HYBRIDISING COLLABORATIVE FILTERING AND TRUST-AWARE RECOMMENDER SYSTEMS

Charif Haydar[1], Anne Boyer[2] and Azim Roussanaly[2]
[1]*Womup, University Nancy 2, Tours, Nancy, France*
[2]*Loria Laboratory, University Nancy 2, Nancy, France*

Keywords: Recommender Systems, Trust, Reputation, Users Similarity.

Abstract: Recommender systems (RS) aim to predict items that users would appreciate, over a list of items. In evaluation of recommender systems, two issues can be defined: accuracy of prediction which implies the satisfaction of the user, and coverage which implies the percentage of satisfied users. Collaborative filtering (CF) is the master approach in this domain, but still has some weaknesses especially about coverage. Trust-aware approach is today another promising approach in RS within social environments, whose prediction exceeds the quality of (CF). In this paper we propose several strategies to hybridize both approaches in order to improve prediction accuracy and coverage.

1 INTRODUCTION

Recommender systems (RS) (Resnick and Varian, 1997) aim to recommend users some items they would appreciate, over a list of items. Collaborative filtering (CF) (Resnick et al., 1994) is the most popular approach of recommender systems. The main idea behind CF is to recommend to a user, called current user, the items appreciated by users who are similar to him/her in the term of preferences. These similar users are called neighbors in this context.

CF suffers from many weaknesses, such as: cold start (Maltz and Ehrlich, 1995), data sparsity (Shin et al., 2008), fragility to malicious attacks (Mobasher et al., 2007; Burke et al., 2005) and recommendation acceptability by users (Herlocker et al., 2000).

In social systems, users can not only express their opinions about items, but also about other users, whereby they rate their credibility and trustworthiness. The literature has proposed to replace user similarity by trust relationships, which resulted in trust-aware recommenders (Massa and Bhattacharjee, 2004; Golbeck and Hendler, 2006). Trust-aware approaches have the advantages of alleviating the precedent weaknesses of CF recommenders, without bringing the recommendation accuracy down (Massa and Bhattacharjee, 2004). What these systems really offer to their users is the possibility to choose their list of neighbors themselves, these neighbors are called friends in this context. Choosing friends manually comes out to more accurate lists. Nevertheless, a sizeable percentage of users are still out of this social range. They still rate items but not other users. Even though trust-aware approaches surpass user similarity in recommendation quality they are unable to recommend items to those users, while users similarity approaches are still able to. This fact leads us to think that hybridising both approaches will allow the RS to recommend to a larger percentage of users.

Hybrid recommenders (Burke, 2007), which hybridise several recommendation approaches to generate recommendation, were widely proposed as solutions to the case when different approaches suffer from contradictory weaknesses. This allows to profit of their advantages together. To the best of our knowledge, opinion similarity and trust similarity have always been considered as independent notions. The main contribution of this paper is to exploit both notions altogether. We propose to hybridise them in one recommender system in the aim of yielding it more profitable by a larger set of users, ensuring a minimal recommendation accuracy.

The outline of the paper is organized as follows: in section 2 we present the user-based collaborative filtering recommenders, and trust-aware recommenders. In section 3 we address to the hybridization strategies to extract a set of them applicable our case. Last section is dedicated to discussing the experiments and results.

Haydar C., Boyer A. and Roussanaly A..
HYBRIDISING COLLABORATIVE FILTERING AND TRUST-AWARE RECOMMENDER SYSTEMS.
DOI: 10.5220/0003937406950700
In *Proceedings of the 8th International Conference on Web Information Systems and Technologies* (WEBIST-2012), pages 695-700
ISBN: 978-989-8565-08-2

2 GENERAL FRAMEWORK

2.1 Recommender Systems

Recommender systems utilize a wide range of techniques in order to recommend pertinent items to users, the skeleton part of the recommender system is the prediction function. This function estimates how much a given user will like a given item. A prediction function might exploit information about other users (user based) (Resnick et al., 1994) or information about other items (item based) (Basu et al., 1998) or both of them (hybrid) (Burke, 2007) to generate the recommendations.

In this paper we are interested in user based recommenders, and more specifically in collaborative filtering and trust-aware recommender systems.

2.2 Collaborative Filtering Recommenders

CF is the most popular recommendation approach. The prediction function is based on the similarity of users' ratings.These ratings are stored in a $m \times n$ matrix called rating matrix, where m is the number of users and n is the number of items. An element $v_{u_a i}$ of this matrix is the rating user u_a to the item i. Users' similarity is computed by using this matrix. Many metrics in the litterature satisfy this definition of similarity in the CF recommenders (Resnick et al., 1994; Breese et al., 1998).

In this paper, we will consider the Pearson correlation coefficient (Resnick et al., 1994) as a similarity metrics, which varies between the values -1 (completely opposite users) and $+1$ (completely similar users). Our choice is found on the wide popularity and the efficiency of this metrics.

It is defined in the equation 1:

$$f_{simil}(u_a,u_j) = \frac{\sum_{i\in I_a \bigcap I_j}(v_{(u_a,i)}-\overline{v_{u_a}})(v_{(u_j,i)}-\overline{v_{u_j}})}{\sqrt{\sum_{i\in I_a \bigcap I_j}(v_{(u_a,i)}-\overline{v_{u_a}})^2 \sum_{i\in I_a \bigcap I_j}(v_{(u_j,i)}-\overline{v_{u_j}})^2}} \quad (1)$$

Where:
u_a, u_j are two users.
$v_{(u_a,i)}$: is the rating of user u_a to the item i.
$\overline{v_{u_a}}$: is the average rate of the user u_a.

In order to predict how much the current user u_a will rate an item r the system exploits the rating of similar users to u_a (equation 2) from the group of users who rated r (U_r).

$$p(u_a,r) = \overline{v_{u_a}} + \frac{\sum_{u_j\in U_r} f_{simil}(u_a,u_j)\times(v_{(u_j,i)}-\overline{v_{u_j}})}{card(U_r)} \quad (2)$$

Where:
U_r: the set of users who have rated r.
$card(U_r)$: is the number of users in U_r.

CF is incapable to generate accurate recommendations to new users who have not rated items yet. Paradoxically, the system needs to attract this kind of users (Massa and Bhattacharjee, 2004). This context is called the cold start problem. Data sparsity problem is observed in datasets containing large archives of items, where even an active user can't rate 1% of the total number of items. By consequence, the majority of the rating matrix values are empty. In sparse matrix, the probability that users rate the same items becomes smaller. As a result, similarity between users becomes rare and the accuracy of the prediction becomes very low.

2.3 Trust Aware Recommenders

The study of trust as a computational phenomenon started in the last decade. Many models were proposed (Abdul-Rahman, 2004; Kruknow, 2006; Mui, 2002). We are interrested in the qualitative definition of trust (called also local-trust), where personal bias is taken into account, and trust is represented as user to user relationship (Ziegler and Lausen, 2004b).

A correlation between trust and users similarity was found in (Ziegler and Lausen, 2004a) and (Lee and Brusilovsky, 2009). Replacing user similarity by trust relationships is the object of several "trust-aware RS" (Golbeck and Hendler, 2006; Massa and Avesani, 2004). In a social network, such as eopinion[1] and filmtrust[2], when a user A expresses that he trusts another user B RS considers that B is eligible to recommend items to A.

Trust-aware systems apply the same prediction function as in CF, with only one difference, which is replacing similarity values by trust values. The trust values can be given explicitly by the current user, or computed by a trust propagation algorithm.

Using trust in RSs allows to alleviate several of CF's weaknesses. A cold start user has only to express at least one trust relation to start to receive recommendation, this is, anyway, less costly than rating dozen of items. The propagation algorithms allow to increase the connectivity of users. This can reduces the impact of the data sparsity. While users choose themselves their friends this allows to easily identify ma-

[1] http://www.eopinion.com
[2] http://trust.mindswap.org/FilmTrust/

licious attackers and to isolate them to protect other users. Finally trust aware recommendation is clearer and easier to explain to the user, which improves the acceptability.

Many models were presented to model trust and it's propagation (Massa and Bhattacharjee, 2004; Golbeck, 2005; Ziegler and Lausen, 2004b; Kuter and Golbeck, 2010). All those models consider trust propagation problem as a graph traversal problem. The difference between those models is about their strategies in traversing the graph and the choice of path between the source and destination nodes.

In most models, trust is expressed either as real or integer value within a given range, whereas it is a binary value in our studied case. That is why we choose for our experiments the model MoleTrust (Massa and Bhattacharjee, 2004). In MoleTrust, each user has a domain of trust where he adds his trustees, that is to say, user can either trusts fully other user or he does not trust him at all. The only initializing parameter is the maximal propagation distance d.

If user A added user B to his domain, and B added C, then the trust of A in C is given by the equation:

$$Tr(A,C) = \frac{(d-n+1)}{d}$$

Where n is the distance between A and C ($n = 2$ as there two steps between them; first step from A to B, and the second from B to C).

3 HYBRIDISATION

In (Burke, 2007) the author identifies seven strategies to hybridise multiple recommendation approaches, he argues that there is no reason why recommenders from the same type could not be hybridized. Some of these strategies require that at least one of the hybridised approaches to be an item-based recommender which is beyond the scope of this paper, other strategies require that the approaches are applied in separated recommenders, while in some strategies the hybridisation is done in the prediction function level. In this paper we are interested in the last type only.

We present here the chosen strategies:

- **Weighted:** The score of both similarity and trust are combined numerically according to predefined weights:

$$score(u_a,u_j) = \alpha \times simil(u_a,u_j) + (1-\alpha) \times trust(u_a,u_j) \quad (3)$$

- **Mixed:** Each approach produces her list of recommendation independently, the lists then are mixed and send to the user as one list. As for our case:

$$score(u_a,u_j) = max(simil,trust)$$

- **Probabilistic:** This strategy privileges neighbors who are both similar and trustee over those who are either only similar or only trustee, with respect to the similarity and the trust values. The formula here is:

$$score = 1-(1-simil)(1-trust)$$

- **Switching:** The system selects a single recommendation approach, and switch to the second one only when the quality of recommendation of the first one is not satisfactory.

$$score(u_a,u_j) = \begin{cases} simil & simil \neq null \\ trust & simil = null \end{cases}$$

- **Cascade:** The idea here is to create a strictly hierarchical hybrid, in which a weak recommender can not overturn the decisions taken by a stronger one, but can merely refine them. In order to be selected by the prediction function the user must be similar to and trusted by the active user. In other words; the first approach chooses trustee users, the second approach chooses similar users within the trustees set.

  The score that we apply in this case is:

$$score(u_a,u_j) = \begin{cases} trust & simil > 0 \\ 0 & otherwise \end{cases}$$

Notice that the last two strategies are sensitive to the order of hybridised approaches, so we shall test each of them twice with alterning the approaches' order.

In order to have the similarity and trust value in the same range, all similarity values in the experiments are normalized.

4 EXPERIMENTS

4.1 DataSet

We use the eopinion.com dataset. eopinion.com is a consumers opinion website where users can rate items in a range of 1 to 5, and write reviews about them. Users can also express their trust towards reviewers whose reviews seem to be interesting to them. Eopinion dataset contains 49,290 users who rated a total of 139,738 items. The total number of ratings is 664,824. The dataset contains also 487,182 binary

trust ratings. It is important also to mention that 3,470 users have neither rated an item nor trusted a user, these users are eliminated from our statistics. Thus the final number of users is 45,820 users.

4.2 Methodology of Evaluation

In (Massa, 2006), authors showed on this corpus how does replacing similarity metrics by trust-aware metrics improve both accuracy and coverage. The improvement of coverage was limited because of the fact that some users are active in rating items but not in rating reviewers. 11,858 users have not trusted anybody in the site (25.8% of the total number of users). Those users have made 75,109 ratings, on average of 6.3 ratings by user. This high average value means that recommendations can be generated to this category of users by a similarity based approach, and not by a trust-aware apraoch. On the other hand, 5,655 users have not rated any item in the site (12.3% of the total number of users). The average of trust relation by user in this set is 4.07 which is not negligible, those users suffer from the same problem with the similarity approaches while trust based approach can predict item to them. We are convinced that each of similarity and trust approaches is suitable to a particular set of users. This is why we think that hybridising these approaches can satisfy a larger set of users, while providing accurate recommendations.

Our objective is to find the suitable hybridisation formula, that leads to improve the satisfaction of the system. The satisfaction in this context is represented by both accuracy and coverage.

We divide the corpus in two parts randomly, 80% for training and 20% for evaluation (a classical ratio in the literature). We respected this ratio by user, so every user has 80% of his ratings in the training corpus and 20% in the evaluation corpus, this is important when we want to measure satisfaction by user.

In each experiment, a rating matrix is formed from the training corpus, while the ratings of the evaluation corpus are considered as empty values. The recommender must predict those values and complete the matrix with them. the resulting matrix is compared to the original full rating matrix (formed from the entire corpus ratings). The evaluation of the recommender depends on how many values could it predict (coverage), and how much the predicted values are close to the real ones (accuracy).

To measure accuracy, we use the mean absolute error metrics (MAE) (Herlocker et al., 2004), which is a widely used predictive accuracy metrics. MAE measures the average absolute deviation between the predicted values of ratings and the real values supplied by the user.

MAE focuses on ratings but not on users (Massa and Avesani, 2004). User mean absolute error (UMAE) (Massa and Avesani, 2004) is the version of MAE which consider the accuracy by user. It consists in computing the MAE to the predictions of every user. And then computing the average of all MAEs.

The aim of RS is to predict appreciable items to user. A prediction function that succeeds in predicting low and middle ratings of the user but fail in predicting high ones can not generate recommendations. High MAE (Baltrunas, 2007) is the version of MAE dedicated to evaluate the ability of the system to recommend and not to predict. This version of MAE takes into account only high ratings (usually 4 and 5 for systems of rating range between 1 and 5).

It is also vital to know the percentage of ratings and users the system can cover. We employ three forms of coverage metrics, compatible with the three forms of MAE. These metrics are:

- Coverage of prediction: the number of predicted ratings divided by the total number of ratings in the evaluation corpus.
- Coverage of users: the number of users who received predictions divided by the total number of users.
- Coverage of high-ratings: the number of predicted high ratings divided by the total number of high ratings in the evaluation corpus.

4.3 Results and Discussion

The weighted hybridisation is the only proposed strategy to have a parameter (α). To adjust this parameter, we associated it with 5 different values between 0.1 and 0.9 with a step of 0.2. Table 1 illustrates the MAE changes according to α values. It is obvious that the change is slight. For what concern the coverage it is stable for any value of α, which is normal while changing α affects the value of the prediction but not the ability to predict. Nevertheless, we will choose the value of $\alpha = 0.3$ in our upcoming comparisons, because it has the best MAE score (MAE = 0.821).

Table 1: α and MAE for weighted hybridisation strategy.

α	0.1	0.3	0.5	0.7	0.9
MAE	0.8219	0,821	0,8212	0,823	0,8294

All results are illustrated in table 2, to facilitate the analysis we group each couple of metrics in a figure while discussing them.

Figure 1 illustrates the MAE and the coverage of prediction metrics for all tested strategies. Almost all

hybridisation strategies improve the prediction coverage by approximately 7% compared to the trust-aware approach, and about 15% compared to similarity approach. The only exception is the cascade strategy, although its accuracy is not very far from that of other strategies, it has a very low coverage. This is because cascade is severe in eliminating neighbors.

This result shows how the hybrid system uses each approach to predict ratings inpredictable by the other approach, which allows to cover much ratings than any of the two approaches.

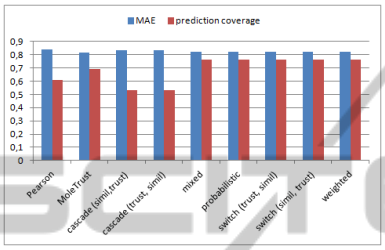

Figure 1: MAE and coverage.

It is obvious now that hybridisation ensures more predictions, the next question is what kind of predictions? and what is the impact of this improvement on the recommendation?

As we argued above, HMAE is a metrics that replies to this question. Figure 2 shows that hybridisation enhence the ability to predict good items by about 5% compared to MoleTrust and 12% compared to Pearson correlation similarity. Once again we exclude the cascade.

This result shows that hybrid system can recommend more items than trust-aware and similarity based recommenders, with ensuring the approximate accuracy to that of trust-aware approach, and farther better than similarity based approach.

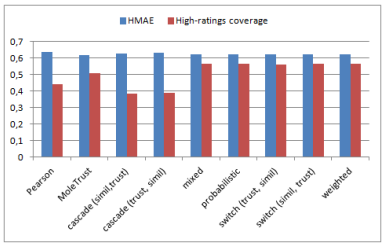

Figure 2: HMAE and high-ratings coverage.

Another important question is: Does this augmentation in prediction coverage concern a limited number of users?

Figure 3 shows the values of UMAE by strategies. Hybridisation strategies cover 10% of users more compared to MoleTrust and 15% more compared to Pearson similarity. This ratio is not negligible and shows that satisfaction is shared by a considerable propotion of users, so hybrid systems are capable to predict to a large variety of users

Notice that in cascade, ordering trust before similarity allowed us to gain 0.08 in UMAE. In the first strategy we use the trust score of similar users, while already similars are rare, not a lot of trustees are available. In the second, we use the similarity of trustee users, while trustees are usually numerous, this give more chance to find similar among them.

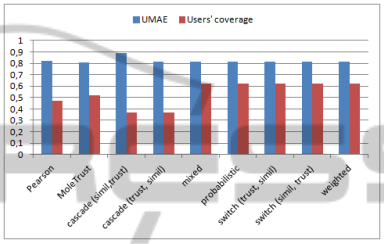

Figure 3: UMAE and users' coverage.

Finally, our results show that hybridisation increases coverage in RS, with keeping the accuracy in a reasonable range. This improvement includes ratings predictions, recommendations and users. It also adapts with larger variety of users behavior. For what concern the choice of hybridisation strategy, it is likely to employ those who tend to aggregate information of many approaches, and avoid severity in eliminating information.

5 CONCLUSIONS AND FUTURE WORK

In this paper we showed that even though trust-aware recommenders can usually improve the accuracy and the coverage of prediction of CF recommenders over usual similarity metrics, it is still unable to recommend to ceratain categories of users. Hybridising these approaches is a promising strategy to improve the coverage without significant decrease in accuracy.

Our results where proved using one corpus, which mean that it can be specific to it. We are convinced that validation must be done on other corpora when available. The nature of current corpus restricted our choice of trust metrics. We hope that upcoming tests to be done on datasets with numeric trust values, which allow to test other trust metrics.

Finally, similarity metrics allows the detection of non similar users. Current trust metrics do not treat

Table 2: The three forms of MAE and coverage.

Strategy	MAE	coverage	HMAE	High coverage	UMAE	User coverage
Pearson correlation	0.84	61.15%	0.6364	44.44%	0.8227	47.46%
MoleTrust	0.8165	69.28%	0.6185	51.12%	0.8079	52.21%
Cascade (Simil,Trust)	0.8315	53.19%	0.6263	38.57%	0.8905	36.78%
Cascade (Trust,Simil)	0.8358	53.44%	0.6339	38.94%	0.8126	36.97%
Mixed	0.8208	76.34%	0.6218	56.43%	0.8124	62.15%
Probabilistic	0.8206	76.31%	0.6212	56.44%	0.8124	62.07%
Switch (Trust,Simil)	0.8217	76.38%	0.622	56.19%	0.8161	61.96%
Switch (Simil,Trust)	0.8220	76.38%	0.623	56.44%	0.8148	62.06%
Weighted ($\alpha = 0.3$)	0.8210	76.38%	0.6214	56.42%	0.8124	62.22%

the distrust issue. We would like to extend our work to integrate this aspect which we find important.

REFERENCES

Abdul-Rahman, A. (2004). *A Framework for Decentralised Trust Reasoning. phD thesis*. PhD thesis, Computer Sceince, University College London.

Baltrunas, R. (2007). Dynamic item weighting and selection for collaborative filtering. In *Web mining 2.0 Workshop*.

Basu, C., Hirsh, H., and Cohen, W. (1998). Recommendation as classification: using social and content-based information in recommendation. In *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*.

Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical analysis of predictive algorithm for collaborative filtering. In *Proceedings of the 14 th Conference on Uncertainty in Artificial Intelligence*, pages 43–52.

Burke, R. (2007). Hybrid web recommender systems. In Brusilovsky, P., Kobsa, A., and Nejdl, W., editors, *The adaptive web*, pages 377–408. Springer-Verlag, Berlin, Heidelberg.

Burke, R., Mobasher, B., Zabicki, R., and Bhaumik, R. (2005). Identifying attack models for secure recommendation. In *in Beyond Personalization: A Workshop on the Next Generation of Recommender Systems*.

Golbeck, J. (2005). Personalizing applications through integration of inferred trust values in semantic web-based social networks.

Golbeck, J. and Hendler, J. (2006). FilmTrust: movie recommendations using trust in web-based social networks. In *Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE*.

Herlocker, J. L., Konstan, J. A., and Riedl, J. (2000). Explaining collaborative filtering recommendations. *Proceedings of the 2000 ACM conference on Computer supported cooperative work CSCW 00*.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., John, and Riedl, T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22:5–53.

Kruknow, K. (2006). *Towards of trust for the global Ubiquitous Computer. PhD thisis*. PhD thesis, University of Aartus.

Kuter, U. and Golbeck, J. (2010). Using probabilistic confidence models for trust inference in web-based social networks. *ACM Trans. Internet Technol.*

Lee, D. H. and Brusilovsky, P. (2009). Does trust influence information similarity?

Maltz, D. and Ehrlich, K. (1995). Pointing the way: active collaborative filtering. In *Proceedings of the SIGCHI conference on Human factors in computing systems*.

Massa, A. (2006). Trust-aware bootstrapping of recommender systems. In *ECAI Workshop on Recommender Systems*.

Massa, P. and Avesani, P. (2004). Trust-aware collaborative filtering for recommender systems. In *In Proc. of Federated Int. Conference On The Move to Meaningful Internet: CoopIS, DOA, ODBASE*, pages 492–508.

Massa, P. and Bhattacharjee, B. (2004). Using trust in recommender systems: An experimental analysis. In *iTrust'04*, pages 221–235.

Mobasher, B., Burke, R., Bhaumik, R., and Williams, C. (2007). Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology*.

Mui, L. (2002). *Computional Models of Trust and Reputation: Agents, Evolutionary Games,and Social Networks. PhD thesis*. PhD thesis, Massechusetts Institute of Technology.

Resnick, P., Iacovou, N., Sushak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In *1994 ACM Conference on Computer Supported Collaborative Work Conference*.

Resnick, P. and Varian, H. R. (1997). Recommender systems. *Commun. ACM*, 40(3):56–58.

Shin, H., Kim, N. Y., Kim, E. Y., and Lee, M. (2008). Behaviors-based user profiling and classification-based content rating for personalized digital tv. In *Consumer Electronics, 2008. ICCE 2008. Digest of Technical Papers. International Conference on*.

Ziegler, C.-N. and Lausen, G. (2004a). Analyzing correlation between trust and user similarity in online communities. In *Proceedings of Second International Conference on Trust Management*, pages 251–265. Springer-Verlag.

Ziegler, C.-N. and Lausen, G. (2004b). Spreading activation models for trust propagation. In *Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04)*.