A Symbolic Hybrid Approach to Face the New User Problem in Recommender Systems

Byron Bezerra and Francisco de A. T. de Carvalho

Centro de Informatica - CIn / UFPE,
Av. Prof. Luiz Freire, s/n - Cidade Universitaria,
CEP 50740-540 Recife - PE, Brazil
{bldb, fatc}@cin.ufpe.br

Abstract. Recommender Systems seek to furnish personalized suggestions automatically based on user preferences. These preferences are usually expressed as a set of items either directly or indirectly given by the user (e.g., the set of products the user bought in a virtual store). In order to suggest new items, Recommender Systems generally use one of the following approaches: *Content Based Filtering*, *Collaborative Filtering* or *hybrid filtering methods*. In this paper we propose a strategy to improve the quality of recommendation in the first user contact with the system. Our approach includes a suitable plan to acquiring a user profile and a hybrid filtering method based on *Modal Symbolic Data*. Our proposed technique outperforms the *Modal Symbolic Content Based Filter* and the standard *kNN Collaborative Filter based on Pearson Correlation*.

1 Introduction

Recommender Systems (RS) allow E-commerce websites to suggest products to their costumers by providing relevant information to assist them in shopping tasks. This system has also increased its importance in entertainment domains [7]. In both cases, two recommendation tasks have been mainly employed by information systems: *Annotation in Context* (providing a score for an item) and *Find Good Items* (building a ranked list of items) [5]. The latter has been widely used in virtual stores.

Whatever the *RS* task is, it must collect user preferences to provide good suggestions. The more information collected, the better the provided suggestions are. The user, however, often has little time for supplying information about him/herself. It is necessary to learn about users with as little data as possible. This problem is all the more challenging during the first system usage, when there is no user information. In such cases, a suitable strategy for acquiring user preferences is quite valuable.

After acquiring user preferences, *RS* may adopt one of the following filtering approaches to build suggestions: *Content Based* (*CB*) *Filtering* (based on the correlation between the user profile and item content), *Collaborative Filtering* (based on the user profile correlation) or hybrid filtering techniques [1,3,4,5,7].

In this paper, we describe a suitable strategy for achieving better recommendation lists in first system usage based on a new hybrid information filtering method (see section 2). Basically, the idea is to ask the user to evaluate at least one item of each

G.I. Webb and X. Yu (Eds.): AI 2004, LNAI 3339, pp. 1011–1016, 2004.

possible evaluation grade. The descriptions of the evaluated items are used to build a modal symbolic profile of the user. This profile is then compared with other user profiles in order to perform recommendations in a collaborative fashion. This novel strategy was experimentally tested and compared in the movie domain (see section 3), where the user can evaluate an item with a grade between 1 (worst) to 5 (best).

2 Collaborative Filtering Based on Modal Symbolic User Profiles

As described in the previous section, our strategy in the user profile acquisition phase is to request the user to evaluate at least one item of each possible evaluation grade. Regardless of the acquisition methodology, the following steps are executed to generate recommendation lists in the *CF* algorithm based on *MS* user profiles:

1. *Construction of the modal symbolic descriptions of the user profile.* This step can be done incrementally without degrading the memory usage.
2. *Weight all users based on their similarity with the active user.* Similarity between users is measured by a function which compares the *MS* descriptions of each user.
3. *Select the k closest users as neighbors of active user.* The closeness is defined by similarity between some candidate neighbor and the active user.
4. *Generation of a ranked list of items after computing predictions from a weighted combination of the selected neighbors' ratings.*

Although, the steps 2–4 are standard in *CF* algorithms, the 2nd one is done in a *CB* manner through the *MS* user profiles built in 1st step. Before detailing all phases of our algorithm we need to introduce modal symbolic data [2] (see *www.jsda.unina2.it*). Let D_j be a finite set of categories. A modal variable y_j with domain D_j defined in the set $E=\{a, b,\ldots\}$ of objects is a multi-state variable where, for each object $a \in E$, not only is a subset of its domain D_j given, but also for each category m of this subset, a weight $w(m)$ is given that indicates how relevant m is for a. Formally, $y_j(a) = (S_j(a), q_j(a))$ where $q_j(a)$ is a weight distribution defined in $S_j(a) \subseteq D_j$ such that a weight $w(m)$ corresponds to each category $m \in S_j(a)$. $S_j(a)$ is the support of the measure $q_j(a)$ in the domain D_j. Therefore, a symbolic description of an item is a vector where there is a weight distribution in each component given by an *MS* variable.

2.1 Building the Modal Symbolic User Profile

According to [1], the construction of the *MS* descriptions of the user profile involves two steps: (a) *pre-processing* and (b) *generalization*. The general idea is (a) to build an *MS* description for each item evaluated by the user and (b) then aggregate these descriptions in some *MS* descriptions where each one represents a user interest.

The *pre-processing* step is necessary for both constructing the set of MS descriptions used to represent the user profile and comparing the user profile with a new item (in CB filtering) or with another user profile (important to step 2 of our recommendation algorithm). Let $x_i = (X_i^1,\ldots, X_i^p, C(i))$ be the description of an item i (i=1,...,n), where $X_i^j \subseteq D_j$ (j=1,...,p) is a subset of categories of the domain D_j of the

variable y_j and $C(i) \in D = \{1,\ldots,5\}$ indicates the user evaluation (grade) for this item. For each category $m \in X_i^j$, we can associate the following weight:

$$w(m) = \frac{1}{\left|X_i^j\right|} \tag{1}$$

where $|X_j|$ is the number of elements belonging to X_j (its cardinality). Then, the *MS* description of item i is $\tilde{x}_i = (\tilde{X}_i^1,\ldots,\tilde{X}_i^p, C(i))$, where $\tilde{X}_i^j = \tilde{X}_j(i) = (S_j(i), q_j(i))$ and $\tilde{X}_j$ is a *MS* variable. $S_j(i) = X_i^j$ is the support of the weighted distribution $q_j(i)$.

The *generalization* step aims to construct a suitable symbolic description of the user profile. In our approach, each user profile is formed by a set of sub-profiles. Each sub-profile is modeled by an MS description that summarizes the entire body of information taken from the set of items the user has evaluated with the same grade.

Formally, let u_g be the sub-profile of user u which is formed by the set of items that have been evaluated with grade g. Let $y_{u_g} = (Y_{u_g}^1,\ldots,Y_{u_g}^p)$ be the *MS* description of the sub-profile u_g, where $Y_{u_g}^j = (S_j(u_g), q_j(u_g))$, with $S_j(u_g)$ being the support of the weighted distribution $q_j(u_g), j = 1,\ldots,p$.

If $\tilde{x}_i = (\tilde{X}_i^1,\ldots,\tilde{X}_i^p, C(i))$, where $\tilde{X}_i^j = (S_j(i), q_j(i))$ (j=1,…,p), is the *MS* description of the item i belonging to u_g, the support $S_j(u_g)$ of $q_j(u_g)$ is defined as

$$S_j(u_g) = \bigcup_{i \in u_g} S_j(i) \tag{2}$$

Let $m \in S_j(u_g)$ be a category belonging to D_j and $|u_g|$ be the number of elements belonging to the set u_g. Then, the weight $W(m) \in q_j(u_g)$ of the category m is:

$$W(m) = \frac{1}{\left|u_g\right|} \sum_{i \in u_g} \delta(i,m), \quad where \quad \delta(i,m) = \begin{cases} w(m) \in q_j(i), & if\ m \in S_j(i) \\ 0, & otherwise \end{cases} \tag{3}$$

2.2 Comparing Modal Symbolic Profiles

The step compares two *MS* user profiles through a suitable function that measures the similarity between each *MS* description of user profiles. This function is then used to define the neighborhood of an active user.

Let $y_{u_g} = (Y_{u_g}^1,\ldots,Y_{u_g}^p)$ be the *MS* description of the sub-profile u_g of an active user. Also, let $y_{v_g} = (Y_{v_g}^1,\ldots,Y_{v_g}^p)$ be the *MS* description of the sub-profile v_g of a candidate neighbor for the active user. The comparison between the active user u and the candidate neighbor v is achieved through the following similarity function:

$$\psi(u,v) = \frac{\sum_{g \in \{1,2,3,4,5\}} h_g(y_{u_g}, y_{v_g}) * \left(1 - \phi(y_{u_g}, y_{v_g})\right)}{5} \tag{4}$$

where $h_1(y_{u_1}, y_{v_1}) = 3$, $h_2(y_{u_2}, y_{v_2}) = 2$, $h_3(y_{u_3}, y_{v_3}) = 1$, $h_4(y_{u_4}, y_{v_4}) = 4$, $h_5(y_{u_5}, y_{v_5}) = 5$ if $y_{u_5} \neq \varnothing$ and $y_{v_5} \neq \varnothing$, otherwise $h_g(y_{u_g}, y_{v_g}) = 0$. Although we have fixed the values of g due to our case study, this model may be easily adapted for other domains.

There are two hypotheses considered by function h_g. First, we agree that positive items are more useful in defining the neighbors of a user, as they may provide better suggestions than users who have similarities with the active user concerning negative

preferences. Additionally, we know that items with grade 5 are preferred over items with grade 4 and, also, items with grade 1 are more disliked than items with grade 2 or 3. We take this second hypothesis into account when measuring the similarities between users through different weights for each grade.

The function $\phi(y_{u_g}, y_{v_g})$ has two components: a context free component, which compares the sets $S_j(u_g)$ and $S_j(v_g)$; and a context depend component, which compares the weight distributions $q_j(u_g)$ and $q_j(v_g)$. This function is defined as:

$$\phi(y_{u_g}, y_{v_g}) = \frac{1}{p}\sum_{j=1}^{p}\left[\frac{\phi_{cf}(S_j(u_g), S_j(v_g)) + \phi_{cd}(q_j(u_g), q_j(v_g))}{2}\right] \tag{5}$$

where ϕ_{cf} measures the difference in position in cases where sets $S_j(u_g)$ and $S_j(v_g)$ are ordered; and ϕ_{cd} measures the difference in content between y_{u_g} and y_{v_g}.

Table 1 expresses the agreement (α and β) and disagreement (γ and δ) between the weight distributions $q_j(u_g)$ and $q_j(v_g)$.

Table 1. Comparison between the weight distributions $q_j(u_g)$ and $q_j(v_g)$

		User u_g	
		+ (Agreement)	- (Disagreement)
User v_g	+	$\alpha = \sum_{m \in S_j(u_g) \cap S_j(v_g)} w(m) \bullet \beta = \sum_{m \in S_j(u_g) \cap S_j(v_g)} W(m)$	$\gamma = \sum_{m \in \overline{S_j(u_g)} \cap S_j(v_g)} w(m)$
	-	$\delta = \sum_{m \in S_j(u_g) \cap \overline{S_j(v_g)}} w(m)$	

The context dependent component ϕ_{cd} is defined as:

$$\phi_{cd}(q_j(u_g), q_j(v_g)) = \frac{1}{2}\left(\frac{\gamma+\delta}{\alpha+\gamma+\delta} + \frac{\gamma+\delta}{\beta+\gamma+\delta}\right) \tag{6}$$

If the domain D_j of the categorical variable y_j is ordered, let $m_L = min(S_j(u_g))$, $m_U = max(S_j(u_g))$, $c_L = min(S_j(v_g))$ and $c_U = max(S_j(v_g))$. The join [6] $S_j(u_g) \oplus S_j(v_g)$ is defined as:

$$S_j(u_g) \oplus S_j(v_g) = \begin{cases} S_j(u_g) \bigcup S_j(v_g), & \text{if the domain } D_j \text{ is non ordered} \\ \{\min(m_L, c_L), \max(m_U, c_U)\}, & \text{otherwise} \end{cases} \tag{7}$$

The context dependent component ϕ_{cf} is defined as:

$$\phi_{cf}(S_j(u_g), S_j(v_g)) = \begin{cases} 0, & \text{if } S_j(u_g) \bigcap S_j(v_g) \neq \varnothing \\ \dfrac{\left|S_j(u_g) \oplus S_j(v_g)\right| - \left|S_j(u_g)\right| - \left|S_j(v_g)\right|}{\left|S_j(u_g) \oplus S_j(v_g)\right|}, & \text{otherwise} \end{cases} \tag{8}$$

2.3 Generating a Ranked List of Items

Now that we are able to compute the similarity between the active user u with each user in the database, we can do the 3rd step in a straightforward manner. Based on the

user neighborhood defined in the 3rd step, we can compute predictions for each unknown item in the repository, according to the following function:

$$\rho(u,i) = \bar{r}_u + \frac{\sum_{v=1}^{k} \left(r_{v,i} - \bar{r}_v\right) * \psi(u,v)}{\sum_{v=1}^{k} \psi(u,v)} \tag{9}$$

where u is the active user, i is an unknown item and k is the neighborhood size. We can present the ranked list of items according to the values produced by equation 9.

3 Experimental Evaluation

We use the Movielens (*movielens.umn.edu*) dataset joined with a content database crawled from IMDB (www.imdb.com) to perform experimental tests. This prepared dataset contains 91,190 explicit ratings between 1 to 5 from 943 different users for 1,466 movies. In this dataset, we selected all users that had evaluated at least 100 items of 1,466 available movies. These users were used in a test set to perform four different experiments concerning the type of training sets T={extratified (*E*), non-extratified (*NE*)} and the number m={5,10} of items provided in the training set for each user. The value of 30 was chosen for k following a recommendation of [4].

We ran an adapted version of the standard 10 fold cross-validation methodology. This adaptation consisted of arranging the training set and test set, respectively, in the proportion of 1 to 9 instead of 9 to 1 as done in the standard schema. This is compatible with the fact that the user does not furnish a sufficient amount of information in his/her first contact with the system.

The subject of our experimental analysis focused on the *Find Good Items* task, motivated by the hypothesis that this task is more useful than other available RS tasks in an E-commerce environment [5,7]. According to [5], the *half-life utility* [3] is the most appropriate metric for this type of task. Thus, it was adopted in our analysis. The following algorithms were executed in our tests:

1. (MSA) – Content-Based Information Filtering based on *MS* Data;
2. (CFA) – *kNN-CF* based on the Pearson Correlation;
3. (CMSA) - Collaborative Filtering based on Modal Symbolic User Profiles.

Table 2 displays the average ($\bar{x}$) and standard deviations (s) of *half-life utility* metric for all algorithms grouped by T={*E,NE*} and m={5,10}.

As seen in Table 2, the proposed methodology ($CMSA_{T=E}$) achieves the best accuracy recommendation lists. Moreover, we show with a confidence level of 0.1% that by giving just one item of each class (grade), the user gets better recommendation lists than those produced by CFA or MSA algorithms, even if they use the same acquiring strategy as in our methodology. This result is very interesting, as having good recommendations with just 5 items can help systems maintain loyal customers and get new ones. Another interesting result is that the observed standard deviation of the CFA and CMSA diminishes when the size of user profile is increased to 10. The reason for this behavior is that as more items are added to the user profile, precision increases in the estimation of user neighborhood. Consequently, better

recommendations can be provided by the system to users whose the profile was obscure when there was just 5 items. The most remarkable result is that CSMA reaches low standard deviations, thus implying more stable systems.

Table 2. Results of experiments grouped by T (type of training sets) and m (number of items in user profile) according to *half-life utility* metric

		MSA		CFA		CMSA	
T	*m*	$\bar{x}$	*s*	$\bar{x}$	*s*	$\bar{x}$	*s*
E	5	34,346	0,826	40,206	4,325	**63,924**	2,224
	10	31,786	1,161	58,088	1,991	63,589	2,001
NE	5	37,335	0,444	58,738	0,593	**61,482**	0,194
	10	32,467	0,657	59,731	0,333	60,000	0,157

4 Conclusions

In this paper we presented a suitable strategy for minimizing the problem of learning a user profile during first system usage. We demonstrate how our new method improves the quality of recommendation lists when there is little information on the user. As a possible future work we propose the comparison of our strategy with some *active learning* approaches. Another exciting work would be the combination of our strategy for acquiring preferences with other hybrid information filtering algorithms.

Acknowledgments. The authors would like to thank CNPq (Brazilian Agency) for its financial support.

References

1. Bezerra, B.L.D. and De Carvalho, F.A.T.: A symbolic approach for content-based information filtering. Information Processing Letters, Vol. 92 (1), 16 October 2004, 45-52.
2. Bock, H.H. and Diday, E.: Analysis of Symbolic Data. Springer, Heidelberg (2000).
3. Breese, J., Heckerman, D., and Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (1998) 43-52.
4. Herlocker, J., Konstan, J.A., Borchers, A., and Riedl, J.: An algorithmic framework for performing collaborative filtering. Proceedings of SIGIR (1999) 230-237.
5. Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.: Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems, Vol. 22, Issue 1 (2004) 5-53.
6. Ichino, M., Yaguchi, H.: Generalized Minkowsky Metrics for Mixed Feature Type Data Analysis. IEEE Transactions on System, Man and Cybernetics, Vol. 24 (1994) 698–708.
7. Schafer, J.B., Konstan, J.A., and Riedl, J.: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery, Vol. 5. (2001) 115-153.