Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems

School of Information Systems

11-2009

Trust relationship prediction using online product review data

Nan MA

Ee Peng LIM
Singapore Management University, eplim@smu.edu.sg

Viet-An NGUYEN

Aixin SUN
Nanyang Technological University

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Databases and Information Systems Commons, and the Numerical Analysis and Scientific Computing Commons

Citation

Trust Relationship Prediction Using Online Product Review Data

Nan Ma
Sch of Computer Engineering
Nanyang Technological University
mana0004@ntu.edu.sg

Ee-Peng Lim
School of Information Systems
Singapore Management University
eplim@smu.edu.sg

Viet-An Nguyen
School of Information Systems
Singapore Management University
vanguyen@smu.edu.sg

Aixin-Sun
Sch of Computer Engineering
Nanyang Technological University, Singapore
axsun@ntu.edu.sg

Haifeng Liu
IBM China Research Lab
Beijing, China
liuhf@cn.ibm.com

ABSTRACT

Trust between users is an important piece of knowledge that can be exploited in search and recommendation. Given that user-supplied trust relationships are usually very sparse, we study the prediction of trust relationships using user interaction features in an online user generated review application context. We show that trust relationship prediction can achieve better accuracy when one adopts personalized and cluster-based classification methods. The former trains one classifier for each user using user-specific training data. The cluster-based method first constructs user clusters before training one classifier for each user cluster. Our proposed methods have been evaluated in a series of experiments using two datasets from Epinions.com. It is shown that the personalized and cluster-based classification methods outperform the global classification method, particularly for the active users.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Data Mining

General Terms

Measurement, Experimentation

Keywords

Trust prediction, Web of trust

1. INTRODUCTION

1.1 Motivation

With the popularity of online social networking sites, many users construct user profiles online and explicitly maintain their relationships with others using the user-friendly interface provided by these sites. Many of such relationships are casual friendships like in the case of Facebook and MySpace. Others are more serious, e.g., a list of favorite sellers maintained by each buyer in eBay[1]. All these relationships involve different degrees of trust and can potentially be used for personalizing search and e-commerce services to users. In [4, 5, 6], trust-aware recommender systems were proposed using trust relationships. When deriving recommendations from similar users, the trust-aware systems assign higher weights to recommended items from trusted users so as to personalize the recommendations to the user's trust network. In P2P file sharing, Kamvar et.al proposed the Eigentrust model to derive the reputation of each node in a P2P network by aggregating the local trust among peer nodes [7]. By making the reputation of nodes publicly available, users can select the right peers to share and download files.

Despite the many applications of trust relationships in e-business, explicit specification of trust relationships are usually not common among all users. The distribution of such user-specified trust relationships can often be modeled by a power law function with a small proportion of users specifying many trust relationships while the large majority specifying very few or none trust relationships (see Section 3.1). This can be attributed to the large proportion of passive users who do not contribute efforts to identify their trustees. Even for the active users, a complete coverage of their trust relationships may not be possible unless they are meticulous and timeliness in their relationship updates.

To overcome the sparseness in user-specified trust relationships, it is important to derive trust relationships from other available data. Researchers have introduced trust propagation techniques to derive trust relationships using known trust relationships [8, 9, 10, 11]. Such techniques are able to predict trust relationship between two users when there

---

[1]http://www.ebay.com.

are existing trust relationships connecting them indirectly. In practice, however, we may not always predict correctly by such connectivity information. The trust connectivity also may not be high enough for applying propagation techniques. In this paper, we will therefore draw upon user interaction data to predict trust relationships instead.

1.2 Objective and Contribution

In this paper, we focus on user trust in Epinions.com[2], a well known product review Web 2.0 site. We study how inter-user trust in Epinions can be derived from observed interactions that occur among them. These include users writing product reviews which are subsequently rated and commented by other users. Epinions data has been used in trust propagation and trust-based recommendation research [11, 12, 6]. Trust prediction using interaction data as features on the Epinions and related data has only been recently investigated by [13, 14].

This paper gives a detailed analysis of two Epinions datasets and the use of SVM to predict trust in the datasets. The first dataset was constructed by crawling the video category data from Epinions. The second dataset is made available publicly for trust research. The unique contribution of this work includes the cluster specific and personalized classification methods which have shown improved prediction accuracy over the standard global classification method. Our results have also shown that trust relationships involving more active trustors can be predicted more accurately than those of less active trustors. This finding can be observed throughout cluster specific and personalized classification methods. This suggests that an active trustor's choice of trustee can be more easily determined from the user-generated features of his/her interactions with the trustee compared with the less active ones.

1.3 Paper Outline

The remainder of the paper is organized as follows. Section 2 reviews the related work. An overview of the Epinions datasets used in this work and the trust prediction task is given in Section 3. Our proposed cluster-centric approach is given in Section 4. Section 5 covers our experimental study and results. Finally, Section 6 concludes this paper.

2. RELATED WORK

Trust can be either *global* or *local*. Global trust, often known as *reputation* or *trustworthiness*, refers to the overall trust a user community has on an individual and is therefore a global property of a user. Local trust, on the other hand, is a relationship from a trustor to his or her trustee. The models that are designed for global trust include Eigentrust [7], Peertrust [15], etc.. The user reputation system implemented by eBay is also an example of global trust. The global trust research works do not address local trust relationship prediction. Instead, they may derive global trust from local trust information.

Inferring trust from known trust relationships among users relies on an already known Web of trust that captures users expressing trust on the other users. Guha, Kumar, Raghavan and Tomkins proposed a trust propagation model to predict trust between two users with or without mutual interaction through such a Web of trust [8]. Other works on

Table 1: Dataset Statistics

Description	EpinionsVideo	EpinionsTrustlet
# users	44,089	405,176
# trustors	23,756	95,318
# trustees	21,065	84,601
# trustors and trustees	18,208	48,091
# trustors or trustees	26,613	131,828
# Web of trust relationships	231,759	717,667
# Web of distrust relationships	N.A.	123,705
# review writers	37,334	326,983
# review raters	24,474	120,492
# review writers and raters	19,788	66,570
# review writers or raters	42,020	380,905
# (u_1, u_2) pairs where u_1 rates some review by u_2	2,818,159	4,835,208

propagating trust through the Web of trust include [9] and [11]. Since these approaches rely on a known Web of trust, their performance is affected by trust sparsity.

Compared with earlier works, this paper takes a supervised learning approach to trust prediction using features derived from pairs of users who interact. Based on the SVM-based classification approach first introduced in [13], we develop new classification methods (also based on SVM) to consider users of different degrees of activeness, and groupings of trustors.

3. BACKGROUND AND OVERVIEW

3.1 Epinions Datasets

We have selected two Epinions datasets for evaluating and comparing trust prediction methods. Epinions is a product review website with users contributing reviews (as **review writers**), ratings (as **review raters**) and comments (as **review commentors**) on reviews. Each Epinions user can also explicitly maintain a set of trust (or distrust) relationships to other users he or she trusts (does not trust). The users who trust others are known as **trustors**. The trusted users are known as **trustees**. The same user can be both a trustor and trustee in different trust relationships, but not in the same one. A set of trust relationships connecting a set of users is also known as Web of trust.

Trust relationships are publicly accessible in Epinions but not the distrust relationships perhaps due to sensitivity of the latter. We constructed the first dataset (called **EpinionsVideo**) by crawling all the product reviews of "Videos & DVDs" category on April 15, 2008. The dataset also includes all ratings and comments on these product reviews, writers, raters and commenters of these reviewers, and the Web of trust relationships among these users. The second column of Table 1 shows the statistics of EpinionsVideo. The second Epinions dataset (called **EpinionsTrustlet**) was made available by Massa for trust research [16]. Unlike EpinionsVideo, the latter contains product reviews and review ratings from all categories before May 30, 2002, and both Web of trust and distrust relationships before August 12, 2003. The third column of Table 1 shows the statistics of EpinionsTrustlet. As the distrust relationships are hidden from public access, we were not able to crawl them for EpinionsVideo. On the other hand, EpinionsTrustlet contains anonymized ids, it is not easy to identify the user accounts or product reviews involved.

Sparseness of trust/distrust relationships. On the whole, both EpinionsVideo and EpinionsTrustlet have very sparse trust (and distrust for the latter) relationships among

their users. As shown in Figures 1(a) and 1(b), the number of trust relationships owned by each user is highly skewed. There are very few users who trust others, while the large majority maintains very few trust relationships. In the case of EpinionsVideo, most users have very few (< 8.48) trust relationship given that mean=8.48, while the more active trustor(s) has 1614 trustees. The skewness is even more prevalent for EpinionsTrustlet.

Distribution of write-rate interactions. Other than expressing trust and distrust relationships, users in Epinions can interact with one another in the following forms: (a) one reads the reviews written by another, (b) one rates the reviews written by another, (c) one comments on the reviews written by another, (d) one reads the ratings by another, (e) one reads the comments by another. While the reading interactions ((a), (d) and (e)) are not available at the Epinions website, we focus on inter-user trust relationships built based upon other forms of interactions (i.e., (b) and (c)).

We define the **write-rate writer count** of a user as the number of review writers whose reviews have been rated by the user. Figures 1(c) and 1(d) depict the distributions of write-rate writer counts in EpinionsVideo and EpinionsTrustlet respectively. Most of EpinionsVideo users have less than 63.92 write-rate writer counts. The skewness can be more clearly found in EpinionsTrustlet.

We say that a user is **active** if he/she has a large write-rate writer count. A user is **popular** if he/she has a large write-rate rater count. As shown in the figures, there are small number of highly active users who have rated many reviews by other users. In EpinionsTrustlet, there are also users without write-rate interactions as the dataset was obtained not by crawling from product reviews. We have therefore excluded these users from trust prediction.

The main challenge of predicting trust relationships of less or non-active users is that there are not much historical data for one to learn the way they trust others. Predicting the trust relationships of these users as trustors is non-trivial. Further, it is not clear whether predicting trust relationships of highly active users is easier compared to that for the less active ones. We will revisit these two issues in Section 5.

3.2 Trust Relationship Prediction Task

In this paper, we consider the trust relationship prediction task as a classification problem. The classification involves each candidate trustor-trustee pair to be assigned either a *trust* or *no-trust* label. For simplicity, we call a trustor-trustee pair a **user pair** and require the pair to involve some write-rate interactions. We would like to focus on predicting trust relationships built upon these interactions. As one rates the reviews written by another, the former should develop trust or non-trust on the latter. It is possible for trust (and non-trust) to form through other forms of interactions, we have excluded them from our study due to non-availability of such data.

The classification approach to predict trust requires features to be identified and training data to be given. Sections 3.3 and 3.4 will describe the features extracted from write-rate interactions and classification method introduced in [13].

3.3 User-Generated Features

Our methods, similar to [13, 14], represent each user pair using both **user** and **interaction features**. User features refer to those that are specific to users. Interaction features are derived from interaction between a candidate trustor and his/her trustee. For trust prediction in Epinions data, the interaction features that relate a trustor to his/her trustee include ratings the former gives to reviews written by the latter. Table 2 lists the user features (3-4, 6-9, 11-12, 15, and 17) and interaction features (1, 2, 5, 10, 13, 14, 16, 18, 19) that have been shown to yield good trust prediction accuracy in [13]. Additional features are also given in [14] but we could not implement them due to a lack of their detailed description. We will therefore only use those in Table 2 in our work.

Each review consists of a **product rating score** ranging from 1 to 5 stars. It also has an aggregated **helpfulness rating** which has ordinal values "off-topic", "not-helpful", "somewhat helpful", "helpful", and "very helpful" obtained by averaging the helpfulness ratings from all users rating the review. All product rating scores of 1 to 5 stars and helpfulness rating values have been mapped into 0.2 to 1.0 (with 0.2 interval) accordingly. Product ratings of ≥ 0.8 are considered good. Feature 7 involves reviews with good product rating. Reviews with average helpfulness rating values of 0.8 or above are considered helpful. Hence, features 1, 2 and 4 involve helpful reviews. Feature 19 refers to the largest difference in helpfulness rating between u_1 and other ratings assigned to the same review for all reviews by u_2 rated by u_1. This feature tells if u_1 has large rating score disagreement with other raters of u_2's reviews.

## 3.4 Global Trust Relationship Classification (GC)

Once a given user pair is represented by the above interaction features, a straightforward prediction approach is to train a global classifier using a set of training user pairs with known labels. SVM with linear kernel is used as the underlying classification technique throughout this work [17]. In the case of EpinionsVideo, there are only few user pairs labeled with trust relationships and the remaining user pairs are not labeled. In EpinionsTrustlet, distrust user pairs are available but they exist in very small number. For simplicity, we shall focus on classifying trust pairs from non-trust pairs and ignore the distrust pairs. We denote this global classification method by **GC**.

4. CLUSTER-CENTRIC AND PERSONALIZED TRUST RELATIONSHIP CLASSIFICATION

4.1 Cluster-Centric Classification (CC)

The cluster-centric trust relationship classification is designed based on the hypothesis that similar users develop trust relationships with others in a similar way. There are several advantages taking the cluster-centric classification approach, namely:

- Unlike global classification, cluster-centric classification attempts to customize classifiers for different groups of trustors. Each classifier may perform more accurately than global classifier.

- Compared with personalized classification, cluster-centric classification has training instances from similar users combined together so as to provide a larger pool of
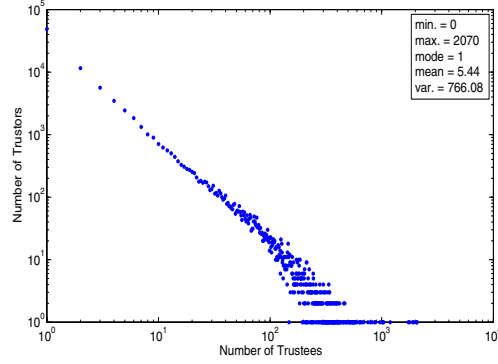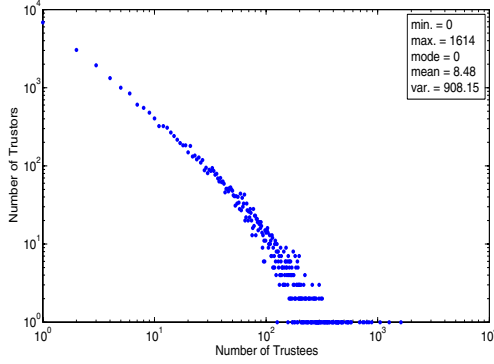
(a) Distrib. of Trust Relationships in EpinionsVideo (b) Distrib. of Trust Relationships in EpinionsTrustlet
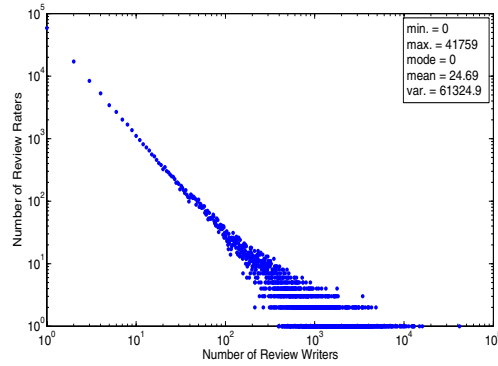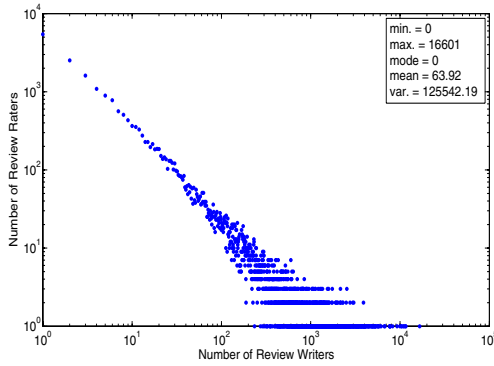
(c) Distrib. of Write-Rate Writer Counts in Epin- (d) Distrib. of Write-Rate Writer Counts in Epinion-
ionsVideo sTrustlet

Figure 1: Distribution Statistics

Table 2: Selected User-Generated Features for User Pair (u_1, u_2)

Id	Features
1	# ratings from u_1 to u_2's reviews that are ≥ 0.8 ("Very Helpful" or "Most helpful")
2	# reviews by u_2 with average rating score ≥ 0.8 and rated by u_1
3	# ratings given to u_2's reviews
4	# ratings given to u_2's reviews that are ≥ 0.8
5	# ratings from u_1 to u_2's reviews
6	# reviews by u_2
7	# reviews by u_2 with product score ≥ 0.8
8	Standard deviation of product scores by u_2
9	Standard deviation of # competing ratings given to u_2's review
10	Frequency of ratings from u_1 to u_2's reviews
11	Standard deviation of ratings on u_2's reviews
12	# reviews per day by u_2
13	Standard deviation of ratings on u_2's reviews before u_1 gives his rating
14	Standard deviation of lengths of reviews by u_2
15	Average product scores of reviews by u_2
16	Max # ratings given to u_2's reviews rated by u_1
17	Max # ratings given to u_2's reviews
18	Max # ratings given to the u_2's review before u_1 gives rating to u_2's review
19	Min agreements of u_1's ratings with other ratings given to u_2's reviews

50

training data for each classifier. Ideally, this can potentially achieve good prediction accuracy.

To form user clusters, inter-user similarity measure is required. We can classify inter-user similarity measures to be *attribute-* and *link-based* [18]. The former derives similarity based on similar user attribute values. Examples are Levenstein and Euclidean distance for string and numeric attributes respectively. Link-based similarity is defined based on neighborhood linked to the users to be compared, e.g. [19]. Most of these measures are mainly for entity resolution. In this work, we do not intend to resolve distinct users. Instead, we would like to cluster them by the density of connectivity in their neighborhood within the Web of trust.

For a given Web of trust relationships, we adopt an efficient **divisive hierarchical clustering** method using **normalized minimum cut** to measure the dissimilarity between two user clusters [20]. The directions of trust relationships are ignored in this clustering method. When two users are linked by some trust relationships, we create an edge linking them and assign an edge weight of 2 if the users trust each other (i.e., the link involves two trust relationships), 1 if only one of them trusts another, and 0 if no trust relationship exists between them. A cut of two sets of users U_A and U_B, denoted by $cut(U_A, U_B)$, is thus the sum of weights of edges linking U_A and U_B. The normalized cut of a set of users U into two disjoint node clusters U_A and U_B is defined as:

$$Ncut(U_A, U_B) = \frac{cut(U_A, U_B)}{assoc(U_A, U)} + \frac{cut(U_A, U_B)}{assoc(U_B, U)}$$

where $assoc(U_A, U)$ (or $assoc(U_B, U)$) denotes the sum of weights of edges between users in U_A (or U_B) and users in U. By bisecting the Web of trust with the minimum normalized min cut repeatedly, a set of clusters can be found.

Once users are grouped into clusters, the CC method trains a SVM classifier for each cluster of users. Given a user cluster U_i, we use user pairs (u_i, u_j) where $u_i \in U_i$ as training instances. To classify a user pair (u_i, u_l), one has to use the classifier of the cluster u_i belongs to. In CC, there is the additional overhead of maintaining user clusters and their classifiers as new users join or existing users leave the site, and retraining the classifiers when there are changes to the existing user pair features due to their new interactions. The evaluation of this overhead is beyond the scope of this paper but will be a topic for future investigation.

4.2 Personalized Classification (PC)

The personalized classification method (denoted by **PC**) creates SVM classifiers customized to individual styles of developing trust relationships. For each user u_i, we use user pairs $(u_i, *)$'s for training a personalized classifier. To classify a user pair (u_k, u_l), one will apply the u_k's personalized classifier. In PC, retraining of classifiers is still required but there is no maintenance overheads for user clusters. The disadvantage here is potentially a very large number of personalized classifiers.

5. EXPERIMENTS AND RESULTS

5.1 Experiment Setup

Given the above GC, CC and PC methods, we would like to evaluate and compare their prediction accuracies in a series of experiments. Ideally, we would like to compare them

for all user pairs in our Epinions datasets. This however will incur too much training overheads especially for CC and PC. The high overhead can be attributed to: (a) large number of user pairs that can be used for training (particularly the negative pairs), and (b) large number of trustors for clustering and training classifiers in the case of CC and PC. We have therefore chosen to conduct experiments on subsets of trustors representing different levels of activeness. In particular, our experiments aim to answer the following research questions:

- Is trust prediction easier for very active users compared with the less active ones? How are the prediction accuracies compared with the baseline prediction methods?

- Does personalized trust classification method (PC) work better than global (GC) and cluster-based classification (CC) methods? Do they perform differently for very active and less active users? In the case of CC, what should be the choice number of clusters?

- Is there any significant performance differences when performing trust prediction on EpinionsVideo and EpinionsTrustlet datasets?

We compare the three classification methods: (a) global classification (GC), clustered classification (CC), and personalized classification (PC). For CC, we experimented for different number of clusters k from 2 to 10. We also include a baseline method based on random selection.

As mentioned in Section 3.1, explicit trust relationships are sparse and the distribution of write-rate interactions is very skewed. To ensure sufficient training data for our classification-based method evaluation and comparison, here we have only included users who have ≥ 50 write-rate writer count and ≥ 25 trustees among the writers they interact with. In order to reduce the overhead for training classifiers as well as to compare the classification performances for **highly active** and **less active** users, we have selected top 500 and bottom 500 active users measured by write-rate writer count and evaluated the classification methods for the two groups of users denoted by U^t and U^b.

Table 3 shows the detailed statistics for these two groups of users in EpinionsVideo and EpinionsTrustlet. It is interesting to note that highly active users rate about 3 times more distinct review writers than less active users in both EpinionsVideo and EpinionsTrustlet. A highly active trustor in EpinionsVideo (or EpinionTrustlet) rates on average 36,142 (or 62,343) distinct review writers compared to 13,800 (or 19,325) for less active users. On the other hand, the number of distinct trustees trusted by highly active users is not very different from that by less active users in both EpinionsVideo and EpinionsTrustlet. For example, the highly active users in EpinionsVideo trust 7304 distinct trustees while the less active ones trust 7451 distinct trustees. However, highly active users share many more trustees compared with less active users. For example, we only have 7304 distinct trustees out of 62,851 trust relationships for highly active users compared with 7451 out of 22,163 for less active users. Hence, the number of distinct users trusted by highly active users is not much different from that of less active users.

We evaluate the prediction performance by **F1** measure as defined by $\frac{2PR}{P+R}$ where P and R represent the precision and recall respectively of the user pairs whose trust relationships are to be predicted. As GC, CC, and PC are evaluated on

Table 3: Highly Active and Less Active User Statistics

Active Users	EpinionsVideo	EpinionsTrustlet
$|U^t|$	500	500
# users with reviews rated by users in U^t	36,142	62,343
# (u_1, u_2) pairs where $u_1 \in U^t$ and u_1 rates some u_2's review	1,263,983	1,373,362
# users trusted by users in U^t	7304	8104
# (u_1, u_2) pairs where $u_1 \in U^t$ and u_1 trusts u_2	62,851	59,391
Less Active Users	EpinionsVideo	EpinionsTrustlet
$|U^b|$	500	500
# users with reviews rated by users in U^b	13,800	19,325
# (u_1, u_2) pairs where $u_1 \in U^b$ and u_1 rates some u_2's review	67,103	59,665
# users trusted by users in U^b	7451	8445
# (u_1, u_2) pairs where $u_1 \in U^b$ and u_1 trusts u_2	22,163	19,917

the highly active users and less active users, we have six different Method-[Active/LessAct] classification results denoted by GC-Active, CC-Active, PC-Active, GC-LessAct, CC-LessAct and PC-LessAct.

The write-rate interaction pairs from users to other users whose reviews are rated are divided into 5 folds. For each user involved as a rater u, we select all other users u has trust relationships with and randomly an equal number of other users u has no trust relationships with. These two sets of users form the positive and negative training pairs for u. To measure $F1$ of a classification method, we use 5-fold validation, i.e., 4 folds for training classifier and the remaining fold for test. This leads us to expect F1=0.5 as the default baseline classification performance.

5.2 Results

Performance on EpinionsVideo. Figure 2(a) shows the F1 performance of GC, CC and PC methods for both highly active and less active users in EpinionsVideo. They all performed better than the F1=0.5 expected from a baseline random method. For the CC method, we show the F1 values for different k ($1 \leq k \leq 10$) clusters. CC with $k = 1$ was identical to GC. The F1's of the GC and PC methods were not affected by k. On the whole, we observed that the F1's for highly active users were better than those of less active users. For the same classification method, the performance differences were more than 0.1. E.g., GC-Active=0.8413 and GC-LessAct=0.7182.

The figure also shows that PC outperformed both GC and CC for both highly active and less active users. This is not a surprise since PC has training data specific to users. The F1's of PC-Active and PC-LessAct were 0.8604 and 0.7659 respectively.

In the case of active users, GC performed slightly better than CC for $2 \leq k \leq 4$ clusters but worse than CC for $5 \leq k \leq 10$ clusters. The former was a small surprise since we had expected CC to be better. This was perhaps due to poor clusters formed for small k. For less active users, CC always outperformed GC for all k's. In fact, CC had F1 increases with larger k's. This was reasonable given that CC was expected to achieve the performance of PC when $k = 500$. This result however did not suggest an ideal k

which remains to be elusive very much like other clustering based methods.

Performance on EpinionsTrustlet. Figure 2(b) shows the F1 performance for active and less active users in EpinionsTrustlet. Similar to EpinionsVideo, we observed that (a) F1 values of active users were better than those of less active users; and (b) PC-Active and PC-LessAct outperformed the GC and CC methods. Unlike in EpinionsVideo, CC performed better than GC for all k's and improved as k got larger.
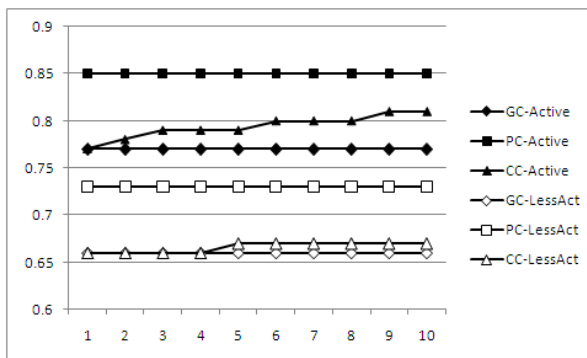
Performance of Personalized Classification (PC). For personalized classification, we further examined the distribution of classifier performance as shown in Figures 3(a) and 3(b) for EpinionsVideo and EpinionsTrustlet respectively. The lines connecting shaded and empty boxes depicts the F1 distribution of personalized classifiers for active and less active users respectively. The Y-axis of the figures shows the number of personalized classifiers for different F1 performance intervals (with interval size of 0.01) in the X-axis. The figures show that majority of personalized classifiers, 318 of them (63.6%), performed better than the global classification for active users in EpinionsVideo. The same observation also holds for less active users in EpinionsVideo. The number of such personalized classifiers was even larger, 413 (82.6%) for active users and 333 (66.6%) for less active users, in EpinionsTrustlet.

Performance for Randomly Selected Users. To further compare the performance of GC, CC and PC classification methods, we evaluated them on a set of randomly chosen 500 users, denoted by U^o. The statistics of these users is shown in Table 4. For both EpinionsVideo and EpinionsTrustlet, we noticed the consistent trend that the more active users have rated more other review writers and formed more trust relationships. The number of distinct trustees however was not far from that of that of less active users. Due to space constraint, we do not show the performance chart.

Our experiments showed that the same performance difference between global and personalized classification also existed for this group of users. For EpinionsVideo, the F1 values of GC and PC were 0.7584 and 0.7944 respectively. For EpinionsTrustlet, the F1 values of GC and PC were 0.7221 and 0.7639 respectively. CC for this group of users
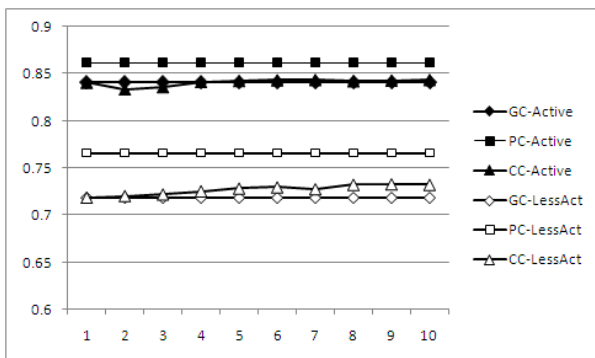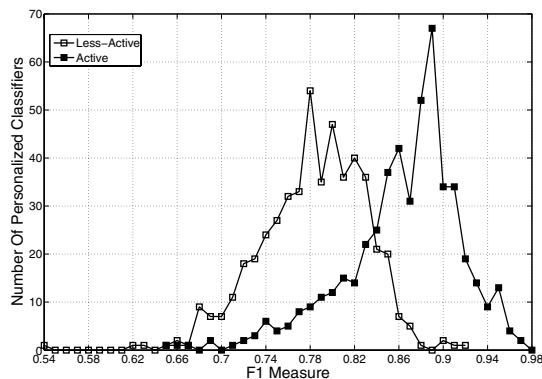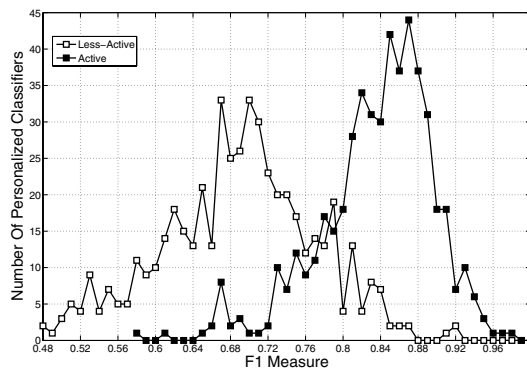
(a) EpinionsVideo (b) EpinionsTrustlet

Figure 2: F1 Performance

(a) EpinionsVideo (b) EpinionsTrustlet

Figure 3: Performance of Personalized Classifiers

also had performance sandwiched between GC and PC, and demonstrated increasing F1's as k increased.

Comparison with Trust Prediction Using Propagation. In the above experiments, we only evaluate trust prediction methods using classification techniques. To compare with earlier trust prediction methods that are based on trust propagation, we implemented Moletrust, a trust propagation based prediction method developed by Massa [11]. We applied Moletrust on the EpinionsTrustlet dataset using 5 fold validation. For candidate pairs that are not connected by trust relationships, we leave them out from F1 measurement since there is no way Moletrust can predict them by trust propagation. As shown in Table 5, Moletrust did not perform well compared to classification methods. This shows that classification methods are more superior in trust prediction accuracy.

6. CONCLUSION

Trust prediction using classification techniques is relatively under explored in contrast with much research works already done in trust propagation. This paper shows that user and interaction features generated from users rating other users' reviews allow us to build classifiers that predict trust relationships with performance significantly better (40 to 70% improvement) than the random baseline method. Our ex-

Table 5: Results of Moletrust and Classification Methods

User Types	Moletrust	GC	PC	CC (k=10)
Active	0.548	0.77	0.85	0.81
LessAct	0.547	0.66	0.73	0.67
Others	0.549	0.72	0.76	

periments have shown that cluster-based and personalized classification methods can achieve even better accuracies by selecting the training examples more relevant to trustors whose trust relationships are to be predicted. These experiments have been obtained on real Epinions datasets.

We plan to extend our work in trust prediction by investigating how cluster classification method can be made feasible so as to avoid using trust relationships. For example, instead of clustering users by trust relationships, we may cluster them by the intensity of their write-rate interactions. The overheads of clustering and cluster maintenance are interesting topics to be studied. Beyond trust prediction, we will also attempt to use the predicted trust relationships for trust-aware recommendation and search.

Table 4: Other Users' Statistics

Other Users	EpinionsVideo	EpinionsTrustlet
$|U^o|$	500	500
# users with reviews rated by users in U^o	23,835	40,851
# (u_1, u_2) pairs where $u_1 \in U^o$ and u_1 rates some u_2's review	231,654	288,513
# users trusted by users in U^o	7577	8497
# (u_1, u_2) pairs where $u_1 \in U^o$ and u_1 trusts u_2	33,183	34,524

Acknowledgment

7. REFERENCES

[1] Jason A. Colquitt, Brent A. Scott, and Jeffery A. LePine. Trust, trustworthiness, and trust propensity: A meta-analytic test of their unique relationships with risk taking and job performance. *Journal of Applied Psychology*, 2(4), 2007.

[2] Dan J. Kim, Donald L. Ferrin, and H. Raghav Rao. A trust-based consumer decision-making model in electronic commerce: The role of trust, perceived risk, and their antecedents. *Decision Support Systems*, 44:544–564, 2008.

[3] Audun Josang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43.

[4] Raph Levien and Alexander Aiken. Attack-resistant trust metrics for public key certification. In *7th Conference on USENIX Security Symposium*, 1998.

[5] Jennifer Golbeck. Generating predictive movie recommendations from trust in social networks. In *International Conference on Trust Management*, pages 93–104, 2006.

[6] Paolo Massa and Paolo Avesani. Trust-aware recommender systems. In *ACM Conference on Recommender Systems*, pages 17–24, 2007.

[7] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *12th International World Wide Web Conference*, pages 640–651, 2003.

[8] R. Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th International Conference on World Wide Web*, pages 403–412, New York, USA, May 2004.

[9] Jennifer Golbeck and James Hendler. Inferring binary trust relationships in web-based social networks. *ACM Transactions on Internet Technology*, 6:497–529, 2006.

[10] Cai-Nicolas Ziegler and Georg Lausen. Propagation models for trust and distrust in social networks. *Information Systems Frontiers*, 7(4-5):337–358, 2005.

[11] Paolo Massa and Paolo Avesani. Controversial users demand local trust metrics: An experimental study on epinions.com community. In *AAAI*, pages 121–126, 2005.

[12] John O'Donovan and Barry Smyth. Trust in recommender systems. In *10th International Conference on Intelligent User Interfaces*, 2005.

[13] Haifeng Liu, Ee-Peng Lim, Hady Wirawan Lauw, Minh-Tam Le, Aixin Sun, Jaideep Srivastava, and Young Ae Kim. Predicting trusts among users of online communities: an epinions case study. In *Proceedings of the 9th ACM Conference on Electronic Commerce*, 2008.

[14] Yutaka Matsuo and Hikaru Yamamoto. Community gravity: measuring bidirectional effects by trust and rating on online social networks. In *WWW*, 2009.

[15] Li Xiong and Ling Liu. Peertrust: supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16:843–857, 2004.

[16] Paolo Massa and Kasper Souren. Trustlet, open research on trust metrics. In *BIS 2008 Workshops Proceedings: Social Aspects of the Web*, 2008.

[17] T. Joachims. Making large-scale svm learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT-Press, 1999.

[18] Indrajit Bhattacharya and Lise Getoor. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 2007.

[19] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[20] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.