

# Trust-aware Clustering Collaborative Filtering: Identification of relevant items

Cosimo Birtolo, Davide Ronca, and Gianluca Aurilio

Poste Italiane – Information Technology  
Research and Development - R&D Center  
Piazza Matteotti 3 – 80133 Naples, Italy  
{birtoloc,roncadav,aurilio5}@posteitaliane.it

**Abstract.** Identifying a customer profile of interest is a challenging task for sellers. Preferences and profile features can range during the time in accordance with current trends. In this paper we investigate the application of different model-based Collaborative Filtering (CF) techniques and in particular propose a trusted approach to user-based clustering CF. We propose a Trust-aware Clustering Collaborative Filtering and we compare several approaches by means of Epinions, which contains explicit trust statements, and MovieLens dataset, where we have implicitly defined a trust information. Experimental results show an increased value of coverage of the recommendations provided by our approach without affecting recommendation quality. To conclude, we introduce a tool, based on recommender systems, able to assist merchants in delivering special offers or in discovering potential interests of their customers. This tool allows each merchant to identify the products to suggest to the target customer in order to best fit his profile of interests.

**Key words:** Trust, Recommender System, Collaborative Filtering, Clustering, Web Intelligence

## 1 Introduction

In the last decade, the concept of trust has become very important with the increase of Social Network platforms and with the use of social community in e-Commerce platforms. Trust can be considered as a remarkable element influencing a user’s decision-making process. Furthermore, the concept of trust includes both the cognitive and the emotional dimensions.

Webs of trust are networks through which a trust-aware system can ask a user to evaluate other users already known. For example, Epinions suggests to put in a user’s web of trust “those reviewers whose reviews and ratings resulted to be extremely useful”. Online interpersonal relations are becoming one of the major characteristics of the Web 2.0, and are also useful for social aspects (MySpace, Msn, Facebook), working connections (LinkedIn) and information (Slashdot.org, Epinions.com) besides, obviously, commercial purposes (eBay.com, Amazon.Com).

In general, trust is a directional relationship between two parties that can be called trustor and trustee. In the e-Commerce domain, trust is applied to a specific purpose, such as mutual trust between customers and sellers. According to Liu [1] trust and reputation management research is highly interdisciplinary, involving researchers from networking and communication, data management and information systems, e-commerce and service computing.

In this paper we associate the concept of trust with similarity between users based on shopping lists. We propose a model-based approach able to combine trust and similarity in order to improve both the quality and the coverage of the suggestions. Furthermore, we present a tool able to assist the merchant in discovering item potentially interesting for customers.

The remainder of this work is organized as follows: Section 2 provides a brief overview of recommendation and trust-aware systems; Section 3 describes how the problem has been modeled; Section 4 presents experimental results; and Section 5 outlines conclusions and future directions.

## 2 Recommendation System

Recommendation systems are aimed at helping users in the search of interesting items among a large set of items within a specific domain by using knowledge about user's preferences in the domain. In other words, Recommendation System is a way for improving personalization by giving personalized suggestions and have largely been adopted in different domains. Almost every e-Commerce site (e.g., Amazon) has its own recommendation engine; different Web sites are focusing on suggesting a personalized content such as a movie (e.g., *MovieLens* and *Netflix*) or a song (*Yahoo!Music*).

Recommendation systems were introduced in 1992 by means of the Tapestry project [2]. Several different approaches have recently been proposed in order to increase the accuracy of the predicted values, thus minimizing the prediction error and improving the quality of the recommendation while taking into account performance issues. However, some issues related to the quality of recommendations and to computational aspects still arise because collaborative systems rely solely on users preferences to make recommendations. Among recommendation techniques, Collaborative Filtering has gained great success in the practical application of e-Commerce [3] and has been proven to be one of the most successful techniques. Recently model-based CF, such as Clustering CF, which uses the user-item database to infer a model which is then applied for predictions, are investigated and discussed in order to outperform traditional memory-based CF. Different works [4, 5] prove that these algorithms improve the quality of predictions, in particular when sparse data is considered.

Moreover, recent works have proved some benefits in terms of an increased quality of suggestions, by including social factors [6] (e.g., user's past behaviors and reputation) or trust information [7] in recommendation systems. These arising systems are called *Trust-based Recommendation Systems* [8] and combine the potentialities of a traditional recommendation system with a trust-aware system.

They adopted the social ties established among online users and extended traditional recommender systems by considering a trust matrix in addition to the ratings matrix.

On the other hand, J. Sun et al. [9] proposed a *Trust-Aware Recommender Model* which utilizes trustworthy experts and their search experiences to recommend their search histories to the common user according to the profile similarity between common users and experts. Furthermore, the same authors extended the model [10] by replacing similarity weight by trust information which is evaluated in a direct and indirect way and decreases along its propagation in the network.

Burke [11] investigated some techniques of hybrid recommendation systems. He compared four different recommendation techniques and several different hybridization strategies. According to his findings, it is possible to distinguish four different hybridization techniques: (i) *Weighted*, (ii) *Mixed*, (iii) *Switching*, and (iv) *Cascade*. In the first approach the score is a weighted linear combination of two recommendation techniques; the second one is a merge based on predicted rating or on recommender confidence; in the third one, the score is equal to one of the two engines according to an external selection criteria that establishes the best confidence; and in the last one, the idea is to create a strictly hierarchical hybrid, one in which a weak recommender cannot overturn decisions made by a stronger one, but can merely refine them.

Although different hybrid algorithms there exist, the major problems are the time necessary to explicitly define the online relations among users and, first and foremost, the small number of social links defined by users themselves, aspect that leads to a scarce quality of recommendations. The second problem, instead, is related to the definition of users' trust that seems to be a challenging task.

### 3 TRACCF: Trust-aware Clustering CF

In Collaborative Filtering [12], once the items to consider are defined, a prediction  $p_i(u)$  for the active user  $u$  is generally evaluated by:

$$p_i(u) = \bar{r}(u) + \frac{\sum_{v \in N_u} sim(u, v) \cdot (r_i(v) - \bar{r}(v))}{\sum_{v \in N_u} |sim(u, v)|} \quad (1)$$

where  $r_i(u)$  is the rating given to item  $i$  by the user  $u$ ,  $\bar{r}(u)$  is the average rating given by the user  $u$ ,  $N_u$  is the set of neighbors of user  $u$  among users, and  $sim(u, v)$  is the similarity function between user  $u$  and user  $v$ .

In this paper we propose a model-based collaborative filtering algorithm which takes into account at the same time clustering and trustworthy information. We introduce  $trust(u, v)$  between user  $u$  and user  $v$  in order to emphasize trustworthy information which comes from a subset of trusted users. This index is evaluated starting from the common viewed or bought items and it is calculated according to the Eq. 2.

$$trust(u, v) = \frac{card(A_{u,v})}{card(A_u)} \quad (2)$$

where  $A_{u,v}$  is the set of common viewed or bought items by user  $u$  and user  $v$ , and  $A_u$  is the set of items viewed or bought by user  $u$ .

The proposed approach, named Trust-aware Clustering CF (TRACCF), evaluates the prediction of a rating  $p$  of item  $i$  for the user  $u$  as expressed by Eq. 3.

$$p_i(u) = \bar{r}(u) + \frac{\sum_{v \in V_u} (\sigma \cdot trust(u, v) + (1 - \sigma) \cdot sim(u, v)) \cdot (r_i(v) - \bar{r}(v))}{\sum_{v \in V_u} |\sigma \cdot trust(u, v) + (1 - \sigma) \cdot sim(u, v)|} \quad (3)$$

where  $r_i(u)$  is the rating given to item  $i$  by the user  $u$ ,  $\bar{r}(u)$  is the average rating given by the user  $u$ ,  $trust(u, v)$  and  $sim(u, v)$  are respectively trust index and similarity function between user  $u$  and user  $v$ ;  $\sigma$  is a weight ranging in the  $[0, 1]$  interval, and  $V_u$  is the set of users belonging to the same cluster of user  $u$ . The adopted hybridization techniques take into account the weighted model proposed by Burke [11], where the score is a weighted linear combination of the trust and of the similarity between two users.

Depending on the  $\sigma$  value, the proposed TRACCF becomes a traditional Clustering Collaborative Filtering (CCF) when *sigma* is equal to 0. While, when *sigma* is equal to 1, it becomes a Clustering-based version of the Trust-aware Recommender System (TaRS) proposed by Massa [8]. Consequently, the proposed approach arises from the hybridization of CCF and Clustering-based TaRS (CTaRS).

The proposed recommendation system is based on four steps: (i) User Clustering, (ii) Elicitation of trust information, (iii) Prediction Engine, and (iv) Top-N CF. In *User Clustering* step, we select a clustering method in order to group together users in clusters, minimizing the dissimilarity between users assigned to the same cluster. The second step elicits the trust information, while the *Prediction Engine* predicts ratings for items similar to rated ones and by taking into account trust information, and finally, *Top-N CF* selects the Top-N items to recommend.

## 4 Experimentation

### 4.1 Dataset

In the experimentation we consider MovieLens and Epinions datasets. The MovieLens dataset was collected by the GroupLens Research Group [13] and consists of 1,000,204 ratings for 3,900 movies by 6,040 users. Each user in the dataset has rated at least 20 movies.

The Epinions dataset has been extracted by [8] from *Epinions.com* which is a consumer opinion website where people can rate products and can express

their Web of Trust, i.e. users whose ratings they have consistently found to be valuable. The extracted dataset provides 13,668,319 ratings by 132,000 users on 1,560,140 articles. Furthermore, Epinions dataset collects 841,372 statements divided in 717,667 trusts and 123,705 distrusts. In both datasets, the ratings are integer numbers on a 1(“bad”)-to-5(“excellent”) scale.

In order to assess the goodness of a dataset we evaluate the sparsity level<sup>1</sup>. The overall sparsity of MovieLens and Epinions is quite high and as the majority of ratings datasets (such as BookCrossing and Netflix), it presents a distribution that follows Zipf’s power law distribution [14] (a lot of ratings for few items and few ratings for the remaining items). The value of sparsity is 0.9575 for MovieLens dataset and 0.9993 for Epinions one, entailing respectively 95.75% and 99.93% of unrated items.

The datasets are randomly divided into a training set, which ranges from 80% to 95% of the ratings per user and a testing set (the remaining ratings). Starting from the training set recommendation algorithms predict unknown ratings, while the testing set is used to evaluate the accuracy of the predictions.

## 4.2 Evaluating the quality of a recommendation system

The quality of a recommendation system can be decided on the basis of evaluation. The type of metric adopted depends on the type of applications [12]. Indeed, Herlocker [12] highlighted the problem of comparing different recommendation algorithms and defined some metrics that had been used to evaluate them.

The commonly-used metrics are accuracy metrics as Root Mean Squared Error (RMSE) which is becoming popular partly because it is commonly used for the evaluation of movie recommendation performance. It amplifies the contributions of the absolute errors between the predictions and actual ratings and is defined as:

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{u,i} (p_i(u) - r_i(u))^2} \quad (4)$$

where  $N$  is the total number of ratings over all users,  $p_i(u)$  is the predicted rating for user  $u$  on item  $i$ , and  $r_i(u)$  is the actual rating. The lower values of  $RMSE$  entail better predictions.

Another kind of metrics in recommendation system is the coverage that can be defined as “the percentage of a dataset that the recommender system is able to provide prediction for” [12].

## 4.3 Results

Analysis of performance is aimed at comparing the results offered by Trust-based Recommendation System (TRS), and Clustering Collaborative Filtering (CCF).

<sup>1</sup> Sparsity level of a dataset is evaluated by means of  $1 - \frac{N}{m \cdot n}$  where  $n$  is the number of items,  $m$  is the number of users and  $N$  is the total number of ratings over all users.

TRS takes into account the different formulations proposed in the Section 3, while CCF is implemented by k-means algorithm.

According to [5], we measure the similarity with Pearson correlation coefficient between users  $u$  and  $v$  as:

$$sim(u, v) = \frac{\sum_{i \in A_{u,v}} (r_i(u) - \bar{r}(u)) (r_i(v) - \bar{r}(v))}{\sqrt{\sum_{i \in A_{u,v}} (r_i(u) - \bar{r}(u))^2} \sqrt{\sum_{i \in A_{u,v}} (r_i(v) - \bar{r}(v))^2}} \quad (5)$$

where  $\bar{r}(u)$  is the average rating of user  $u$ .

We adopt two trust indexes, the first one is explicitly provided by Epinion dataset, while the other is evaluated according to Eq.2.

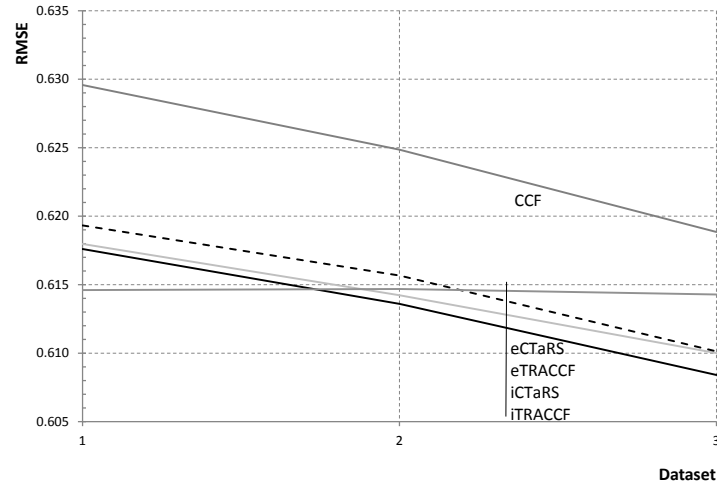
We consider Epinions (dataset ID ranging from 1 to 3) and MovieLens (dataset ID ranging from 4 to 6) dataset with increased size of training set (i.e., 80%, 90%, and 95% of the entire dataset) and setup the algorithms with the following parameters: (i) number of clusters  $k = 20$ , (ii) trust evaluated according to Eq.2, and (iii) similarity function as in (5)). We experimented 5 different approaches: Clustering-based TaRS with explicitly defined values of trust (eCTaRS), Clustering-based TaRS with implicitly defined values of trust (iCTaRS), Clustering CF (CCF) and two versions of the proposed hybrid algorithm, i.e., eTRACCF and iTRACCF which consider at the same time trust and similarity information, but differ for the criteria adopted to evaluate the trust information: in the first approach the trust is explicitly provided by the users (only for the Epinion dataset), while in the second one, it is evaluated starting from the common viewed or bought items without taking into account the expressed rating.

In Fig. 1 and in Fig. 2, we compared the different algorithms at varying training size. Charts respectively refer to the average of RMSE and the average of coverage of 5 different runs (in each run, ratings per user in training set are randomly selected).

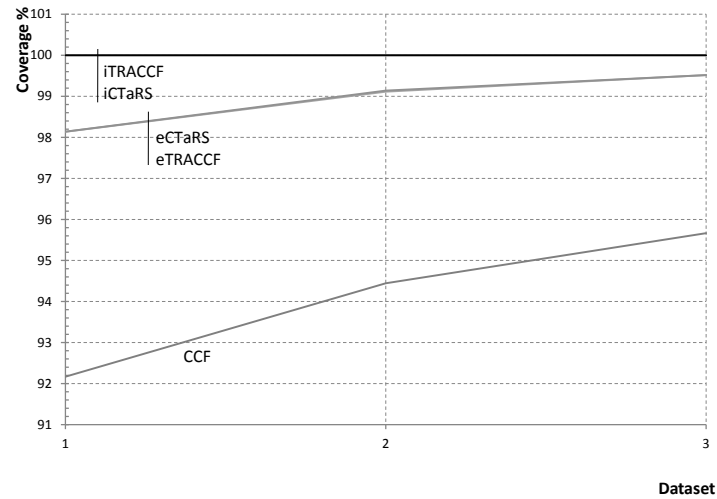
Fig. 3 shows a bar chart, where a comparison between the trust approach and pure clustering approach is depicted, while Fig. 4 compares explicit and implicit trust within the proposed algorithm.

On the one hand, iCTaRS and iTRACCF seems to outperform in term of accuracy (a lower value of RMSE). On the other hand, iCTaRS and iTRACCF adopt the implicit evaluation of trust which guarantees a higher coverage of predictions as proved in Fig. 2 where the Epinions dataset is taken into account and in Fig. 5 where the MovieLens dataset is also investigated. These findings can be useful for the adoption of trust information even if they are not explicitly provided.

Investigating these results more deeply, we consider a pairwise (two-sided) Wilcoxon signed-rank test, reporting results in Table 1, where the average value of RMSE of 15 different runs per technique (5 repetitions per training size) is shown. Assuming 0.05 as upper limit to reject the null hypothesis, we can affirm that there is statistical difference between iTRACCF and other RS approaches

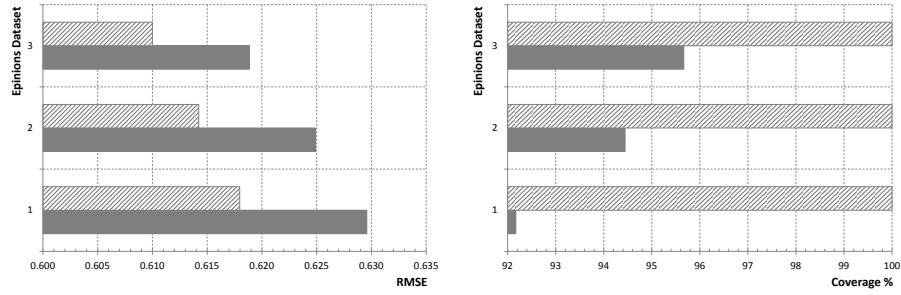


**Fig. 1.** Comparison of eCTaRS, iCTaRS, CCF, eTRACCF, and iTRACCF by means of average RMSE at varying the training size (from 80% to 95% of the Epinions dataset).

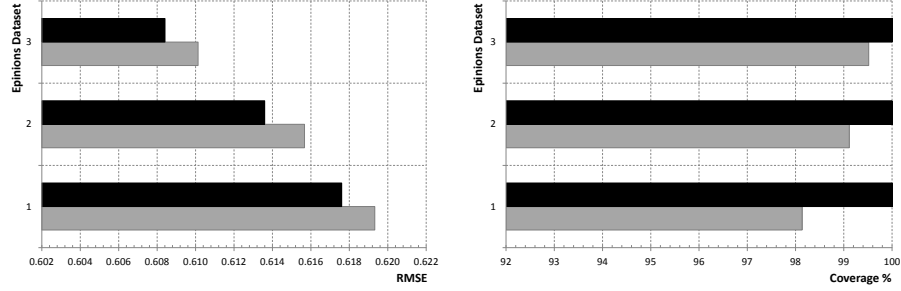


**Fig. 2.** Comparison of eCTaRS, iCTaRS, CCF, eTRACCF, and iTRACCF by means of average coverage at varying the training size (from 80% to 95% of the Epinions dataset).

(i.e., CCF and iCTaRS). We prove that eTRACCF outperforms CCF, but we cannot reject the null hypothesis when eTRACCF and eCTaRS is compared.



**Fig. 3.** Comparison of iCTaRS (light-grey bar) and CCF (dark-grey bar) by means of average RMSE and average coverage of 5 different runs.



**Fig. 4.** Comparison of eTRACCF (grey bar) and iTRACCF (black bar) by means of average RMSE and average coverage of 5 different runs.

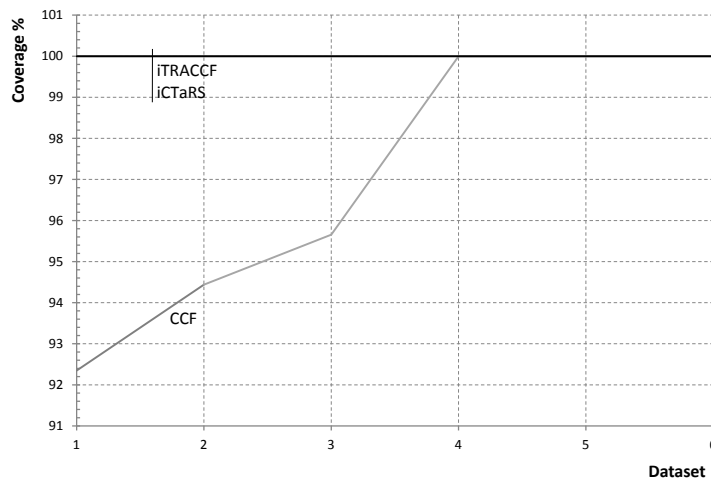
**Table 1.** Wilcoxon signed-rank test on the Epinions dataset: Average RMSE and p-values.

Algorithm	Average		p-value			
	RMSE	CCF	eCTaRS	iCTaRS	eTRACCF	iTRACCF
CCF	0.6244	-	1.526e-03	6.104e-05	6.104e-05	6.104e-05
eCTaRS	0.6145	1.526e-03	-	6.387e-01	8.469e-01	5.614e-01
iCTaRS	0.6141	6.104e-05	6.387e-01	-	1.245e-02	3.052e-04
eTRACCF	0.6150	6.104e-05	8.469e-01	1.245e-02	-	6.104e-05
iTRACCF	0.6132	6.104e-05	5.614e-01	3.052e-04	6.104e-05	-

To sum up, the proposed approach provides three main advantages: first of all, in every experiment conducted on the same dataset iTRACCF outperforms iCTaRS and CCF, as shown in Fig. 1, then even if there is no explicit trust information, it is possible to evaluate it by means of user web usage mining, as



depicted in Fig. 3 and in Fig. 4, and finally, trust information can increase the coverage of recommendation, as proved by Fig. 2 and Fig. 5.



**Fig. 5.** Comparison of iCTaRS, CCF, and iTRACCF at varying the adopted dataset.

#### 4.4 Example of application

In order to assist the merchant in suggesting interesting items to his customers, we developed a tool able to optimize web pages employing the algorithm described above. The tool interface (see Fig. 6) is organized in four main panels.

The first panel on the top allows to specify the input files which is preliminarily divided into training and testing. The input file contains all the ratings expressed by all the users for the items in the catalogue. The second panel, named Parameter, allows to specify the recommendation algorithm and related parameters. Selecting a model-based algorithm it is possible to set the required parameters, otherwise a default value is taken into account. Once a cluster-based algorithm is selected, the tool allows the visualization of users in the different clusters. The third panel, named Recommendation, is divided into two sections: (i) on the left, the list of users is shown, where each user is colored by green if recommendations are available for that user, otherwise he is colored by red; (ii) on the right, once a user is selected, the recommended items and predicted rating are shown. Double-clicking on user opens a pop-up dialog that shows details on the selected user such as his expressed ratings (History section), a graphical representations of the user preferences, and the items recommended by means of the selected algorithm. Fig. 6 shows the results for the users 1641 of MovieLens dataset, his rated movies and the top suggestions.

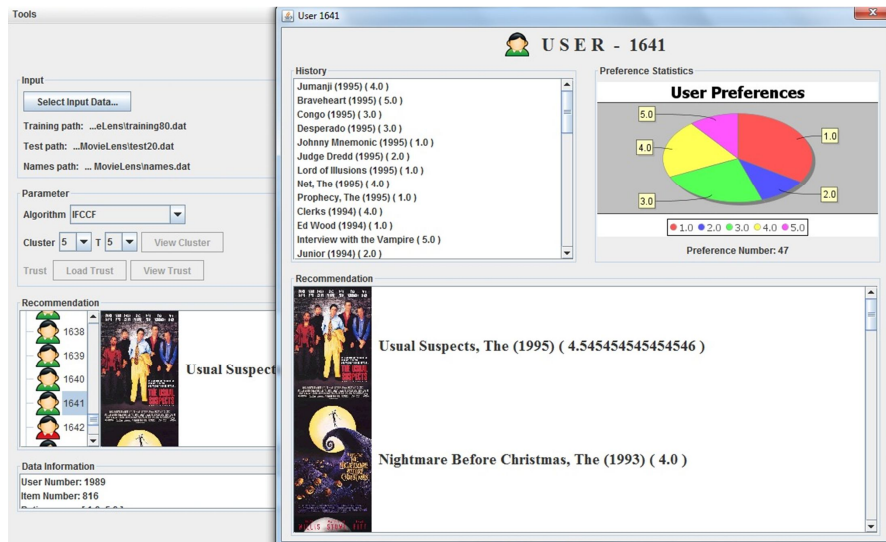


Fig. 6. Tool for selecting contents.

The proposed tool gives the merchant different features in e-Commerce domain. First of all, by selecting a target user it is possible to discover customer's profile of interest, his interaction history and a set of candidate items that he can potentially consider very interesting. In other words, this tool allows knowledge extraction regarding customers and suggests implicitly which products add in some personalized marketing campaigns enhancing customer loyalty.

## 5 Conclusions and future work

In this paper we investigate the application of different collaborative filtering techniques as a viable approach to identify interesting products for customers.

We evaluated the trust by means of shopping chart of the customers and proved the benefits of this information in recommending items. We proposed a hybrid model-based CF which takes into account estimated trust and user similarity. Similarity and trustworthy information is adopted in order to elaborate the rating predictions.

Experimental results showed that the proposed approach slightly increases the accuracy of the predicted values when the same dataset is considered. Moreover, we proved that trust information increase the coverage of predictions without affecting recommendation quality.

However, we aim at extending the proposed approach in the future, taking into account other information arising from social network analysis such as the relationship level (e.g., friend, colleague), the number of common friends, and the number of common social interaction (e.g., thematic groups, content sharing).

These factors could lead to a more accurate measurement of trust entailing further improvement of suggestions.

**Acknowledgments.** This work was partially supported by MSE under the Intelligent Virtual Mall (InViMall) Project MI01-00123.

## References

1. Liu, L., Shi, W.: Trust and reputation management. *Internet Computing*, IEEE **14**(5) (sept.-oct. 2010) 10–13
2. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. *Commun. ACM* **35** (December 1992) 61–70
3. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering*, IEEE Transactions on **17**(6) (June 2005) 734–749
4. Huang, C., Yin, J.: Effective association clusters filtering to cold-start recommendations. In: *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh Int. Conf. on*. Volume 5. (August 2010) 2461–2464
5. Birtolo, C., Ronca, D., Armenise, R., Ascione, M.: Personalized suggestions by means of collaborative filtering: A comparison of two different model-based techniques. In: *NaBIC*, IEEE (2011) 444–450
6. Kitisin, S., Neuman, C.: Reputation-based trust-aware recommender system. In: *Securecomm and Workshops*, 2006. (28 2006-sept. 1 2006) 1–7
7. Liu, B., Yuan, Z.: Incorporating social networks and user opinions for collaborative recommendation: local trust network based method. In: *Proc. of the Workshop on Context-Aware Movie Recommendation. CAMRa10*, New York, NY, USA, ACM (2010) 53–56
8. Massa, P., Avesani, P.: Trust-aware recommender systems. In: *Proceedings of the 2007 ACM conference on Recommender systems. RecSys '07*, New York, NY, USA, ACM (2007) 17–24
9. Sun, J., Yu, X., Li, X., Wu, Z.: Research on trust-aware recommender model based on profile similarity. In: *Computational Intelligence and Design*, 2008. *ISCID '08. International Symposium on*. Volume 1. (oct. 2008) 154–157
10. Wu, Z., Yu, X., Sun, J.: An improved trust metric for trust-aware recommender systems. In: *Education Technology and Computer Science*, 2009. *ETCS '09. First International Workshop on*. Volume 1. (march 2009) 947–951
11. Burke, R.D.: Hybrid web recommender systems. In Brusilovsky, P., Kobsa, A., Nejdl, W., eds.: *The Adaptive Web*. Volume 4321 of *Lecture Notes in Computer Science*., Springer (2007) 377–408
12. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* **22**(1) (January 2004) 5–53
13. Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., Riedl, J.: GroupLens: An open architecture for collaborative filtering of netnews. In: *Proc. of ACM 1994 Conference on Computer Supported Cooperative Work*, ACM (1994) 175–186
14. Zaier, Z., Godin, R., Faucher, L.: Evaluating recommender systems. In: *Proceedings of the 2008 International Conference on Automated solutions for Cross Media Content and Multi-channel Distribution*, Washington, DC, USA, IEEE Computer Society (2008) 211–217