A Comparative Analysis of Trust-Enhanced Recommenders for Controversial Items

Patricia Victor and **Chris Cornelis**
Dept. of Applied Mathematics & Computer Science, Ghent University
Krijgslaan 281 (S9), 9000 Gent, Belgium

Martine De Cock* and **Ankur M. Teredesai**
Institute of Technology, University of Washington, Tacoma
1900 Pacific Ave, Tacoma, WA, USA

Abstract

A particularly challenging task for recommender systems (RSs) is deciding whether to recommend an item that received a variety of high and low scores from its users. RSs that incorporate a trust network among their users have the potential to make more personalized recommendations for such controversial items (CIs) compared to collaborative filtering (CF) based systems, provided they succeed in utilizing the trust information to their advantage. In this paper, we formalize the concept of CIs in RSs. We then compare the performance of several well-known trust-enhanced techniques for effectively personalizing the recommendations for CIs versus random items in the RS. Furthermore, we introduce a new algorithm that maximizes the synergy between CF and its trust-based variants, and show that the new algorithm outperforms other trust-based techniques in generating rating predictions for CIs.

Introduction

The wealth of information available on the web has made it increasingly difficult to find what one is really looking for. This is particularly true for exploratory queries where one is looking for opinions and views. Hence, it comes as no surprise that personalization applications to guide the search process are gaining tremendous importance. One particulary interesting set of applications that address this problem are online *recommender systems* (RSs), which, based on information about their users' profiles and relationships, suggest items of interest (Resnick & Varian 1997).

RSs are often used to accurately estimate the degree to which a particular user (from now on termed the target user) will like a particular item (termed the target item). Most widely used methods for making recommendations are either content-based or collaborative filtering (CF) methods. Content-based methods suggest items similar to the ones that the user previously indicated a liking for. Hence, these methods tend to have their scope of recommendations limited to the immediate neighbourhood of the users' past purchase history or rating record for items. RSs can be improved significantly by (additionally) using CF (Resnick et al. 1994), which typically works by identifying users whose tastes are similar to those of the target user (i.e., neighbours) and by computing predictions that are based on the ratings of these neighbours. The advanced recommendation techniques that we discuss in this paper adhere to the CF paradigm, in the sense that a recommendation for a target item is based on ratings by other users for that item, rather than on an analysis of the content of the item.

The growing popularity of open social networks and the trend to integrate e-commerce applications with RSs have generated a rising interest in *trust-enhanced RSs*. Recommendations generated by such systems are based on information coming from an (online) trust network, i.e., a social network in which the members of the community can express how much they trust each other. A typical example is the e-commerce site Epinions.com, which maintains a trust network by asking its users to indicate which members they trust, i.e., their personal 'web of trust' (WOT). Trust-enhanced RSs use the knowledge that originates from such networks to generate recommendations: users receive recommendations for items rated highly by people in their WOT, or even by people who are trusted by these WOT members, etc. (see e.g. (Golbeck et al. 2005; Massa et al. 2007; O'Donovan et al. 2005))

If all users who have rated an item liked it very much, it is reasonable to assume that a new user might like it too. In such cases, a trivial algorithm can achieve high accuracy. However, the more challenging items are those that receive a variety of high and low scores, reflecting disagreement about them. We call such items *controversial items* (CIs). More than in any other case, a recommendation from a target user needs to be truly personalized when the target item under consideration is controversial; i.e., when an item has both 'ardent supporters' and 'motivated adversaries'.

In order to be effective as well as efficient, a recommender system needs to be able to identity whether the target item is controversial or not, and to apply the most suitable recommendation method. Our first step in this direction is the proposition of an operational definition of the CI concept that is applicable to a wide variety of RSs. Furthermore, we compare the performance of CF and several trust-enhanced algorithms on CIs, including the proposals by (Golbeck 2005), (Massa & Avesani 2007) and (O'Donovan & Smyth

*On leave from Ghent University

2005), along with a new approach that combines aspects of CF with trust information.

In the following section, we analyze the controversiality level of two Epinions data sets and explain why a classical measure like standard deviation is insufficient to detect their true CIs. In the third section, we discuss the rationale behind the aforementioned algorithms, while their coverage and accuracy performance is analyzed in the fourth section, on a set of CIs and of randomly selected items.

Controversial Items in Epinions

Epinions.com is a popular e-commerce site where users can write reviews about consumer products and assign a rating to the products and the reviews. Two Epinions data sets are often used for experimenting with trust-enhanced RSs. The first one was collected by (Massa & Bhattacharjee 2004) in a 5-week crawl and contains 139 738 products that are rated by 49 290 users in total. The second data set was compiled by (Guha *et al.* 2004): this large set contains 1 560 144 reviews that received 25 170 637 ratings by 163 634 different users. Both products and reviews are rated on a scale from 1 to 5. Most items receive very high scores, in fact, 45.3% of all products in Massa's data set received the highest possible evaluation, and over 75% of all ratings in Guha's data set are 'most helpful'. This means that a trivial algorithm that always predicts 5, or that uses the average score for the item as its prediction, can achieve high accuracy. However, such recommendation strategies have difficulties coping with CIs.

A straightforward way to detect a controversial item in a data set is to inspect the standard deviation of the ratings for each item i (see e.g. (Massa & Avesani 2007)); we denote this by $\sigma(i)$. However, $\sigma(i)$ does not convey the full picture of controversiality. E.g., consider the ratings for items i_1, i_2 and i_3 in Table 1; $f_i(k)$ denotes the number of times item i received rating k. Intuitively, item i_2 seems the most controversial since it received ratings all over the range, while there is more agreement on i_1 and i_3 that are liked by a majority of the users. Still, in this example the most controversial item according to intuition has the lowest σ, which illustrates that by itself standard deviation does not always reflect the controversiality of an item adequately.

We propose a new measure that looks at how often adjacent scores appear w.r.t. the total number of received ratings. The underlying intuition is that different scores that are close to each other reflect less disagreement than different scores that are on opposite ends of the scale. In a system with discrete ratings on a scale from 1 to M, the size of the window in which adjacent scores are being considered can vary from 1 to M. In the definition below, the granularity of the window is controlled by a parameter Δ.

Definition 1 (Level of Disagreement) *For a system with discrete ratings on a scale from* 1 *to* M*, let* $\Delta \in \{1, \ldots, M\}$*. The* Δ*-level of disagreement for an item* i *is defined as*

$$(\alpha@\Delta)(i) = 1 - \max_{a \in \{1,\ldots,M-\Delta+1\}} \left(\frac{\sum_{k=a}^{a+\Delta-1} f_i(k)}{\sum_{k=1}^{M} f_i(k)} \right)$$

Table 1: Example of three items and their ratings.

	$f_i(1)$	$f_i(2)$	$f_i(3)$	$f_i(4)$	$f_i(5)$	$\sigma(i)$	$(\alpha@2)(i)$
i_1	1	1	0	3	5	1.34	0.20
i_2	1	2	3	2	1	1.15	0.44
i_3	1	0	0	4	4	1.20	0.11

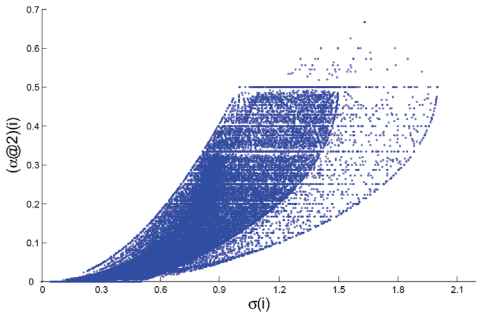

Figure 1: $\alpha(i)$ vs. $\sigma(i)$ in Guha et al.'s data set

with $f_i(k)$ *the number of times that item i received rating k.*

A window size of $\Delta = 1$ means that scores are considered in isolation. A window size of $\Delta = 2$ means each score is considered with a neighbouring score, i.e. scores are considered in groups of 2. If $\Delta = M$, then $(\alpha@\Delta) = 0$, since there can be no disagreement when all ratings are considered together. The last column of Table 1 displays the 2-level of disagreement for items i_1, i_2 and i_3, indicating that there is more disagreement on i_2 than on i_1 and i_3.

Fig. 1 depicts the standard deviation (horizontal axis) and the 2-level of disagreement (vertical axis) of items in Guha's data set. While a small $\sigma(i)$ typically entails a small $\alpha@\Delta 2)(i)$, there is considerable variation for high values of σ (and vice versa). This highlights that σ and $\alpha@\Delta$ are significantly different measures that can be used together to define the concept of a controversial item.

Since the controversiality of items with few ratings may be due to chance, we include a popularity threshold in our definition (in which f_i denotes the number of times item i has been evaluated) to ensure real controversiality:

Definition 2 ($(\sigma^\star, \alpha^\star, \beta^\star)$-controversial) *We call item* i $(\sigma^\star, \alpha^\star, \beta^\star)$*-controversial iff* $\sigma(i) \geq \sigma^\star$*,* $(\alpha@2)(i) \geq \alpha^\star$ *and* $f_i \geq \beta^\star$*.*

Applying this definition to the data set requires a parameter selection that is adapted to its characteristics, e.g., the predominance of rating value 5. For example, for Massa's data set, we choose a $\sigma^\star$ value of 1.4, an $\alpha^\star$ value of 0.4, and a $\beta^\star = 20$ times, which yields 266 CIs.

Recommendation Strategies

RSs come in many flavours, including content-based, collaborative filtering and trust-based methods; the latter two being the ones most relevant to our current efforts.

In CF algorithms (Resnick *et al.* 1994), a rating of

target item i for target user a can be predicted using a combination of the ratings of the neighbours of a (similar users) that are already familiar with item i. The classical CF-formula is given by (CF). The unknown rating $p_{a,i}$ for item i and target user a is predicted based on the mean $\overline{r}_a$ of ratings by a for other items, as well as on the ratings $r_{u,i}$ by other users u for i. The formula also takes into account the similarity $w_{a,u}$ between users a and u, usually calculated as Pearson's Correlation Coefficient (PCC) (Herlocker *et al.* 2004). In practice, most often only users with a positive correlation $w_{a,u}$ who have rated i are considered. We denote this set by R^+.

$$p_{a,i}^{(1)} = \overline{r}_a + \frac{\sum_{u \in R^+} w_{a,u}(r_{u,i} - \overline{r}_u)}{\sum_{u \in R^+} w_{a,u}} \qquad \text{(CF)}$$

Trust-enhanced RSs often use information coming from a trust network in which users are connected by trust scores indicating how much they trust each other; in general, $t_{a,u}$ is a number between 0 and 1 indicating to what extent a trusts u.

Trust-based weighted mean refines the baseline recommendation strategy of simply computing the average rating for the target item, it is natural to assign more weight to ratings of highly trusted users. See (T1), in which R^T represents the set of users who evaluated i and for which the trust score $t_{a,u}$ exceeds a given threshold value. This formula is at the heart of Golbeck et al.'s TidalTrust (2005).

$$p_{a,i}^{(3)} = \frac{\sum_{u \in R^T} t_{a,u} r_{u,i}}{\sum_{u \in R^T} t_{a,u}} \qquad \text{(T1)}$$

Another class of trust-enhanced systems is tied more closely to the collaborative filtering algorithm. O'Donovan et al.'s *trust-based filtering* (2005) adapts Formula (CF) by only taking into account trustworthy neighbours, i.e., users in $R^{T+} = R^T \cap R^+$.

$$p_{a,i}^{(4)} = \overline{r_a} + \frac{\sum_{u \in R^{T+}} w_{a,u}(r_{u,i} - \overline{r_u})}{\sum_{u \in R^{T+}} w_{a,u}} \qquad \text{(T2)}$$

In other words, we only consider the users who are trusted by the target user a, and have a positive correlation with a.

Instead of a PCC-based computation of the weights, one can also infer the weights through the relations of the target user in the trust network, as in (T1). We call this alternative for CF *trust-based CF*; see (T3) which adapts (T2) by replacing the PCC weights $w_{a,u}$ by the trust values $t_{a,u}$.

$$p_{a,i}^{(5)} = \overline{r_a} + \frac{\sum_{u \in R^T} t_{a,u}(r_{u,i} - \overline{r_u})}{\sum_{u \in R^T} t_{a,u}} \qquad \text{(T3)}$$

This method is known as Massa et al.'s MoleTrust (2007).

A very important feature of trust-enhanced RSs is their use of *trust propagation operators*: mechanisms to estimate the trust transitively by computing how much trust an agent has in another agent. Both TidalTrust and MoleTrust invoke trust propagation to expand the set R^T. However, the way they implement this operation differs significantly, see (Golbeck 2005) and (Massa & Avesani 2007). Although trust propagation is not used in (T2) (O'Donovan & Smyth 2005), it is of course possible to do so; since trust scores are not used explicitly in (T2), we only need to specify how propagation enlarges the set R^T.

Table 2: Performance of trust-based algorithms

ALGORITHM	*Controversial items (CIs)*			*Random items (RIs)*		
	COV	*MAE*	*RMSE*	*COV*	*MAE*	*RMSE*
CF	81	1.34	1.58	79	0.84	1.12
Trust-based weighted mean	41	1.33	1.70	34	0.87	1.24
Trust-based filtering	25	1.35	1.71	22	0.85	1.18
Trust-based CF	40	1.32	1.65	34	0.86	1.19
EnsembleTrustCF	84	1.32	1.57	81	0.83	1.11
Prop Trust-based weighted mean	76	1.37	1.69	72	0.90	1.23
Prop Trust-based filtering	57	1.36	1.64	53	0.86	1.16
Prop Trust-based CF	76	1.32	1.56	72	0.84	1.12
Prop EnsembleTrustCF	84	1.32	1.57	81	0.83	1.11

It has been demonstrated that including trust in the recommendation process significantly improves accuracy (Golbeck 2005; Massa & Avesani 2007). On the other hand, the coverage of algorithms (T2) and (T3) remains lower than their classical counterpart (CF) (Massa & Avesani 2007). In order to maximize the synergy between CF and its trust-based variants, we propose *EnsembleTrustCF*:

$$p_{a,i}^{(6)} = \overline{r_a} + \qquad \text{(T4)}$$
$$\frac{\sum_{u \in R^T} t_{a,u}(r_{u,i} - \overline{r_u}) + \sum_{u \in R^+ \setminus R^T} w_{a,u}(r_{u,i} - \overline{r_u})}{\sum_{u \in R^T} t_{a,u} + \sum_{u \in R^+ \setminus R^T} w_{a,u}}$$

The rationale behind this strategy, which combines (CF) and (T3), is that we take into account all possible ways to obtain a positive weight for a user who has rated the target item, favouring a trust relation over a PCC-based one; in particular, if a user can be reached by a direct or indirect trust relation, we use this value instead of the PCC to obtain the user's weight. In this way, we retain the accuracy benefit by first looking at the trusted users, while on the other hand the coverage can increase by taking into account neighbours for which no trust information is available. This new strategy is guaranteed to perform as least as good as (CF) and (T3) in terms of coverage.

Experimental Results

In Epinions, users can evaluate other users by including them in their WOT, which is a list of users whose reviews and ratings were consistently found to be valuable. Massa's data set contains 487 003 such trust statements. Due to space restrictions, we only report results on this set; results on Guha's can be found in (Victor *et al.* 2009). Note that the data sets only contains binary trust values, hence in our experiments $t_{a,u}$ in (T1)–(T4) can take on the values 0 (absence of trust) and 1 (full presence) only.

To measure the performance of RSs, we work with the leave one out method. In particular, we use two well-known accuracy measures, viz. mean absolute error (MAE) and root mean squared error (RMSE) (Herlocker *et al.* 2004). Since reviews are rated on a scale from 1 to 5, the extreme values that MAE/RMSE can reach are 0 and 4. Besides

accuracy, we also consider coverage: during the leave one out we count how many predictions can be generated for the hidden scores. To compare the performance achieved for CIs with the performance that can be obtained in general, we also present the average coverage and accuracy for 266 randomly selected 'popular' items (RIs) (that have been evaluated at least 20 times). Table 2 shows the relative coverage (COV) and accuracy (MAE, RMSE) for Massa's data set. For simplicity, we only consider one-step propagation in this paper. For (PT1) and (PT3), we maintained the propagation strategy used in TidalTrust and MoleTrust respectively, while for (PT2) we added a user to R^T if he belongs to the WOT of the target user a, or is directly trusted by a WOT member of a. For (PT4), we assign gradual propagated trust weights $t_{a,u} = (PCC + 1)/2$. In this way, users u who cannot be reached through a direct trust relation are still rewarded for their presence in a's propagated WOT.

Without propagation, it is clear that the coverage of (CF) and (T4) is superior to that of the others, and approaches the maximal value. This is due to the fact that PCC information is, in general, more readily available than direct trust information (there are normally more users for which a positive correlation with the target user a can be computed than users in a's WOT). Our new algorithm EnsembleTrustCF (T4) is most flexible, since having either some trust or a positive correlation is sufficient to make a prediction. On the other hand, (T2), which also uses PCC weights, is the most demanding strategy because it requires users in a's WOT who have already rated two other items in common with a (otherwise the PCC can not be computed). In between these extremes, the coverage for (T1) is a bit higher than that of (T3) because the latter can only generate predictions for target users who have rated at least two items (otherwise the average rating for the target user can not be computed).

This ranking of approaches in terms of coverage still applies when propagated trust information is taken into account, but note that the difference with CF has shrunk considerably. In particular, thanks to trust propagation, the coverage increases with more than 30%, except for EnsembleTrustCF, for which the unpropagated version continues to score better than the propagated versions of (T1)–(T3).

It is clear that generating good predictions for controversial items is much harder than for randomly chosen items. When focusing on the MAE for CIs, we notice that, without propagation, almost all trust-enhanced approaches yield better results than CF, which is in accordance with the observations made in (Golbeck 2005; Massa and Avesani 2007). This can be attributed to the accuracy/coverage trade-off: a coverage increase is usually at the expense of accuracy, and vice versa. It also becomes clear when taking into account trust propagation: as the coverage of (PT1–3) nears that of (CF) and (T4), so do the MAEs. However, the RMSEs give us a different picture: those of the trust-enhanced approaches are generally higher than that of CF; recall that a higher RMSE means that more large prediction errors occur. Also remark that the propagated algorithms achieve lower RMSEs than their unpropagated counterparts.

We can also observe that our new algorithm is a valuable asset in the domain of trust-enhanced techniques. EnsembleTrustCF beats or matches all other algorithms on accuracy; MAE and RMSE, with and without propagation, and for both CIs and RIs. Moreover, taking into account the much higher coverage that (T4) achieves (for unpropagated algorithms at least the double), it is fair to state that EnsembleTrustCF is the real winner on Massa's data set.

Conclusions

We have provided a comparative analysis of the performance of collaborative filtering (CF) and trust-enhanced recommendation algorithms for controversial and random items. A coverage and accuracy based comparison shows no clear winner among the three state-of-art trust-enhanced strategies proposed by (Massa & Avesani 2007), (O'Donovan & Smyth 2005), and (Golbeck 2005). However, by combining the best of both the CF and the trust world, we have introduced EnsembleTrustCF, a new algorithm that achieves higher coverage than other trust-based methods while it still retains the accuracy benefit of the latter strategies.

Acknowledgements

Patricia Victor and Chris Cornelis thank the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT) and the Research Foundation-Flanders (FWO) resp. for funding their research. Part of this work was carried out during a visit of Patricia Victor and Martine De Cock at UW Tacoma, supported by IWT and FWO resp.

References

Golbeck, J. 2005. *Computing and applying trust in web-based social networks*. Ph.D. Dissertation.

Guha, R.; Kumar, R.; Raghavan, P.; and Tomkins, A. 2004. Propagation of trust and distrust. In *Proc. of WWW04*, 403–412.

Herlocker, J.; Konstan, J.; Terveen, L.; and Riedl, J. 2004. Evaluating collaborative filtering recommender systems. *ACM T Inform Syst* 22:5–53.

Massa, P., and Avesani, A. 2007. Trust-aware recommender systems. In *Proc. of RECSYS*, 17–24.

Massa, P., and Bhattacharjee, B. 2004. Using trust in recommender systems: an experimental analysis. *LNCS* 2995:221–235.

O'Donovan, J., and Smyth, B. 2005. Trust in recommender systems. In *Proc. of IUI2005*, 167–174.

Resnick, P., and Varian, H. 1997. Recommender systems. *Commun ACM* 40:56–58.

Resnick, P.; Iacovou, N.; Suchak, M.; Bergstorm, P.; and Riedl, J. 1994. Grouplens: An open architecture for collaborative filtering of netnews. In *Proc. of CSCW08*, 175–186.

Victor, P.; Cornelis, C.; Cock, M. D.; and Teredesai, A. 2009. Trust- and distrust-based recommendations for controversial reviews. In *Proc. of WEBSCI09*.