Integrating User Data and Collaborative Filtering in a Web Recommendation System

Paolo Buono, Maria Francesca Costabile, Stefano Guida, Antonio Piccinno, Giuseppe Tesoro

Dipartimento di Informatica, Università di Bari, via Orabona 4, 70125 Bari, Italy
{*buono, costabile, piccinno, tesoro*}*@di.uniba.it, stpag@libero.it*

Abstract

Web-based applications with a large variety of users suffer from the inability to satisfy heterogeneous needs. A remedy for the negative effects of the traditional "one-size-fits-all" approach is to enhance the system's ability to adapt its own behaviour to the users characteristics, such as goals, tasks, interests, that are stored in user profiles. Filtering techniques are used to analyse profile data and provide recommendation to the users to help them navigating in the site and retrieving information of interest. However, techniques such as collaborative filtering are very time consuming and for this are not suitable to be implemented in small systems. We describe here the approach we have adopted in FAIRWIS (Trade FAIR Web-based Information Services), a system that offers on-line innovative services to support the trade fair business processes and a great number of exhibitors organized in a Web-based virtual fair. The approach is based on the integration of data the system collects about users, both explicitly and implicitly, and a classical collaborative filtering technique in order to provide appropriate recommendations to the user in any circumstances during the visit of the on-line fair catalogue.

1. Introduction

Web-based applications with a large variety of users suffer from the inability to satisfy heterogeneous needs. For example, a Web bookstore offers the same selection of best sellers to customers with different reading preferences. A Web museum offers the same "guided tour" and the same narration to visitors with very different goals and interests.

A remedy for the negative effects of the traditional "one-size-fits-all" approach is to enhance the system's ability to adapt its own behaviour to the goals, tasks, interests, and other features of individual users. In the last years, many research teams have been investigating ways of modelling features of the users of hypermedia systems. This has led to a number of interesting proposals of adaptation techniques and adaptive hypermedia systems, which are especially challenging for the Web and therefore are pushing many researchers to work on this topic [1].

Personalisation is a process of gathering and storing information about visitors of a web site, analysing the stored information, and, based on this analysis, delivering the right information to each visitor at the right time. A personalisation component should be capable to recommend documents and/or other web sites, promote products, make appropriate advice, target e-mail, etc. Personalisation is increasingly used as a mean to expedite the delivery of information to a visitor, making the site useful and attractive so that the visitor is stimulated to return to it. For this, personalisation is becoming an expected feature of e-business web sites.

A personalisation component builds and exploits models or profiles of the users interacting with the system. A user profile is a (possibly structured) representation of characteristics of that user, in order to take into account his or her needs, goals, and interests. The term user profiling is also used to refer to a software module that acquires personal data of a user, process these data to obtain additional information, and uses it to modify either content aspects or navigation capabilities of web pages.

Big companies use different methods to personalize their Web sites. Many successful sites, such as Amazon.com, Yahoo.com, and CNN.com, use rich profile information as the basis for providing valuable services and they are considered models for those who want to personalise their site. However, most Web sites do not provide yet any personalisation feature.

The work presented here is related to the project FAIRWIS (Trade FAIR Web-based Information Services), founded by EU. This project aims at offering on-line innovative services to support the trade fair business processes and a great number of exhibitors organised in a Web-based virtual fair. The whole concept of trade fairs is transferred into an electronic form, and visualisation techniques, including virtual reality, are used in order to provide “reality” feelings to the users of trade fair information systems. In recent years, some Web-based information sites have been made available, providing information both on trade fair events and on companies participating in these fairs. However, these data are not organised in an integrated, homogeneous and comprehensive way, since are usually presented in a rigid pre-designed company oriented style. Moreover, currently available Web sites exploit static data that it is difficult to update and to put on-line in an appropriate format. FAIRWIS has a real time connection with an underlying database to guarantee coherence of data and up-to-date

status. Another unique feature of FAIRWIS is provided by the User Profile Engine (UPE) that is the personalisation module. In the analysis, we have performed of fair web sites worldwide, none was found to show any personalisation feature. Therefore, UPE provides a significant added value towards a system that can fit the users needs as better as possible.

In order to build the user profiles and provide recommendations, the approach we have implemented in UPE is based on the integration of data the system collects about users both explicitly and implicitly and a classical collaborative filtering technique in order to provide appropriate recommendations to the user in any circumstances during the visit of the on-line fair catalogue.

The paper is organized as follows. In section 2, we describe general approaches for collecting and exploiting user information. Section 3 discusses the ratings provided by the FAIRWIS users about the web pages they visit. Section 4 presents UPE in more details, and in Section 5 the current prototype is described. Section 6 concludes the paper.

2. Collecting and exploiting user information

The objective of collecting user information is to create a profile that describes user characteristics. The more common techniques are explicit profiling, implicit profiling, and use of legacy data:

- *Explicit profiling*: each user is asked to fill in a form when visiting the web site; this method has the advantage of letting users specify directly their interests.
- *Implicit profiling*: the user's behaviour is tracked automatically by the system. This method is generally transparent to the user. Often, user registration is saved in what is called a cookie that is kept at the browser and updated at each visit. Behaviour information is generally stored in a log file.
- *Legacy data*: they provide a rich source of profile information for known users.

The above methods can be combined to produce comprehensive profiles. This is what we have done in FAIRWIS.

The generated user profiles are analysed in order to present or recommend documents, items, or actions to the user. Making recommendations is a very challenging

step. Rule-based and filtering techniques are the best known for analysing profile data and making appropriate recommendations.

Rule-based techniques exploit a set of rules specified in the system in order to drive personalisation. Cross-selling is an e-business example of the rule-based technique: a rule could be specified to offer product X to a customer who has just bought product Y; for example, a customer of a book might be interested in current or previous books by the same author or in books on the same subject.

Filtering techniques employ algorithms to analyse profile data and drive presentations and recommendations. The three most common filtering techniques are: simple filtering, content-based filtering, and collaborative filtering [2].

- *Simple filtering* relies on predefined classes of users to determine what content should be displayed or what service should be provided. For example, employees of the Research department may have access to some functionality that may not be available for employees of other departments. Therefore, specific pages will be presented to the employees of the Research department.

- *Content-based filtering* works by analysing the content of the objects to generate a representation of the user's interests. The analysis needs to identify a set of key attributes for each object and then fill in the attribute values. For example, in e-commerce users are often asked to provide ratings for each attribute of a product. In this way, content-based filtering analyses the ratings provided by the users to determine, for any product, which other product of the same category has the closest ratings and could then be recommended to a user who got interest in the first product. This technique is most suitable when the objects are easily analysed by the computer and the user's decision about object suitability is not very much subjective. However, recommendations are limited to objects related to those the visitor has considered during his or her navigation, and there is no provision for user qualification.

- *Collaborative filtering* collects visitor opinions on a set of objects, using ratings provided explicitly by the users or implicitly computed, to form peer groups and then learns from the peer groups to predict a particular user's interest in a item. Instead of finding objects similar to those a user liked, as in content-based filtering,

collaborative filtering develops recommendations by finding visitors with similar tastes. Recommendations produced by collaborative filtering are qualified based on the peer group's response and are not restricted to a simple profile matching. However, this method requires a large user base in order to find a peer group for each visitor. This might imply a long learning curve, because at the beginning, when the number of participating visitors is small, the quality of recommendations will be low. The results improve gradually as the number of participating users increases. The more objects two users have rated similarly, the closer the two users are.

For examples of systems incorporating a personalisation components based on content-based and/or collaborative filtering see [3-10].

3. User ratings for providing recommendations

UPE (User Profile Engine) is the Personalisation Module implemented in FAIRWIS. In the current implementation, UPE works as a recommender system that provides personalized suggestions about pages users might find interesting in the on-line fair catalogue available in the system. The recommendations are generated on the basis of different types of ratings that the system gets from the user interaction or computes trough an algorithm of collaborative filtering. As illustrated in the previous section, collaborative filtering works on the idea of analysing "human" evaluations (also called ratings) on items of certain domain and join users who share same tastes.

The ratings collected by the system may be both implicit and explicit. They are explicit if users tell the system what they think about an item. For example, the user may give a rate of 5 to an element of the fair catalogue he or she has found very interesting by filling an appropriate form shown on the screen.

Even if explicit rating is fairly precise, it has disadvantages, such as: 1) stopping to enter explicit ratings can alter normal patterns of browsing and reading; 2) unless users perceive that there is a benefit providing the ratings, they may stop providing them.

Implicit rating is much more difficult to determine. Oard and Kim divide implicit ratings into three categories [11]: rating based on examination, when a user examines an item; rating based on retention, when a user saves an item; rating based on reference, when a user links all or part of an item into an other item.

How can user preferences with implicit ratings be determined? Some criteria were established [12]. In FAIRWIS, by taking into account the structure of the system currently developed by the other partners of the project, we may only consider the following events, and each one has been associated with a weight that highlights the importance of that event for collecting information about the user interests:

- access to a web page (we gave different weights to the home page and the other pages);
- print and/or save action (the user that does this is highly interested on that page or item);
- download of specific files included in download areas;
- image zoom;
- access by search (if the system includes a search engine).

Even if implicit ratings are difficult to determine, they have the following advantages: 1) every interaction with the system (and every absence of interaction) can contribute to implicit rating; 2) can be gathered for free; 3) can be combined with several types of implicit ratings for a more accurate rating; 4) can be combined with explicit ratings for an enhanced rating.

Indeed, the method that is quite effective is the mixed technique implicit/explicit rating and we implemented it in FAIRWIS, as it will be described in the next section. However, especially in the case of sites with many pages, we can be in a situation that some pages have not been evaluated by the current user (neither explicit nor implicit ratings are available). To overcome this situation, algorithms of collaborative filtering may be used. They predict user interests on an item not evaluated by taking into account the historical data set on rates of a users community stored into a database of existing rating provided by other users. Such a database is a set of rates $u_{i,j}$ corresponding to the evaluation of user *i* on the item *j*. If I_i is the set of items on which user *i* has expressed a rate, then it's possible to define the average rate for user *i* as:

$$\bar{u}_i = \frac{1}{|I_i|} \sum_{j \in I_i} u_{i,j} \qquad (1)$$

It is also possible to compute the evaluation of the current user (indicated with a subscript *a*) based on information on the current user and on a set of weights calculated

from the user database. We can assume that the predicted rate of current user expected for item *j*, i.e. $p_{a,j}$, is a weighted sum of rates of other users:

$$p_{a,j} = \bar{u}_a + k\sum_{i=1}^{n} w(a,i)(u_{i,j} - \bar{u}_a) \qquad (2)$$

where *n* is the number of user in the database with non zero weight. Weights *w(a,i)* can reflect distance, correlation, or similarity between each user *i* and the current user. *k* is a normalisation factor such that the absolute values of the weights sum to unit. The expression that identifies the weight *w(a,i)* which relates the current user a with user *i* is:

$$w(a,i) = \frac{\sum_j (u_{a,j} - \bar{u}_a)(u_{i,j} - \bar{u}_i)}{\sqrt{\sum_j (u_{a,j} - \bar{u}_a)^2 \sum_j (u_{i,j} - \bar{u}_i)^2}} \qquad (3)$$

where summations over *j* are calculated on all items which have been evaluated by the users *a* and *i*.

4. The User Profile Engine

We describe in this section the current use in FAIRWIS of the User Profile Engine (UPE). UPE has been developed as an independent component so that it can be portable to other systems. It uses Windows NT, Java and SQL-Server 7.0.

The main types of users addressed by FAIRWIS are: i) fair organisers; ii) exhibitors, namely responsibles of companies that exhibit their products or activities in the fair; iii) professional visitors, who visit the fair for business reasons rather than fun.

The user profiles managed by UPE have a static component and a dynamic one. The static component consists of a set of information that identifies each user and doesn't change (or change rarely). For example: name, nationality, type of users. The information sources come primarily from the registration forms that some FAIRWIS users, namely exhibitor and professional visitors, are required to fill.

The dynamic component of user profile is the changing part of user data. The set of user preferences is part of the dynamic profile. UPE obtains this information by using different type of ratings: explicit ratings, for instance interest ratings for catalogue pages; implicit ratings, obtained by tracking user navigation; computed ratings, obtained by collaborative filtering techniques to supply the preferences not expressed by users, either implicitly or explicitly.

UPE stores individual user profiles, but also assigns the users to a kind of stereotype [13], each stereotype characterised by specific values of the attributes considered in the user profile; these stereotypes are called “segments”. Segments are used because they group users with similar characteristics, so that if the individual user profile is non yet complete, it is still possible to provide recommendations to a user based on his or her stereotype. Indeed, the recommendations for users of a segment are calculated using all ratings available for every user belonging to that segment. In this way, UPE is able to provide recommendations even to a user who just registered, because in the registration form the user provided enough data to be assigned to a segment, and thus UPE gives as recommendations those relative to the segment the user belongs to.

It may also happen that the segment doesn’t contain enough information for recommendations. In this case, UPE provides recommendations based only on the current page the user is visiting, calculating them by taking into account the behaviour of the other users that have already visited that page. In other words, the system will suggest the pages indicated as interesting by most users who were also interested in the page the user is currently looking at. This peculiarity of UPE is suitable also in the case of an unknown user, i.e. user not yet registered or any user who is surfing in the site.

The integration in UPE of a collaborative filtering algorithm permits to predict possible preferences of the current user on the basis of the evaluations provided by other users and stored in the UPE DB. As it is well known, these algorithms are useful but also very time consuming. We are trying to reduce the algorithm complexity by using some heuristics. For example, UPE doesn’t re-calculate all the weights in formula (3), but it does it only for those pairs of correlated users, in which at least one of the two users has interacted with the web site and has modified a number of ratings higher than a certain threshold.

The rates in UPE DB are updated automatically in a scheduled way, or with an explicit request of the system administrator.

Fig. 1 shows the main modules of UPE. The predictive algorithm is responsible of computing the ratings the user did not provide (explicitly or implicitly). The recommendation module is the UPE main component, which manages the user profiles and computes the recommendations. To do this, UPE needs to communicate with the FAIRWIS Core system, which manages the web interface.

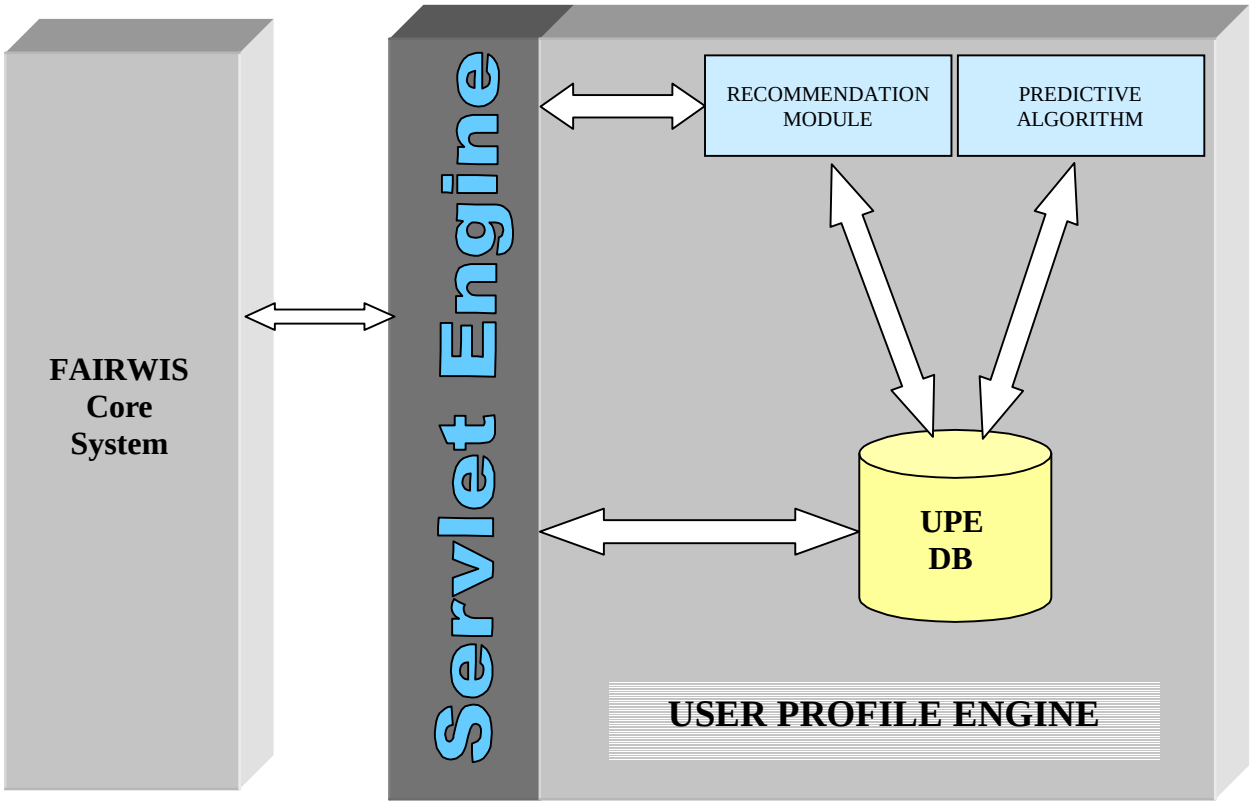

Figure 1. UPE architecture and the communication with FAIRWIS Core system.

The communication between FAIRWIS Core and UPE is based on Java Servlet technology. Servlets are programs that run on a Web server; they are designed to work according to a request/response processing model. In this model, a client sends a request message to a server and the server responds by sending back a reply message. Requests can come in the form of an HTTP, URL, FTP, or a custom protocol. As shown in Fig. 1 the communication module is called Servlet Engine. It is composed by servlets devoted to the communication between FAIRWIS Core and UPE.

More specifically, servlets running on the web server perform the following tasks:

1. insert static user information into UPE DB; this information is obtained from the registration form filled by the user in the FAIRWIS web site and sent by FAIRWIS Core to UPE Servlet Engine;
2. update UPE DB with the dynamic information coming from the user interaction, sent by FAIRWIS Core to UPE Servlet Engine;

3. give the recommendation for a user, by replying to a specific request coming from FAIRWIS Core.

5. The prototype

We here report some examples that describe the interaction of users with a prototype web site in order to show the UPE behaviour. The recommendations provided to the user are shown in an area of the fair web pages. In the current prototype, which refers to the fair on Information Technology MITE 2001, this area is located at the bottom-left area of all web pages but the home page, as shown in Fig. 2.

As we already said, exhibitors and professional visitors are required to fill the form for user registration that includes data required by UPE.

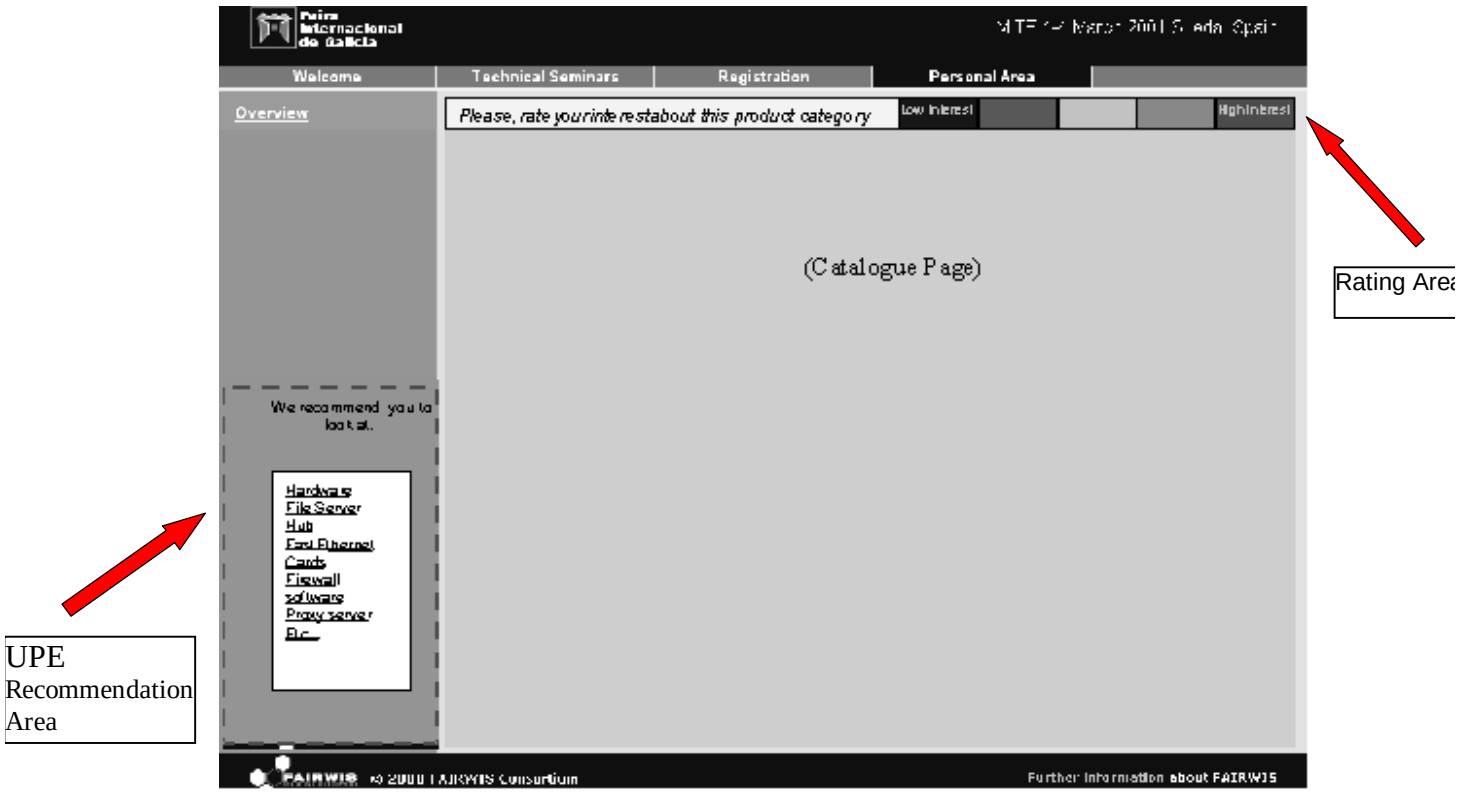

Figure 2. Example from the MITE 2001 prototype.

Example 1: Explicit rating request

When the registered user is visiting a catalogue web page (see Fig. 2), FAIRWIS asks him or her to rate the current page, showing a frame at the top of the web page.

Users can rate the web page by acting in an explicit rating box located at the top of

the main window (see Fig. 2): he or she selects the vote for the current page by clicking on a specific area of the rating box, according to his or her own interest. Of course, the user is free to ignore the request of explicit rating and keep navigating through the catalogue.

Example 2: FAIRWIS provides recommendation for a registered user

When a registered user logs in the system, the page shown to the user is appropriately personalised and the system suggests some links worth to be examined. The recommendations are links to other web pages. The user can click on one recommendation or keep navigating among web pages in a "traditional" way.

6. Conclusions

In this paper, we have described the FAIRWIS personalization component that currently works as a recommendation system. The approach is based on the integration of data collected both explicitly and implicitly from the user interaction and a collaborative filtering technique.

It is worth nothing that UPE has been designed to be an independent module that can be integrated in any system, with which the communication is done through Java Servlet technology. Moreover, UPE manages a complex user profile that can be better exploited to provide different types of personalisation. The current behaviour as a recommender system is due to constraints imposed by the actual structure of the FAIRWIS Core system.

Acknowledgement

The support of European Commission through grant FAIRWIS IST-1999-12641 is acknowledged.

References

[1] AH2000, Proceedings of the International Conference on "Adaptive Hypermedia and Adaptive Web-Based Systems", Trento, Italy, August 28-30, 2000.

[2] "Web Site Personalisation". IBM High-Volume Web site team, January 2000, http://www-4.ibm.com/software/developer/library/personalization/index.html.

[3] Balabanovic, M. Exploring versus exploiting when learning user models for next recommendation. User Modeling and User-Adapted Interaction, 8(1), pp. 71-102, 1998.

[4] Basu, C., Hirsh, H., Cohen, W. Recommendation as classification: Using social and content-based information in recommendation. Proceeding of the Fifteenth National Conference on AI, Madison, WI, AAAI Press, pp. 714-720, July 1998.

[5] Boone, G. Concept features in Re:Agent, an intelligent e-mail agent. Proceedings of the Second International Conference on Autonomous Agents, New York, ACM Press, pp. 141-148, 1998.

[6] Dent, L., Boticario, J., McDermott, T., Zabrowski, D. A personal learning apprentice. Proceedings of the Tenth National Conference on AI, San Jose, CA: AAAI Press, pp. 96-103, July 1992.

[7] Hinkle, D., and Toomey, C. N. CLAVIER: Applying case-based reasoning on to composit part fabrication. Proceeding of the Sixth Innovative Application of AI Conference, Seattle, WA, AAAI Press, pp. 55-62, 1994.

[8] Lang, K. NEWSWEEDER: Learning to filter news. Proceeding on the 12th International Conference on Machine Learning, Lake Tahoe, CA, Morgan Kaufmann, pp. 331-339, 1995.

[9] Pazzani M. J., Muramatsu J., Billsus D.: Syskill & Webert: Identifying Interesting Web Sites. Proceedings on Thirteenth National Conference on Artificial Intelligence, Portland, OR, AAAI/IAAI Press, Vol. 1, pp. 54-61, 1996.

[10] Shardanad, U., Maes, P. Social Information Filtering: Algorithms for automating "word of mouth". Proceedings of the Conference on Human Factors in Computing Systems, Denver, CO, ACM Press, pp. 210-217, 1995.

[11] Oard, D., Kim, J. Implicit Feedback for Recommender Systems. In Proceedings of AAAI Workshops on Recommender Systems. July 1998.

[12] Claypool, M., Le, P., Waseda, M., Brown, D. Implicit Interest Indicator. Computer Science Technical Report Series, WPI Computer Science Department, Worcester, Massachussets, July 2000.

[13] E. Rich, “Stereotypes and user modeling”, *User Models in Dialogue Systems*, A. Kobsa and W. Wahlster, eds. Springer-Verlag, pp. 35-51, 1989.