

Recommending WWW information sources using Feature Guided Automated Collaborative Filtering

Gabriela Polčicová and Pavol Návrat

Slovak University of Technology, Department of Computer Science and Engineering,
Ilkovičova 3, 812 19 Bratislava, Slovakia
{polcicova, navrat}@dcs.elf.stuba.sk

Abstract. Exponential growth of the number of information sources available on the web hampers effective identification of those items that are likely to be of interest to the user. To overcome the problem of information overload, techniques of information filtering aim at recommending items of potential interest based on a profile of user needs, preferences or opinions. In the paper, a recommender system is described that utilizes Feature Guided Automated Collaborative Filtering for recommending relevant HTML-documents to the users. While browsing the web, user expresses his/her opinions on documents by rating them. The system "learns" user's opinions and searches for like-minded users in order to recommend him unseen relevant documents of interest. The system's architecture is based on two kinds of agents - a recommendation agent and a communication agent - that work for each user.

1 Introduction

Exponential growth of the number of information sources available on the web hampers effective identification of those items that are likely to be of interest to the user. To overcome the problem of information overload, techniques of information filtering aim at recommending items of potential interest based on a profile of user needs, preferences or opinions on one hand, and based on an analysis of the item on the other hand.

Recommender systems can either analyse information items by themselves, employing one of the techniques of content-based filtering, or rely on opinions of other users and employing one of the collaborative filtering techniques. Techniques for recommending appropriate information using opinions of users with similar preferences are known as Automated Collaborative Filtering (ACF) [7, 9]. The recommender system that is described in this paper uses a particular

kind of ACF called Feature Guided Automated Collaborative Filtering (FGACF).

We shall shortly describe the technique in the next section. In section 3, we shall present the recommender system that we currently work on. It should be noted that our work makes use of some results of G. Polčicová's masters thesis that have been presented in [6].

2 Feature Guided Automated Collaborative Filtering

ACF approach is based on the hypothesis that users who had similar preferences in the past will probably have similar preferences again [8]. Recommender systems that use ACF collect users' opinions on a set of items in the form of their ratings. They search through the community of users to identify users with similar opinions. Similarity is defined in terms of ratings of respective items. Opinions of those like-minded users about items that a particular user with a similar interest has not seen are used to predict his/her ratings. Items with positively predicted ratings are recommended to the user.

An important advantage of this approach, as compared to other filtering techniques, is that content of items being filtered need not be analysed at all. With regard to the assumption that agreement of preferences in one topic (e.g. in literature) does not imply agreement of preferences in another topic (e.g. in sports), it is necessary to recommend items of each topic (category) separately. This method of recommendation is known as Feature Guided Automated Collaborative Filtering (FGACF).

A variety of algorithms and systems for ACF have been reported in the literature. Most of the systems, such as the system Tapestry [4] that filters e-mails, Ringo [9] recommending music albums and artists, GroupLens [7] using Usenet newsgroups as a domain,

or Fab [1] recommending HTML-documents, make use of correlation coefficient. Other approaches for ACF, e.g. method based on finding like-minded users by clustering them according to the filtered items content (system Yenta [3]), or method utilizing neural networks [2], have been reported.

3 Recommender system

We have proposed a recommender system for the web that utilizes the FGACF method. The system consists of users' agents (in particular, communication and recommendation agent) and servers (Fig. 1).

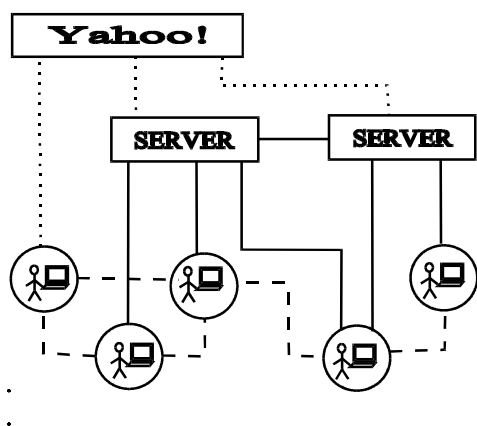


Fig. 1. The schema of the recommender system. Circles represent users' agents. Dashed lines represent communication between recommendation agents through profile documents; solid lines represent communication between servers and between servers and agents. Dotted lines represent communication between servers or agents with Yahoo!.

For each user the agents perform the following tasks:

- collect user's ratings of HTML-documents (communication agent),
- find like-minded users (recommendation agent),
- use preferences of like-minded users to recommend HTML-documents to the user (recommendation agent).

All these tasks are accomplished for each topic represented by a category. Both kinds of agents acquire necessary information from servers. Both servers and agents can request information from Yahoo! search engine, serving as a kind of default. Fig.2 shows communication among agents, servers and Yahoo!. The system was implemented in Java.

3.1 Communication agent

Communication agent (CA) performs two functions: to collect user's ratings and to offer recommended documents to the user. While browsing, users rate documents on a scale from 1 to 7 (with 7 being the best rating). The category a document belongs to has to be determined, too, by either requesting a server (or consulting Yahoo!) or a user. The CA writes all the ratings in the user's profile (which is itself a HTML-document accessible through the Internet). The profile is divided into categories. Each category consists of three parts:

- *a name* of the category,
- *links* - a list of ratings and rating predictions for the documents classified into the category, each represented either by a pair - document's URL and the rating - or by a triplet - document's URL, rating prediction and the word "prediction",
- *similar profiles* - a list of like-minded users' profiles for the given category that is represented by the pair - URL of like-minded user's profile and a degree of similarity.

List of predictions and similar profiles, as well as the list of recommendations (Fig. 2) is generated by the recommendation agent (RA) and offered to the user by CA.

3.2 Recommendation agent

Recommendation agent (RA) compares the profile of its user with profiles of other users for each category in which its user has rated some item. This process consists of three tasks:

1. reading the profile of the agent's user - it reads ratings, previously computed predictions and it also creates a list of previously found like-minded users,
2. reading profiles of the like-minded users and comparing them with the user's profile,
3. reading profiles of other users with unclear preferences and comparing them with the user's profile - the agent acquires URLs of other users' profiles by sending a message to the server (*send me URLs of the profiles*). If there is no server available, the RA can acquire the list of registered profiles by requesting them from Yahoo!.

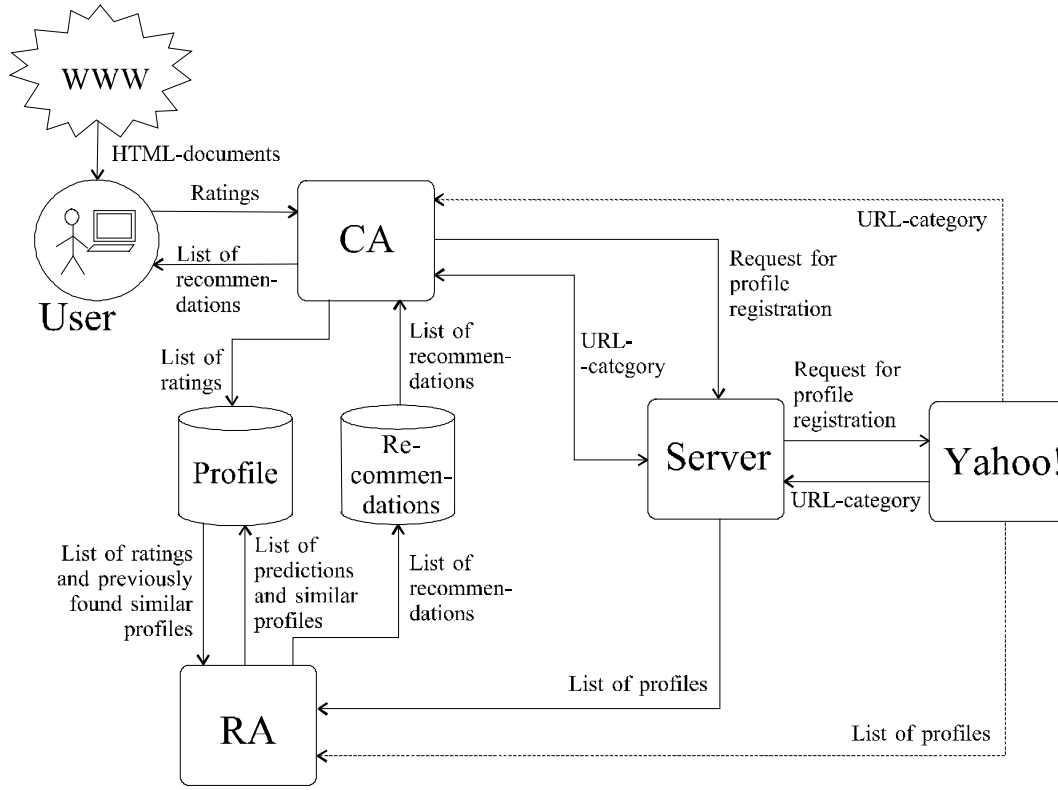


Fig. 2. Communication among user, communication agent (CA), recommendation agent (RA), server and Yahoo! search engine (solid lines). In the case that no server is available, CA and RA communicate directly with Yahoo! (dashed lines).

The user determines the amount of time the tasks 2 and 3 can take. Profiles are compared on how similar they are for each category that both users have rated. The degree of similarity is determined using the Pearson correlation coefficient [9]

$$k_{xy} = \frac{\sum_{j \in I_{xy}} (r_{xj} - \bar{r}_x)(r_{yj} - \bar{r}_y)}{\sqrt{\sum_{j \in I_{xy}} (r_{xj} - \bar{r}_x)^2} \sqrt{\sum_{j \in I_{xy}} (r_{yj} - \bar{r}_y)^2}}, \quad (1)$$

where I_{xy} is a set of documents rated by both users x and y , r_{xj} (r_{yj}) is the rating of document j by the user x (the rating of document j by the user y) and \bar{r}_x (\bar{r}_y) is average of user x (y) ratings.

Predictions of ratings for the documents not rated by the user x are computed from the ratings of the other users y and their degree of similarity (k_{xy}) with the user x . Rating prediction (p_{xj}) for document j is given by [7]

$$p_{xj} = \bar{r}_x + \frac{\sum_{y \in U_j} (r_{yj} - \bar{r}_y) k_{xy}}{\sum_{y \in U_j} |k_{xy}|}, \quad (2)$$

where U_j is the set of users who rated document j and whose profiles are used for prediction computation. This is done for each category.

The user may set whether for computing of predictions only ratings or both ratings and predictions will be used.

Predictions of ratings are written into the user's profile. Documents with predicted rating higher then the threshold (in our system, 4) are written to the file *Recommendations* (Fig. 2). CA uses this file for recommending documents to the user.

3.3 Server

The server maintains two lists – a list consisting of URLs of the agents' profiles and a list of pairs - URL

of the rated document and the category the document belongs to. Function of the server is to respond to the following agents' messages:

- *register new profile* - it adds the URL of the profile to the list and sends a request for profile registration to Yahoo! search engine,
- *send me the category for the document* - if a pair - URL of the document, document category – exists in the list maintained on the server or if Yahoo! determined the category, the server sends the category to the CA,
- *the document belongs to the category* - it adds the pair - URL of the document, document category - to the list,
- *send me URLs of the profiles* - it sends the list of profiles URLs to the RA.

Servers can also communicate with each other (Fig. 1) in order to complete their lists (*send me your lists* message).

4 Conclusions

The ambition of the proposed recommender system is to help the user in orientation in the number of information sources available on the web on the basis of recommendations from like-minded users for particular categories of items.

An important feature of the system is that for the computation of new predictions, not only ratings but previously computed predictions may be used as well. This is an attempt to eliminate the well-known limitation of the ACF i.e., profiles of the users who did not rate the same documents are not comparable. It enables the system to generate more recommendations (even less accurate). This might be helpful at the beginning of system operation when only a few documents are rated.

The system consists of agents and servers. Agents form preference profiles and generate recommendations for each user. That is why the system can work (with the above mentioned constraints) even without servers that just maintain and provide information for agents. This may be considered as an advantage of the presented recommender system.

On the other side, some of the shortcomings of the pure collaborative filtering techniques can be reduced by integrating content-based ratings [8]. We are currently working on proposal of the method, where instead of past predictions generated by ACF tech-

nique, predictions of content-based filtering method are used. In order not to use too many predictions, only those are taken, which are missing in Pearson coefficient computation. That means only predictions for those documents are used, that the other user has rated. Combining collaborative filtering with personal agents into hybrid recommender systems has been discussed also in [5].

Acknowledgements

The research has been supported by the INCO-Copernicus programme project No. 979069, and by Slovak Science Grant Agency project No. G1/7611/20.

References

1. Balabanovic, M., Shoham, Y.: Fab: Content-based, Collaborative Recommendation. *Communications of the ACM*, 40(3):29–38 (1997)
2. Billsus, D., Pazzani, M.: *Learning Collaborative Information Filters*. Proceedings of the International Conference on Machine Learning. Morgan Kaufmann Publishers. Madison (1998)
3. Foner, L.: Yenta: A multi-agent, referral based match-making system. Proceedings of the first international conference on Autonomous agents, 301–307 (1997)
4. Goldberg, D., Nichols, D., Oki, B., Terry, D.: Using collaborative filtering to weave an information tapestry. *Communication of the ACM*, 35 (12): 61–70, (1992)
5. Good, N., Schafer, J.B., Konstan, J., Borchers, A., Sarwar, B., Herlocker, J., and Riedl, J.: Combining collaborative filtering with personal agents for better recommendations. In: Proceedings of the 1999 Conference of the AAAI, 439–446 (1999)
6. Polčicová, G.: Recommending HTML-documents using Feature Guided Automated Collaborative Filtering. In: Eder, J., Rozman, I. and Welzer, T.: *Advances in Databases and Information Systems, Proceedings of the 3rd East European ADBIS'99 Conference, Short Papers*, 86–91, (1999)
7. Resnick, P., Iacovou, N., Sushak, M., Bergstrom, P., Riedl, J.: GroupLens: An open architecture for collaborative filtering of netnews. Proceedings of the Conference on CSCW, ACM Press, 175–186 (1994)
8. Sarwar, B., Konsta, J., Borchers, A., Herlocker, J., Miller, B., and Riedl, J.: Using filtering agents to improve prediction quality in the GroupLensResearch Collaborative Filtering System. In: Proc. CSCW'98, Seattle, ACM Press, 345–354 (1998)
9. Shardanand, U., Maes, P.: Social information filtering: Algorithms for automating "word of mouth". Proceedings of the 1995 ACM Conference on Human Factors in Computing Systems, 210–217, New York, ACM (1995)