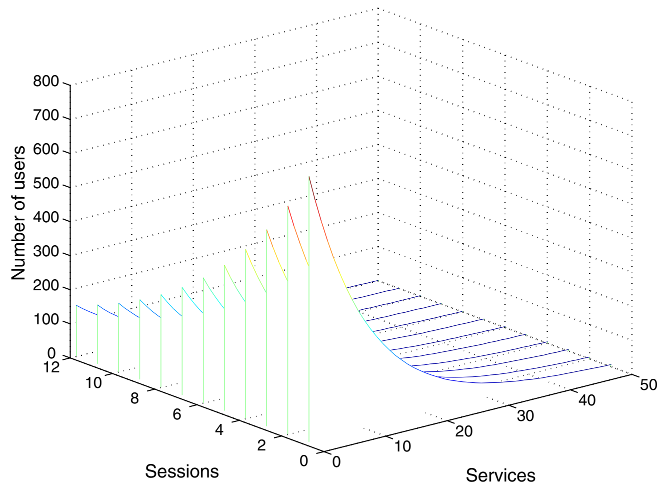

Fig. 2. A simulated Web site with 50 services and 12 sessions is represented. An exponential decrease of the number of users with respect to the logged-in sessions and the clicked services is shown.

$$N = N_{\mathrm{M}} \cdot \mathrm{e}^{-(\alpha\cdot(x-1)+\beta\cdot(y-1))} \quad (1)$$

where N_{M} is the maximum number of users (those logging in the first session and accessing at least one object), and α and β are constants whose values determine the slope of the exponential decrease. Fig. 2 shows these restrictions for a particular case generated by the user model. In Fig. 3(a), the percentage of users vs the number of logged-in sessions and (b) the percentage of users vs length of sessions in a real citizen Web portal are shown. A strong similarity between the simulated restrictions and the real conditions can be observed.

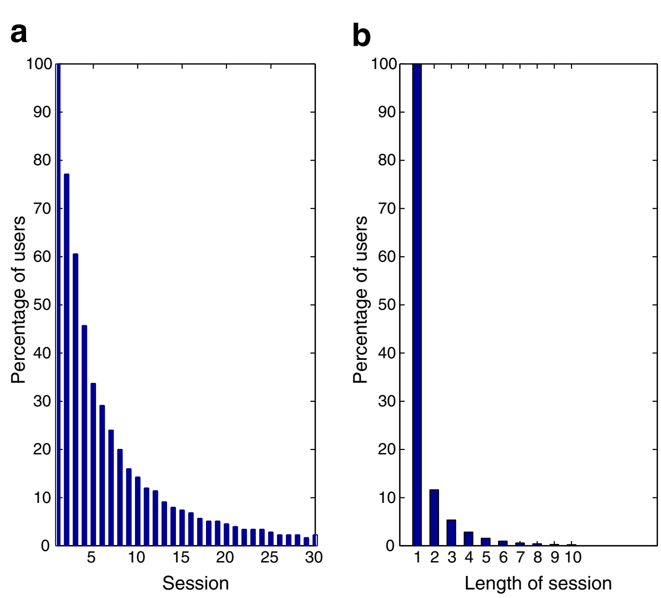

Fig. 3. Histograms (normalized to percentages) representing accesses to the citizen Web portal *Infoville XXI* (http://www.infoville.es). (a) Represents the percentage of users vs the number of logged-in sessions; (b) Represents the percentage of users vs the length of the session, i.e., the number of clicked services within a session.

The user model works in a space of reduced dimensionality because it can be very difficult to find useful inter-user similarities in a space of high dimensionality. Since the quantity of objects that can be clicked on in a Web portal may be very large, it is not recommended to generate users in a space defined by services; it is preferable to do it in a reduced space instead. It must be taken into account that working with approximately the same or even fewer users than the dimensionality of the space is useless in terms of knowledge discovery. Also, inter-user similarities cannot be found in such a space, either. Therefore, we defined some labels that gather several services with similar characteristics, which led to a lower dimensionality space. These labels are often called "page categories" or "descriptors"; for instance, in an electronic newspaper, one can consider several pages or objects that are grouped under subject labels like "Sport", "Politics" and so on (Cadez, Heckerman, Meek, Smyth, & White, 2001). However, since descriptors may be unavailable in some cases, the user model offers information about users in a space defined by services as well.

The user model consists of two main parts, as shown in Fig. 4: first, sets of users are generated in a descriptor space, providing a vector for each user. The components of these vectors indicate the a priori probability of accessing the descriptors. After this step, the service accesses can be obtained from the relationship between labels and services, and also from the constraints of the user model. Information about label and service accesses is coded into two tensorial matrices. In Fig. 4, T_{D} is a tensor that records accesses to the different descriptors in each session. Its dimension is $N \times N_{\mathrm{D}} \times N_{\mathrm{Smax}}$, where N is the number of users, N_{D} the number of descriptors and N_{Smax} the maximum number of sessions that can be logged-in by the same user. Let us consider an example to understand the storage of data in T_{D}. Assume a portal whose $N_{\mathrm{D}} = 3$, and that we want to know the accesses corresponding to user #9 in his/her fourth session. This information is stored in the components $(9,k,4)$ of the tensor T_{D}, where $k = 1,2,\ldots,N_{\mathrm{D}}$. If, for instance, $T_{\mathrm{D}(9,k,4)} = [3,2,2]$, it means that user #9 has accessed seven objects during his/her fourth session, three of which correspond to descriptor D_1, two to D_2 and the other two to D_3. Moreover, T_{S} is the tensor that records

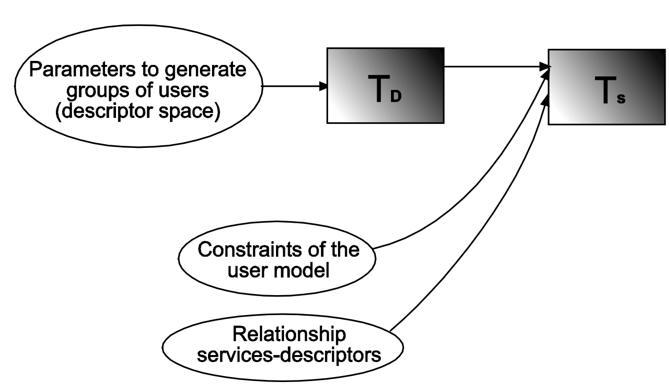

Fig. 4. Block diagram showing the stages of the user model.

accesses to the different services of the portal in each session. In this case, the dimension of the tensor is $N \times L_{\max} \times N_{\mathrm{Smax}}$, where $L_{\max}$ is the maximum length of a session, i.e., the maximum number of services that can be clicked on in only one session. If, analogously to the previous example, we want to know the services accessed by user #9 during his/her fourth session, the result might be $T_{\mathrm{S}(9,l,4)} = [43, 27, 2, 6, 22, 19, 5, 0, \ldots, 0]$, where $l = 1, 2, \ldots, L_{\max}$. It means that in his/her fourth session user #9 has clicked on service #43 first, and then on #27, #2, #6, #22, #19 and #5. Therefore, the last selected service is #5, and the user ends his/her navigation in the portal during the fourth session in service #5. The vector is completed with zeros in order to store efficiently the data of users with click-streams of different lengths.

2.1.2. Characteristics of the artificial data sets

Six artificial data sets were selected in order to test the clustering algorithms. They represent common situations that can occur in Web portals since they have been derived from empirical Web portal access data (Balaguer & Palomares, 2003), and follow characteristics that are similar to other sets used in the literature (Banerjee & Ghosh, 2002; Ghosh, Strehl, & Meregu, 2002). The clusters were assumed to follow a normal distribution, so they could be described by the location of their centroids and their covariance or their standard deviation matrix. The artificial data sets were generated in a space defined by the probability of access to descriptors. The main characteristics of each data set are presented in Table 1.

Data set #1 is a very simple data set, with just two clusters in a space defined by two descriptors. In contrast to the other data sets, it is not inspired in real-Web-portal-access data, but serves purely as a baseline to test the clustering ability in a simple task. Data set #2 is considerably more complex, especially because the clusters are very close to each other, showing a high overlap. Data sets #3 and #4 are similar since they consist of eight groups of users in a space of five descriptors; the difference between them stems from the overlap, which is higher in the case of data set #4. Finally, data sets #5 and #6 represent accesses to Web portals in a high dimension, since eight descriptors are taken into account. Data sets with a higher number of descriptors were also created. However, clustering algorithms showed very similar results to those obtained with data sets #5 and #6. Therefore, they were not selected for benchmarking of clustering algorithms. In other words, data sets #5 and #6 are a good-enough representation of high-dimensional data sets.

Table 1
Main characteristics of the artificial data sets

	N_{D}	N_{C}	Overlap
Data set #1	2	2	No
Data set #2	3	4	High
Data set #3	5	8	Slight
Data set #4	5	8	High
Data set #5	8	12	High
Data set #6	8	12	Slight

N_{D} represents the number of descriptors and N_{C} the number of clusters. Moreover, the degree of overlap among the different clusters is also shown (we consider a slight overlap when less than 20% of the patterns are overlapped, whereas a high overlap means that more than 20% of the patterns are overlapped among different clusters).

2.2. A real data set: accesses to the Web portal *Infoville XXI*

2.2.1. Characteristics of the data set

In spite of simulated data sets are very useful for carrying out an analysis about algorithm's performance in different situations, real data become absolutely necessary as a final test. In this work, we focus on citizen Web portals, an interactive gateway between citizens and the public administration. They involve citizens in the Information Society by offering a growing number of services on the Internet, creating a new model for service delivery to the public as a result of the interaction between the basic services provided by the Government and private entities, which ends up at the citizen who made the request. The success and acceptance of these portals depend largely on their ability to attract the citizens, and the public and private entities in the area. Finding out inter-user similarities and, in turn, creating groups of users with similar tastes helps in the customization of the portal. This is an easy way to make the site attractive to the majority of the people. In this work, the suitability of customization is analysed by means of a recommender. This analysis provides information about the possible benefits of carrying out such a customization.

We profiled user accesses to the region Web portal *Infoville XXI*, http://www.infoville.es. This is an official Web site supported by the Valencian Government, which provides citizens from Valencia, Spain, with more than 2000 services, grouped into 22 descriptors, namely, public administration, agenda/events, children's area, town councils, street maps, channels (this descriptor consists of information on four specific matters: education, job-hunting, setting up a business and housing), shopping, Infoville community (this descriptor enables the communication of people who access the portal by e-mail, fora, postcards, bulletin boards, etc.), Infoville diary, education and training, finance, information for citizens, internal, register (internal and register are descriptors used for administration purposes), Lanetro (local information about where to eat, drink, dance,...), SMS messages, entertainment, electronic newspapers, tourism in Valencia, national and international tourism, searcher and user utilities (personal agenda, site customization, personal Web page, helping guide,...). Furthermore, the term *Infoville*, which was coined by the Generalitat Valenciana,[1] is currently part

[1] Generalitat Valenciana is the name of the autonomous government of Valencia.

of an European project. In fact, this term is used for citizen Web portals from Germany, Italy, England, Denmark and France.

2.2.2. Preprocessing

We have used accesses from June 2002 to February 2003. The data recorded consists of user ID, session ID and service ID, together with the date and time corresponding to each access. A preprocessing procedure was carried out to eliminate data which did not provide useful information for our goals, and also to build sets for clustering and analysis of the recommendations. This preprocessing procedure involved the following steps:

- *Removing administrators*. The administrators of the portal create a great number of fictitious users for test purposes. These users are useless in terms of knowledge discovery and, therefore, they were eliminated from the data set.
- *Removing anomalous users.* Those users who accessed the site only once in all the months included in the study can be considered as lost users, and therefore, they were removed from the data set. Besides, more than 95% of the users logged in fewer than 30 sessions, being removed those users who accessed the portal more than 30 times.
- *Removing high and low accessed descriptors.* Since the clustering is carried out in the descriptor space, it is important to analyse the information provided by the descriptors. Those descriptors that record a very low number of accesses should be removed because they do not contain an important amount of information. Descriptors that record a very high number of accesses should also be removed, since they can bias the clustering considerably. After this preprocessing procedure, six descriptors were eliminated, with 16 descriptors remaining in the data set. It must be emphasized that these descriptors were removed for clustering tasks, but the services that belonged to them were all taken into account for recommendation.
- *Removing users who logged in fewer than three times.* Those users who logged in less than three times were removed from the data set, since it would be difficult for the clustering algorithms to find similarities among users with so little information. The final number of users after the preprocessing procedure, was 4800 users.
- *Final preparation for clustering.* Accesses were encoded in a probability notation in order to be processed by the clustering algorithms. Furthermore, data was split into two sets: a first set was used to carry out the clustering (it consisted of 17,404 accesses corresponding to the first half of the months taken into account) and a second set was used to analyse recommendations (14,079 accesses corresponding to the second half of the period of time taken into account). The latter analyses whether a recommendation based on the clustering achieved would match the actual services accessed by users. It must be emphasized that this second data set was not used at all for clustering purposes, hence, it enabled us to carry out a recommendation evaluation, and, in turn, to show the robustness of the clustering achieved.

3. Recommendations based on clustering

3.1. Clustering with ART

The ART model was originally proposed by Carpenter and Grossberg (1987) to model fast adaptive learning in the initial stages of human visual processing. Hence it is termed an artificial neural network. In its initial form, ART1, the model applied only to clustering of binary vectors. It remains among few clustering methods specifically designed for quantized data. The model was then extended to continuous-valued vectors in ART2 (Carpenter & Grossberg, 1991). These networks cluster inputs by using unsupervised learning.

ART operates as a two stage process. Each time a pattern is presented, an appropriate cluster unit is chosen, and that cluster's weights are adjusted to let the cluster unit learn the pattern. The weights on a cluster unit are considered to be a prototype for the patterns assigned to that cluster. The second and crucial stage of the recognition process is to test whether the prototype forms and adequate representation of the input pattern. Once a good-enough winning prototype has been selected, the process is referred to a vigilance test. From this, either the prototype is updated to form a running average of the input vector, or a new prototype is initiated.

As a computational tool, ART networks allow the user to control the degree of similarity of patterns placed on the same cluster; once this choice is done, it is not necessary to choose the number of clusters in advance, but the network finds the number corresponding to the degree of similarity chosen. During training, each data pattern is presented several times. A pattern may be placed on one cluster unit the first time it is presented and then placed on a different cluster when it is presented later (due to changes in the weights for the first cluster if it has learned other patterns in the meantime). A stable network will not return a pattern to a previous cluster, i.e., a pattern oscillating among different cluster units at different stages of training indicates an unstable network. Some self-organized neural network models achieve stability by gradually reducing the learning rate as the same set of training patterns is presented many times (Kohonen, 1997). However, this does not enable the network to learn rapidly a new pattern that is presented for the first time after a number of training epochs have already taken place. The ability of a network to respond to a new pattern equally well at any stage of learning is called *plasticity*. ART networks are designed to be both stable and plastic.

In this work, we used ART2 network[2] first to cluster patterns from the artificial data sets presented earlier. Since artificial data sets enable us to analyse clustering performance, we benchmark the clustering achieved by ART2 with that obtained by using the classical K-means. As it is shown later, ART provides much better results, thus showing its capabilities to cluster this kind of data sets. Afterwards, ART2 was also applied to cluster users from the Web portal *Infoville XXI*; since in this case we are working with real data, evaluation of the clustering is carried out by studying the meaning of the clusters found and also analysing the success of recommendations based on clustering (prediction analysis).

3.2. Procedure of recommendations

Clustering of users of the citizen Web portal Infoville XXI was used to carry out a kind of collaborative filtering, i.e., the most likely service of the user group was recommended, provided that this service had not yet been accessed. Services based on clustering could not be recommended for the first accesses, since there was not enough information to assign users to a certain cluster. Instead, the most likely services of the portal were recommended for these first accesses, providing that they had not yet been accessed.

After the recommendations, a test was performed to determine whether or not users actually clicked on the recommended object; since ART clustering was much more accurate than K-means clustering in all scenarios, recommendations should be based on ART. The success ratio (SR) achieved in the prediction of the accessed services by using an ART clustering was benchmarked with that SR obtained by recommending only the most likely service of the whole portal that had not yet been accessed. Therefore, the effectiveness of our methodology was measured in terms of the improvement in the SR with respect to a methodology which did not use clustering.

Although the clustering is carried out in a space defined by the probability of accesses to descriptors, the analysis of viability of developing a recommender was carried out in a service domain. This analysis involved a two-step process. First, for the m first accesses of a user, the most probable service of the whole portal not previously accessed by this user was considered for recommendation. Second, in the nth access to the portal ($n \geqslant m+1$), the previous $n-1$ accesses were used to measure the distance between user's behaviour and the clusters found by the algorithm, selecting the one which shows the minimum distance as the *winner* cluster. After this, the most likely service within the winner cluster is chosen (provided that it has not yet been accessed), and considered for recommendation. We consider a success to be when the object considered for recommendation is actually clicked on. We consider different values of m, and also different values of l, being l the number of accesses for which we analyse the prediction ($l \geqslant m+1$).

4. Results

4.1. Clustering of artificial data sets

In order to evaluate the clusters achieved, two approaches were taken into account. On the one hand, we considered whether or not the number of clusters found by the algorithm was correct, and, on the other, how good these clusters were.

Therefore, we compared the number of prototypes found by the algorithms with the correct number that we knew in advance, and afterwards, the goodness of the clustering was measured by the Mahalanobis distance from each cluster found by the algorithm to the nearest known cluster' centre.[3] The advantage of using this distance measure is that it takes into account the covariance of the group, hence it does not depend on the shape of the cluster. Any cluster whose Mahalanobis distance from the nearest known cluster centre was >1, did not match the corresponding centre properly, and was removed from the set of correct groups. Finally, we took account of the number of patterns in each cluster in the final measure to evaluate the quality of the clustering:

$$D = \frac{1}{N}\sum_{i=1}^{M} N_i d_i \tag{2}$$

In Eq. (2), D provides information about the distance from the cluster found to the nearest actual centre. The smaller the value of D, the closer the match to the known cluster. N is the whole number of patterns, M the number of correct clusters found, N_i the number of patterns belonging to the ith cluster found, and d_i the Mahalanobis distance from the ith cluster found to the corresponding centre.

The number of clusters found by K-means and ART2 are compared in Fig. 5. When the dimensionality is low, similar results were achieved by both algorithms, but ART2's behaviour was much better when dealing with a high number of clusters in a high dimensionality space. It is important to emphasize that K-means had information about the number of clusters in advance, which ART2 did not have. Nevertheless, ART2 achieved better results than K-means clustering. This seems to be an advantage of the two-stage similarity used by ART2, which successfully filters similarities among data, the final number of clusters being a natural result of these similarities; on the other hand, K-means tries to find a certain number of groups, which mix natural clusters or break them up with unnecessary intermediate clusters.

The values of the parameter D for the six artificial data sets, using K-means and ART2 networks are shown in

[2] A detailed procedure of the algorithm is shown in Appendix A.

[3] The distance was measured in the space defined by the frequency of accesses to descriptors.

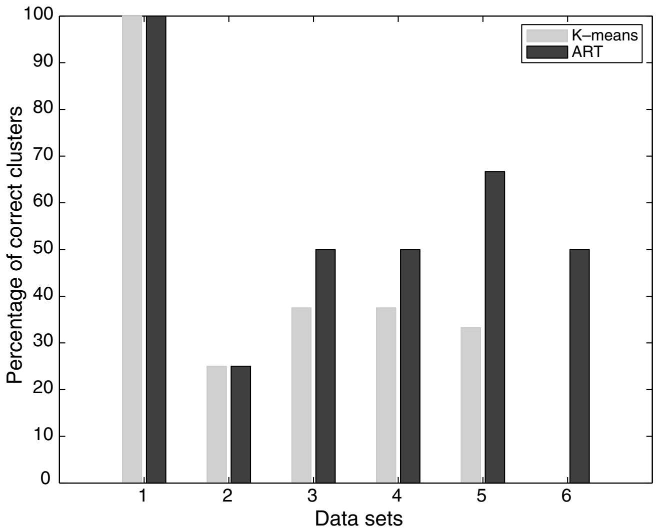

Fig. 5. Percentage of correct clusters found by the ART2 network and by the K-means algorithm. The six artificial data sets are represented in the x-axis.

Table 2. This measure is used to determine the goodness of the clustering together with the percentage of right clusters found. Apart from the simple data set #1, the empirical outcomes obtained with ART2 are clearly better, since D had a smaller value except with data sets #3 and #4; nevertheless, the percentage of correct clusters found by ART2 with these data sets was considerably higher than those found by K-means (50% and 66.7% by ART2, and 37.5% in both cases by K-means). Therefore, the conclusion is that K-means works more or less well with a small number of groups but ART2 best captured the structure of the known clusters in the tests with artificial data. It is important to point out that with ART2 the values of D were very similar for the different data sets (except #1, which was very simple), indicating the robustness of this algorithm, since it was able to find correct clusters for different dimensionalities and actual numbers of clusters. Obviously, an overall assessment of the algorithm must also take account of the percentage of correct clusters.

A final test of the algorithms' robustness was carried out by analysing the normality of the clusters achieved, given that the artificial data sets were generated with multivariate Gaussian distributions. For this purpose we can use measures of *skewness* and *kurtosis* (Hair, Anderson, Tatham, & Black, 1998). Skewness is a measure of symmetry, or more precisely, the lack of symmetry. On the other hand, kurtosis is a measure of whether the data are peaked or flat relative to a normal distribution. A statistical test based on skewness and kurtosis values was carried out, testing for a normality to a confidence of 95.5%. All clusters found by either method for data sets #1 and #2 were consistent with normality. However, with the higher dimensional data sets, K-means showed a high percentage of non-normal clusters of 25% and 50% compared with 8% and 25% for ART2, respectively. This indicates that ART2 more closely captured normality of the artificial clusters.

Table 2
Normalized Mahalanobis distance between the actual centres and the correct clusters found by the K-means algorithm and an ART2 network

	K-means	ART2 network
Data set #1	0.0330	0.0330
Data set #2	0.6818	0.2492
Data set #3	0.1498	0.2314
Data set #4	0.2227	0.2747
Data set #5	0.2858	0.2701
Data set #6	–	0.2411

The distances are measured in the descriptors' probability space.

4.2. Preliminary clustering of accesses to Infoville XXI

First, a preliminary study was carried out just to know the capabilities of the algorithms to find useful and understable clusters for this citizen Web portal. This issue was assessed by selecting a small group from the available descriptors. A reduced data set (November 2002–January 2003) was used. First, the access frequencies of each descriptor were analysed to remove those descriptors that provided the slightest information. From the remaining descriptors, five were selected by Tissat, S.A.[4] as the most significant ones: public administration, town councils, channels, shopping and entertainment. This resulted in 1676 users for this study.

The results were analysed in terms of the interpretability of the clusters obtained. This was possible because the clustering was done in a five-dimension space, in which the meaning of all the components was known. The clustering achieved by K-means was not easy to understand, and the clusters did not represent logical behaviours of people, indeed. However, the ART2 clustering was quite straightforward, since they clustered the data into seven different groups: five of them were clearly focused on each one of the five different descriptors, whereas the other two clusters contained people who were interested in the leisure items of the portal or in the administrative ones. In particular, one of the clusters was centred between the descriptors "shopping" and "entertainment". Therefore, it clustered individuals who mainly accessed the portal for leisure purposes. The other cluster was centred between the descriptors "public administration" and "town councils", and it also presented a small membership to the descriptor "channels". Therefore, people clustered in this group clearly accessed the portal for administrative purposes. These seven clusters demonstrate two important facts: on the one hand, ART2 seems to be suitable as a clustering tool for this portal; on the other hand, the usefulness of the portal is clearly demonstrated, since it was basically designed to accomplish these two requirements, i.e., to accelerate administrative paperwork, and to provide a fast gateway for the leisure interests of citizens.

[4] Tissat, S.A. is the company responsible for developing the portal.

4.3. Final clustering and viability of recommendations in *Infoville XXI*

Finally, we clustered the data set formed by the users of *Infoville XXI* described in Section 2.2. The results of clustering achieved with artificial data sets, and also the preliminary clustering of accesses to *Infoville XXI* both suggest the use of ART2 as clustering tool. However, in order to carry out a last comparison we also clustered these data set by using K-means. Since the clusters were not known in advance, the evaluation described for the artificial data sets could not be carried out, nor was it feasible to analyse the interpretability of the groups obtained due to the high-dimensional space in which the clustering was performed. The evaluation of the clustering could be assessed by means of analysing the success of the recommendations based on this clustering. This is an approach which can be used not only to evaluate the clustering, but also, to study the feasibility of a recommender before its actual implementation.

Clustering was used to carry out a kind of collaborative filtering, i.e., the most likely service of the user group was recommended, provided that this service had not yet been accessed. Services based on clustering could not be recommended for the first accesses, since there was not enough information to assign users to a certain cluster. Instead, the most likely services of the portal were recommended for these first accesses, provided that they had not yet been accessed.

Afterwards, a test was performed to determine whether or not users actually click on the recommended object. Finally, the success ratio (SR) achieved in the prediction of the accessed services by using our methodology (collaborative recommendation based on clustering) was benchmarked with that SR obtained by recommending only the most likely service of the whole portal that had not yet been accessed (Naïve–Bayes recommendation). Therefore, the effectiveness of our methodology was measured in terms of the improvement in the SR with respect to the methodology which did not use clustering. ART2 yielded a clustering formed by 12 groups of users, which corresponded with a vigilance parameter $\rho = 0.8$; slight differences in this value led to a considerably different number of clusters. Therefore, we considered 12 groups as a natural number of clusters for this data set, and hence, we assumed 12 groups for K-means clustering, as well.

The average success ratio (ASR) over the 14,076 accesses used for the evaluation is benchmarked in Table 3 for different values of m (number of accesses needed to carry out a prediction) and l (depth of the prediction) and for ART clustering recommendation and a naïve recommendation. Results with K-means were not included since they were similar to naïve recommendations, and much worse than those obtained by ART2, as it was expected from the results of the previous tests with artificial data sets and with a reduced version of the real data set. It can be observed that our methodology based on using ART2 clustering information yields higher ASRs than the methodology that does not take into account clustering information. As more accesses are used to cluster, better results are obtained; this is expected, since the information gathered by the clustering algorithms is more extensive. Besides this, the importance of the clustering appears to be more relevant in the first accesses starting from the $(m + 1)$th one; as the number of accesses increase, the difference between using clustering information or not becomes smaller. Therefore, clustering appears to be particularly important in the first accesses of the users, when they must be attracted in order to establish their loyalty to the portal.

Table 3
Average success rate, ASR (%) measuring the goodness of service prediction as a preliminary step in the development of a recommender

m	l	No clustering	ART2 clustering
2	4	6.91	12.84
2	5	10.12	14.57
2	6	13.07	16.48
2	7	16.13	18.74
3	4	3.47	13.73
3	5	7.04	15.11
3	6	10.31	16.81
3	7	13.70	18.97
4	6	7.56	18.06
4	7	11.32	19.94
5	7	8.16	20.94

Prediction with and without clustering is benchmarked for different values of m and l.

5. Discussion

Recommender systems are one of the most prolific fields of research and publication of user modelling. In this work, we focus our efforts on recommendation systems for Web sites, although their application to other areas is also possible with some small changes. A good recommender is undoubtedly useful since users can achieve the objects searched for in less time, or even better, find something interesting that they would not have found by themselves. It is also useful for the company which exploits the site, since obvious economical profits can be obtained from useful recommendations. Finally, a good recommender also provides an indirect benefit, which is the improvement of the Web site.

However, the development of such systems is not easy, and in addition, it may involve a high economic investment. Until now, recommender systems used to be developed and then evaluated; in this work, we propose an approach which consists of carrying out an evaluation of the intended methodology for the design of the recommender system before being implemented in order to analyse its feasibility and to tune its performance. It optimises predictive accuracy over a range of artificial data models, before testing the system on retrospective test data. If this prediction works, then the users have been successfully profiled. Besides, it is logical to believe that a recommender system using a similar strategy would work even better, since attractive recommendations can affect user behaviour, making the users click on such recommendations. Therefore, the success obtained with the prediction could be interpreted as a lower threshold of the success that can be obtained with a similar recommender system.

In particular, we have benchmarked a prediction based on using information about clustering with one that predicts the most likely service of the portal that has not yet been accessed by the recommended user. More specifically, we use clustering of users to classify new users in a certain group, thus finding out which service will be the most likely for this user. In order to make the prediction useful from a recommendation point of view, it is important not to predict/recommend services already accessed by users, since these objects are already known by them, and therefore, they do not provide new information about the portal, which is one of the most important goals of a recommender system.

ART2 and K-means have been benchmarked in some artificial data sets, which have been obtained by a user model; the data sets represent different kinds of Web usage sites. Moreover, these algorithms have also been benchmarked in a real data set, consisting of accesses to the Web portal *Infoville XXI*. All these tests show that ART is a far more suitable technique than the classical K-means.

The results show that using ART2 clustering information provides a much better prediction, showing success rates which are approximately double the rates obtained with respect to prediction using K-means clustering or without any kind of clustering information. Although it might seem obvious, the authors want to point out that the users used for clustering are different from the users used to evaluate the prediction. This demonstrates the robustness of the clustering achieved, and the relevance of the information provided by it. Moreover, the percentage of success in the recommendation can be considered as very important and relevant, since typical recommenders tend to yield percentages of acceptance considerably lower (Geyer-Schulz & Hashler, 2002), and in addition, these results should be understood as a lower threshold of the success that can be obtained with a similar recommender system, actually.

Part of the approach proposed in this work is already implemented in the software iSUM® (http://www.isum.com/), and nowadays, the implementation of all the methodology is being considered.

6. Conclusions

A novel approach to evaluate the viability of implementing a recommender in Web portals is proposed. The first step of the proposed methodology is to cluster user data based on simulations in order to ensure that the collaborative filter is robust across a range of user models. These results demonstrated the predictive accuracy of cluster-based recommender systems using the ART2 neural network algorithm applied to profiles of simulated user data. This predictive accuracy then supports the offer of services that are new to the user. The results on a retrospective sample of real-world data show a considerable improvement with respect to either a prediction based on K-means clustering or a prediction which does not take into account clustering information, indicating that the proposed methodology will add value to the design of cluster-based collaborative recommender systems for users of the citizen information Web portal studied.

The proposed methodology has generic applicability to other Web portals, which include anticipated growth areas, for instance, interactive TV, where the user model would have to be re-estimated.

Future work will be dedicated to carrying out a follow-up of real recommendations once these data are available. This follow-up should be used to improve the recommender system, since feedback of actual recommendations can be used to adapt the system, potentially with adaptive on-line profiling.

Acknowledgements

This work has been partially supported by the research project entitled "NDPG (Neuro-Dynamic Programming Group): Aplicaciones prácticas de programación dinámica y aprendizaje reforzado en minería web y marketing", with reference number GV05/009.

Appendix A. ART2 algorithm

Let

E^k	kth input pattern
p	the dimension of the training examples and prototypes
α	positive number $\leqslant 1/\sqrt{p}$
β	small positive number
θ	normalization parameter, with $0 \leqslant \theta \leqslant 1/\sqrt{p}$
ρ	vigilance parameter, with $0 \leqslant \rho \leqslant 1$

0. Preprocess all training examples using threshold θ.
   - 0a. Normalize all E^k.
   - 0b. Replace every component E_j^k that is $\leqslant\theta$ by 0.
   - 0c. Renormalize all E^k.
1. Start with no prototype vectors.
2. Perform iterations until none of the training examples cause any change in the set of prototype vectors; at this point quit because stability has been achieved. For each iteration take the next training example, E^k, chosen in cyclic order.
3. Find the prototype P_i (if any) not yet tried during this iteration that maximizes $P_i \cdot E^k$.
4. Test whether P_i is sufficiently similar to E^k:

$$P_i \cdot E^k \geqslant \alpha \sum_j E_j^k ?$$

4a. If not then:
 4aa. Make a new cluster with prototype set to E^k.
 4ab. End this iteration and return to step 2 for the next example.

4A. If sufficiently similar, then test for vigilance acceptability:

$$P_i \cdot E^k \geqslant \rho$$

 4Aa. If acceptable then E^k belongs in P_i's cluster. Modify P_i to be more like E^k

$$P_i = \frac{(1-\beta)P_i + \beta E^k}{\|(1-\beta)P_i + \beta E^k\|}$$

and go to step 2 for the next iteration with the next example.

 4AA. If not acceptable, then make a new cluster with prototype set to E^k and return to step 2 for the next example.

References

Andersen, J., Larsen, R. S., Giversen, A., Pedersen, T. B., Jensen, A. H., & Skyt, J. (2000). Analyzing clickstreams using subsessions. Technical report TR-00-5001, Department of Computer Science, Aalborg University.

Balaguer, E., & Palomares, A. (2003). AI recommendation engine of tissat, S.A. Internal Technical Report, Tissat, S.A. Valencia, Spain.

Banerjee, A., & Ghosh, J. (2002). Characterizing visitors to a Web site across multiple sessions. In *Proceedings of NGDM'02: National science foundation workshop on next generation data mining*, Baltimore, USA.

Baudisch, P., & Brueckner, L. (2002). TV scout: lowering the entry barrier to personalized TV program recommendation. In *Proceedings of the 2nd international conference AH2002* (pp. 58–68), Malaga, Spain.

Breese, J. S., Keckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th annual conference on uncertainty in artificial intelligence* (pp. 43–52).

Breslau, L., Cao, P., Fan, L., Phillips, G., & Shenker, S. (1999). Web caching and zipf-like distributions: evidence and implications. *Proceedings of INFOCOM 1999, 1*, 126–134.

Cadez, I., Heckerman, D., Meek, C., Smyth, P., & White, S. (2001). Model-based clustering and visualization and navigation patterns on a web site technical report MSR-TR-0018, Microsoft Research, Microsoft Corporation.

Carberry, S. (2001). Techniques for plan recognition. *User Modeling and User Adapted Interaction, 11*, 31–48.

Carpenter, G. A., & Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics and Image Processing, 1137*, 54–115.

Carpenter, G. A., & Grossberg, S. (1991). *ART2: Self-organization of stable category recognition codes for analog input patterns. Pattern recognition by self-organizing neural networks*. MIT Press.

Cosley, D., Shyong, K. L., Albert, I., Konstan, J., & Riedl, J. (2003). Is seeing believing? How recommender system interfaces affect users' opinions. In *Proceedings of the ACM SIGCHI conference on human factors in computing systems, 2003*. Minneapolis, USA.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. John Wiley and Sons.

Fu, Y., Shandu, K., & Shih, M. (1999). Fast clustering of web users based on navigation pattern. In *Proceedings of SCI'99/ISAS'9*, Orlando, USA.

Geyer-Schulz, A., & Hashler, M. (2002). Evaluation of recommender algorithms for an Internet information based on simple association. In *Proceedings of WEBKDD'02* (pp. 110–114), Edmonton, Canada.

Ghosh, J., Strehl, A., & Meregu, S. (2002). A consensus framework for integrating distributed clusterings under limited knowledge sharing. In *Proceedings of NGDM'02: National science foundation workshop on next generation data mining* (pp. 99–108), Baltimore, USA.

Hair, J. F., Anderson, R. E., Tatham, R. L., & Black, W. C. (1998). *Multivariate data analysis* (5th ed.). Upper Saddle River: Prentice Hall.

Hill, W., Stead, L., Rosenstein, M., & Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. In *CHI'95: Conference proceedings on human factors in computing systems* (pp. 194–201), Denver, USA.

Kim, Y., Ok, S., & Woo, Y. (2002). A case-based recommender system using implicit rating techniques. In *Proceedings of the 2nd international conference AH2002* (pp. 522–526), Malaga, Spain.

Kohonen, T. (1997). *Self-organizing maps* (2nd ed.). Berlin: Springer-Verlag.

Lang, K. (1995). Newsweeder: learning to filter news. In *Proceedings of the 12th international conference on machine learning* (pp. 331–339), Lake Tahoe, USA.

Lee, M., Choi, P., & Woo, Y. (2002). A hybrid recommender system combining collaborative filtering with neural networks. In *Proceedings of the 2nd international conference AH2002* (pp. 531–534), Malaga, Spain.

Martin, J. D., Palomares, A., Balaguer, E., Soria, E., Gomez, J., & Soriano, A. (2006). Studying the feasibility of a recommender in a citizen web portal based on user modeling and clustering algorithms. *Expert Systems with Applications, 30*, 299–312.

McNee, S. M., Lam, S. K., Konstan, J. A., & Riedl, J. (2003). Interfaces for eliciting new user preferences in recommender systems. In *Proceedings of the 9th international conference UM2003* (pp. 178–187), Johnstown, USA.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). An open architecture for collaborative filtering of netnews. In *Proceedings of the conference on computer supported cooperative work* (pp. 175–186), Chapel Hill, USA.

Schafer, J. B., Konstan, J., & Riedl, J. (1999). Recommender systems in E-commerce. In *Proceedings of the first ACM conference on electronic commerce EC'99* (pp. 158–166), Denver, USA.

Shardanand, U., & Maes, P. (1995). Social information filtering: algorithms for automating "word of mouth". In *CHI'95: Conference proceedings on human factors in computing systems* (pp. 210–217), Denver, USA.

Su, Z., Ye-Lu, Q. Y., & Zhang, H. J. (2000). WhatNext: a prediction system for Web requests using N-gram sequence models. In *WISE 2000 proceedings: 1st international conference on web information systems engineering* (pp. 214–221), Hong Kong.

Zhou, D., Weston, J., Gretton, A., & Schölkopf, B. (2003). Ranking on data manifolds. Technical report num. TR-113. Tuebingen, Germany: Max Planck Institute for Biological Cybernetics.

Zukerman, I., & Albrecht, D. W. (2001). Predictive statistical models for user modeling. *User Modeling and User Adapted Interaction, 11*, 5–18.