StereoTrust: A Group Based Personalized Trust Model

Xin Liu
NTU, Singapore
liu_xin@pmail.ntu.edu.sg

Anwitaman Datta
NTU, Singapore
anwitaman@ntu.edu.sg

Krzysztof Rzadca
NTU, Singapore
krz@ntu.edu.sg

Ee-Peng Lim
SMU, Singapore
eplim@smu.edu.sg

ABSTRACT

Trust plays important roles in diverse decentralized environments, including our society at large. Computational trust models help to, for instance, guide users' judgements in online auction sites about other users; or determine quality of contributions in web 2.0 sites. Most of the existing trust models, however, require historical information about past behavior of a specific agent being evaluated – information that is not always available. In contrast, in real life interactions among users, in order to make the first guess about the trustworthiness of a stranger, we commonly use our "instinct" – essentially stereotypes developed from our past interactions with "similar" people.

We propose *StereoTrust*, a computational trust model inspired by real life stereotypes. A user forms stereotypes using her previous transactions with other agents. A stereotype contains certain features of agents and an expected outcome of the transaction. These features can be taken from agents' profile information, or agents' observed behavior in the system. When facing a stranger, the stereotypes matching stranger's profile are aggregated to derive his expected trust. Additionally, when some information about stranger's previous transactions is available, StereoTrust uses it to refine the stereotype matching.

According to our experiments, StereoTrust compares favorably with existing trust models that use different kind of information and more complete historical information. Moreover, because evaluation is done according to user's personal stereotypes, the system is completely distributed and the result obtained is personalized.

StereoTrust can be used as a complimentary mechanism to provide the initial trust value for a stranger, especially when there is no trusted, common third parties.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval – Clustering; H.2.0 [**Information Systems**]: General – Security, integrity, and protection

CIKM'09, November 2–6, 2009, Hong Kong, China.

General Terms

Algorithms, Performance, Theory

Keywords

stereotypes, trust model, group, reputation

1. INTRODUCTION

Trust is an important abstraction used in diverse scenarios including e-commence, distributed and peer-to-peer systems, grid systems and dynamic collaborative systems. By the very nature of the large scale and openness of these systems, one is often required to interact with other agents with whom there are few or no shared past interactions. To assess the risk of such interactions and to determine whether an unknown agent is worthy of engagement, these systems usually offer some trust-management mechanisms.

If a user has sufficient direct experience with an agent, the agent's future performance can be reliably predicted [1]. However, in large-scale environments, direct experience is often not sufficient or even non-existent. In this case, prediction is based on user's "indirect experience" – opinions obtained from other agents [11, 16, 2] (also known as target agent's *reputation*). Simple aggregations (like a seller's ranking on eBay) rely on access to global information like the history of the agent's behavior. Alternatively, *transitive trust models* [2, 9] (or web of trust models) build chains of trust relationships between the user and the target agent. The basic idea is that if A trusts B and B trusts C, then A can derive C's trust using B's referral on C and A's trust in B. In a distributed system, such chains are not trivial to discover. Moreover, they suffer inaccurate reports and "weakest link" [6].

All the mentioned approaches aggregate the same *kind* of information – agents' impressions about the transactions. At the same time, most of the systems provide a vast context for each transaction, including transaction's type, category, or participant's profile. We were curious to see how accurately we could predict trust using only (or – also) the context. Thus, we did not want to propose a better mechanism using the same information, but rather a complementing alternative when information used by existing models is unavailable.

The result of our work is StereoTrust, a trust model that estimates a target agent's trust using stereotypes learned from interactions with other agents. Our work is inspired by [14, 15], which studies the relation between the reputation of a company and its employees: The company's reputation can be modeled as an aggregate of employees' reputations and it can suggest a prior estimate for employee's reputation. In StereoTrust, users form stereotypes by aggregating information from their interaction partners' profile pages, or the context of the transaction. Example stereotypes are "agents selling mobile phones are less honest than others" or "agents living in small towns are more honest". To build stereotypes, a user has to group other agents ("agents selling mobile phones" or "agents living in small towns"). These groups do not have to be mutually exclusive. Then, when facing a new agent, the user estimates the agent's trust using stereotypes on groups to which the new agent belongs ("does she sell mobile phones?", "does she live in a small town?").

StereoTrust has its own weaknesses and limitations. For instance, not in all circumstances a user can determine the profile of an agent. Similarly, a new user cannot form a decent local stereotype, as she does not have enough prior interactions.

However, we think that StereoTrust is interesting both academically and in practice. Academically, as it emulates a real world human behavior by using stereotypes for a first guess about a stranger. In practice, as, firstly, StereoTrust's predictions are personalized; and, secondly, StereoTrust works when global information (required by other models) is not available, inaccurate, or tampered. But when global information is available, StereoTrust can still be used as a mechanism to enhance the prediction (thus a dual context of the word "stereo"). Our experiments show that such an augmentation, called d-StereoTrust, significantly improves the accuracy.

Notice that StereoTrust is generic, and its use of abstract group definitions allow it to be used in very different kinds of applications. Also, the notion of trust itself can be easily adapted to different contexts. In this paper, we adopt the definition from [7]: "*Trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor or enforce it) and in a context in which it affects his own action*".

For an example application, consider judging quality of product reviews from a web site (such as Epinions.com). In such community, users write reviews for products, structured into different categories (e.g., books, cars, music). These reviews are later ranked by other users. Normally, each reviewer has some categories in which she is an "expert" (like jazz albums for a jazz fan). The reviewer is more likely to provide high quality reviews for products in these familiar categories. Of course, users may also write reviews for products from other categories, but their quality might be not so high, because of, e.g., insufficient background knowledge. "Mastery" can be correlated between categories. For instance, audiophiles (people who use top-end music equipment) usually know how to appreciate music, and thus, if they review a jazz album, the review is more likely to be in-depth. The correlation might be also negative, as we do not expect an insightful review of a jazz album from, e.g., a game boy reviewer. When facing an unrated review of a jazz album by an unknown contributor, we can use the information on contributor's past categories (game-boy fan or an audiophile?) and our stereotypes ("noisy" gamers vs. insightful audiophiles) to estimate the quality of the review. In fact, evaluation of our method on Epinions.com dataset (Section 3) indicates that considering reviewer's interests provides a good estimation of the quality of the review.

Consider a very different kind of application, for example, that of a peer-to-peer storage system. If a peer wants to store a new block of data at some peers, it would need to choose a suitable peer to do so. The suitability of a peer may depend on the likelihood that the peer will be available when the data needs to be retrieved (which may depend on its geographic location/time-zone difference), the response time to access the data (which may depend on agreements and infrastructure between ISPs, but the user would need to perceive it by measurements), and so on. Conventional systems design approach models such a scenario as a multi-criterion optimization problem. Such an approach would typically need knowledge about the specific peer in question - for instance, what time does this peer come online and go offline, what is the end to end latency and available bandwidth (which in practice varies over time, and is hard to measure and anticipate), etc. Applying StereoTrust can provide an alternative systems design[1], where a peer in, say Tokyo, can think - my past experiences tell me that peers in Beijing and Hong Kong have more common online time with me compared with peers in London and New York. Likewise, peers in New York and Hong Kong with specific IP prefix provide reliable and fast connections, while the others don't. Based on such information, the peer would be able to make a first guess that a peer in Hong Kong is likely to be its best bet, if it has to choose between a peer in Hong Kong and London, without needing to know the history of the specific peer in question. Of-course, a second-order optimization in due course by studying the history of specific peers will be necessary to optimize such a system (as well as refine StereoTrust's accuracy).

2. DESCRIPTION OF GROUP-BASED TRUST MODEL

We refer to a participant in the system as an agent. We denote by $\mathcal{A}$ the set of all agents in the system; and by $\mathcal{A}_x$ the set of agents known to agent a_x. An agent can provide services for other agents. A transaction in the system happens when an agent accepts another agent's service. To indicate the quality of a service, an agent can rank the transaction. Θ_{a_x,a_y} denotes the set of transactions between service provision agent a_y and service consumption agent a_x. $\theta_{a_x,a_y} = |\Theta_{a_x,a_y}|$ denotes the number of such transactions.

2.1 Group Definition

In StereoTrust, a group is a community of agents who show some common properties or behave similarly in certain aspects. Because of the common properties shared by all the members of this group, we believe that the group can act as a collective entity to represent its member agents (to a certain extent). For instance, people may consider a programmer working in a well-known software company as skilled, even if they do not personally know the person. People trust the company based on the quality of produced software, thus they also trust the programmers who create the software. On the other hand, a company employing

[1]We are not claiming that it will provide the best possible system design, but merely that it opens the opportunity for alternative designs.

skilled (i.e., trusted) programmers can release high-quality products, and thus gain high reputation. Such interplay between the group's and its members' reputation is the basis of our work. We derive the trust of an agent according to the trusts of its corresponding groups.

Groups are defined subjectively by the agent a_x that uses StereoTrust to derive trust to other agents. A group G_x^i is a set of agents. We denote by $\mathcal{G}_x = \{G_x^1, G_x^2, \ldots, G_x^n\}$ the set of all groups defined by a_x. Based on a_x's previous experience, stereotypes, and any other information, a_x formulates grouping functions $M_x(G_x^i, a) : [\mathcal{G}_x, \mathcal{A}_x] \rightarrow [0,1]$, that, for each group G_x^i, map agent a to the probability that a is the member of this group. Thus, in the most general model, a group is a fuzzy set of agents. If $M_x(G_x^i, a) = 1$, it is certain that a is member of G_x^i ($a \in G_x^i$); if $M_x(G_x^i, a) = 0$, it is certain that a does not belong to G_x^i ($a \notin G_x^i$).

Note that in this paper we do not discuss how to derive the grouping functions $M_x(G_x^i, a)$. However, during our experiments, we propose how to formulate such functions for Epinions.com dataset. For instance, a group can gather all agents responsible for the same type of task; or all agents interested in a certain topic; or all agents living in the same location. Depending on the type of criteria used, groups may overlap (an agent belongs to multiple groups simultaneously) or be disjoined (each agent belongs only to one group).

2.2 Modeling Trust

A computational trust model models the complex notion of trust by a variable (binary, discrete, continuous, etc.). We assume (after [7]) that trust indicates the probability that an agent will perform a particular, expected action during a transaction. Thus, agent's trust rating is a real number from range $[0,1]$, where 0 indicates that the agent is absolutely untrustworthy and 1 indicates that the agent is absolutely trustworthy.

The Beta distribution is commonly used to model uncertainty about probability p of a random event (including agent's reputation [10, 5]). We model a series of transactions between a pair of agents as observations of independent Bernoulli trials. In each trial, the success probability p is modeled by Beta distribution with parameters α and β (we start with $\alpha = \beta = 1$, that translate into complete uncertainty about the distribution of the parameter, modeled by the uniform distribution: $Beta(1,1) = U(0,1)$). After observing s successes in n trials, the posterior density of p is $Beta(\alpha + s, \beta + n - s)$ [4].

The following definition defines trust function between *entities* (an individual agent or a group) based on a beta function. By E_t, we denote the entity participating in the trust calculation.

Definition 1 (Trust Function). *Entity E_1 evaluates entity E_2. From the viewpoint of E_1, S_{E_1,E_2} and U_{E_1,E_2} represent, respectively, the number of successful transactions and unsuccessful transactions between E_1 and E_2 ($S_{E_1,E_2} \geq 0$ and $U_{E_1,E_2} \geq 0$). Trust function $T_{E_1,E_2}(p|S_{E_1,E_2}, U_{E_1,E_2})$ mapping trust rating p ($0 \leq p \leq 1$) to its probability is defined by:*

$$T_{E_1,E_2}(p|S_{E_1,E_2}, U_{E_1,E_2}) = \frac{\Gamma(S_{E_1,E_2} + U_{E_1,E_2} + 2)}{\Gamma(S_{E_1,E_2} + 1)\Gamma(U_{E_1,E_2} + 1)} \cdot \cdot p^{S_{E_1,E_2}}(1-p)^{U_{E_1,E_2}}. \tag{1}$$

The expected value of the trust function is equal to:

$$E_{E_1,E_2}(T_{E_1,E_2}(p|S_{E_1,E_2}, U_{E_1,E_2})) = \frac{(S_{E_1,E_2} + 1)}{(S_{E_1,E_2} + U_{E_1,E_2} + 2)} \tag{2}$$

2.3 StereoTrust Model

StereoTrust model uses only agent's local information to derive another agent's trust.

Consider a scenario where an agent a_x, a service requestor encounters a potential service provider a_y with whom a_x had no prior experience. We assume that a_x can obtain some meta-information about a_y, such as a_y's interests, location, age etc. Such information as well as other information like a_x's previous experience is used by a_x to form groups that help derive a_y's trust.

In the basic model, StereoTrust considers only groups for which the membership is certain. We denote these groups by $\mathcal{G}_{x,y} = \{G_{x,y}^1, G_{x,y}^2, \ldots\}$ such that $M_x(G_{x,y}^i, a_y) = 1$ (for the sake of simplifying the notation, we will use G^i in place of $G_{x,y}^i$ when the context is clear). The trust between a_x and each of these groups G^i is derived based on past interactions with agents that belong to G^i with certainty. Thus, from the set of all agents a_x has previously interacted with ($\mathcal{A}_x = \{a_1, a_2, \ldots\}$), a_x extracts those that belong to G^i (i.e., $G^i = \{a : M_x(G^i, a) = 1\}$). Then, a_x counts the total number of successful S_{a_x,G^i} transactions with G^i by summing up the successful transactions with G^i's members: $S_{a_x,G^i} = \sum_{a \in G^i} S_{a_x,a}$. The total number of unsuccessful transactions U_{a_x,G^i} is computed similarly. Finally, a_x uses Eq. (1) to derive G^i trust function.

To derive agent's a_y trust value, a_x combines her trust towards all the groups $\mathcal{G}_x$ in which a_y is a member. The trust is computed as a weighted sum of groups' trust with weights proportional the fraction of transactions with that group. For group G^i, weight factor $W_{x,y}^i$ is calculated as a number $\theta_{x,y}^i$ of a_x's transactions with $G_{x,y}^i$ members ($\theta_{x,y}^i = |\Theta_{a_x,a}|$ such that $a \in G_{x,y}^i$); divided by the total number of a_x's transactions with members of any $\mathcal{G}_{x,y}$ group. Obviously, the higher the number of transactions regarding one group, the more likely is a_x to interact with agents of this group, so this group contributes more to represent a_y's trust from viewpoint of a_x. We define weight factor $W_{x,y}^i$ for G_i as:

$$W_{x,y}^i = \frac{\theta_{x,y}^i}{\sum_j \theta_{x,y}^j} \tag{3}$$

Using the estimated weights, we combine all group trusts to derive a_y's trust. The process of trust calculation is illustrated in Figure 1.

We propose two approaches to calculate and combine group trusts.

SOF Approach (Sum Of Functions)

In this approach, we first calculate probability density of trust rating for each group using trust function (Eq. (1)) and then combine them to produce a_y's probability density

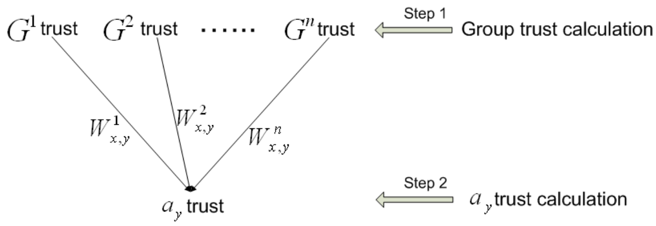

Figure 1: Process of trust calculation. Weighted sum of each group G^i's trust by assigning corresponding weight factor $W^i_{x,y}$

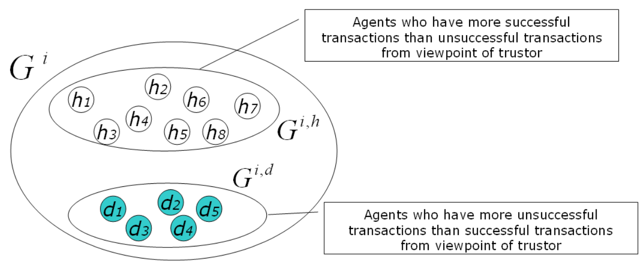

Figure 2: Structure of group G^i in d-StereoTrust

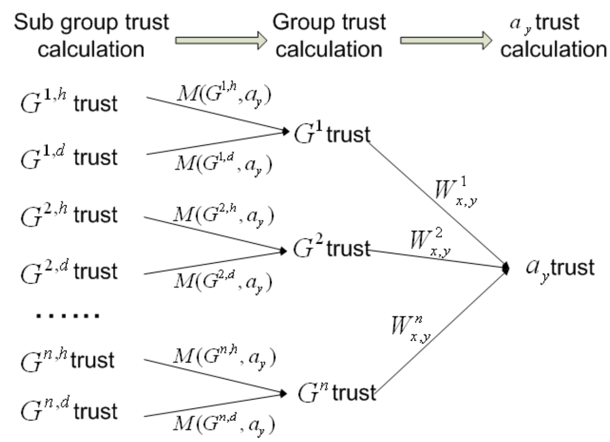

Figure 3: Process of trust calculation. Trusts of honest sub group $G^{i,h}$ and dishonest sub group $G^{i,d}$ of each group G^i are firstly combined using closeness and then trusts of all groups G^i are combined using weight factor $W^i_{x,y}$ to derive target agent's trust.

of trust rating $TD_{a_x,a_y}(p)$ using Eq. (3):

$$TD_{a_x,a_y}(p) = \sum_i W^i_{x,y} \cdot T_{a_x,G^i}(p|S_{a_x,G^i}, U_{a_x,G^i}), \qquad (4)$$

where S_{a_x,G^i} and U_{a_x,G^i} are aggregated numbers of successful and unsuccessful transactions between a_x and members of group G^i.

SOP Approach (Sum Of Parameters)

In this approach, we use only one trust function by setting the parameters, i.e. numbers of corresponding successful and unsuccessful transactions.

$$TD_{a_x,a_y}(p) = T_{a_x,a_y}(p|\sum_i W^i_{x,y} \cdot S_{a_x,G^i}, \sum_i W^i_{x,y} \cdot U_{a_x,G^i}), \qquad (5)$$

where S_{a_x,G^i} and U_{a_x,G^i} are defined as in SOF approach.

2.4 Dichotomy Based Enhanced Model

StereoTrust model simply groups agents based on agents' profiles. This makes StereoTrust model difficult to accurately predict the performance of an agent who behaves quite differently from the other agents of the same groups. For instance, consider a case when most of the agents that a_x has interacted are honest, while the target agent is malicious. StereoTrust will derive high trust for the malicious target agent.

To improve prediction accuracy, we propose dichotomy-based enhancement of StereoTrust (called d-StereoTrust). The main idea is to construct sub groups that divide agents on a finer level than groups based on agents' profiles. Additional information (third party information) is needed in this case. Figure 2 illustrates this kind of grouping.

In d-StereoTrust, each top-level group G^i is further divided into two sub groups, an *honest* $G^{i,h}$ and a *dishonest* $G^{i,d}$ sub group (hence dichotomy-based). a_x assigns an agent $a \in G^i$ to either sub group by analyzing history of her transactions with a. The basic criterion we use is that if a_x has more successful than unsuccessful transactions with a, a is added to the honest sub group $G^{i,h}$ (and, consequently, in the alternative case a is added to $G^{i,d}$). Several alternative criteria are possible, for instance, the average rating of transactions with a.

After dividing a group G^i into sub groups ($G^{i,h}$, $G^{i,d}$) and determining a_x's trust towards the sub groups (computed as in the previous section), d-StereoTrust computes how similar is the target agent a_y to the honest and the dishonest sub group. If a_y "seems" more honest, a_x's trust towards aggregated G^i should reflect more a_x's trust towards the honest sub group $G^{i,h}$; similarly, if a_y "seems" more dishonest, the dishonest sub group $G^{i,d}$ should have more impact on a_x's aggregated trust towards G^i. This process is illustrated on Figure 3.

The closeness, which can be measured by membership $M_x(G^{i,\cdot}, a_y)$ of target agent a_y to sub group $G^{i,\cdot}$ (where $\cdot$ represents d or h) is based on other agents' opinions about a_y. Note that we cannot group a_y as any other agent $a \in \mathcal{A}_x$, because the grouping described above is based on a_x history with a, and, obviously, there are no previous transactions between a_x and a_y. Thus, both $M_x(G^{i,h}, a_y)$ and $M(G^{i,d}, a_y)$ are fuzzy (in $[0, 1]$).

Agent a_x obtains opinions about a_y by requesting a certain metric from other agents. For instance, a_x can ask other agent a_k about the percentage (denoted by $m_{k,y}$) of successful transactions she had with a_y. a_x will seek opinions from honest agents from the group $G^{i,h}$; and also from agents interested in G^i, but with no transactions with a_x (based on their profile information, these agents could be classified as members of G^i, but they had no transactions with a_x). Obviously, the agents who have no transactions with a_x may be dishonest thus may provide false reports.

Note that the amount of historic information needed from other agents in d-StereoTrust is a small subset of information required in models based on feedbacks or transitive/Eigentrust. To collect feedbacks or form transitive trust paths, Eigentrust-like algorithms must explore the whole network (take into account all the available historic transactions). In contrast, in d-StereoTrust, a_x only asks the agents who are interested in the corresponding groups.

Based on all opinions $m_{k,y}$ received, a_x computes an aggregated opinion m_y, which is used to measure the closeness of a_y to sub groups as a simple average of $m_{k,y}$.

To characterize sub groups in a similar way, a_x computes similar aggregation of *her* opinions towards sub groups $G^{i,h}$ and $G^{i,d}$. Aggregated opinion m_h about sub group $G^{i,h}$ is equal to the simple average of $m_{x,j}$ (percentage of successful transactions a_x had with a_j), where j is the index of agent $a_j \in G^{i,h}$. The aggregated opinion m_d about sub group $G^{i,d}$ is derived in the same way.

Finally, the closeness between a_y and each of the sub groups is computed as the fraction of the distance between m_y from one side and m_h and m_d from the other:

$$M_x(G^{i,h}, a_y) = \frac{1/(|m_y - m_h|)}{1/(|m_y - m_h|) + 1/(|m_y - m_d|)} \quad (6)$$

$$M_x(G^{i,d}, a_y) = \frac{1/(|m_y - m_d|)}{1/(|m_y - m_h|) + 1/(|m_y - m_d|)} \quad (7)$$

This procedure has a straightforward interpretation. If other agents have similar opinions about target agent a_y as a_x has about the dishonest sub group, then the target agent is most likely dishonest, so the dishonest sub group trust should more influence a_y's trust in the context of group G^i. Similarly, if other agents have experienced similar performance with a_y as a_x with the honest group, then a_y is most likely honest.

Note that we do not use the opinions provided by other entities to directly calculate a_y's trust. Instead, we use them as metrics to measure closeness between a_y and the sub groups. In other words, we do not ask other agents "is a_y honest?", but rather we ask about quantified experience they had with a_y. This allows us, firstly, to be more objective; and, secondly, to easily extend d-StereoTrust to use multiple metrics (and to combine them with, e.g., Euclidean distance). Simulation results show that d-StereoTrust derived trust is more accurate than that is derived directly using others' opinions.

Also note that when other agents' opinions are not available, d-StereoTrust model degrades to StereoTrust model.

After calculating closeness, we combine groups' trusts to derive a_y's trust. Similarly to the original StereoTrust, there are two approaches to combine various trust sources.

SOF Approach (Sum Of Functions)

Using Eq. (1)(3)(6) and (7) we have probability density of target agent (a_y)'s trust rating $TD_{a_x,a_y}(x)$:

$$TD_{a_x,a_y}(p) = \sum_i W^i_{x,y} \cdot (M_x(G^{i,h}, a_y) \cdot T_{a_x,G^{i,h}}(p|S_{a_x,G^{i,h}}, U_{a_x,G^{i,h}}) + M_x(G^{i,d}, a_y) \cdot T_{a_x,G^{i,d}}(p|S_{a_x,G^{i,d}}, U_{a_x,G^{i,d}})), \quad (8)$$

Where $S_{a_x,G^{i,h}}/S_{a_x,G^{i,d}}$ and $U_{a_x,G_{i,h}}/U_{a_x,G^{i,d}}$ are aggregated numbers of successful and unsuccessful transactions of each member of G^i's sub group $G^{i,h}/G^{i,d}$ from viewpoint of agent a_x.

SOP Approach (Sum Of Parameters)

Using Eq. (1)(3)(6) and (7) we have probability density of agent a_y's trust rating $TD_{a_x,a_y}(x)$:

$$TD_{a_x,a_y}(p) = T_{a_x,a_y}(p|\sum_i W^i_{x,y} \cdot (M_x(G^{i,h}, a_y) \cdot S_{a_x,G^{i,h}} + M_x(G^{i,d}, a_y) \cdot S_{a_x,G^{i,d}}), \sum_i W^i_{x,y} \cdot (M_x(G^{i,h}, a_y) \cdot U_{a_x,G^{i,h}} + M_x(G^{i,d}, a_y) \cdot U_{a_x,G^{i,d}}), \quad (9)$$

Where $S_{a_x,G^{i,h}}/S_{a_x,G^{i,d}}$ and $U_{a_x,G^{i,h}}/U_{a_x,G^{i,d}}$ have the same meanings with that in SOF approach.

3. EVALUATION

In this section, we conduct experiments to evaluate the performance of (d-)StereoTrust. We first discuss methodology in 3.1. In 3.2 and 3.3, we present our results that use Epinions dataset and synthetic dataset respectively.

3.1 Methodology

To evaluate performance of proposed models, we compare our models with some other algorithms. We consider two factors: the *accuracy of prediction* that compares the result of the algorithm with some ground truth; and the *coverage* – fraction of the population that can be evaluated by the trustor, given trustor's limited knowledge.

We compare StereoTrust with the following algorithms.

Feedback Aggregation In this model, If trustor does not know target agent, it asks other agents across the network and aggregate feedbacks to derive target agent's trust. Please note that as trustor may not have experience with the probed agent, it can not identify the dishonest agents, thus may suffer false feedbacks.

EigenTrust EigenTrust [11] uses transitivity of trust and aggregates trust from peers by having them perform a distributed calculation using eigenvector of the trust matrix. Trustor first requests its trusted friends about target agent's trust. Each opinion of a friend is weighted with the friend's global reputation. To get the wide view of target agent's information, trustor will continue asking its friends' friends, and so on, until the difference of two derived trusts in two subsequent iterations is smaller than a threshold (convergence). Pretrust agents (with high global reputation) are used in this model.

Transitive Trust (Web of Trust) This model is based on transitive trust chain. If trustor doesn't know target agent, it asks its neighbors and its neighbors will ask their neighbors if they do not know target agent either, so the trust graph is formed.

Shortest Path In this variation, agent only chooses the shortest path and ignore the trustworthiness of agents along the path. If multiple shortest trust paths exist, trustor will choose the most reliable one (the agents along the path are the most reliable).

Most Reliable Path In this variation, agent will choose its most reliable neighbor who has the highest trust rating to request for target agent's trust. If this neighbor does not know target agent, it continues requesting its own most reliable neighbor. So the most reliable path is found. To avoid infinite requesting, number of hops is limited, in this experiment, is 6. That is to say, if no agent knows target agent within 6 hops, this model fails to derive the target agent's trust.

Group Feedback Aggregation d-StereoTrust uses opinions reported by the agents who are the the members of honest sub groups and agents who are also interested in the corresponding groups as the metrics to measure closeness between target agent and the sub groups. we compare the accuracy of trust value derived using other agent's opinions (called group feedback aggregation) with that derived using d-StereoTrust to validate whether such third party information is used by d-StereoTrust judiciously. Note that different from feedback aggregation we described above, group feedback aggregation only uses the feedbacks provided by the agents that are interested in the corresponding groups.

Dichotomy-only d-Stereotrust divides each group into an honest and dishonest sub groups. To evaluate the impact of the initial grouping in d-Stereotrust, we compare d-Stereotrust with a similar, dichotomy-based algorithm, but without the initial grouping (without the stereotypes). In dichotomy-only, agent a_x classifies all the agents it has previously interacted with into two groups: honest and dishonest ("honest agents" having more successful transcations with a_x). To evaluate agent a_y, similarly to d-Stereotrust, a_x asks honest agents about their trust information to a_y, and using these feedbacks, calculates the distance between a_y and the two groups. If dichotomy-only's results are similar to d-Stereotrust, the "stereotypes" used in d-Stereotrust (the initial groups) are not needed, as they do not increase the accuracy of the model.

Please note that we only compare StereoTrust model with the existing models using synthetic dataset because we lack the information of how users in Epinions dataset carry out these existing models.

To estimate the accuracy of each algorithm, we compare the value of trust computed by the algorithm for a pair of agents with the ground truth. Then, we aggregate these differences over different pairs using Mean Absolute Error (MAE).

We present the results in two formats. Firstly, to measure overall performance of an algorithm, we show MAE aggregated over the whole population of agents (e.g., Table 1). Secondly, to see how the algorithm performs in function of agent's ground truth, we construct figures presenting the derived trust for a subset of agents (e.g., Figure 4). To avoid cluttering, we randomly choose 50 target agents. Y-axis represents the trust rating of the agents. X-axis represents the index of the evaluated agent. For clarity, agents are ordered by decreasing ground truth.

Ideally, the ground truth of an agent represents agent's objective trustworthiness. However, as we are not able to measure it, we have to estimate it using the available data. We will discuss how to derive the ground truth when mapping each dataset. Besides prediction accuracy, we also measure the performance of algorithms using coverage – percentage of agents in the system that can be evaluated by trustor.

The complete Epinions dataset we crawled contains 5,215 users, 224,500 reviews and 5,049,988 ratings of these reviews. For our experiments, we selected 20 trustors and 150 target agents randomly (we repeated the experiments with different agents and got the similar results). On the result plots (e.g., Figure 4), error bars are added to show the deviation of predictions by each trustor for the same target agent.

In the experiments using synthetic dataset, we choose one honest agent randomly as trustor to predict behavior of other agents in the system (there are totally 200 agents in the system). Each experiment is repeated 10 times (each running uses different synthetic dataset) and error bars are added to indicate the deviation of each running (e.g., Figure 8).

3.2 Epinions Dataset

Epinions.com is a web site where users can write reviews about the products and services, such as books, cars, music, etc (later on we use the generic term "product"). A review should give the reader detailed information about a specific product. Other users can rate the the quality of the review by specifying whether it was *Off Topic*, *Not Helpful*, *Somewhat Helpful*, *Helpful*, *Very Helpful* or *Most Helpful*. For each review, Epinions.com shows an average score assigned by users.

Epinions.com structures products in tree-like categories. Each category (e.g., books) can include more specific categories (e.g., adventure, non-fiction, etc.). The deeper the level, the more specific category the product belongs to.

Epinions community provides a good scenario to test our proposed model, where users write reviews or rate reviews of products they are interested. This gives the intuitive grouping criteria, that is, users are grouped if they are interested in the same product/category. We use Epinions dataset to test the performance of StereoTrust and see whether the enhanced model d-StereoTrust performs better.

3.2.1 Modeling of Epinions to StereoTrust Model

To map Epinions.com to StereoTrust model, we treat each user as an agent. Epinions.com categories provide a natural representation of *interested in* relation. A user is *interested in* a (sub)category if she wrote or rated at least one review of a product under this category. Groups are formed according to agents' interested in relations. Consequently, each Epinions.com category corresponds to a group of agents, each of whom is *interested in* (wrote or rated a review for) this category. A transaction between agents a_x and a_y occurs when a_x rates a review written by a_y. To map Epinions.com ratings to StereoTrust binary outcome, we assume that the transaction is successful only if the assigned rate is *Very Helpful* or *Most Helpful*. We set the threshold so high to avoid extreme sparsity of unsuccessful transactions (over 91% review ratings are *Very Helpful* or *Most Helpful*).

We compute the ground truth of an agent as the average rating of the reviews written by this agent. For instance, if an agent wrote 3 reviews, the first review was ranked by two users as 0.75 and 1.0 respectively, while the second and the third received one ranking each (0.75 and 0.5), the ground truth for that user is equal to $(0.75+1.0+0.75+0.5)/4$. Note that the "ground truth" computed with this simple method only approximates the real trustworthiness of an agent, as we do not adjust the scores to counteract, e.g., positive or negative biases of the scoring agents.

3.2.2 *Results*

Figure 4 shows the performance of StereoTrust model. SOF/SOP on the legend indicates that the trust rating is calculated using SOF/SOP approach respectively. From the figure we can see that both SOF and SOP approaches fail to provide a good fit to the ground truth. This is because in Epinions dataset, most ratings given by the agents are positive (*Very Helpful* or *Most Helpful*). So trustors are in a very friendly environment, which makes them difficult to identify the agents who write somewhat low quality reviews simply using local information.

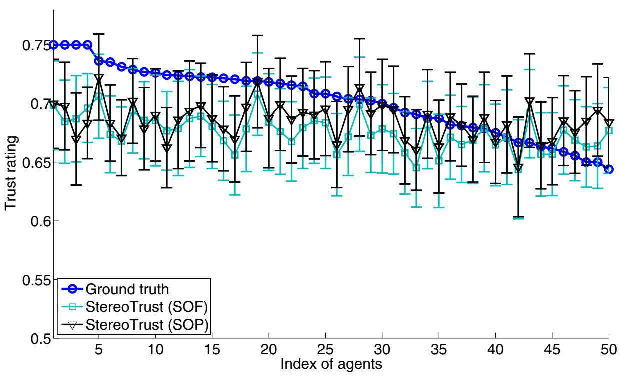

Figure 4: Comparison of StereoTrust model and ground truth using Epinions.com dataset.

Figure 5 show the performance of d-StereoTrust model. We can see both SOF and SOP derived trust ratings are more accurate than feedbacks derived trust rating (group feedback aggregation), which supports that our model outperforms that simply aggregates other agents' feedbacks. SOF approach gives a better fit to the ground truth than SOP approach. Comparing Figure 4 and 5, We observe that d-StereoTrust is obviously better than StereoTrust (d-StereoTrust provide better fit to ground truth), so d-StereoTrust improve the accuracy of prediction of target agent's performance as we expected.

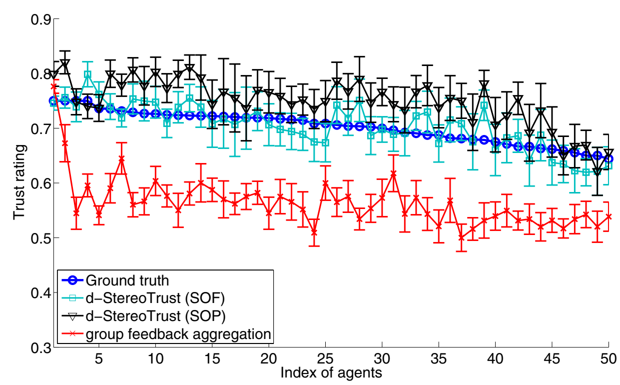

Figure 5: Comparison of d-StereoTrust and ground truth using Epinions.com dataset.

Figure 6 compares d-StereoTrust model with dichotomy-only (StereoTrust is omitted as it is worse than d-StereoTrust in terms of prediction accuracy). Error bars are removed for clarity and only SOF approach, which outperforms SOP approach is showed for each model. From the figure we see that d-StereoTrust model provides more accurate prediction than dichotomy-only does. This proves that considering both interests based group and some global information can predict target agent's behavior more accurately.

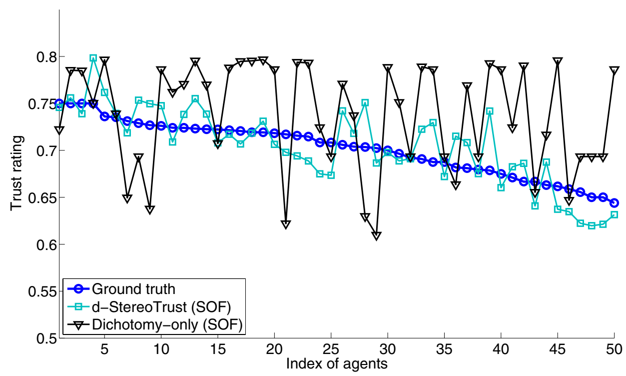

Figure 6: Comparison of d-StereoTrust model and dichotomy-only using Epinions.com dataset.

Table 1 lists the calculated MAE along with 95% confidence interval. Although plots only show 50 target agents, MAE/95% C.I. is computed using all $20 \cdot 150$ (trustor, target agent) pairs. Confidence interval is computed using the expression $\bar{x} \pm S_E \cdot 1.96$, where $\bar{x}$ is the sample mean, S_E is the standard error for the sample mean, and 1.96 is the 0.975 quantile of the normal distribution.

Table 1: Mean Absolute Error (with 95% Confidence Interval)

	MAE	95% C.I.
StereoTrust (SOF)	0.1114	(0.1067,0.1161)
StereoTrust (SOP)	0.1177	(0.1136,0.1218)
d-StereoTrust (SOF)	0.0632	(0.0586,0.0678)
d-StereoTrust (SOP)	0.1299	(0.1245,0.1353)
dichotomy-only (SOF)	0.1365	(0.1307,0.1423)
dichotomy-only (SOP)	0.1750	(0.1690,0.1810)
group feedback aggregation	0.1452	(0.1386,0.1518)

3.2.3 *Discussion*

From the results we can see that simply using only "stereotype" (category information) to form group (StereoTrust model) does not predict target agent's performance accurately because Epinions is a friendly community. Users are likely to give high ratings to reviews written by others, which makes successful transactions dominate the groups. In such environment, StereoTrust model more probably derives a high rating for the target agent. Due to the same reason, dichotomy-only does not work well either. On the contrary, d-StereoTrust improves the prediction accuracy. This proves that using groups and small amount of trust information from other agents helps provide more accurate prediction of target agent's performance.

3.3 Synthetic Dataset

Epinions.com has a friendly community with few dishonest agents. To test StereoTrust in a more hostile environment, we generated a synthetic dataset simulating a hostile version of Epinions.com-like community.

3.3.1 Synthetic Dataset Generation

In the synthetic dataset, 40% of the population of 200 agents are dishonest. A honest agent provides a high quality review (with real rating = 0.6 or 0.8 or 1.0) or a true feedback with a probability 0.9. A dishonest agent provides a low quality review (real rating = 0.0, 0.2, 0.4) or a false feedback with a probability 0.9. Note that if a true feedback is a value of λ, the corresponding false feedback is a value of $1 - \lambda$. Both number of reviews written by an agent and number of ratings of a review are generated by a Normal distribution ($\mu = 10$, $\sigma = 4$). Note that the agents who assign ratings to a review are selected randomly from a set of agents who are also interested in the category that this review is about.

We simulate an environment with 12 categories (indexed $1, 2, ..., 12$) and 20 products in each category. Honest and dishonest agents are biased towards different categories. A honest agent with probability 0.7 writes a review for a product from categories $1, 2, 3, 4$; with probability 0.21 for products from categories $9, 10, 11, 12$; and with probability 0.03 for products from categories $5, 6, 7, 8$. A dishonest agent with probability 0.7 writes a review for products from categories $5, 6, 7, 8$; with probability 0.21 for products from categories $9, 10, 11, 12$; and with probability 0.03 for products from categories $1, 2, 3, 4$.

We compute the ground truth of an agent as the average rating of the reviews written by this agent. Different from ground truth in Epinions dataset, in synthetic dataset, rating of one review is determined by the design of dataset, so this rating represents the real quality of the review, thus the calculated ground truth more approximates the objective trustworthiness of one agent.

3.3.2 Results

Figure 7 shows the performance of StereoTrust model. We can see that the model derived trust rating fits the ground truth in general but not very closely. However, the trend looks better than that in real Epinions dataset.

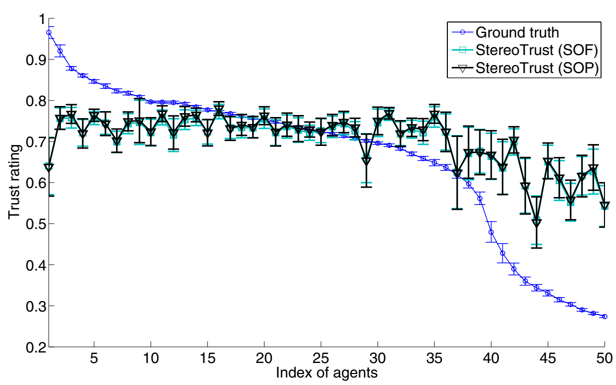

Figure 7: Comparison of StereoTrust model and ground truth using synthetic dataset.

Figure 8 shows the performance of d-StereoTrust. Obviously, d-StereoTrust provides more accurate prediction than StereoTrust model (more fits to ground truth). This is because, d-StereoTrust forms groups in a finer granularity thus local trust information and third party information are properly used to represent target agent's trust. Similar to Epinions dataset, feedback derived trust (group feedback aggregation) is less accurate than that derived by SOF/SOP.

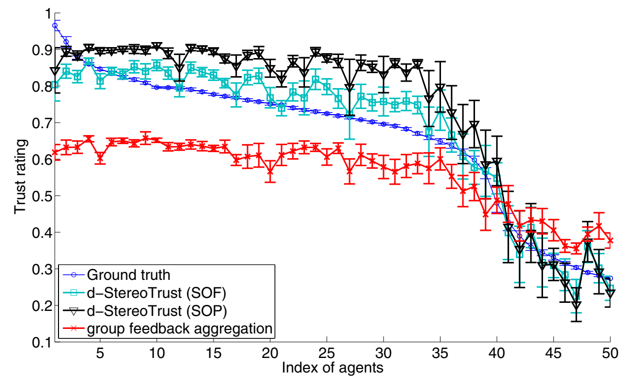

Figure 8: Comparison of d-StereoTrust and ground truth using synthetic dataset.

Figure 9 compares d-StereoTrust model (using SOF) with dichotomy-only (using SOF) and various existing algorithms (described in section 3.1). Error bars are removed for clarity. From the figure we observe that the existing algorithms predict target agent's trust less accurately than d-StereoTrust does. These existing algorithms show obvious gaps between ground truth and derived rating for honest target agents part (like EigenTrust and most reliable path transitive trust) or dishonest target agents part (like shortest path transitive trust) or all the target agents (like feedback aggregation).

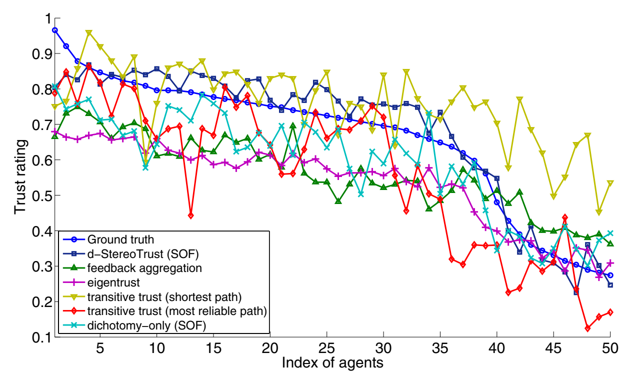

Figure 9: Comparison of all the algorithms using synthetic dataset.

Table 2 summaries MAE (with 95% confidence interval for all the agents) and coverage of each model involved in comparison. For each model, we show MAE for evaluating honest target agents part, dishonest target agents part and all target agents respectively. Note that for StereoTrust, d-StereoTrust and dichotomy-only, we only show the results using SOF approach, which outperforms SOP approach.

3.3.3 Discussion

Feedback aggregation and both variations of transitive trust models are significantly worse than d-StereoTrust in

Table 2: Mean Absolute Error and Coverage (for synthetic dataset)

	Honest agents	Dishonest agents	All agents (with 95% C.I.)	Coverage
d-StereoTrust (SOF)	0.1154	0.1099	0.1126 (0.1036,0.1216)	95.5%
EigenTrust	0.1487	0.1002	0.1263 (0.0966,0.1510)	96.4%
Dichotomy-only (SOF)	0.1326	0.1215	0.1288 (0.1202,0.1374)	96.3%
StereoTrust (SOF)	0.1377	0.2641	0.1884 (0.1300,0.2753)	96.9%
Feedback aggregation	0.1450	0.1642	0.1535 (0.1432,0.1678)	99.9%
Transitive trust (shortest path)	0.1547	0.3319	0.2304 (0.1424,0.3384)	99.3%
Transitive trust (most reliable path)	0.1468	0.1678	0.1552 (0.1416,0.1688)	82.1%

terms of prediction accuracy even if feedback aggregation and transitive trust (shortest path) have the best coverage. Transitive trust model (most reliable path) has the worst coverage. EigenTrust performs almost as good as d-StereoTrust but requires more third party information thus incurring higher communication overhead. Additionally, the assumption of pre-trusted agents is not realistic, which is essential to this model. Dichotomy-only, as a baseline, proves that considering "stereotypes" improves the accuracy prediction of target agent's behavior definitely. So to sum up, d-StereoTrust, which has the highest prediction accuracy (MAE is the smallest) at the cost of losing a bit coverage (95.5%) and incurring medium communication overhead (trustor only asks agents that are also interested in corresponding categories) outperforms among all the models. This also proves that d-StereoTrust is more robust to large portion of malicious agents (up to 40% in the experiments) than other algorithms. Both real and synthetic dataset proves that StereoTrust model does not work well as d-StereoTrust model. However, StereoTrust has its own advantage in term of communication overhead because it only uses local information to derive trust of target agent. This is very promising in some scenarios where by carefully forming groups, honest and dishonest agents are put into groups which seldom overlap, thus group trust can represent individual trust accurately.

4. RELATED WORK

Past mutual interactions information can be used to predict an agent's future behavior (e.g., [1]), but such an approach is unsuitable in distributed systems where one may need to assess trustworthiness of an agent with whom there have been no past personal interactions.

Instead of using only local experience, many works derive the trust of interaction partners based on reputation – information gathered from third parties. Abdul-Rahman et al [2] and Jøsong et al [9] used transitive trust path to derive participant's trust. However, transitive trust is not always true in real world (some conditions must be fulfilled) and it has several drawbacks: (i) This method does not handle wrong recommendations properly, which affect the accuracy of derived trust seriously. (ii) This method does not provide a mechanism for updating trust efficiently in a dynamic system. (iii) Establishing a trust path, even if such a path exists, is nontrivial.

EigenTrust [11] is a reputation system developed for P2P networks. It tries to fulfill the goals of self-policing, anonymity, no profit for newcomers, minimal overhead and robust to malicious collectives of peers. EigenTrust uses transitivity of trust and aggregates trust from peers by having them perform a distributed calculation to determine the eigenvector of the trust matrix over peers. The main drawback of EigenTrust is that it relies on some pre-trusted peers, which are supposed to be trusted by all peers. This assumption is not always true in real world. First, these pre-trusted peers become points of vulnerability for attacks. Second, even if these pre-trusted peers can defend attacks, there are no mechanisms to guarantee that they are reliable/trustworthy permanently. Additionally, EigenTrust (and some other reputation systems like [16]) is designed based on Distributed Hash Tables and thus imposes system design complexity and deployment overheads. Our proposed model does not need an overlay for trust management.

REGRET [13] combines direct experience with social dimension of agents, that also includes so-called system reputation. System reputation is based on previous experience with other agents from the same *institution*. Unlike StereoTrust's groups, REGRET's institutions exists outside the system and there is objective function to assign agents to institutions. REGRET also assumes that an agent belongs only to one institution.

Ravichandran et al [12] proposed a trust system built on top of a peer group infrastructure. The group in this paper is formed based on a particular interest criterion and members must follow a set of rules of its group. The authors assumed that a group leader creates the group and controls the membership. To calculate trust, the authors introduced Eigen Group Trust, which is an aggregative version of EigenTrust [11]. In Eigen Group trust, all the transactions rely on the group leaders, who are assumed to be trusted and resourceful. Their notion of groups is thus very different from ours.

Different from existing works, we define groups using agent's local information according to certain group criteria (not merely *interests*). These locally defined groups may overlap or be disjoined depending on the criteria used. Moreover, different agents may have entirely different criteria.

While our technique is novel in the context of evaluating trust – and provides a new paradigm of using stereotypes for trust calculation instead of using feedbacks or web of trust – it bears resemblance with collaborative filtering techniques. The primary difference is that StereoTrust uses only local information in a decentralized system. However, the similarities also mean that while our work proposes a new paradigm to determine trust, the methodology we use is not out of the blue. Also, we anticipate that sophisticated collaborative filtering techniques can be adopted to further improve StereoTrust's performance.

StereoTrust also has parallels to web search engines' ranking mechanisms. Using the group information is analogous to using the content of the web pages to rank them. Transitive trust models resemble "pure" PageRank, that uses only links between pages. Similarly to web search, where using both content and links together gives better results, we derive an enhanced method (d-StereoTrust), that uses both groups and (limited) trust transitivity.

5. CONCLUSIONS

We consider the problem of predicting the trust between a trustor and an unknown agent in a large-scale distributed setting. Traditional approaches to this problem derive unknown agent's trust essentially by combining trust of third parties to the agent with the trustor's trust of these third parties; or simply by aggregating third parties' feedbacks about the unknown agent. In contrast, StereoTrust uses different *kind* of information: that of *semantic* similarity of the unknown agent to other agents that the trustor personally knows. In StereoTrust, a trustor builds stereotypes that aggregate and summarize the experience she had with different kinds of agents. The basis of the grouping to construct stereotypes is very flexible. For instance, stereotypes can be based on information from agents' personal profiles, or the class of transactions they make. Facing a possible transaction with an unknown agent, the trustor builds its trust by cumulating the experience from the stereotypes to which the unknown agent conforms.

The stereotypes are based entirely on the local perspective and local information of the trustor, and, therefore, are naturally suited for large-scale systems; personalized for each trustor; and less susceptible to false or unsuitable information from third parties. The rationale for the StereoTrust approach is to determine an alternative and complementary mechanism (than existing techniques) to compute trust even in absence of (global) information that is likely to be unavailable under some circumstances, and instead using some other class of information (stereotypes) which can be established by local interactions.

When some of third parties' opinions about an agent are available, we propose an enhancement (d-StereoTrust), which creates a "good" and a "bad" subgroup inside each stereotype. The trustor assigns each one of her previous transaction partners to one of these groups based on the her personal experience with the partner (e.g., the ratio of failed transactions). Then, the trustor uses the aggregated third parties' opinions about the unknown agent to determine how similar is the agent to the "good" and "bad" subgroup. Third parties' opinions are a small subset of information used by traditional mechanisms (such as feedback aggregation or Eigentrust-type algorithms). However, according to our experiments, by combining stereotypes with these partial historic information, d-StereoTrust predicts the agent's behavior more accurately than Eigentrust and feedback aggregation.

StereoTrust can not only be personalized for the trustor, but also, it can be used to determine an agent's trustworthiness for specific type of interactions (classified by groups). We are currently working on such extensions of StereoTrust, as well as exploring possible concrete applications to employ it on, including in designing a p2p storage system like we explained in the motivation.

Acknowledgment

This work has been partly supported by the HP Labs Innovation Research Program research grant.

6. REFERENCES

[1] L. Mui and M. Mohtashemi. A computational model of trust and reputation. In *In Proceedings of the 35th HICSS*, 2002.

[2] A. Abdul-Rahman and S. Hailes. A distributed trust model. In *NSPW '97: Proceedings of the 1997 workshop on New security paradigms*, pages 48–60, New York, NY, USA, 1997. ACM.

[3] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, 2001.

[4] R. Bartoszynski and M. Niewiadomska-Bugaj. *Probability and statistical inference.* Wiley-Interscience, 2008.

[5] S. Buchegger and J.-Y. L. Boudec. A robust reputation system for p2p and mobile ad-hoc networks. In *Proceedings of P2PEcon*, 2004.

[6] A. Datta, M. Hauswirth, and K. Aberer. Beyond "web of trust": Enabling p2p e-commerce. In *Proceedings of the IEEE CEC*, pages 24–27, 2003.

[7] D. Gambetta. Can we trust trust? In D. Gambetta, editor, *Trust: Making and Breaking Cooperative Relations.* Basil Blackwell, 1988.

[8] M. Gupta, P. Judge, and M. Ammar. A reputation system for peer-to-peer networks. In *Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video*, 2003.

[9] A. Jøsang, E. Gray, and M. Kinateder. Analysing topologies of transitive trust. In *Proceedings of the Workshop of Formal Aspects of Security and Trust (FAST)*, page 150, 2003.

[10] A. Jøsang and R. Ismail. The beta reputation system. In *Proceedings of the 15th Bled Electronic Commerce Conference*, June 2002.

[11] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *WWW '03: Proceedings of the 12th international conference on World Wide Web.* ACM Press, 2003.

[12] A. Ravichandran and J. Yoon. Trust management with delegation in grouped peer-to-peer communities. In *SACMAT '06: Proceedings of the eleventh ACM symposium on Access control models and technologies*, pages 71–80, New York, NY, USA, 2006. ACM.

[13] J. Sabater and C. Sierra. Regret: reputation in gregarious societies. In *AGENTS '01: Proceedings of the fifth international conference on Autonomous agents*, pages 194–195, New York, USA, 2001.

[14] S. Tadelis. Firm reputation with hidden information. *Economic Theory*, 2003.

[15] J. Tirole. A theory of collective reputations (with applications to the persistence of corruption and to firm quality). *Review of Economic Studies*, 1996.

[16] L. Xiong, L. Liu, and I. C. Society. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16:843–857, 2004.

[17] Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing. *IEEE Trans. Parallel Distrib. Syst.*, 18(4):460–473, 2007. Member-Zhou, Runfang and Fellow-Hwang, Kai.