

Singapore Management University Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems

School of Information Systems

4-2010

Do you trust to get trust? A study of trust reciprocity behaviors and reciprocal trust prediction

Viet-An NGUYEN

Singapore Management University

Ee Peng LIM

Singapore Management University, eplim@smu.edu.sg

Hwee Hoon TAN

Singapore Management University, hhtan@smu.edu.sg

Jing JIANG


Singapore Management University, jingjiang@smu.edu.sg

Aixin SUN

Nanyang Technological University

DOI: <https://doi.org/10.1137/1.9781611972801.7>

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

 Part of the [Databases and Information Systems Commons](#), [Numerical Analysis and Scientific Computing Commons](#), and the [Organizational Behavior and Theory Commons](#)

Citation

NGUYEN, Viet-An; LIM, Ee Peng; TAN, Hwee Hoon; JIANG, Jing; and SUN, Aixin. Do you trust to get trust? A study of trust reciprocity behaviors and reciprocal trust prediction. (2010). *Proceedings of the 2010 SIAM International Conference on Data Mining*. 72-83. Research Collection School Of Information Systems.

Available at: https://ink.library.smu.edu.sg/sis_research/510

This Conference Proceeding Article is brought to you for free and open access by the School of Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email libIR@smu.edu.sg.

Do You Trust to Get Trust? A Study of Trust Reciprocity Behaviors and Reciprocal Trust Prediction

Viet-An Nguyen, Ee-Peng Lim, Hwee-Hoon Tan, Jing Jiang* Aixin Sun†

Abstract

Trust reciprocity, a special form of link reciprocity, exists in many networks of trust among users. In this paper, we seek to determine the extent to which reciprocity exists in a trust network and develop quantitative models for measuring reciprocity and reciprocity related behaviors. We identify several reciprocity behaviors and their respective measures. These behavior measures can be employed for predicting if a trustee will return trust to her trustor given that the latter initiates a trust link earlier. We develop for this reciprocal trust prediction task a number of ranking method and classification methods, and evaluated them on an Epinions trust network data. Our results show that reciprocity related behaviors provide good features for both ranking and classification based methods under different parameter settings.

1 Introduction

Trust reciprocity refers to two users trusting each other. This two-way trust usually represents a stronger relationship between the connected users compared to one-way trust. A user network with many two-way trust links is likely to be more robust than that with few such links. Such strong and stable trust relationships can also be very useful in developing the next generation search and recommendation applications that aim to provide more personalized and accurate results [9].

Trust reciprocity can be caused by the mindset of: “You scratch my back (trust me), and I will do the same on yours (trust you back)”. A trustor may trust someone hoping to gain trust in return. A trustee returns trust to her trustor to show appreciation at the same time expanding the trustee’s web of trust. The above mindset (e.g., exchanging gifts among friends, returning favors, etc.) exists in many different cultures and leads to a range of trust reciprocity related behaviors. In this paper, we specifically study four interesting trust reciprocity related behaviors: A user who initiates trust with many other users in a non-discriminative manner is said to demonstrate the **trust initiating (I)** behav-

ior. A user who returns trust to anyone who trusts them is said to demonstrate the **trust returning (R)** behavior. A user who gains trust from many others without having to initiate trust earlier is said to demonstrate the **trust attracting (A)** behavior. Finally, a user who can get trustees to trust her back is said to demonstrate the **successful trust seeking (S)** behavior.

1.1 Research Objectives In this paper, we study the phenomenon of trust reciprocity and the above four reciprocity related behaviors. We also aim to show that such behaviors can be effectively used to predict trust links among users. To study trust reciprocity related behaviors, we introduce several models to measure behaviors. To predict trust links using behaviors, we introduce the reciprocal trust prediction task and develop prediction methods using features derived from reciprocity related behaviors. This research is conducted on a publicly available real dataset from Epinions providing us the empirical evidence of trust reciprocity as well as the data for the behavior study and prediction task.

We now summarize our contributions as follows:

- We measure **trust reciprocity** using link reciprocity which has been studied in the context of Web network, email networks, word networks, food networks, etc. [2]. The previously proposed link reciprocity is a global measure as it calibrates reciprocity for a given network. We develop a new **user reciprocity measures** to determine the users with high reciprocity. We show that both global and user trust reciprocities exist in our Epinions trust network data (see Sections 3.2 and 3.3).
- We then develop three models for the four types of trust reciprocity related behaviors. Each model allows us to quantitatively assign values to each user behavior based on a set of principles. This is the first attempt the trust reciprocity related behaviors (i.e., I, R, S and A behaviors) and their measures are systemically studied. We believe this opens a new data mining research direction for trust reciprocity related behaviors.

*Singapore Management University.

†Nanyang Technological University.

- To illustrate the usefulness of trust reciprocity related behaviors, we consider them in the **reciprocal trust prediction** task. In reciprocal trust prediction, we determine the likelihood of having a trustee return trust to his/her trustor. Reciprocal trust prediction is useful in trust recommendation applications that prioritize trust links with other users when prior trust links from these users exist, and trust links with other users with the motive to get returning links from them. Both unsupervised (i.e., ranking) and supervised (i.e., classification) methods of reciprocal trust prediction have been proposed. Our experiments on the Epinions trust network show that the both unsupervised (i.e., BASICRANK, DELAYRANK, REPUTATIONRANK) and supervised methods using trust reciprocity related behavior measures (i.e., SVMBASIC, SVMDELAY, SVMREPUTATION, and SVMALL) perform well in this task compared with trust prediction method using topological features. This suggests that reciprocity related behaviors play an important role in trust prediction.

1.2 Epinions Dataset We conduct this research on a real dataset that consists of trust statements (or trust links) of Epinions users from 17 January 2001 to 12 August 2003 (938 days). In Epinions, users write reviews on products and the reviews can be rated or commented by other users. These user generated content serves as useful reference for users comparing and evaluating products. In addition, Epinions users can specify other users they trust using trust statements. Each trust statement represents a trustor conferring trust on a trustee and is assigned a timestamp. We use $z = 1$ to 938 to denote the timestamps of trust statements from 17 January 2001 to 12 August 2003 respectively. The dataset also consists of a set of non-timestamped trust statements that exist prior to 17 January 2001. We use $z = 0$ to cover this time period.

Table 1 shows the main statistics of this dataset. It shows that about $\frac{34,355}{114,222}$ or 30% of users who have trust links also have reciprocal trust links. $\frac{172,244}{506,934}$ or about 34% of non-timestamped trust statements (i.e., at $z = 0$) are reciprocal links. This proportion is similar for the combined non-timestamped and timestamped trust statements from $z = 0$ to 938 which has $\frac{249,076}{506,934+210,195}$ or 35% reciprocal links. Both the user and trust reciprocal link proportions indicate that high degree of trust reciprocity exists in the trust network.

1.3 Paper Outline For the rest of the paper, we first review the related work in Section 2. We then present the existing network level reciprocity measures and a

Table 1: Statistics of Epinions Dataset

| Description | Number |
|--|---------|
| $ \mathbf{U} $ = # users with trust links | 114,222 |
| $ \mathbf{U}^{\leftrightarrow} $ = # users with reciprocal trust links | 34,355 |
| $ \mathbf{T}_0 $ = # trust links for $z = 0$ | 506,934 |
| $ \mathbf{T}_{[1,938]} $ = # trust links for $z \in [1, 938]$ | 210,195 |
| $ \mathbf{T}_{all} $ = $ \mathbf{T}_0 \cup \mathbf{T}_{[1,938]} $ | 717,129 |
| $ \mathbf{T}^{\leftrightarrow}_0 $ = # reciprocal trust links for $z = 0$ | 172,244 |
| $ \mathbf{T}^{\leftrightarrow}_{938} $ = # reciprocal trust links up to $z = 938$ (including $z = 0$) | 249,076 |

new user level reciprocity measure in Section 3. Our proposed trust reciprocity related user behavior models and their empirical evaluation results are covered in Sections 4 and 5 respectively. Reciprocal trust prediction and our proposed solution methods using reciprocity related behaviors are given in Section 6. Section 7 shows the results of the various prediction methods. We finally conclude the paper in Section 8.

2 Related Work

As will be mentioned in Section 3, link reciprocity in graphs have been studied in [2, 11]. These earlier works are very different from ours in that: (a) we study reciprocity mainly at the user level; and (b) we focus on modeling reciprocity related user behaviors from the timestamped trust links. Furthermore, we have applied these user level reciprocity behavior models to reciprocal trust prediction.

Reciprocal trust prediction is a special kind of trust prediction task and the latter has been studied in several previous papers. Trust prediction can be addressed by non-supervised and supervised techniques. The non-supervised techniques include those based on node proximity and attribute similarity between candidate trustors and trustees [6, 12], and those based on propagating trust weights [4] (e.g., Moletrust [8] and Tidaltrust [3]). The supervised techniques require training data for constructing prediction models [7, 10]. Features are first defined for representing candidate trustor-trustee pairs before the prediction models are trained using labeled trustor-trustee pairs. Finally, the learnt prediction models are required to classify unlabeled user pairs.

Reciprocal trust prediction is different in that it requires a user to first initiate trust link to another user whereas general trust prediction predicts trust between two users without such an initiating link condition. The latter is obviously a harder problem. Nevertheless, solving the reciprocal trust prediction can potentially help to improve the accuracy of general trust prediction.

3 Reciprocity in Directed Networks

3.1 Notations Before we discuss further in trust reciprocity, we first list the notations to be used in our subsequent discussion. They are:

- $\mathbf{U}[z]$: a set of users involved in some trust link(s) up to time z
- $\mathbf{U}_{i*}[z]$: Set of users to whom u_i creates trust links up to time z
- $\mathbf{U}_{*i}[z]$: Set of users who create trust links to u_i up to the time z
- $\mathbf{U}_i[z] (= \mathbf{U}_{i*}[z] \cup \mathbf{U}_{*i}[z])$: Set of users having trust links with u_i up to the time z
- $\mathbf{U}_{i*}^{\mathcal{I}}[z]$: Set of users to whom u_i initiates trust links up to time z . ($\mathbf{U}_{i*}^{\mathcal{I}}[z] \subseteq \mathbf{U}_{i*}[z]$)
- $\mathbf{U}_{*i}^{\mathcal{I}}[z]$: Set of users who initiate trust links to u_i up to time z . ($\mathbf{U}_{*i}^{\mathcal{I}}[z] \subseteq \mathbf{U}_{*i}[z]$)
- z_{ij} : The time stamp when u_i creates a trust link to u_j

To keep discussion simple without causing any confusion, z may be omitted from all the notations (except z_{ij}).

3.2 Global Reciprocity Our research starts by examining the extent of reciprocity in a trust network. A trust network is essentially a directed network with users as nodes and trust from a trustor u_i to a trustee u_j represented by an edge (u_i, u_j) . In [11], the link reciprocity r_g of a directed graph is defined by:

$$(3.1) \quad r_g = \frac{|\mathbf{T}^{\leftrightarrow}|}{|\mathbf{T}|}$$

where $\mathbf{T}^{\leftrightarrow}$ and \mathbf{T} refer to the set of reciprocity links and the set of all links respectively. The subscript g in the notation distinguishes the above measure from the user reciprocity measure. When using trust links up to time z to derive r_g , we denote it by $r_g[z]$.

Figure 1 depicts the r_g at daily interval accumulating the trust links in our Epinions data from $z = 0$ to 938. Throughout the entire time period, r_g varies between a narrow range between [0.335 and 0.35] suggesting that the global reciprocity remains rather stable throughout. Incidentally, this global reciprocity value of Epinions trust network is similar to that of Wikipedia which consists of articles linking to one another [13].

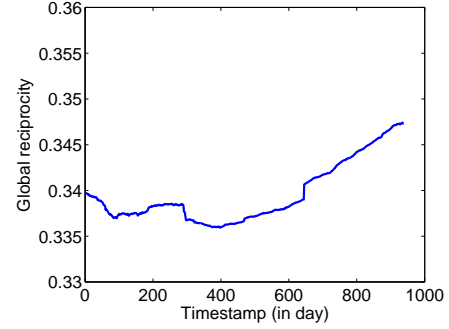


Figure 1: Global reciprocity r_g from $z = 0$ to 938

3.3 User Reciprocity The global link reciprocity r_g is a measure for the entire graph and does not allow us to identify the users involved in reciprocity. Such users are the ones with high reciprocity in their neighborhoods. We therefore define a new user reciprocity measures r_{u_i} for user u_i as follows:

$$(3.2) \quad r_{u_i} = \frac{|\mathbf{U}_{i*} \cap \mathbf{U}_{*i}|}{|\mathbf{U}_i|}$$

The r_{u_i} measure, with a value in $[0,1]$, denotes the ratio of the number of bi-directional neighbors to the total number of neighbors of user u_i . For users with $|\mathbf{U}_i| = 0$, we assign 0 to r_{u_i} .

Figure 2 depicts the average, median and 75 percentile of $\{r_{u_i}\}$'s at daily interval accumulating the trust links in the dataset from $z = 0$ to 938. Note that the maximum r_{u_i} among the Epinions users is 1 throughout the period and we do not show them in the figure. This suggests that there exists a very small proportion of users enjoying very high reciprocity with their neighbors. We observe that the average and median user reciprocity change very little. Due to many users having zero r_{u_i} values, the lower 50 percentile values are zeros. The 75 percentile curve shows that the higher user reciprocity values converge more towards average over time. This means that users from 50 to 70 percentiles have less than 20% neighbors linking back.

4 Reciprocity Related User Behavior Models

In this section, we design quantitative models for user behaviors related to trust reciprocity. The reciprocity measures in Section 3 allow us to measure the amount of reciprocity links in a given trust network falling short of giving an explanation of how the reciprocity is formed. This part of the research thus aims to explain reciprocity by investigating the following user behaviors.

- **Trust initiating (I) behavior:** This behavior refers to how active a user is engaged in initiating

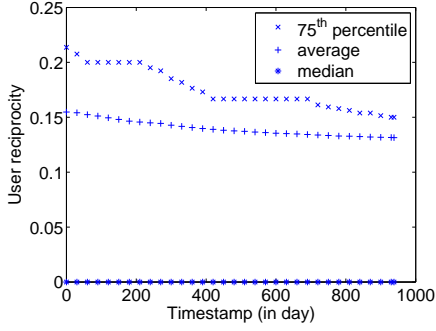


Figure 2: User reciprocity r_{u_i} 's from $z = 0$ to 938

trust links with other users. By initiating many trust links, one may maximize the growth of her trust links and the number of her trustees returning trust.

- **Trust successful seeking (S) behavior:** This behavior refers to how successful or easy a user gets returning trust links from users she earlier *initiates* trust links with. Successful seeking behavior, unlike initiating behavior, is concerned with the likelihood of returning trust. Hence, a high trust initiating user may not be successful in seeking returning trust. On the other hand, a highly successful user in seeking returning trust may not initiate many trust.
- **Trust attracting (A) behavior:** This behavior refers to how good a user is at getting trust links *initiated* from others. The trust attracting behavior is very much the opposite of trust initiating behavior. For one reason or another (e.g., reputation, review writing with strong opinions), some users may tend to attract trust links initiated by others.
- **Trust returning (R) behavior:** This behavior refers to the tendency of a user returning trust to others. Similar to initiating behavior, trust returning behavior can allow a user to grow trust links quickly. It is however different from initiating behavior by considering only return trust links. A high trust returning user may not have high trust attracting behavior. A high trust attracting user also may not return many trust links.

The above behaviors are different from global and user reciprocity measures as these behaviors divide trust links into initiating and returning links. This is only possible by exploiting the timestamps (ie., z_{ij} 's) of trust links. We now propose different quantitative models for the above behaviors based on different underlying principles.

4.1 Basic Behavior Models A straightforward approach to model I , S , A and R behaviors is to consider the sets of initiating and returning trust links. The set notations used have been defined in Section 3.1. A trust link from u_i to u_j is said to be *initiating* if no prior trust link exist between them. A trust link u_j to u_i is said to be *returning* if the trust link u_i to u_j exists beforehand. Given a time point z , the basic reciprocity related behaviors of user u_i are defined as follows:

- **Initiating:** Proportion of u_i initiated trust links among trust links from u_i .

$$(4.3) \quad I_i[z] = \frac{|\mathbf{U}_{i*}^I[z]|}{|\mathbf{U}_{i*}[z]|}$$

- **Attracting:** Proportion of incoming trust links to u_i that are initiated by others.

$$(4.4) \quad A_i[z] = \frac{|\mathbf{U}_{*i}^I[z]|}{|\mathbf{U}_{*i}[z]|}$$

- **Successful seeking:** Proportion of u_i initiated trust links that are returned.

$$(4.5) \quad S_i[z] = \frac{|\mathbf{U}_{i*}^I[z] \cap \mathbf{U}_{*i}[z]|}{|\mathbf{U}_{i*}^I[z]|}$$

- **Returning:** Proportion of trust links to u_i initiated by others that are returned by u_i .

$$(4.6) \quad R_i[z] = \frac{|\mathbf{U}_{*i}^I[z] \cap \mathbf{U}_{i*}[z]|}{|\mathbf{U}_{*i}^I[z]|}$$

4.2 Delay-Aware Models The delay-aware models, unlike the basic ones, consider a non-initiating trust link from u_i to u_j to be returning depending on how soon the link is created after the trust link from u_j to u_i . A decay function is defined to assign a **returning weight** w_{ij} to the link from u_i to u_j as shown in Equation 4.7. Every initiating trust link is assigned a zero returning weight. A non-initiating trust link is assigned high returning weight when it is created soon after a corresponding initiating trust link. The weight reduces as the time delay between initiating and returning links enlarges. This weighting scheme is designed to treat a long overdue returning link as an initiating link.

We define a returning weight w_{ij} in $[0,1]$ for a trust link from u_i to u_j as follows:

$$(4.7) \quad w_{ij}[z] = \begin{cases} \frac{1}{1+\alpha \cdot (z_{ij} - z_{ji})}, & \text{if } u_i \in \{\mathbf{U}_{j*}^I[z] \cap \mathbf{U}_{*j}[z]\}; \\ 0, & \text{otherwise.} \end{cases}$$

where α is in the form $\frac{1}{m}$ and $w_{ij} = \frac{1}{2}$ when z_{ij} is m days later than z_{ji} . A small m will therefore increase the decaying rate. In our experiments, we have used $\alpha = 0.1$ (or $m = 10$ days).

The delay-aware user behavior models are thus defined as follows¹:

- **Initiating:** This is defined by the average of the inverse of returning weights of trust links from u_i among all trust links from u_i . The inverse of returning weight of a trust link tells how likely the trust link is initiated from u_i .

$$(4.8) \quad I_i^d[z] = \text{Avg}_{u_j \in \mathbf{U}_{i*}[z]} (1 - w_{ij}[z])$$

- **Attracting:** This is defined by the average of the inverse of returning weights of trust links to u_i among all trust links to u_i .

$$(4.9) \quad A_i^d[z] = \text{Avg}_{u_j \in \mathbf{U}_{*i}[z]} (1 - w_{ji}[z])$$

- **Successful seeking:** This is defined by the average of returning weights of trust links to u_i from users with whom u_i has initiated trust links.

$$(4.10) \quad S_i^d[z] = \text{Avg}_{u_j \in \mathbf{U}_{i*}^T[z]} w_{ji}[z]$$

- **Returning:** This is defined by the average of returning weights of trust links from u_i to users who have initiated trust links with u_i .

$$(4.11) \quad R_i^d[z] = \text{Avg}_{u_j \in \mathbf{U}_{*i}^T[z]} w_{ij}[z]$$

4.3 Reputation-Aware Models So far, we do not distinguish users creating trust links seriously versus other users creating trusts frivolously. Reputation-aware model thus scrutinizes the reputation of users so as to determine trust initiating, attracting, successful seeking and returning behaviors that involve non-serious trust links. For example, a user is considered active in initiating trust links if most of the links involve trustees who are perceived to be less reputable than him.

The **reputation of a user u_j perceived by user u_i** at time z is denoted by $t_{ij}[z]$. We determine the absolute reputation of a user u_j by $|\mathbf{U}_{*j}|$, the number of incoming trust links. The perceived reputation $t_{ij}[z]$ is thus determined by the positive difference in reputation $\Delta_{ij}[z] = |\mathbf{U}_{*j}[z]| - |\mathbf{U}_{*i}[z]|$. To keep this value in the

range $[0, 1]$ we define the normalized reputation of u_j perceived by u_i at time z as follows:

$$(4.12) \quad t_{ij}[z] = \frac{\Delta_{ij}[z] - \min_{i'j'}\{\Delta_{i'j'}[z]\}}{\max_{i'j'}\{\Delta_{i'j'}[z]\} - \min_{i'j'}\{\Delta_{i'j'}[z]\}}$$

The reputation-aware user behavior model thus extends the basic model as follows²:

- **Initiating:** This is defined by the weighted proportion of u_i initiated trust links among trust links from u_i with weights determined by the inverse perceived reputation of the trustees by u_i at the times the links were initiated by u_i .

$$(4.13) \quad I_i^r[z] = \frac{1}{|\mathbf{U}_{i*}[z]|} \cdot \sum_{u_j \in \mathbf{U}_{i*}^T[z]} (1 - t_{ij}[z_{ij}])$$

- **Attracting:** This is defined by the weighted proportion of u_i incoming trust links initiated by other trustors among all the incoming links to u_i . The weights are the inverse perceived reputations of u_i by the trustors at the times the incoming links were created.

$$(4.14) \quad A_i^r[z] = \frac{1}{|\mathbf{U}_{*i}[z]|} \cdot \sum_{u_j \in \mathbf{U}_{*i}^T[z]} (1 - t_{ji}[z_{ji}])$$

- **Successful seeking:** This is defined by the weighted proportion of returning trust links to u_i among the trust links initiated by u_i . The weights are the inverse perceived reputations of u_i by the users returning the trust links at the times the returning links were created.

$$(4.15) \quad S_i^r[z] = \frac{1}{|\mathbf{U}_{i*}^T[z]|} \cdot \sum_{u_j \in \mathbf{U}_{i*}^T[z] \cap \mathbf{U}_{*i}[z]} (1 - t_{ji}[z_{ji}])$$

- **Returning:** This is defined by the weighted proportion of the returning trust links from u_i among the trust links initiated by other users. The weights are the inverse perceived reputations of these other users by u_i at the time the returning trust links were created.

$$(4.16) \quad R_i^r[z] = \frac{1}{|\mathbf{U}_{*i}^T[z]|} \cdot \sum_{u_j \in \mathbf{U}_{*i}^T[z] \cap \mathbf{U}_{i*}[z]} (1 - t_{ij}[z_{ij}])$$

¹The superscript d used for the behavior notations denotes delay-aware model.

²The superscript r used for the behavior notations denotes reputation-aware model.

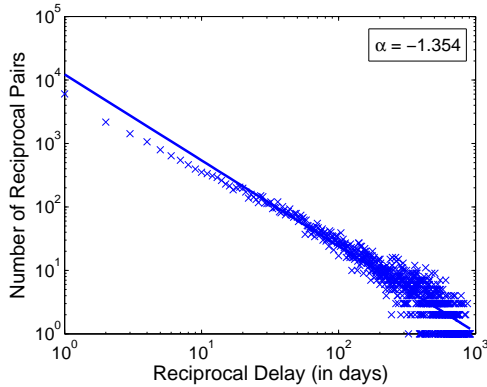


Figure 3: Reciprocal Delay Distribution

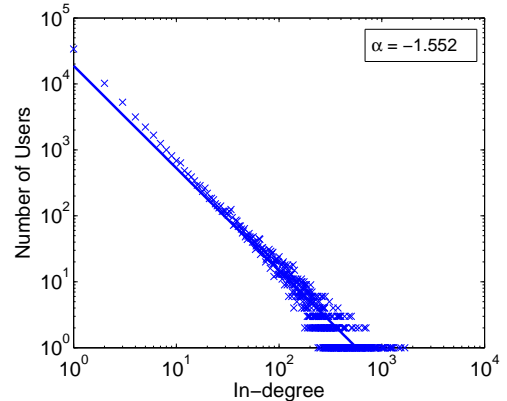


Figure 4: Indegree Distribution

5 Empirical Results of Reciprocity Related Behaviors

In this section, we show the distribution of reciprocal delay of users and difference in indegree of trust user pairs in the Epinions dataset. They are used in the Delay-Aware and Reputation-Aware Behavior Models. We also give an empirical comparison of our proposed behavior models.

5.1 Reciprocal Delay Figure 3 shows the distribution of delay for user pairs with reciprocal trust links (34,910 of them) which follows power law. Among the pairs, 9445 have zero delay, i.e., they form bi-directional links on the same day. The maximum, 75 percentile, and median delays are 913, 23, and 2 days respectively. This shows that most delays in returning links are small. There are however some returning links that have large delays which may not necessary be treated as responses to the corresponding initiating links. The Delay-Aware Model will see them more as non-returning links.

5.2 Difference in Indegree The perceived reputation of u_j by u_i , t_{ij} , is measured by the indegree difference between u_j and u_i , ($|\mathbf{U}_{*j}| - |\mathbf{U}_{*i}|$). In our dataset, there are 114,222 users with trust links. We first show a distribution of indegree of these users in Figure 4. This distribution follows power law.

Figure 5 shows the indegree difference distribution for initiating trust links, returning trust links and any trust links. Each trust link involves a user pair. The y-axis show the number of trust links in log scale. The distribution shows that there are many users initiating and returning trust links to others having lower reputations (with -ve indegree differences). The number of initiating trust links for different indegree difference represented by cross symbol peaks at zero.

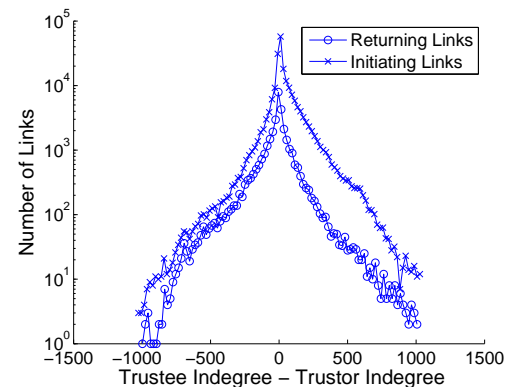
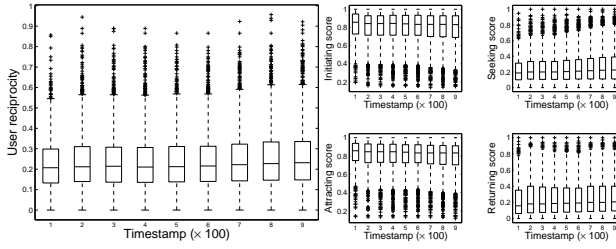


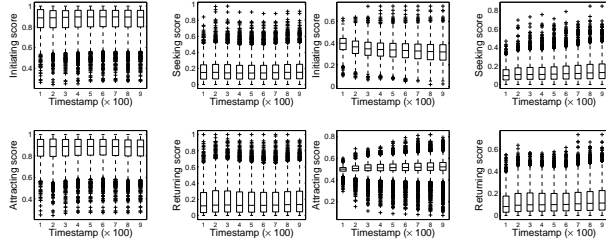
Figure 5: Indegree Difference Distribution

The same can be said for returning links represented by circles. The figure shows that the number of initiating trust links for positive indegree differences is larger than that for negative indegree differences of the same magnitude. On the other hand, the number of returning trust links for positive indegree differences is smaller than that for negative indegree differences of the same magnitude. This is reasonable since the corresponding initiating links are likely to have positive indegree differences. Since there are many more initiating links than returning ones, the combined distribution of both initiating and returning links is very similar to that of initiating links.

5.3 Comparison of Behavior Models To examine the distributions of the different trust related behavior values and user reciprocity over time, we show the box-plots of behavior scores from $z = 1$ to 938 in Figure 6. We only include users with at least 5 incoming trust links and 5 outgoing trust links to ensure that the computed behavior values are meaningful.



(a) User Reciprocity (b) Basic User Behaviors



(c) Delay-Aware User Behaviors (d) Reputation-Aware User Behaviors

Figure 6: Box Plots of Behavior Scores from $z = 1$ to 938

The figure shows that the medians, upper and lower quartiles of behavior and user reciprocity values are generally stable throughout consistent with the earlier observation on user reciprocity (see Section 3.3). There are few outlying values that vary across time.

Finally, we examine the correlations between user behavior values and user reciprocity using Pearson Correlation coefficient in the following tables. Only behavior and user reciprocity values at $z = 938$ are used and the minimum 5 incoming and 5 outgoing trust links filter still applies. As shown in these tables, Basic and Delay-Aware models are very strongly correlated across all the 4 behaviors. Reputation-Aware model is strongly correlated with Basic and Delay-Aware models in Initiating, Successful Seeking and Returning behaviors but not Attracting behavior. We notice that attracting scores have been affected much by uneven distribution of perceived reputation of incoming links among different users. This results in most attracting values of Reputation-Aware model concentrate in a narrow band and are less consistent with those generated by other behavior models.

The user reciprocity values are not strongly correlated with most of our proposed behavior models for most trust reciprocity related behaviors except the returning behavior. This suggests that users with high reciprocity are more likely to depend on their good returning behavior than successful seeking behavior to achieve good reciprocity with neighbors.

Table 2: Initiating scores

| | User Rec. | Basic | Delay-Aware | Rep.-Aware |
|-------------|-----------|---------|-------------|------------|
| User Rec | 1.0000 | -0.6108 | -0.6428 | -0.6138 |
| Basic | - | 1.0000 | 0.9014 | 0.9812 |
| Delay-Aware | - | - | 1.0000 | 0.8958 |
| Rep.Aware | - | - | - | 1.0000 |

Table 3: Seeking scores

| | User Rec. | Basic | Delay-Aware | Rep.-Aware |
|-------------|-----------|--------|-------------|------------|
| User Rec | 1.0000 | 0.6361 | 0.6236 | 0.6471 |
| Basic | - | 1.0000 | 0.9427 | 0.9962 |
| Delay-Aware | - | - | 1.0000 | 0.9386 |
| Rep.-Aware | - | - | - | 1.0000 |

6 Reciprocal Trust Prediction Problem

6.1 Problem Statement To study how the known reciprocity user behaviors affect trust link creation, we now explore the use of these behaviors in the **reciprocal trust prediction problem**. In this problem, we predict if a returning trust link from u_j to u_i will be created at $z_{ji} \geq T$ for an initiating link with $z_{ij} = T$. Reciprocal trust prediction is useful when one is interested to know if an initiating trust link will be returned in the future. This therefore helps one to select the users to trust.

Reciprocal trust prediction is a variant of link prediction problem that has not been studied before. In the following, we describe several solution methods that are broadly classified into *ranking* and *classification* approaches. The former uses reciprocity related user behaviors directly to score the likelihood of returning trust links. The latter uses all the reciprocity related user behaviors as features in training a classifier for predicting returning trust links.

6.2 Ranking Approach The ranking approach is non-supervised and requires a scoring function that derives a score for a returning link given each initiating trust link. Suppose u_i has already initiated a trust to u_j . We want to predict if u_j returns another trust to u_i . Among the four different user behaviors (i.e., initiating (I), attracting (A), successful seeking (S) and returning (R)), the combinations that involve u_i 's successful seeking and u_j 's returning are most relevant. We expect u_i who is high in successful seeking can solicit returning trust more easily than those with low successful seeking

Table 4: Attracting scores

| | User Rec. | Basic | Delay-Aware | Rep.-Aware |
|-------------|-----------|---------|-------------|------------|
| User Rec | 1.0000 | -0.4673 | -0.4194 | 0.2549 |
| Basic | - | 1.0000 | 0.9490 | -0.3681 |
| Delay-Aware | - | - | 1.0000 | -0.3438 |
| Rep.-Aware | - | - | - | 1.0000 |

Table 5: Returning scores

| | User Rec. | Basic | Delay-Aware | Rep.-Aware |
|-------------|-----------|--------|-------------|------------|
| User Rec | 1.0000 | 0.8213 | 0.7624 | 0.8190 |
| Basic | - | 1.0000 | 0.9491 | 0.9986 |
| Delay-Aware | - | - | 1.0000 | 0.9475 |
| Rep.Aware | - | - | - | 1.0000 |

behavior. Similarly, u_j with high returning behaviors is also more likely to return trust links. Initiating and attracting behaviors, on the other hand, have no direct relevance in reciprocal trust prediction.

In this paper, we therefore confine ourselves to the three combinations of successful seeking and returning behaviors derived by the basic, delay-aware and reputation-aware models, namely BASICRANK, DELAYRANK and REPUTATIONRANK. Each method assigns a score $score(i, j)$ representing the likelihood of a returning trust link given an initiating link from u_i to u_j as follows:

- BASICRANK Method: $score^s(i, j) = S_i[z_{ij}] \cdot R_j[z_{ij}]$
- DELAYRANK Method: $score^d(i, j) = S_i^d[z_{ij}] \cdot R_j^d[z_{ij}]$
- REPUTATIONRANK Method: $score^r(i, j) = S_i^r[z_{ij}] \cdot R_j^r[z_{ij}]$

Based on user reciprocity, we also propose a ranking method based on the assumption that two users with neighborhood of high reciprocity are more likely to return a trust link.

- **UserReciprocityRank** Method: $score^u(i, j) = r_{u_i}[z_{ij}] \cdot r_{u_j}[z_{ij}]$

Once a score is assigned to each initiating trust link, we can rank all initiating trust links by score and predict the top ones as the ones with highest likelihood to have returning trust links. The criteria used to select top scores can be based on some pre-defined threshold or expected fraction of returning links among initiating links.

6.3 Classification Approach The classification approach is supervised and it trains a classifier to assign a returning or non-returning label to an initiating trust link. We use Support Vector Machine (SVM) classifiers in this study. We represent each initiating trust link by a vector of feature values. The features involved can be grouped into (a) topological features; (b) user reciprocity features; and (c) reciprocity related user behavior features. These features are shown in Table 6.

The topological features are a set of coefficients defined using different types of common neighbors between

trustor u_i and trustee u_j of an initiating trust link and the coefficients include common neighbor count, Jaccard coefficient, Adar/Adamic coefficient, and Preferential Attachment coefficient. For a user u_i , we define $\Gamma^{in}(u_i)$ and $\Gamma^{out}(u_i)$ as the set of trustors and set of trustees of u_i respectively. We also define $\Gamma(u_i) = \Gamma^{in}(u_i) \cup \Gamma^{out}(u_i)$ as the set of trustors and trustees of u_i .

The user reciprocity features include the user reciprocities of the trustor and trustee of the initiating link, and the USERRECIPROCITYRANK score. The reciprocity related behavior features include initiating and successful seeking behavior values of u_i , attracting and returning behavior values of u_j , as well as the different scores of BASICRANK, DELAYRANK and REPUTATIONRANK.

7 Experiments and Results

7.1 Experiment Setup We would like to evaluate the performances of different ranking and classification methods using our proposed behavior models on predicting whether there exists a returning link given an initiating link. From time point 1 to 938, there are 171,779 initiating links and 19,852 returning links for these initiating links. In general, the network grows over time with new nodes as well as edges added. Thus, it is not reasonable to measure the behaviors of a new user who just joins the system and does not leave enough evidence to express his/her behaviors. Hence, we only include an initiating link (u_i, u_j) created at time z in our prediction task if both u_i and u_j have at least d_{min} out-links and d_{min} in-links created during $[1, z]$.

There are two sets of experiments conducted for the reciprocal trust prediction task:

- *Prediction of returning link during $[z, 938]$* : This experiment predicts the returning link formed anytime between the creation of initiating link at time z and last time point in our dataset, i.e., 938. This variant of prediction does not care about how soon after the initiating link the returning link will be created.
- *Prediction of returning link during $[z, z + \Delta z_{max}]$ where Δz_{max}* : In some applications, one may be interested in returning links formed soon after the initiating links. We therefore introduce Δz_{max} as the **delay threshold** to include only returning links created at most Δz_{max} days after the initiating link. This reduces the number of returning links to be predicted making the prediction task more challenging.

In the second set of experiments, we vary Δz_{max} from 5 to 50. For simplicity, we say the first set of experiments

Table 6: Features of Initiating Trust Link from u_i to u_j

| Features | Description |
|---|---|
| Topological Features | |
| Common in-in-neighbors | $ \Gamma^{in}(u_i) \cap \Gamma^{in}(u_j) $ |
| Common out-out-neighbors | $ \Gamma^{out}(u_i) \cap \Gamma^{out}(u_j) $ |
| Common neighbors | $ \Gamma(u_i) \cap \Gamma(u_j) $ |
| Common out-in neighbors | $ \Gamma^{out}(u_i) \cap \Gamma^{in}(u_j) $ |
| Common in-out neighbors | $ \Gamma^{in}(u_i) \cap \Gamma^{out}(u_j) $ |
| JC ^a of in-in-neighbors | $\frac{ \Gamma^{in}(u_i) \cap \Gamma^{in}(u_j) }{ \Gamma^{in}(u_i) \cup \Gamma^{in}(u_j) }$ |
| JC of out-out-neighbors | $\frac{ \Gamma^{out}(u_i) \cap \Gamma^{out}(u_j) }{ \Gamma^{out}(u_i) \cup \Gamma^{out}(u_j) }$ |
| JC of neighbors | $\frac{ \Gamma(u_i) \cap \Gamma(u_j) }{ \Gamma(u_i) \cup \Gamma(u_j) }$ |
| JC of out-in neighbors | $\frac{ \Gamma^{out}(u_i) \cap \Gamma^{in}(u_j) }{ \Gamma^{out}(u_i) \cup \Gamma^{in}(u_j) }$ |
| JC of in-out neighbors | $\frac{ \Gamma^{in}(u_i) \cap \Gamma^{out}(u_j) }{ \Gamma^{in}(u_i) \cup \Gamma^{out}(u_j) }$ |
| AA ^b of in-in-neighbors | $\sum_{u_k \in \Gamma^{in}(u_i) \cap \Gamma^{in}(u_j)} \frac{1}{\log(\Gamma(u_k))}$ |
| AA of out-out-neighbors | $\sum_{u_k \in \Gamma^{out}(u_i) \cap \Gamma^{out}(u_j)} \frac{1}{\log(\Gamma(u_k))}$ |
| AA of neighbors | $\sum_{u_k \in \Gamma(u_i) \cap \Gamma(u_j)} \frac{1}{\log(\Gamma(u_k))}$ |
| AA of out-in neighbors | $\sum_{u_k \in \Gamma^{out}(u_i) \cap \Gamma^{in}(u_j)} \frac{1}{\log(\Gamma(u_k))}$ |
| AA of in-out neighbors | $\sum_{u_k \in \Gamma^{in}(u_i) \cap \Gamma^{out}(u_j)} \frac{1}{\log(\Gamma(u_k))}$ |
| PA ^c of in-in-neighbors | $ \Gamma^{in}(u_i) \cdot \Gamma^{in}(u_j) $ |
| PA of out-out-neighbors | $ \Gamma^{out}(u_i) \cdot \Gamma^{out}(u_j) $ |
| PA of neighbors | $ \Gamma(u_i) \cdot \Gamma(u_j) $ |
| PA of out-int neighbors | $ \Gamma^{out}(u_i) \cdot \Gamma^{in}(u_j) $ |
| PA of in-out neighbors | $ \Gamma^{in}(u_i) \cdot \Gamma^{out}(u_j) $ |
| User Reciprocity Features | |
| User reciprocity of u_i | r_{u_i} |
| User reciprocity of u_j | r_{u_j} |
| USERRECIPROCITYRANK Score | $score^u(i, j)$ |
| Reciprocity Related User Behavior Features | |
| Initiating behaviors of u_i | $I_i/I_i^d/I_i^r$ |
| Attracting behaviors of u_j | $A_j/A_j^d/A_j^r$ |
| Successful seeking behavior of u_i | $S_i/S_i^d/S_i^r$ |
| Returning behaviors of u_j | $R_j/R_j^d/R_j^r$ |
| BASICRANK Score | $score^s(i, j)$ |
| DELAYRANK Score | $score^d(i, j)$ |
| REPUTATIONRANK Score | $score^r(i, j)$ |

^aJaccard Coefficient.^bAdamic/Adar coefficient.^cPreferential attachment coefficient

has $\Delta z_{max} = \infty$.

7.2 Evaluation Measurement To evaluate the performances of different ranking and classification methods for reciprocal trust prediction, we use **F1** and **AUC PRC** (Area under Precision Recall Curve). Suppose there are N initiating links to be predicted, out of which N_r have returning links. We rank N initiating links in descending order in terms of its returning score generated by each ranking/classification method. For SVM based methods, SVM-light [5] is used and we use 5-fold

validation to train and test on initiating links. The precision, recall and F1 measure of these predicted results are identical and defined as:

$$F1 = \frac{\# \text{ returning links among top } N_r \text{ ranked initiating links}}{N_r}$$

In addition, we also use AUC PRC[1] which measures the area under the Precision-Recall curve as a performance metric. An important disadvantage when using F1 is that F1 only examines a subset of top initiating links with highest scores which might be the easiest ones to predict. This disadvantage might lead to bias results when comparing the performances of different methods. In contrast, AUC PRC looks at an algorithm's overall performance to rank all the links. The F1 of a Random predictor is expected to be $\frac{N_r}{N}$.

7.3 Overall Results We want to examine the overall prediction accuracies of our proposed methods with other baseline methods. Our proposed methods to be evaluated include: (a) Ranking based methods: BASICRANK, DELAYRANK, REPUTATIONRANK and USERRECIPROCITYRANK; and (b) Classification based methods: SVMTOPOLOGY, SVMBASIC, SVMDELAYRANK, SVMUSERRECIPROCITY, and SVMALL.

The baseline methods to be compared with are 20 ranking methods each using a topological feature in Table 6. For each of these baseline methods, the feature value is used as $score(i, j)$ for ranking initiating links. For example, the Common in-in-neighbors ranking model will give an initiating link from u_i to u_j a score of $score(i, j) = |\Gamma^{in}(u_i) \cap \Gamma^{in}(u_j)|$ as the likelihood that u_j will return a trust link to u_i at a later time. Several of these baseline methods have been found effective in link prediction research [6]. Finally, we also include a Random prediction method that randomly predicts initiating links to have returning links.

Performance of Baseline Ranking Methods We first examine the performance of our 20 baseline ranking methods using topological features and the random baseline method, for predicting returning links. In addition to $\Delta z_{max} = \infty$, we use $\Delta z_{max} = 5$ so as to predict returning links within 5 days from the time the corresponding initiating links are created. We use $d_{min} = 5$ so that the behavior models compute reciprocity related behaviors based on at least 5 in-links and 5 out-links before one user initiates link to another user. This d_{min} setting confines our prediction task to 57,579 initiating links. The numbers of returning links based on the two Δz_{max} settings are 19,852 and 11,798 respectively.

The baseline results for both $\Delta z_{max} = \infty$ and

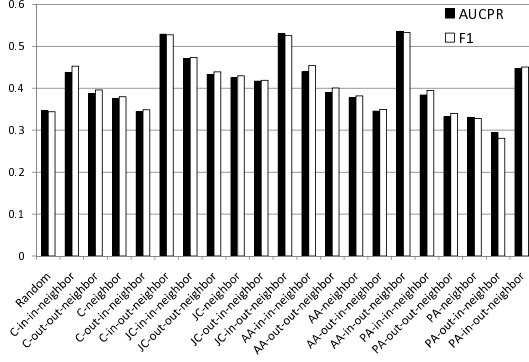


Figure 7: Performance of baseline ranking methods ($d_{min} = 5$ and $\Delta z_{max} = \infty$)

$\Delta z_{max} = 5$ cases are shown in Figures 7 and 8 respectively. For $\Delta z_{max} = \infty$, we observe that Adamic/Adar coefficient (AA) of in-out-neighbors outperforms other baseline methods both in AUC PRC and F1. The AUC PRC and F1 of this topological baseline ranking method are more than 50% improvement over those of random baseline. The performance of common in-out-neighbor and JC in-out-neighbor methods are just slightly lower than AA-in-out-neighbors.

The above results are reasonable since the more users who are both the trustors of user u_i initiating a link and trustees of user u_j receiving the link, the more likely u_j returns a link back to u_i . We also observe that in-out-neighbor type methods generally outperform the in-in-neighbor type methods which in turns outperform the out-out-neighbor and neighbor type methods. The out-in-neighbor methods yield the worst performance. In fact, the methods using preferential attachment (PA) coefficient of out-out-neighbors, neighbors, and out-in-neighbors have performed worse than random.

The results for $\Delta z_{max} = 5$ are similar except that the AUC PRC and F1 values are lower for all methods since the task is harder. AA-in-out-neighbor remains to be the best among the baseline methods. As AA-in-out-neighbor baseline ranking method gives the best performance, we will only include it for subsequent performance evaluation.

Performance of Proposed Ranking and Classification Methods We now examine the performance of our proposed ranking and classification methods for $\Delta z_{max} = \infty$ in Figure 9. We fix $d_{min} = 5$ in these experiments. Among the ranking methods, DELAYRANK yields the best performance and is very closely followed by BASICRANK and REPUTATIONRANK. USERRECIPROCITYRANK, on the other hand, performs only slightly better than the best baseline,

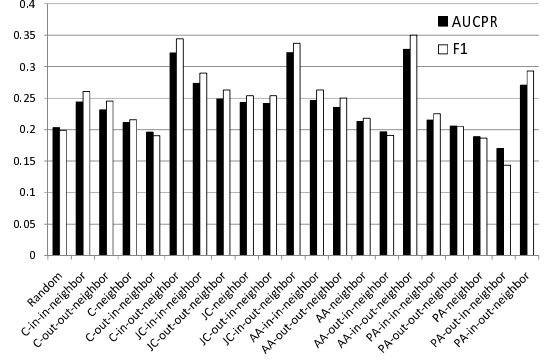


Figure 8: Performance of baseline ranking methods ($d_{min} = 5$ and $\Delta z_{max} = 5$)

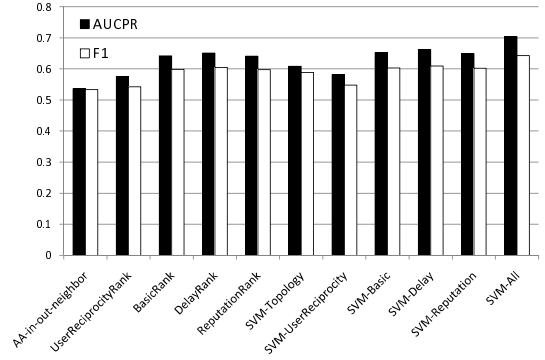


Figure 9: Performance of proposed ranking and classification methods ($d_{min} = 5$ and $\Delta z_{max} = \infty$)

AA-in-out-neighbor. Interestingly, SVM using all topological features performs no better than DELAYRANK, BASICRANK and REPUTATIONRANK which are non-supervised methods. When all features are included, SVM-All outperforms all ranking methods. The above observations are also found in the case of $\Delta z_{max} = 5$ as shown in Figure 10.

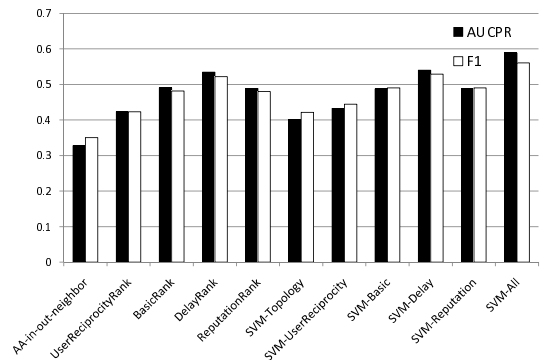


Figure 10: Performance of proposed ranking and classification methods ($d_{min} = 5$ and $\Delta z_{max} = 5$)

Table 7: Number of initiating and returning links by varying d_{min} .

| d_{min} | # initiating links | # return- ing links | % | # return- ing links | % |
|-----------|--------------------|-----------------------------|------|------------------------|------|
| | | $(\Delta z_{max} = \infty)$ | | $(\Delta z_{max} = 5)$ | |
| 0 | 171,779 | 34,910 | 20.3 | 20,962 | 12.2 |
| 5 | 57,579 | 19,852 | 34.5 | 11,798 | 20.5 |
| 10 | 39,617 | 15,100 | 38.1 | 8,967 | 22.6 |
| 15 | 29,593 | 12,042 | 40.7 | 7,175 | 24.2 |
| 20 | 22,664 | 9,817 | 43.3 | 5,859 | 25.9 |
| 25 | 18,032 | 8,033 | 44.5 | 4,802 | 26.6 |
| 30 | 14,461 | 6,589 | 45.6 | 3,981 | 27.5 |
| 35 | 11,646 | 5,457 | 46.9 | 3,336 | 28.7 |
| 40 | 9,585 | 4,570 | 47.7 | 2,804 | 29.3 |
| 45 | 7,894 | 3,805 | 48.2 | 2,360 | 29.9 |
| 50 | 6,410 | 3,134 | 48.9 | 1,958 | 30.6 |

7.4 Results by varying d_{min} We now focus on evaluating performances for different d_{min} values, i.e., $d_{min} \in \{5, 10, \dots, 50\}$ for both $\Delta z_{max} = \infty$ and $\Delta z_{max} = 5$. The number of initiating and returning links with different d_{min} values are shown in Table 7. As shown, the larger the value of d_{min} the higher the percentage of returning links, implying that we are more likely to find returning links among more serious and experienced users (given their higher number of in- and out-links).

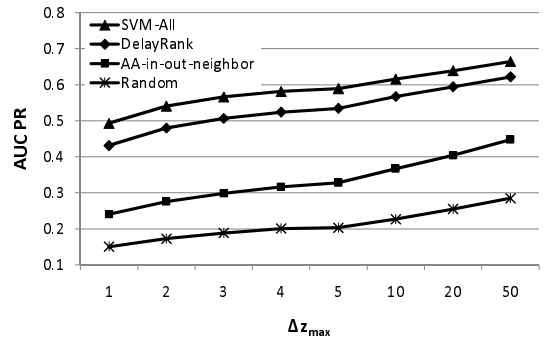
Instead of showing the performances of all methods, we only include the representative methods from baseline ranking, our proposed reciprocity behavior based ranking, and classification approaches. They are AA-in-out-neighbor, DELAYRANK and SVM-All. Figures 11 and 12 show the performances of these 3 methods for different d_{min} 's for $\Delta z_{max} = \infty$ and 5. From the figures, we observe that the performances of the 3 methods are not very much affected by d_{min} . The relative order by performance of SVM-All, DELAYRANK and AA-in-out-neighbor remains unchanged.

7.5 Performances with different Δz_{max} So far, we have tried $\Delta z_{max} = 5$ other than $\Delta z_{max} = \infty$. We thus evaluate the performance of the three representative methods for $\Delta z_{max} \in \{1, 2, 3, 4, 5, 10, 20, 50\}$ while fixing $d_{min} = 5$. With different Δz_{max} values, we have different numbers of returning links as shown in Table 8.

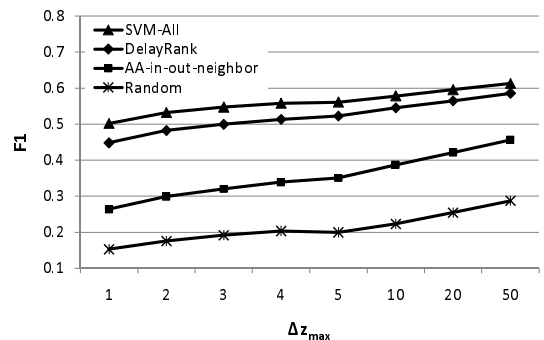
As shown in Figures 13(a) and (b), AA-in-out-neighbor, DELAYRANK and SVMALL improve in AUC PRC and F1 slightly as Δz_{max} increases. In practical applications, we know that Δz_{max} is expected to be small, say < 10 . In this range, both SVMALL and DELAYRANK outperform AA-in-out-neighbor by wide margin.

Table 8: Number of returning links for different Δz_{max} 's ($d_{min} = 5$).

| Δz_{max} | # returning links | % returning links |
|------------------|-------------------|-------------------|
| 1 | 8,716 | 43.9 |
| 2 | 9,960 | 50.2 |
| 3 | 10,761 | 54.2 |
| 4 | 11,383 | 57.3 |
| 5 | 11,798 | 59.4 |
| 10 | 13,150 | 66.2 |
| 20 | 14,602 | 73.6 |
| 50 | 16,395 | 82.6 |
| ∞ | 19,852 | 100 |



(a) AUC PRC



(b) F1

Figure 13: Performances with different Δz_{max} 's ($d_{min} = 5$)

8 Conclusion

This paper studies the trust reciprocity behaviors within an online review user community, and deploys them in solving reciprocal trust prediction problem. Built upon the existing work on network level link reciprocity measures, we develop node level user reciprocity measure and a number of new measures for trust reciprocity related behaviors, namely trust initiating, returning, successful seeking, and attracting. We make observations about these behavior measures in an Epinions dataset. These measures have been further used in sev-

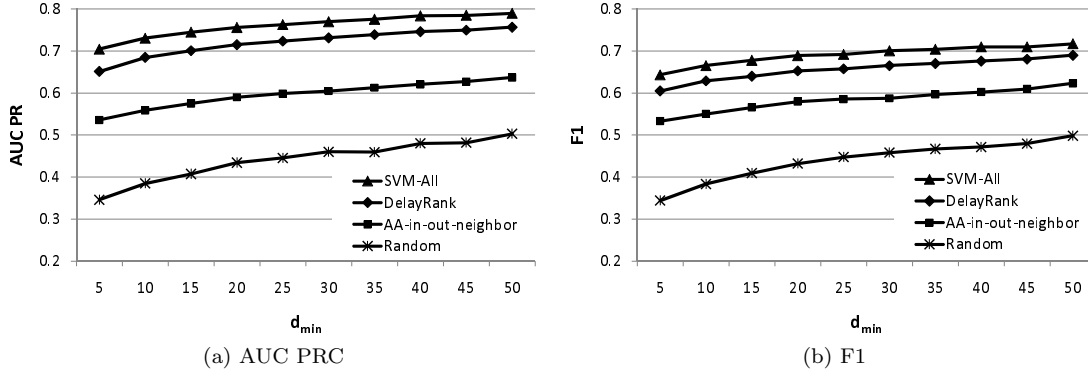


Figure 11: Performances with different values of d_{min} ($\Delta z_{max} = \infty$)

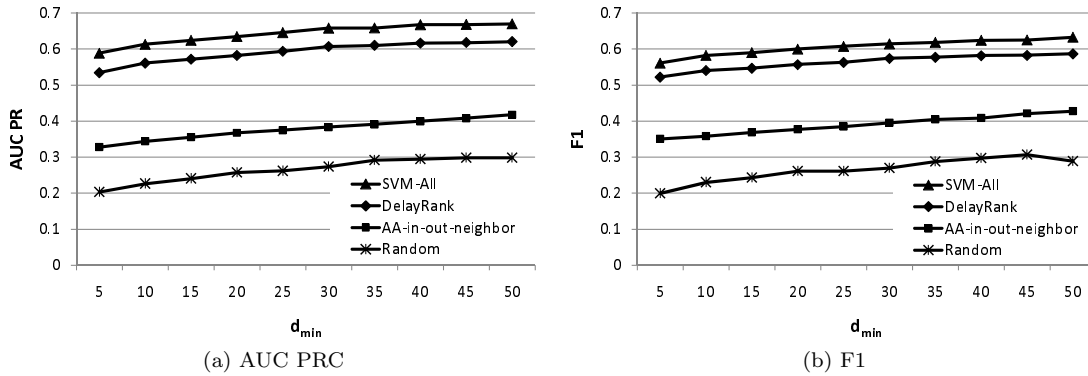


Figure 12: Performances with different values of d_{min} ($\Delta z_{max} = 5$)

eral ranking (non-supervised) and classification (supervised) methods to address the reciprocal trust link prediction problem effectively. This paper represents an early data mining work on trust reciprocity, a relatively novel research topic. There are several interesting directions for the future work. In particular, link reciprocity exists in many different network data. It is useful to evaluate reciprocity related behaviors and reciprocal link formation in other large networks that record link creation timestamps. To create network models that resemble real life networks, one may incorporate reciprocity related behaviors of users and the effect of reciprocal links.

References

- [1] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML*, 2006.
- [2] D. Garlaschelli and M.I. Loffredo. Patterns of link reciprocity in directed networks. *Physical Review Letters*, 2004.
- [3] J. Golbeck. Generating predictive movie recommendations from trust in social networks. In *iTrust*, 2006.
- [4] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *WWW*, 2004.
- [5] T. Joachims. *Making Large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning*. MIT-Press, 1999.
- [6] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.
- [7] H. Liu, E.-P. Lim, H. W. Lauw, M.-T. Le, A. Sun, J. Srivastava, and Y. A. Kim. Predicting trusts among users of online communities: an opinions case study. In *ACM EC*, 2008.
- [8] P. Massa and P. Avesani. Controversial users demand local trust metrics: An experimental study on opinions.com community. In *AAAI*, 2005.
- [9] P. Massa and P. Avesani. Trust-aware recommender systems. In *RecSys*, pages 17–24, 2007.
- [10] Y. Matsuo and H. Yamamoto. Community gravity: measuring bidirectional effects by trust and rating on online social networks. In *WWW*, 2009.
- [11] S. Wasserman and K. Faust. *Social Network Analysis*. Cambridge University Press, 1994.
- [12] C.-N. Ziegler and J. Golbeck. Investigating interactions of trust and interest similarity. *Decision Support Systems*, 43, 2007.
- [13] V. Zlatić, M. Božicević, H. Stefancić, and M. Domazet. Wikipedia: Collaborative web-based encyclopedias as a complex networks. *Physical Review E*, 74, 2006.