Self-adjusting Hybrid Recommenders Based on Social Network Analysis

Alejandro Bellogín, Pablo Castells, Iván Cantador
Universidad Autónoma de Madrid, Escuela Politécnica Superior
Ciudad Universitaria de Cantoblanco, 28049 Madrid, Spain
{alejandro.bellogin, pablo.castells, ivan.cantador}@uam.es

ABSTRACT

Ensemble recommender systems successfully enhance recommendation accuracy by exploiting different sources of user preferences, such as ratings and social contacts. In linear ensembles, the optimal weight of each recommender strategy is commonly tuned empirically, with limited guarantee that such weights are optimal afterwards. We propose a self-adjusting hybrid recommendation approach that alleviates the social cold start situation by weighting the recommender combination dynamically at recommendation time, based on social network analysis algorithms. We show empirical results where our approach outperforms the best static combination for different hybrid recommenders.

Categories and Subject Descriptors: H.3.3 Information Search and Retrieval – *information filtering*.

General Terms: Algorithms, Performance, Experimentation.

Keywords: Hybrid recommender systems, social networks, graph theory, link analysis.

1. INTRODUCTION

Recommender Systems (RS) aim at automatically finding the most useful products or services for a particular user, providing a personalized list of items according to different input and attributes of users and items. State-of-the-art recommender systems are usually based on ratings and implicit feedback given by users about the items. Recently, due to the large number of social systems appearing in the so called Web 2.0, where friendship relations between people are explicit, "social contexts" exploitation has started to receive significant interest. In particular, social recommenders have started to be investigated that exploit social links between users in a community to suggest interesting items [2], [6]. On the other hand, Collaborative Filtering (CF) is one of the most popular strategies in RS [1], which makes use of explicit preferences to find similar-minded people according to the observed user interaction with the items, and presents suggestions on this basis.

Although social recommenders perform very well in some situations, they usually cannot provide any suggestion to users without a social context (social cold start). A solution to this problem is to exploit both sources of information, social and collaborative, by a hybrid ensemble recommendation approach. One of the most common forms of ensemble recommenders simply consists in a linear combination of recommenders. In this approach, the weights in the linear combination are generally the same for all users. However, not all users are equally active in terms of social connectivity. Thus, in this paper, we explore the adaptive adjustments of the ensemble coefficients on a user basis. More specifi-

cally, we explore a novel approach for self-adjusting the weight of each recommender by using social network analysis to balance the influence of each recommender. We report empirical results showing that our dynamic ensemble approach outperforms the best static combination in different hybrid recommenders.

2. SELF-ADJUSTING RECOMMENDERS

In [3], a detailed taxonomy is presented where hybrid recommenders are classified according to how they combine different recommendation techniques. In the following, we focus on the so called weighted hybrid recommenders, in which the scores of n individual techniques are aggregated by a linear combination. Without loss of generality, we focus on the case with $n = 2$, as follows:

$$s(u_m, i_n) = (1 - \lambda) \times s_1(u_m, i_n) + \lambda \times s_2(u_m, i_n)$$

with λ ranging between 0 and 1. This type of hybridization weights each recommender in a static way; i.e., once the value of λ is fixed, recommendations from each technique receive the same weight, independently of the target user. In this context, two main shortcomings arise. First, the optimal weight has to be found empirically by relying on current recommender performance, dataset characteristics, etc., which are subject to change [3]. Second, the optimal weight may not be the same for all users since the system gathers a different amount of information from each user, and thus, a recommender may be more useful than others in different situations.

In this paper, we explore a self-adjusting hybrid recommendation approach that makes use of adjusting factors to boost one of the combined recommenders for certain users. In our case, the value of λ is fixed at the user level, i.e. $\lambda = \lambda(u_m)$, aiming to promote the recommender that is expected to perform best for each particular user, e.g. because there is more meaningful available information for this recommender than for others. For this purpose, we analyze a particular hybrid configuration in which we combine social and CF recommenders. In this context, we propose to use graph-based measures as adjusting factors of the users' strength in the network, by balancing the influence of each recommender. More specifically, we apply these measures to set the weight of the social recommender, in such a way that when a user is very socially "active" (e.g., she plays a special role in the community), the recommendations from the social recommender are assumed to be more important than those from the CF recommender, and vice versa.

The adjusting factors used herein are based on vertex measures from graph theory. More specifically, we use user *degree*, i.e., the number of friends of each user in the network, *average neighbor degree*, and size of *two-hop neighborhood*, both described in [4]; *PageRank* and *HITS* scores, well-known measures of connectivity relevance within a social network; betweenness *centrality*, which is an indicator of whether a user can reach others on relatively short paths [5]; *clustering coefficient*, which measures the probability that the user's friends are friends themselves [8]; and *ego components size*, which is the number of connected components remaining when the user and her friends are removed [4].

3. EXPERIMENTS

We use the dataset provided in the social track of the CAMRa Challenge [7]. This dataset was gathered by the Filmtipset community, and contains social links between users, movie ratings, movie comments, and other attributes of users and movies. The set of users presented in the test set turns out to be very specific: every user has at least two friends, something not plausible in a real setting, where not every user has a social network. Because of that, we simulate a more realistic scenario to empirically compare social and CF recommenders in a fair environment where users with and without social contacts in their profiles are considered. We sampled randomly the same number of test users in the original test set (i.e., 439 users), but forcing them to have no friends, and include them in the new test set used in our experiments.

We evaluated a number of hybrid ensemble recommendation approaches combining CF and two different social recommenders. User-based CF [1] with a neighborhood size of 10 is denoted as *UB10*. One of the social recommenders is denoted as *PureSocial*. It is inspired by the approach presented in [6], and incorporates social information into the user-based CF model, by replacing the set of nearest neighbors with the active user's (explicit) friends. The other social recommender (denoted as *Personal*) is described in [2], where distances between users in the social graph are explicitly incorporated in the scoring formula as: $s(u_m, i_n) = \sum_{v \in X(u_m, L)} K^{-d(u_m, v)} rat(v, i_n)$. In this equation, $X(u_m, L)$ denotes the social tree of user u_m up to level L, and K is an attenuation coefficient of the social network that determines the extent of the effect of $d(u_m, v)$, i.e., the impact of the distance between two users in the social graph, weighted by the rating given by user v to item i_n, i.e., $rat(v, i_n)$. In our experiments, we used as distance d the one obtained by Dijkstra's algorithm, $K = 2$ and $L = 6$.

We evaluated three different recommender ensembles, depending on which social recommender was combined with *UB10*, and a threshold parameter in the number of common items required to decide when two users are considered neighbors. We denote the above hybrids as follows: H1 (*UB10* and *Personal*, with a threshold value of 2), H2 (*UB10* and *PureSocial*, threshold is 1), and H3 (the same as in H2 but with a threshold value of 2).

Table 1 shows the results obtained with the proposed self-adjusted hybridization technique, using the adjusting factors presented in the previous section. We only show P@5 and nDCG@5 results due to space constraints, but results are consistent for different cutoff values other than 5. In the table, we compare the results from the self-adjusted hybrid recommendations against those of the best statics, and a static with $\lambda = 0.5$, since this value is the natural choice (and best prior on average) in the absence of information about the recommender systems to be combined. We also include the performance for the extreme cases: $\lambda = 1$ (only social) and $\lambda = 0$ (collaborative).

In the table, we observe that the self-adjusted recommenders get a general advantage over the static configurations. For instance, the PageRank-based hybrid is better than statics in all cases (without statistical significance in H3 though), and all methods, except Centrality and Clustering on H3, improve over the 0.5 static –and the best posterior static in most cases. The improvements are less significant on the H3 ensemble. We hypothesize this can be due to the fact that the social algorithm in H3 performs significantly and systematically better than the CF, whereby any ensemble is suboptimal with respect to the best component (here, social). It is important to note that the best static is different for each ensemble (namely, $\lambda = 0.9$ for H1, 0.1 for H2, and 0.8 for H3), which further highlights the importance of this result, in that the best static is

not actually a real configuration: a best static configuration would require a manual tuning of λ, and even so, the table shows the posterior best static, which a manually tuned λ would not guarantee. Nonetheless, although no self-adjusting hybrid method is significantly better in every situation, *PageRank*, *HITS*, and *ego components size* factors seem to be safe alternatives.

Table 1. Performance results for the three ensembles tested. The best absolute value is underlined. Improvements over the best static are shown in bold font, and over the static 0.5 with italics. Statistical significant (p<0.05) differences between self-adjusted hybrid recommenders and static 0.5, best static, and both are marked with *, †, and ‡, respectively.

	P@5			nDCG@5		
	H1	H2	H3	H1	H2	H3
Average Neigh Deg	0.219*	0.092*	0.199	0.240*	0.097*	0.215
Centrality	0.222*	0.106‡	0.188†	0.242*	0.111‡	0.204†
Clustering coef	0.211*	0.094*	0.188†	0.231*	0.100*	0.202†
Degree	0.233‡	0.095*	0.197	0.256‡	0.099*	0.213
Ego Comp Size	0.227‡	0.096*	0.201‡	0.249‡	0.101*	0.215
HITS	0.225*	0.110‡	0.197	0.248*	0.114‡	0.212
PageRank	0.227‡	0.097*	0.200	0.247*	0.101*	0.216
Two Hop Neigh	0.229*	0.093*	0.195	0.250‡	0.100*	0.212
Static 0.5	0.186	0.077	0.189	0.205	0.081	0.206
Best static	0.218	0.091	0.199	0.239	0.096	0.215
CF Component	0.076	0.043	0.076	0.082	0.047	0.082
Social component	0.372	0.055	0.270	0.409	0.053	0.298

Finally, it is worth noting that single approaches obtain lower accuracy values than hybrids on equal terms since they deal poorly with the social cold start situation if we measure the performance for all the users (the social component evaluated in this way is only able to recommend items for 55% of the test users, i.e., its "true" P@5 value for H1 should be 0.206). The empirical results thus suggest that significant improvements can be drawn from the proposed self-adjustment strategy based on graph-based social factors. The herein presented adjusting criteria only make use of social network structures; we shall explore alternative factors in the future, such as distributional properties of user ratings.

4. REFERENCES

[1] G. Adomavicius, A. Tuzhilin. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE TKDE*, **17**(6): 734–749, June 2005.

[2] D. Ben-Shimon, A. Tsikinovsky, L. Rokach, A. Meisles, G. Shani, L. Naamani. Recommender system from personal social networks. In *Proceedings of AWIC'2007*, 47–55, 2007.

[3] R. Burke. Hybrid recommender systems: survey and experiments. *UMUAI* **12**(4): 331–370, November 2002.

[4] M. De Choudhury, W. A. Mason, J. M. Hofman, D. J. Watts. Inferring relevant social networks from interpersonal communication. In *Proceedings of WWW'10*, 301–310, 2010.

[5] L.C. Freeman. A set of measures of centrality based on betweenness. *Sociometry* **40**(1), 35–41, 1977.

[6] F. Liu, H.J. Lee. Use of social network information to enhance collaborative filtering performance. *Expert Systems with Applications* **37**(7): 4772–4778, July 2010.

[7] A. Said, S. Berkovsky, E. W. De Luca. Putting things in context: challenge on context-aware movie recommendation. In *Proceedings of the RecSys'10 CAMRa Challenge*, 2–6, 2010.

[8] D. J. Watts, S. Strogatz. Collective dynamics of 'small-world' networks. *Nature* **393**: 440–442, June 1998.