

Trust-based Agent Community for Collaborative Recommendation*

Jianshu Weng, Chunyan Miao, Angela Goh
School of Computer Engineering
Nanyang Technological University,
Singapore

Zhiqi Shen, Robert Gay
School of Electrical and Electronic Engineering
Nanyang Technological University,
Singapore

{WENG0004, ASCYMiao, ASEGoh, ZQShen, EKLGay}@ntu.edu.sg

ABSTRACT

There exist a number of similarity-based recommendation communities, within which similar users' opinions are collected by users' agents to make predictions of their opinions on a new item. Similarity-based recommendation communities suffer from some significant limitations, such as scalability and susceptibility to the noise. In this paper, we propose a trust-based community to overcome these limitations. The trust-based recommendation community incorporates trust into the domain of item recommendation. Experimental results based on a real dataset show that trust-based community manages to outperform its similarity-based counterpart in terms of prediction accuracy, coverage, and robustness in the presence of noise.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Information filtering

General Terms: Algorithm, Performance

Keywords: Trust Metric, Recommendations

1. INTRODUCTION

The past few years have seen the emergence of the online recommendation communities, such as Epinions¹ and Movielens². Users join the community to collect others' opinions (usually in the form of ratings), which are then used to make predictions of new items' ratings, which recommend the users whether to browse the new items.

Traditionally, similarity-based approach is applied to make predictions in communities, i.e. only opinions from agents³ with similar preferences are taken into account when making the predictions. The similarity between a pair of agents is generally measured according to the "agreements" between

individual items over all the items co-rated by the two agents [6]. However, the similarity-based communities have major limitations, for example:

- (1) Agents usually rate few items [2]. It is thus not unusual that agents do not co-rate the minimum number of items required to compute the similarity. For example, it is required that agents co-rate at least two items to calculate the Pearson correlation coefficient, which is a widely used similarity metric. Hence, the number of similar agents is small. In some cases, agent even cannot find similar agents. As a consequence, a large number of agents are not able to make predictions.
- (2) By intentionally rating only a small number of items, malicious agents are easier to achieve high similarity since the number of co-rated items is small. For example, Pearson correlation gives a similarity of either 1, -1 or 0 when the number of co-rated items is 2. Consequently, malicious agents' noisy ratings⁴ can easily outweigh others' ratings, leading to the biased predictions on given items.
- (3) Typically, there is a centralized server that collects the opinions and makes predictions for the agents. The computation burden of the centralized server increases quickly with the increase of the number of agents.

To overcome the aforementioned limitations, we propose a trust-based community by incorporating *trust* into the domain of item recommendation. This paper adapts Barber's definition[1] which defines *trust* to be the *expectation of technically competent role performance*. More specifically, *trust* is interpreted as an agent's expectation of another agent's competence in providing opinions to reduce its uncertainty in predicting new items' ratings. The main components of the trust-based community are as follows:

- (1) Agents in trust-based communities take opinions of *trustworthy* agents into account when making predictions. A trust metric is designed to help an agent to quantify the degrees of trust it should place on other agents. Unlike the similarity metric, the trust metric is computable on most agents. It is even computable on pairs of agents who only co-rate one item. A new approach is also proposed to make predictions based on trustworthy agents' opinions.
- (2) Each agent collects others' opinions actively when it encounters new items, and then makes predictions of the new items' ratings with local computations. At the

*This work was partially supported by IMSS grant from the Agency for Science, Technology and Research, Singapore.

¹Epinions: <http://www.epinions.com/>

²Movielens: <http://movielens.umn.edu/>

³In the rest of this paper, *agent* is used to denote the user who is the owner of the agent.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.

Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

⁴Presence of noisy ratings means that agents might give unfairly positive or negative opinions for the purpose of biasing other agents' predictions for or against certain items.

Ag_s 's opinions	Ag_t 's opinions					Total
	1	...	j	...	Z	
1	n_{11}	...	n_{1j}	...	n_{1Z}	R_1
...	...	...	...	...	...	...
i	n_{i1}	...	n_{ij}	...	n_{iZ}	R_i
...	...	...	...	...	...	...
Z	n_{Z1}	...	n_{Zj}	...	n_{ZZ}	R_Z
Total	C_1	...	C_j	...	C_Z	N

Table 1: Ag_s 's experience with Ag_t

same time, each agent contributes to the community by sharing its opinions when it is contacted by others for opinions. Hence, each agent is in fact a peer servant, and the community is basically a P2P overlay network. The P2P-based architecture avoids the need for centralized server.

The outline of the rest of this paper is as follows. The main components of the trust-based community are described in Section 2. Experimental results are presented in Section 3. Finally, Section 4 concludes the paper.

2. COMPONENTS OF THE TRUST-BASED COMMUNITY

There are two sets of entities within the communities.

- (1) a set of uniquely identifiable *items* $O = \{O_1, O_2, \dots, O_j, \dots, O_m\}$,
- (2) a set of uniquely identifiable *agents* $Ag = \{Ag_1, Ag_2, \dots, Ag_i, \dots, Ag_n\}$, who have their opinions $r_{i,j}$ on the *items*.

Here $r_{i,j}$ denotes Ag_i 's *opinion* (in the form of rating) on O_j after browsing O_j . Agents' predictions of items' ratings are also given in the form of ratings. Ag_i 's prediction on O_j 's rating is denoted as $Pr_{i,j}$. When contacted by other agents, agents' opinions on the items, but not the predictions, are shared.

As discussed earlier, each agent is treated equally as a peer servant. Without loss of generality, the rest of this paper is presented from the perspective of a particular agent Ag_s .

2.1 Design of the Trust Metric

The proposed trust metric is experience-based, i.e., the trustworthiness of a given agent, say Ag_t , is calculated by Ag_s based on its previous experience with Ag_t ⁵. Ag_s compiles its experience with Ag_t in a table as shown in Table 1. In Table 1, Z denotes the number of possible ratings, and N is the total number of items co-rated by Ag_s and Ag_t . Each cell n_{ij} records the number of co-rated items on which Ag_t 's opinions are j while Ag_s 's opinions (after browsing the items) are i . Each cell is initiated with dummy experience number as $1/Z$. After Ag_t shares opinions as j for the first time, the dummy experience in all the cells of column j are cleared first, then cells in column j are updated based on real experience. The dummy experience is introduced to eliminate the possible appearance of the undefined $0/0$ when calculating the trustworthiness.

$R_i = \sum_{j=1}^Z n_{ij}$, i.e. the number of co-rated items on which Ag_s 's opinions are i ; $C_j = \sum_{i=1}^Z n_{ij}$, i.e. the number of co-rated items on which Ag_t 's opinions are j . $\sum_i R_i = \sum_j C_j = N$. After compiling the experience with Ag_t , agent Ag_s calculates Ag_t 's *trustworthiness* $Tr_{s,t}$ as:

$$Tr_{s,t} = \frac{(1 - \sum_i (R_i/N)^2) - (1 - \frac{1}{N} \sum_i \sum_j (n_{ij}^2/C_j))}{1 - \sum_i (R_i/N)^2} \quad (1)$$

$$= \frac{N \sum_i \sum_j \frac{n_{ij}^2}{C_j} - \sum_i R_i^2}{N^2 - \sum_i R_i^2} \quad (2)$$

⁵ Ag_s has experience with Ag_t means Ag_t 's opinions have ever been collected by Ag_s to make predictions.

As discussed before, $Tr_{s,t}$ is interpreted as Ag_s 's expectation of Ag_t 's competence in providing opinions to reduce its uncertainty in predicting new items' ratings. The reduction of uncertainty can be measured as the reduction of the incorrect predictions given Ag_t 's opinions. In the right hand side of Eq. (1), the first and second term in the numerator are the proportions of the incorrect predictions made without Ag_t 's opinions and with Ag_t 's opinions respectively. Thus, Eq. (1) quantifies agent Ag_t 's trustworthiness by measuring the reduction of incorrect prediction given Ag_t 's opinions. Eq. (1) is further simplified to Eq. (2).

$Tr_{s,t}$ takes values in $[0, 1]$. $Tr_{s,t} = 0$ implies that Ag_t does not reduce Ag_s 's uncertainty at all, whereas $Tr_{s,t} = 1$ implies Ag_s 's perfect predictability of the new items' ratings given Ag_t 's opinions. Every time after a new prediction has been made with Ag_t 's opinions, Ag_s updates its evaluation of Ag_t 's trustworthiness.

2.2 Opinions Collection

Ag_s maintains a set of *direct neighbors*, whom Ag_s contacts directly for opinions when it encounters a new item. Each direct neighbor, when contacted by Ag_s , shares its opinion if it has rating on the target item. Moreover, each direct neighbor returns all its direct neighbors as well as their trustworthiness as *referrals* to Ag_s . Instead of contacting all the referrals, Ag_s applies a probabilistic strategy in determining whether to contact a particular referral at the next step. Suppose Ag_d returns a number of referrals. Ag_s will contact a particular referral Ag_m at the next step with a probability $Tr_{d,m}$, and will not contact Ag_m with a probability $1 - Tr_{d,m}$. $Tr_{d,m}$ is Ag_d 's evaluation of Ag_m 's trustworthiness. Also, Ag_s will not contact an agent if it has been contacted before for the same item. Then each contacted agent carries out the similar actions as Ag_s 's direct neighbors do. The only difference is that it returns referrals only when it has no opinions to share.

A path starting from agent Ag_s connecting all the contacted agents (including the direct neighbors) and ending at the agent who has opinion on the target item is called a *referral chain*. An upper limit (U) is applied on the length of a referral chain to avoid endless opinion collection.

Opinions collection usually discovers more than one agent that can share opinion on a target item. Ag_s can use the proposed trust metric to determine those agents' trustworthiness if it has previous experience with them. Trust propagation is necessary to help Ag_s reason about the trustworthiness of the agents who share opinions but Ag_s has no previous experience with. The trust propagation mechanism proposed in [4] is employed in this work. Suppose in a given referral chain starting from Ag_s , Ag_m is one of Ag_d 's referrals to Ag_s . Ag_s derives Ag_m 's trustworthiness through *trust propagation* as:

$$Tr_{s,m} = Tr_{s,d} * Tr_{d,m} + (1 - Tr_{s,d}) * (1 - Tr_{d,m}) \quad (3)$$

where $Tr_{s,d}$ is Ag_s 's evaluation of Ag_d 's trustworthiness, and $Tr_{d,m}$ is Ag_d 's evaluation of Ag_m 's trustworthiness. Then (3) is computed recursively by Ag_s over all the agents along the referral chain to derive the trustworthiness of the agent at the end of the referral chain.

2.3 Rating Prediction

The new items' ratings are basically derived as weighted sum of the collected opinions, with each opinion-sharing agent's trustworthiness as weight. The collected opinions are adjusted before being used to make predictions if Ag_s has previous experience with the corresponding opinion-sharing

agents. Suppose agent Ag_x shared opinion on the item O_n as $r_{x,n} = v$ and Ag_s has previous experience with Ag_x , then Ag_s 's opinion $r_{s,n} = v$ is adjusted to $r'_{s,n} = i$ with a probability n_{iv}/C_v . If Ag_s has no previous experience with Ag_x , Ag_s 's opinion is used directly without adjustment.

It is possible that process of opinion collection does not collect any opinion. In this case, Ag_s selects a *pseudo-opinion* as $r'_{s,n} = i$ from each direct neighbor Ag_x with a probability R_i/N (R_i and N are based on Ag_s 's experience with each direct neighbor Ag_x).

Ag_s then predicts the new item O_n 's rating $Pr_{s,n}$ as:

$$Pr_{s,n} = \frac{\sum_{Ag_x \in \mathcal{A}_{s,n}} Tr_{s,x} * r'_{x,n}}{\sum_{Ag_x \in \mathcal{A}_{s,n}} Tr_{s,x}} \quad (4)$$

Here $\mathcal{A}_{s,n}$ is the set of opinion-sharing agents (including those share pseudo-opinions). $r'_{x,n}$ is the adjusted opinion or pseudo-opinion shared by agent Ag_x . Each agent in $\mathcal{A}_{s,n}$ is assigned a weight which equals to its trustworthiness $Tr_{s,x}$.

3. EXPERIMENTAL RESULTS

A trust-based movie recommendation community has been simulated based on the MovieLens dataset⁶. A similarity-based community based on same dataset has also been setup to study the improvement introduced by the trust-based community. The Pearson correlation efficient is employed as the similarity metric. Agents with negative similarity are ignored when making prediction. Opinion collection with "similarity propagation" was also implemented for the similarity-based community.

The first experiment studied the improvement in terms of coverage without the presence of noisy ratings. Coverage was measured as the percentage of the dataset that the agents were able to predict rating for. The trust-based community manages to achieve a coverage very close to 100%, whereas the maximum coverage achievable with similarity-based community is less than 80%.

The second experiment studied the improvement in terms of prediction accuracy without the presence of noisy ratings. Prediction accuracy was measured as Mean Absolute Error (MAE). MAE is derived by calculating each agent's mean prediction error between the predicted ratings and real opinions, and then averaging all agents' mean errors. The values of U (limit of the length of the referral chains) and D (number of direct neighbors) are tuned to be: $U \in [3, 8]$, and $D \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, 50, 60, 70, 80\}$. Figure 1 compares the MAE achieved by the trust-based and the similarity-based community respectively when $U = 3$. It is observed that the trust-based community generally achieves higher prediction accuracy (i.e. lower MAE) than its similarity-based counterpart with different settings of D . The same trends are observable with other values of U . Due to the space limitation, comparisons of the MAE with other values of U are omitted here.

The third experiment was to study the community's robustness in the presence of noisy ratings. $D = 10$ and $U = 3$ in the third experiment. The robustness was measured in terms of *power of attack* (POA). Let O_t be the targeted item of the noisy ratings, and Ω be the set of agents who are predicting the item O_t 's rating. The POA for a particular type of noisy rating ATK is given by:

$$POA(\Omega, O_t, ATK) = \frac{\sum_{a \in \Omega} (|r_{target} - r_{a,t}| - |r_{target} - r''_{a,t}|)}{|\Omega|}$$

⁶MovieLens dataset is publicly available at <http://www.cs.umn.edu/research/GroupLens/data/>.

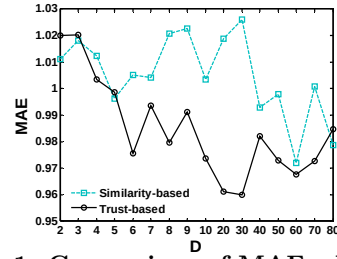


Figure 1: Comparison of MAE when $U = 3$ where $r_{a,t}$ and $r_{a,t}$ are predictions made with and without the presence of noisy ratings respectively. r_{target} is the target rating, e.g. $r_{target} = 5$, or 1 for *push* and *nuke* respectively. A positive value of POA means that the predictions have been forced toward the target rating. Otherwise, POA gives a negative value. Robustness in the presence of different ratios of noisy ratings is shown in Table 2. It is observed that the trust-based community manages to achieve smaller POA than its similarity-based counterpart. It is thus shown that the trust-based community is more robust to the presence of noisy ratings than similarity-based community.

Noisy rating ratios	POA (Push)		POA (Nuke)	
	Trust-based	Similarity-based	Trust-based	Similarity-based
25%	0.0688	0.0410	0.0722	0.0238
50%	0.1105	0.1880	0.1430	0.2145
75%	0.1375	0.2020	0.5305	0.6408
100%	0.1788	0.2211	0.6730	0.7617

Table 2: POA w.r.t. different malicious noisy ratios

4. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a trust-based recommendation community, in which a trust metric has been designed to quantify the degrees of trust an agent should place on others, and a new prediction method has also been proposed.

Work has been reported to introduce trust into the domain of item recommendation to overcome the limitations listed in Section 1, e.g. [3, 5]. However, all of these in existing work do not make clear how an agent can determine the degrees of trust it should place on others. This paper goes beyond existing work in that it proposes a trust metric, with which an agent can practically quantify the degrees of trust it places on others.

Currently, direct neighbors maintained by an agent are not changed during its whole life cycle. It is possible that some direct neighbors become less trustworthy. In this case, those less trustworthy direct neighbors should be replaced by some indirect neighbors that are more trustworthy. The performance of the trust-based community with neighbor replacement will be further investigated.

5. REFERENCES

- [1] B. Barber. *The logic and limits of trust*. Rutgers University Press, 1983.
- [2] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval Journal*, 4(2):133–151, 2001.
- [3] P. Massa and P. Avesani. Trust-aware collaborative filtering for recommender systems. In *Proceedings of International Conference on Cooperative Information Systems 2004*, 2004.
- [4] L. Mui. *Computational Models of Trust and Reputation: Agents, Evolutionary Games, and Social Networks*. PhD thesis, Department of Electrical Engineering and Computer Science, MIT, 2003.
- [5] M. Papagelis, D. Plexousakis, and T. Kutsuras. Alleviating the sparsity problem of collaborative filtering using trust inferences. In *Proceedings of iTrust 2005*, pages 224–239, 2005.
- [6] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186, 1994.