

## From "I like" to "I prefer" in Collaborative Filtering

Armelle Brun, Ahmad Hamad, Olivier Buffet, Anne Boyer

► **To cite this version:**

Armelle Brun, Ahmad Hamad, Olivier Buffet, Anne Boyer. From "I like" to "I prefer" in Collaborative Filtering. International Conference on Tools with Artificial Intelligence - ICTAI 2010, Oct 2010, Arras, France. IEEE, 2010, Proceedings of the 22th International Conference on Tools with Artificial Intelligence - ICTAI 2010. <inria-00535566>

**HAL Id: inria-00535566**

**<https://hal.inria.fr/inria-00535566>**

Submitted on 11 Nov 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# From “I like” to “I prefer” in Collaborative Filtering

Armelle Brun<sup>1</sup> Ahmad Hamad<sup>2</sup> Olivier Buffet<sup>3</sup> Anne Boyer<sup>1</sup>  
<sup>1</sup> LORIA / Nancy-Université – <sup>2</sup> Sailendra SAS – <sup>3</sup> LORIA / INRIA  
 Campus Scientifique – BP 239 / 54506 Vandœuvre-lès-Nancy – France  
 {armelle.brun, ahmad.hamad, olivier.buffet, anne.boyer}@loria.fr

**Abstract**—Collaborative filtering exploits user preferences, generally ratings, to provide them with recommendations. However, the ratings may not be completely trustworthy: the rating scale is usually reduced and the rating values may be influenced by many factors. This paper is a first attempt at studying the expression of preferences under the form of preference relations where users are asked to compare pairs of resources. First experiments show that this new approach compares with, and sometimes improves, the classical one.

## I. INTRODUCTION

The democratization of the Internet has resulted in a so huge volume of accessible information that it is becoming difficult for users to find the information they are looking for. Thus, a critical issue of current Web applications is to incorporate mechanisms for delivering information that fits the users’ interests.

Recommender systems (RS) are such a mechanism. They aim at recommending items to users, linked to their expectations and tastes. An item (also called a resource) can for example be a web page or a movie. To recommend items to a given user, an RS uses the user’s profile, which represents the user’s preferences. RSs generally fall into three categories, based on the information they use to perform recommendations [1]: content-based systems, knowledge-based systems and collaborative-filtering (CF) systems. In this paper we are interested in the CF approach whose popularity has increased over the last few years. The *preferences* of users  $U$  about items  $I$  are known, a preference being broadly defined as “preferences over some domain of possible choices order these choices so that a more desirable choice precedes a less desirable one” [2]. However, these preferences are partially known, as they are those the users have given to the system. CF aims at guessing the missing preferences. The most common representation of preferences is under the form of *utilities*, i.e. quantitative votes (/ratings) provided by users. The RS estimates the votes of the users on the items they have not seen. We argue that using votes has several drawbacks such as impreciseness. This paper is a first attempt to cope with these limitations.

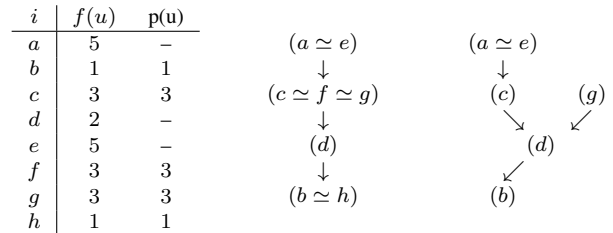
We propose to replace utilities by their qualitative counterpart: *preference relations*. Instead of expressing a quantitative interest (through utilities), the users express qualitatively their interest about resources (through preference relations). In this study, we mainly focus on the decrease in quality of recom-

mendations when exploiting preference relations, due to the loss of the quantitative aspect.

Section II presents the classical CF approach. The following section introduces preference relations. Then, Section IV presents our propositions to exploit preference relations in CF. The next section is an experimental validation. Last, we conclude and present perspectives.<sup>1</sup>

## II. CLASSICAL COLLABORATIVE FILTERING

CF uses the known preferences (ratings) of a given user  $u$  and the preferences of other users to estimate the unknown preferences of  $u$  and thus recommend resources he/she does not know [4]. These votes can be viewed as “quantitative” preferences and can be represented by a *utility* function  $ut : I \rightarrow \mathbb{R}$ . The resource  $j$  is preferred to the resource  $i$  with respect to the utility function  $ut$ . Figure 1-a, column 1, shows a user’s utility function. In CF, the known part of  $u$ ’s preferences is called  $u$ ’s *profile*. Fig. 1-a, col. 2, shows a utility profile corresponding to the utility function. Dashes indicate missing pieces of information.



a- utility function & profile    b- pref. relation    c- compatible profile

Figure 1. (a) a utility function and a corresponding utility profile; (b) a corresponding preference relation; and (c) a profile compatible with this preference relation (an arrow means “strictly preferred to”)

To estimate the unknown preferences of a user, two approaches can be used [4]. The memory-based approach (non parametric) exploits similarities (of preferences) between users. The model-based approach (parametric) pre-processes data to build a model. The memory-based approach is the one we focus on, it is divided into the three following steps:

### A. Collecting User Profiles

Collecting a user’s profile comes down to asking a user to vote for some resources. The votes are usually a positive

<sup>1</sup>An extended version of this work is presented in [3].

integer value in a predefined scale. The resulting information is the user profile.

### B. Computing Similarities Between Users

Once the user profiles are built, the second step computes the similarities  $sim(u, u')$  between all pairs of users based on the users' profiles. Let  $u_1$  and  $u_2$  be two users with profiles on two subsets  $I_1 \subseteq I$  and  $I_2 \subseteq I$ . Several measures can be used to compute similarities between users [5]. We choose here the well-known cosine measure. This measure takes into account the proportion of items present in both profiles, and the distribution of the votes on the items. Two users  $u_1$  and  $u_2$  will be considered as having identical profiles if and only if they are "co-linear".

### C. Recommending to a User

To estimate the utility of a resource  $i$  for  $u$ , the classical approach computes the weighted average of the utilities of  $i$  among the neighboring users. The vote of each user  $u'$  is weighted by the similarity between  $u$  and  $u'$ . In this paper, the similarity measure is instantiated by the cosine measure.

Once these estimates are computed, the resources with the highest estimates are recommended to  $u$ .

## III. FROM UTILITIES TO PREFERENCE RELATIONS

### A. From Utilities

The possible values for a vote (utility) in classical CF are positive integer values, and the scale of possible votes is generally reduced, thus imprecise. In addition, the context, the previously rated resources, etc. may influence the choice of the rating. The resulting votes may thus be imprecise and unreliable, which limits the quality of the computed similarities and therefore the quality of the recommendations.

The objective of an RS is to provide a user with a list of items this user does not know yet, the items being ordered according to the user's expected preferences. In other words, the objective is to complete the top of the user's preference relation. There is no obligation to estimate quantitative information as in classical approaches (the utilities).

### B. To Preference Relations

We propose here to replace utility functions by *preference relations*. A preference relation is a binary relation on  $I$  that is reflexive, transitive and total/complete.

In this case, the user is not asked to vote for resources but to express a qualitative interest about the resources he/she has already seen. We put forward the idea that a preference relation can be more appropriate than votes: in a preference relation the discretization problem is avoided. Comparing two resources is easier than choosing a vote value for each of them among a reduced set of integer values.<sup>2</sup> Moreover, [6] shows that making preference judgements is faster than absolute judgements (ratings). Using preference relations is thus likely to improve

<sup>2</sup>We suppose here, as in most work relying on utilities, that items are always comparable.

the participation rate. In addition, this approach will allow to consider two users as similar when they order resources in the same way even if they do not rate these resources identically or co-linearly.

However, one of the drawbacks of using preference relations is the polynomial increase in the number of comparisons needed in a test collection [7]: placing a resource here requires comparing it to numerous other resources, whereas rating one resource was enough. Plus, even if recommendation only asks for a qualitative result, utility functions are—in theory, i.e., assuming reliable ratings—strictly more informative than preference relations.

## IV. CF USING PREFERENCE RELATIONS

We now adapt the three steps of classical CF to the use of preference relations:

### A. Collecting User Profiles

A user's preference profile can also be collected through question answering. The system presents resource pairs and then asks which resource the user prefers. One problem is the number of questions required to get a full profile, and which subset of questions to ask to obtain an incomplete, but informative, profile.

### B. Computing Similarities Between Users

Let  $\mathcal{I}_u$  be the set of resource pairs  $(i, j)$  in  $u$ 's profile. Let us also define the function  $f_{u_1, u_2}(i, j)$  indicating (by outputting 0 or 1) whether two users  $u_1$  and  $u_2$  agree about their preference on  $i$  and  $j$ . This allows for a cosine similarity measure adapted to preference relations:

$$\cos_{\preceq}(u_1, u_2) = \frac{\sum_{(i,j) \in \mathcal{I}_1 \cap \mathcal{I}_2} f_{u_1, u_2}(i, j)}{\sqrt{|\mathcal{I}_1| \cdot |\mathcal{I}_2|}}. \quad (1)$$

As for utilities, this definition accounts for the proportion of resource pairs present in both profiles, and the distribution of the preferences on the resource pairs.

### C. Recommending to a User

To complete  $u$ 's preference relation, our approach estimates the position of a resource in the preference relation. As in classical CF, similar users  $u'$  are used. We compute the position of an item  $i$  in each neighbor's profile. Then we estimate the position of  $i$  in the profile of  $u$  as the weighted average of the positions of  $i$  in the profiles of users  $u'$ .

The profile  $u'$  being generally partial, we propose to measure the position of  $i$  in the profile of  $u'$  by counting:

- $\#_{u', i}^{\oplus}$  the number of resources that are strictly preferred to  $i$ ;
- $\#_{u', i}^{\approx}$  the number of resources (other than  $i$ ) equally preferred to  $i$ ;
- $\#_{u', i}^{\ominus}$  the number of resources that are strictly less preferred than  $i$ .

The position of  $i$  in the profile of  $u'$  is then:

$$\widehat{ut}_{u'}(i) = \frac{-\#_{u', i}^{\oplus} + \#_{u', i}^{\ominus}}{\#_{u', i}^{\oplus} + \#_{u', i}^{\approx} + \#_{u', i}^{\ominus}}.$$

The resulting value (in  $[-1, +1]$ ) can be seen as a utility.

We can now use the classical prediction formula by exploiting the position of  $i$  in the profile of neighbor users  $u'$  ( $\hat{u}t_{u'}(i)$ ), instead of their vote on  $i$  ( $ut_{u'}(i)$ ):

$$\tilde{u}t_u(i) = \frac{\sum_{u' \in \hat{U}_{i,u,k}} \text{sim}(u, u') \cdot \hat{u}t_{u'}(i)}{\sum_{u' \in \hat{U}_{i,u,k}} \text{sim}(u, u')}$$

where  $\hat{U}_{i,u,k}$  is the set of users  $u'$  in  $U$  that have  $i$  in their profile, and belong to the set of  $u$ 's  $k$  nearest neighbors.

Note that the higher the position of a resource, the better its recommendation to the user. Once the positions of resources  $i$  have been computed, the resources with the highest estimated positions are recommended to the user.

## V. EXPERIMENTS

### A. Experimental Data

Having only access to classical utility-based datasets, we had to produce an artificial preference-relation-based dataset by converting each user utility profile in the corresponding preference profile. This implies that we will not be able to compare the quality of the users' input depending on whether it is under the form of utilities or preference relations. In these conditions, we do not expect to increase the quality of the recommendations but aim at evaluating the loss in performance to assess whether using only qualitative information is a promising approach.

We chose to work on the Movielens (<http://movielens.org>) corpus, made up of a set of user votes (between 1 and 5) about movies. It contains 1682 users, 943 items and 100k preferences, and is divided into training and test sets.

### B. Similarity Between Users

In CF, the RS classically estimates missing utilities. These estimates are then compared to the utilities in the test set. The system accuracy is thus usually evaluated in terms of MAE (Mean Absolute Error).

In this section, we compare user similarities based on utilities vs preference relations. To that end, they are both used with the same classical recommendation step based on ratings.

The resulting MAEs are: 0.71 with utility functions, 0.73 with preference relations. The increase in mean error when using preference relations was predictable as the preference relation profiles are derived from utility profiles by removing (quantitative) information. But this loss is not significant, showing the potential of this new approach.

### C. Recommending to a User

In this section we evaluate the accuracy of the complete recommendation process, i.e. using either utility functions or preference relations in all three steps. We evaluate in terms of precision, the ratio between the number of resources the system judged as being preferred divided by the number of resources actually preferred by the user.

The items we consider here to be preferred by a user are the highly rated ones, i.e., rated 4 or 5 in the MovieLens corpus.

Table I shows experimental results in two cases. These results show that both approaches have similar precisions. We can deduce that exploiting preference relations in the whole process does surprisingly not lead to a decrease in the quality of predictions, despite the loss in information when transforming rating data in preference relations.

Table I  
PRECISION OF THE TWO APPROACHES

Approach	rated 5	rated 4 or 5
Utility functions	0.52	0.75
Preference relations	0.51	0.77

To refine our experimentations, we evaluate the preference relations approach in terms of mean rank. We have computed the mean rank on the set of resources rated 5 in the test set. When using preference relations, the mean rank is 9% lower than with the utility-based approach. This improvement is significant. In terms of mean rank, the recommendations computed when using preference relations are thus more accurate than those from the utility-based approach.

## VI. CONCLUSION AND FUTURE WORK

This paper presents a new approach to represent preferences in a CF-based RS using preference relations instead of ratings (utilities). This approach has been evaluated on a state of the art corpus. It compares with the classical approach and even improves significantly performance in terms of mean rank. Exploiting preference relations to acquire users' preference is thus a highly promising approach.

Future work includes testing the robustness and stability of preference relations compared to ratings, and implementing a complete recommendation system, what includes the tedious task of collecting the users' preferences under the form of preference relations.

## REFERENCES

- [1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art," *Trans. on Knowledge and Data Engineering*, vol. 17, 2005.
- [2] R. Brafman and C. Domshlak, "Preference handling - an introductory tutorial," *AI Magazine*, vol. 30, no. 1, 2009.
- [3] A. Brun, A. Hamad, O. Buffet, and A. Boyer, "Towards preference relations in recommender systems," in *Proc. of the ECML/PKDD workshop on Preference Learning*, 2010.
- [4] J. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. of UAI-98*.
- [5] L. Candillier, F. Meyer, and M. Boullé, "Comparing state-of-the-art collaborative filtering systems," in *Proc. of MLMD-07*.
- [6] B. Carterette, P. Bennett, D. Chickering, and S. Dumais, "Here or there; preference judgments for relevance," in *Proc. of ECIR-08*.
- [7] B. Carterette and P. Bennett, "Evaluation measures for preference judgments," in *Proc. of SIGIR*, 2008.