Trust No One: Evaluating Trust-based Filtering for Recommenders

John O'Donovan, Barry Smyth
Adaptive Information Cluster
Department of Computer Science
University College Dublin
Belfield, Dublin 4, Ireland
john.odonovan@ucd.ie, barry.smyth@ucd.ie

Abstract

To be successful recommender systems must gain the trust of users. To do this they must demonstrate their ability to make reliable predictions. We argue that collaborative filtering recommendation algorithms can benefit from explicit models of trust to inform their predictions. We present one such model of trust along with a cost-benefit analysis that focuses on the classical trade-off that exists between recommendation coverage and prediction accuracy.

1 Introduction

Recommender systems have been developed as a solution to the well documented information overload problem. [Resnick *et al.*, 1994], [Breese *et al.*, 1998]. These systems employ techniques from user profiling, machine learning and information filtering to produce individual recommendations of items to suit users' requirements. Collaborative filtering (CF) recommenders operate on the assumption that similar users share similar tastes; recommendations are generated for a target user by analysing the rating histories of a set of suitable recommendation partners.

The traditional CF approach relies heavily on the similarity between the target user and its partner as a way to weight each partner's predictions [Resnick *et al.*, 1994]. In this paper we propose that, in addition, it is possible to model the trustworthiness of these partners, and to use this as another factor to influence their prediction contributions. Indeed the idea of explicitly modeling and using trust in filtering tasks is becoming increasingly popular. For example, [Golbeck and Hendler, 2004] presents a trust-based email filter, trust scores in this system are calculated through inference and propagation, of the form $(A \Rightarrow B \Rightarrow C) \Rightarrow (A \Rightarrow C)$, where A, B and C are users with interpersonal trust scores. The TrustMail application [Golbeck and Hendler, 2004] looks up an email sender in the reputation/trust network, and provides an inline rating for each mail. These trust values can tell a user if a mail is important or unimportant. Trust values in this system can be defined with respect to a certain topic, or on a general level, in a similar manner to work in [O'Donovan and Smyth, 2005a] and [O'Donovan and Smyth, 2005b].

[Avesani *et al.*, 2004] describe a trust-based recommender system in the skiing domain. However these approaches rely on models of trust that are built from the direct feedback of users; in short, individual users are expected to indicate those partners that they place the most trust in and a trust model is generated from the resulting graph of relationships.

[Massa and Bhattacharjee, 2004] build a trust model directly from explicit user-provided trust ratings. This work is carried out using the popular *epinions.com* service. *Epinions.com* is a web site that allows users to review various items (cars, books, music, etc.). In addition they can assign a trust rating to reviewers based on the degree to which they have found them to be helpful and reliable in the past. [Massa and Bhattacharjee, 2004] argue that this trust data can be extracted and used as part of the recommendation process, especially as a means to relieve the sparsity problem (lack of overlapping user ratings) that has hampered traditional collaborative filtering techniques [O'Sullivan *et al.*, 2002]. [Massa and Bhattacharjee, 2004] argue that it is possible to compare users according to their degree of connectedness in the trust-graph encoded by *Epinions.com*, but do not show that this method of comparison maintains recommendation accuracy.

Our benchmark algorithm uses Resnick's standard prediction formula which is reproduced below as Equation 1; see also [Resnick *et al.*, 1994]. In this formula $c(i)$ is the rating to be predicted for item i in consumer profile c and $p(i)$ is the rating for item i by a producer profile p who has rated i. In addition, $\overline{c}$ and $\overline{p}$ refers to the mean ratings for c and p respectively. The weighting factor $sim(c, p)$ is a measure of the *similarity* between profiles c and p, which is traditionally calculated as Pearson's correlation coefficient.

$$c(i) = \overline{c} + \frac{\displaystyle\sum_{p \epsilon P(i)} (p(i) - \overline{p}) sim(c, p)}{\displaystyle\sum_{p \epsilon P_i} |sim(c, p)|} \tag{1}$$

As we have seen above Resnick's prediction formula discounts the contribution of a partner's prediction according to its degree of similarity with the target user so that more similar partners have a large impact on the final ratings prediction.

We argue that there is another factor which might be used in conjunction with similarity to influence recommendation and prediction. We believe that the reliability of a partner profile to deliver accurate recommendations in the past is another important factor, one that we refer to as the *trust*. Intuitively, if a profile has made lots of accurate predictions in the past, then they can be viewed as more trustworthy than another profile that has made many poor predictions. We describe a computational model of trust that can be generated unobtrusively, during the normal operation of a CF recommender system, by mining the recommendation histories of different recommendation partners. We re-evaluate work in [O'Donovan and Smyth, 2005b] which shows that trust-based methods can improve prediction accuracy when compared to existing CF approaches. However, we describe a more comprehensive cost-benefit analysis by considering three accuracy benefits against changes in recommendation coverage.

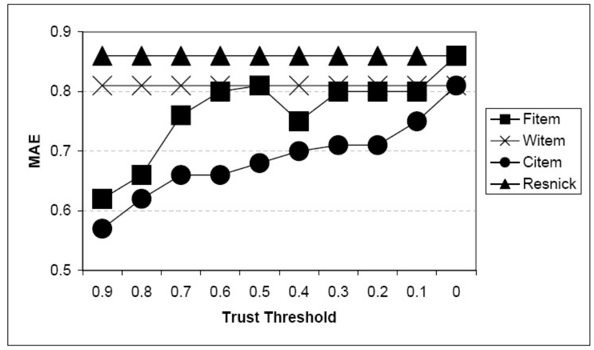

Figure 1: Recommendation Error.

2 A Computational Model of Trust

Intuitively, if a recommendation partner (profile) has made many good predictions in the past, it can be viewed as more trustworthy than one with many poor predictions. The trust model [O'Donovan and Smyth, 2005b] is based on this idea. We differentiate between profiles generating recommendations (*producer profiles*) and those receiving recommendations (*consumer profiles*) in a particular recommendation session. To generate a predicted rating, $p(i)$, for item i for some consumer c, conventional CF systems draw on the services of a number of producer profiles, combining their individual recommendations according to some suitable function, such as Resnick's formula. (see Equation 1). Our trust model depends on whether these predicted ratings are correct relative to the true ratings of the consumer, $c(i)$; see Equation 2.

$$Correct(i,p,c) \Leftrightarrow |p(i) - c(i)| < \epsilon \quad (2)$$

2.1 Item-Level Trust

We define the *item-level* trust for each producer p with respect to a given profile item i to be the percentage of times that p has made a *correct* rating prediction for i across some set of consumers; see Equation 3. To do this we consider the rating that p alone predicts for i, for the consumer in question. We define the $RecSet(p)$ (Equation 4) to be the total set of rating predictions that p has made; each (r_k, i_k) refers to a rating prediction, r_k that p has made for item i_k. Similarly, $CorrSet(p)$ is the subset of these ratings that are considered to be correct; Equation 5.

$$Trust^I(p,i) = \frac{|\{(r_k, i_k) \in CorrSet(p) : i_k = i\}|}{|\{(r_k, i_k) \in RecSet(p) : i_k = i\}|} \quad (3)$$

$$RecSet(p) = \{(r_1, i_1), ..., (r_n, i_n)\} \quad (4)$$

$$CorrSet(p) = \{(c_k, i_k) \in RecSet(p) : Correct(i_k, p, c_k)\} \quad (5)$$

Thus, the trust of p in relation to item i is a measure of how often p's predicted ratings for i have been considered correct in the past. This information can be accumulated during the normal course of operation of a CF recommender system in a variety of ways. For example, users could be asked their opinions of the predicted ratings, or this might be confirmed by evaluating the user's actions on the basis of the rating predictions; if a user buys a highly rated item then we might assume that the high rating was justified.

2.2 Trust-Based Recommendation

We incorporate our trust-model into CF by modifying the standard *Resnick* prediction algorithm in 3 ways to produce 3 different trust-based variations. Resnick's standard prediction formula is given in Equation 1. Normally it uses the similarity between the target user profile and each recommendation partner profile to weight their prediction contributions, shown as $sim(c,p)$ in Equation 1; Equation 6 shows our modifications to this standard equation by adding the $w(c,p,i)$ weighting term. Our first variation (*WItem*) modifies this so that the weighting term is a combination of item trust and profile similarity; we use the harmonic mean of trust and similarity. The *FItem* approach differs in that it uses profile similarity as the weight factor, but filters out profiles that fall below a given trust level for the target item prior to recommendation. Finally, the *CItem* approach uses the obvious combination of *WItem* and *FItem*.

$$r(i) = \overline{r} + \frac{\sum_{p \epsilon P(i)} (p(i) - \overline{p}) w(c,p,i)}{\sum_{p \epsilon P(i)} |w(c,p,i)|} \quad (6)$$

3 Evaluation

For the following evaluation we use the 943 profiles from the MovieLens data-set, split into 80% as training profiles and the remaining 20% as test profiles. During training we use a leave-one-out approach to build our trust model over the training profiles. Briefly, each training profile is used as a consumer with the remaining acting as producers. Item-level trust values are computed on the basis of the correctness or otherwise of the producer predictions. During testing, we evaluate the predictions of the training profiles for each of the items in the test profiles using our 4 basic algorithms (*Resnick*, *WItem*, *FItem*, *CItem*).

In this evaluation we are especially interested in the tradeoff between the coverage of a recommender (the percentage of items that a rating can be predicted for) (Figure 2) and

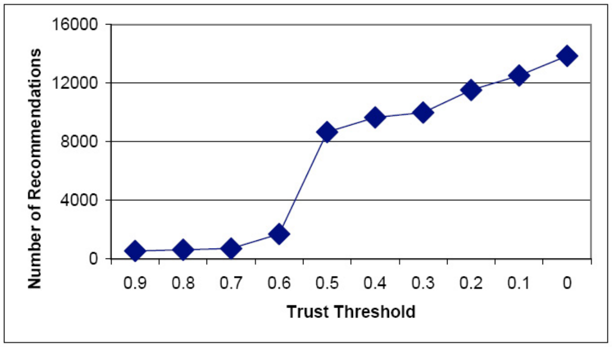

Figure 2: Recommendation Coverage.

the error over these predictions (Figure 1). The error graph shows a positive response to error for the trust-based methods, especially those that employ trust-based filtering and in particular for the higher trust-levels. For example, at a trust level of 0.9 only those profiles that have previously been correct 90% or more of the time that they have been called upon to rate an item, are included as recommendation partners for the filter-based approaches (*FItem* and *CItem*). And for these approaches we see significant error reductions of up to 57% compared to the baseline Resnick. However, the coverage results indicate that these error reductions come at a cost. In particular, the minimal error rates at the highest trust thresholds reduce coverage by over 90%, which is unlikely to be acceptable in most recommendation scenarios. However, for trust thresholds below 0.5 we get significant error reductions while preserving coverage to a reasonable degree. In particular, the error for *CItem* is seen to drop most rapidly up to a trust threshold of 0.2, at which point it offers 85% coverage and an error of 0.71; Resnick's error is 22% higher than this.

4 Conclusions

We believe that computational models of trust can improve the effectiveness of recommender systems. We have shown that by integrating an item-level model of trust into standard collaborative filtering we can increase accuracy by up to 57% by using only the top 1% most trustworthy profiles as recommendation partners. While this benefit comes at a significant coverage cost, more reasonable coverage can be achieved with reduced error rates by less drastic filtering thresholds. In addition to improving prediction accuracy, we believe that this trust-based approach may make recommenders more robust to attack by malicious users, as discussed in [O'Mahony *et al.*, 2002], [Levien, 2003] and [Kushmerick, 2002]. This is a matter that we will investigate as part of future work.

5 Acknowledgments

This material is based on works supported by Science Foundation Ireland under Grant No. 03/IN.3/I361

References

[Avesani *et al.*, 2004] Paolo Avesani, Paolo Massa, and Roberto Tiella. Moleskiing: a trust-aware decentralized recommender system. *1st Workshop on Friend of a Friend, Social Networking and the Semantic Web. Galway, Ireland*, 2004.

[Breese *et al.*, 1998] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Gregory F. Cooper and Serafín Moral, editors, *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pages 43–52, San Francisco, July 24–26 1998. Morgan Kaufmann.

[Golbeck and Hendler, 2004] Jennifer Golbeck and James Hendler. Accuracy of metrics for inferring trust and reputation in semantic web-based social networks. In *Proceedings of EKAW'04*, pages LNAI 2416, p. 278 ff., 2004.

[Kushmerick, 2002] N. Kushmerick. Robustness analyses of instance-based collaborative recommendation. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the European Conference on Machine Learning, Helsinki, Finland.*, volume 2430, pages 232–244. Lecture Notes in Computer Science Springer-Verlag Heidelberg, 2002.

[Levien, 2003] Raph Levien. Attack resistant trust metrics. *Ph.D Thesis, UC Berkeley*, 2003.

[Massa and Bhattacharjee, 2004] Paolo Massa and Bobby Bhattacharjee. Using trust in recommender systems: an experimental analysis. *Proceedings of 2nd International Conference on Trust Managment, Oxford, England*, 2004.

[O'Donovan and Smyth, 2005a] John O'Donovan and Barry Smyth. Eliciting trust values from recommendation errors. In *Proceedings of the 18th International FLAIRS Conference*. AAAI Press, 2005.

[O'Donovan and Smyth, 2005b] John O'Donovan and Barry Smyth. Trust in recommender systems. In *Proceedings of the 10th International Conference on Intelligent User Interfaces*, pages 167–174. ACM Press, 2005.

[O'Mahony *et al.*, 2002] Michael P. O'Mahony, Neil Hurley, and Guenole C. M. Silvestre. An attack on collaborative filtering. In *Proceedings of the 13th Int. Conf. on Database and Expert Systems Applications*, pages 494–503. Springer-Verlag, 2002.

[O'Sullivan *et al.*, 2002] Derry O'Sullivan, David C. Wilson, and Barry Smyth. Improving case-based recommendation: A collaborative filtering approach. In *Proceedings of the Sixth European Conference on Case Based Reasoning.*, pages LNAI 2416, p. 278 ff., 2002.

[Resnick *et al.*, 1994] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM CSCW'94 Conference on Computer-Supported Cooperative Work*, Sharing Information and Creating Meaning, pages 175–186, 1994.