Trust beyond reputation: A computational trust model based on stereotypes

Xin Liu
Nanyang Technological University
liu_xin@pmail.ntu.edu.sg

Anwitaman Datta
Nanyang Technological University
anwitaman@ntu.edu.sg

Krzysztof Rzadca
University of Warsaw
krzadca@mimuw.edu.pl

Abstract

Trust plays important roles in diverse, large scale, open environments. 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 existing trust models, however, require historical information about past behavior of a specific agent in question – information that is not always available. In contrast, in real life interactions, 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” other people. In this paper, we propose StereoTrust, a computational trust model inspired by use of stereotypes in real life. A stereotype contains certain features of agents and an expected outcome of the transaction. These features can be taken from the agents’ profile information, or the agents’ observed behavior. When facing a stranger, the stereotypes matching stranger’s profile are aggregated to derive an expected trust. Additionally, when some information about a stranger’s previous transactions is available, StereoTrust uses it to refine the stereotype matching. In case trustor (i.e., the agent that needs to derive trust on other agents) does not have sufficient past transactions, we propose to construct an overlay network to share local knowledge among trustworthy agents. Such extensions of the StereoTrust model thus allows a seamless incorporation of existing techniques based on aggregation of historical data as well as social (web-of-trust) networks. StereoTrust compares favorably with existing trust models that use different kind of information and more complete historical information for both real as well as synthetic data sets. Moreover, because evaluation is done according to agent’s local personal stereotypes, the system is completely distributed and the obtained recommendation is personalized.

Keywords: trust model, stereotypes, local information, personalized recommendation
1 Introduction

Trust is an important abstraction used in diverse scenarios including e-commerce, distributed and peer-to-peer systems, grid systems and dynamic collaborative systems. Due to 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 [17]. However, in large-scale environments, direct experience is often not sufficient or even non-existent. In such cases, prediction is based on user’s “indirect experience” – opinions obtained from other agents [13, 27, 1] (which essentially determines the 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 [1, 11] (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 from inaccurate reports and “weakest link” [8].

All the mentioned approaches aggregate the same kind of information – agents’ impressions about the transactions. Many systems however 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) such contextual information associated with the agent and/or transaction. Thus, the original objective of this work was not specifically to propose a better mechanism than existing trust models, and using the same information as they do, but rather to explore the feasibility of a complementing alternative when information used by existing models is either unavailable, or in addition to such information.

To that end, we have developed a trust model that estimates target agents’ trust using stereotypes learned by the assessor from her own interactions with other agents with similar profile, where the profile is determined from contextual information associated with the agent/transaction. Our work is inspired by [21, 23], which study 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’ profiles, or the context of the transaction. Example stereotypes are “programmers employed by a reputable software company are more skilled than average” or “people living in affluent areas are richer than those living in other parts of the city”. To build stereotypes, a user has to group other agents (“programmers employed by reputable software company” or “people living in affluent areas”). 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. In case the users do not have sufficient local knowledge, i.e., stereotypes cannot be effectively formed, we propose to construct a stereotypes sharing overlay network (SSON), which allows local information (i.e., calculated stereotypes) to be exchanged between trustworthy agents. Such mechanism helps to bootstrap inexperienced users.

StereoTrust has its own weaknesses and limitations. For instance, not in all circumstances may a user determine the profile of an agent. However, we think that StereoTrust is interesting both academically and in practice. Academically, its novelty lies in emulating a real world human behavior for modeling trust by using stereotypes for a first guess about a stranger. In practice, firstly, StereoTrust may be applicable under certain circumstances even when information used by traditional trust models is not available, or noisy, inaccurate or tampered and hence unreliable. Secondly, StereoTrust facilitates personalized recommendations. Finally, it can be seamlessly extended to incorporate and leverage on global information used by traditional trust models, if available, as well, in order to enhance the prediction (thus, the term “stereo” in the model’s name is a double entendre). Our experiments show that such a refinement, called d-StereoTrust, significantly improves the accuracy.

StereoTrust is generic, and its use of abstract group definitions allow it to be used in very different kinds of applications, and even within a same application, different agents may have their own personal, locally defined groups. Also, the notion of trust itself can be easily adapted to different contexts. In this paper, we adopt the definition from [10]: “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 a 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 indicates that considering a 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 internet service providers), and so on. Conventional systems design models such a scenario as a multi-criteria optimization problem. Such an approach would typically need knowledge about the specific peer in question - for instance, its online pattern, end to end latency and available bandwidth, etc. Applying StereoTrust can provide an alternative systems design, 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 a 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. A ‘mixed’ data placement strategy, using historical information, when available, in addition to using such stereotypes, is expected in turn, to result in even better performance.

This paper extends from our preliminary results reported in [15]. On the basis of the original contributions, i.e., basic StereoTrust model (Section 3) and dichotomy based enhanced model (Section 4), we discuss in depth how the useful features are selected and combined (Section 2) using machine learning techniques and information theory. Such feature prepossesses can help to form the discriminating stereotypes, thus greatly affecting accuracy of the trust assessment. In case trustor does not have sufficient past transactions, we propose to construct an overlay network to share local knowledge among trustworthy agents to help inexperienced agents bootstrap the system (Section 5). Such extensions of the StereoTrust model thus allows a seamless incorporation of existing trust approaches based on aggregation of historical data. More comprehensive evaluation using real Epinions dataset as well as synthetic dataset is conducted in Section 6.

2 Preliminary

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 $\mathcal{A}_a$ the set of agents known to agent $a$. An agent can provide services for other agents. A transaction in the system happens

---

1We are not claiming that it necessarily provides the best possible system design, but merely that it opens the opportunity for alternative designs. We have in fact devised such StereoTrust guided data placement strategy for P2P backup systems, resulting in several desirable properties w.r. to other existing placement strategies [24].
when an agent accepts another agent’s service. To indicate the quality of a
service, an agent can rank the transaction. For simplicity, we assume binary
result of the transaction, i.e., successful or unsuccessful. \( \Theta_{a_x,a_y} \) denotes the
set of transactions between service provision agent \( a_y \) and service consumption
agent \( a_x \) and \( \theta_{a_x,a_y} = |\Theta_{a_x,a_y}| \) denotes the number of such transactions.

2.1 Group Definition

In StereoTrust, a group is a community of agents who show some common prop-
erties 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 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 her corresponding groups.

Groups are defined subjectively by the agent \( a_x \) that uses StereoTrust to
derive trust to other agents. A group \( G^i_x \) is a set of agents. We denote by
\( G_x = \{G^1_x, G^2_x, \ldots, G^n_x\} \) the set of all groups defined by \( a_x \). Based on \( a_x \)’s previ-
ous experience, stereotypes, and any other information, \( a_x \) formulates grouping
functions \( M_x(G^i_x, a) : [G_x, A_x] \rightarrow [0, 1] \), that, for each group \( G^i_x \), 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 \( M_x(G^i_x, a) = 1 \), it is certain that \( a \) is
member of \( G^i_x \) (\( a \in G^i_x \)); if \( M_x(G^i_x, a) = 0 \), it is certain that \( a \) does not belong
to \( G^i_x \) (\( a \notin G^i_x \)).

Note that in this paper we do not discuss how to derive the grouping func-
tions \( M_x(G^i_x, a) \). 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 [10]) 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). 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 \( \alpha \) and \( \beta \) (we start with \( \alpha = \beta = 1 \), that translate into complete uncertainty about the distribution of the parameter, modeled by the uniform distribution: \( \text{Beta}(1, 1) = U(0, 1) \)). After observing \( s \) successes in \( n \) trials, the posterior density of \( p \) is \( \text{Beta}(\alpha + s, \beta + n - s) \) [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:

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

(1)

The expected value of the trust function is equal to:

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

(2)

### 2.3 Feature Combination

Each stereotype is associated with a feature vector. In some cases, a trustor may want to develop new stereotypes that are associated with combined features. For instance, consider that she already has two stereotypes, say, on people from a country \( A \) and on women. When the trustor wants to develop a new stereotype on women from country \( A \), she needs to combine the two features (i.e., in country \( A \) and is female). Given a set of features \( F_c = \{f_1, f_2, ..., f_m\} \) that will be combined, if these features are qualitative, the combined feature is simply represented as \( f_c = f_1 \cap ... \cap f_m \). If these features are quantitative (e.g., we want to evaluate an agent based on the features such as salary, age, number of children, etc.), we use discriminant analysis (DA) [3, 9], which is a popular method to find a linear combination of features that separate two or more classes of events.

We first represent the group of honest agents and group of dishonest agents that are described by feature vector \( F_c \) (i.e., the features that need to be combined) by matrices (see expression [3] and [4]). Note that whether an agent is deemed honest or not depends on an individual trustor’s own criterion. For
instance, if over 60% of her past transactions with an agent are successful, then that agent is considered honest.

$$G_s = \begin{pmatrix}
    f_1^1(s) & f_2^1(s) & \cdots & f_m^1(s) \\
    \vdots & \vdots & \ddots & \vdots \\
    f_1^n(s) & f_2^n(s) & \cdots & f_m^n(s)
\end{pmatrix}$$

(3)

$$G_u = \begin{pmatrix}
    f_1^1(u) & f_2^1(u) & \cdots & f_m^1(u) \\
    \vdots & \vdots & \ddots & \vdots \\
    f_1^u(u) & f_2^u(u) & \cdots & f_m^u(u)
\end{pmatrix}$$

(4)

Note that $f_y^x(\cdot)$ represents $x^{th}$ feature of the $y^{th}$ agent and $s/u$ in the parentheses indicates whether the transaction is successful/unsuccessful. $n_s$ and $n_u$ are the sizes of honest agent group and dishonest agent group respectively. Note that $n = n_s + n_u$. Let $h_x$ be a $x \times 1$ (column) vector of ones, we then calculate centroid of each agent group: $c_s = \frac{1}{n_s} \cdot h_{n_s}^T G_s$, $c_u = \frac{1}{n_u} \cdot h_{n_u}^T G_u$ and the global centroid:

$$c = \frac{1}{n_s + n_u} \cdot h_{n_s + n_u}^T \begin{bmatrix}
    G_s \\
    G_u
\end{bmatrix}$$

(5)

In discriminant analysis, the internal variance (within-class scatter matrix) and external variance (between-class scatter matrix) are used to indicate the degree of class separability, i.e., to what extent are the honest agents distinguished from the dishonest ones. The internal variance and external variance are calculated using Eq. 6 and 7 respectively:

$$S_w = \frac{1}{n_s + n_u} (n_s(G_s - h_{n_s}c_s)^T(G_s - h_{n_s}c_s) + n_u(G_u - h_{n_u}c_u)^T(G_u - h_{n_u}c_u))$$

(6)

$$S_b = \frac{1}{n} (n_s(c_s - c)^T(c_s - c) + n_u(c_u - c)^T(c_u - c))$$

(7)

Discriminant analysis aims to find a projection direction (a transformation) $v$ that maximizes the external class variance and minimizes the internal class variance. Formally, the criterion function (see Eq. 8) is to be maximized.

$$J(v) = \frac{v^T S_b v}{v^T S_w v}$$

(8)

The projection direction $v$ is found as the eigenvector associated with the largest eigenvalue of $S_w^{-1}S_b$. Using $v$, we obtain the combined feature that describes the agents:

$$F' = v^T F_c$$

(9)
2.4 Feature Selection

One premise of our work is that group trust (i.e., stereotypes) can reflect trustworthiness of the group members. So it is ideal that all group members behave consistently, i.e., all the group members are either honest or dishonest. However, this situation does not always exist in reality. So an interesting problem arising from the use of stereotypes for determining trust is, which stereotypes are more discriminating, and which are not. Here we use information gain\(^2\), which is measured by entropy as the criteria to select features\(^3\).

We denote a set of features (original features and combined features) by \(F_s = \{f_1, f_2, ..., \}\). We assume that the categorization is binary (i.e., an agent is honest or dishonest). We denote the proportion of honest and dishonest agents by \(p_h\) and \(p_d\) respectively. Then the entropy of the set of agents that trustor \(a_x\) has interacted with is calculated as:

\[
\text{Entropy}(A_x) = -p_h \log_2(p_h) - p_d \log_2(p_d) \quad (10)
\]

Entropy is used to characterize (im)purity of a collection of examples. From Eq. (10) we can see that the entropy will have the minimum value of 0 when all agents belong to one class (i.e., either all honest or all dishonest) and the maximum value of \(\log_2 2 = 1\) when agents are evenly distributed across the two classes. Using entropy, we now calculate information gain of every feature to determine the best ones. For each feature \(f_i \in F_s\) with respect to a set of agents \(A_{x, f_i}\), we assume it has a set of values (i.e., discrete variable) or intervals (i.e., continuous variable), which is denoted by \(V(f_i)\). For each value \(v \in V(f_i)\), we denote the set of agents that are associated with \(v\) for feature \(f_i\) by \(A_{x, f_i, v}\). The information gain of feature \(f_i\) is then calculated by:

\[
IGain(A_{x, f_i}, f_i) = \text{Entropy}(A_{x, f_i}) - \sum_{v \in V(f_i)} \frac{|A_{x, f_i, v}|}{|A_{x, f_i}|} \text{Entropy}(A_{x, f_i}) \quad (11)
\]

The information gain of a feature measures expected reduction in entropy by considering this feature. Clearly, the higher the information gain, the lower the corresponding entropy becomes. We can then select features based on their information gains. Several methods can be used. For instance, we can first rank the features by information gain (descending order) and choose the first \(K\) features, depending on the specific applications; or we can set a threshold \(\delta\) and select the features whose information gain is higher than \(\delta\). We will demonstrate in evaluation (see Section 6) how such feature selection scheme is applied in Epinions dataset for improving trust prediction accuracy.

\(^2\)Other criterion include gini index of diversity \(^5\), chi-square test \(^14\), etc.
\(^3\)An alternative approach to do so is to rely on machine learning techniques such as decision tree or learning discriminant analysis (refer to our recent works \(^{26, 25}\)).
3 Basic Stereotrust Model

We now present basic StereoTrust model, which only uses agent’s local information to derive another agent’s trustworthiness.

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 features 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, in order to form appropriate groups/stereotypes, StereoTrust first performs feature combination and feature selection, if necessary. Using the processed features, StereoTrust groups agents accordingly. Note that StereoTrust considers only groups for which the membership is certain. We denote these groups by $G_{x,y} = \{G_{i,x,y}^1, G_{i,x,y}^2, \ldots\}$ such that $M_x(G_{i,x,y}, a_y) = 1$ (for the sake of simplifying the notation, we will use $G_i$ in place of $G_{i,x,y}$ 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 ($A_x = \{a_1, a_2, \ldots\}$), $a_x$ 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 $G_x$ 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 agents that belong to $G^i$ with certainty. Thus, from the set of all agents $a_x$ has previously interacted with ($A_x = \{a_1, a_2, \ldots\}$), $a_x$ 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.

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

probability density of trust rating $TD_{a_x,a_y}(p)$ using Eq. (12):

$$TD_{a_x,a_y}(p) = \sum_i W_{x,y}^i \cdot T_{a_x,G^i}(p|S_{a_x,G^i}, U_{a_x,G^i}), \quad (13)$$

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.

$$TD_{a_x,a_y}(p) = T_{a_x,a_y}(p) \sum_i W_{x,y}^i \cdot S_{a_x,G^i} \sum_i W_{x,y}^i \cdot U_{a_x,G^i}, \quad (14)$$

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

**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 subgroups, an \emph{honest} $G^{i,h}$ and a \emph{dishonest} $G^{i,d}$ subgroup (hence dichotomy-based). $a_x$ assigns an agent $a \in G^i$ to either subgroup 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 subgroup $G^{i,h}$ (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 subgroups ($G^{i,h}, G^{i,d}$) and determining $a_x$’s trust towards the subgroups (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 $a_y$’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 assign $a_y$ to a group (similarly to any other agent $a \in A_x$), 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,
Figure 3: Process of trust calculation. Trusts of honest subgroup $G_i^h$ and dishonest subgroup $G_i^d$ of each group $G_i$ are firstly combined using closeness and then trusts of all groups $G_i$ are combined using weight factor $W_{x,y}$ to derive target agent’s trust.

$a_x$ 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:

$$M_x(G_i^h, a_y) = \frac{1/|m_y - m_h|}{1/|m_y - m_h| + 1/|m_y - m_d|}$$ (15)

$$M_x(G_i^d, a_y) = \frac{1/|m_y - m_d|}{1/|m_y - m_h| + 1/|m_y - m_d|}$$ (16)

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. (12) and (15) we have probability density of target agent (a_y)’s trust rating $TD_{a_x,a_y}(x)$:

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

(17)

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. (12) and (15) we have probability density of agent a_y’s trust rating $TD_{a_x,a_y}(x)$:

$$TD_{a_x,a_y}(p) = T_{a_x,a_y}(p) \sum_i W_{x,y} \cdot (M_x(G^{i,h},a_y) \cdot S_{a_x,G^{i,h}} + M_x(G^{i,d},a_y) \cdot S_{a_x,G^{i,d}}) +$$

$$\sum_i W_{x,y} \cdot (M_x(G^{i,h},a_y) \cdot U_{a_x,G^{i,h}} + M_x(G^{i,d},a_y) \cdot U_{a_x,G^{i,d}}),$$

(18)

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.

### 5 Trustworthy sharing of local knowledge

StereoTrust works under the assumption that the trustor has sufficient local knowledge such that appropriate stereotypes can be efficiently formed. However, in some cases, agents’ local knowledge may be very limited (e.g., agents who have recently joined the system or who do not interact with others frequently). To make StereoTrust adaptive to such a scenario, we propose to let agents share
their local knowledge to help inexperienced agents estimate trustworthiness of a stranger. We first investigate what stereotypes are accurate to be shared in Sec. 5.1. Then we present how the stereotypes sharing overlay network (SSON) is constructed in Section 5.2. The issues of external stereotypes combination, and SSON update are discussed in Section 5.2.1 and 5.2.2 respectively.

5.1 Accurate stereotypes

In StereoTrust, a trustor’s stereotype on one group is formed by aggregating individuals’ trusts (i.e., based on past experience) in this group. This raises an interesting question: what stereotypes are accurate enough to be shared with other agents? For instance, an agent herself may have few interactions with agents from a specific group, and hence her stereotypes about that group may not be comprehensive to make good estimate about the group members’ behavior. Thus, even if the agent has no malicious intent, if she shares with others a stereotype which is in fact wrong or inconclusive, that will be detrimental to the system.

To address this issue, we need to investigate the minimum number of transactions used by a trustor to be confident about the collective behavior of the group. We treat a group of agents as an entity, so we model the numbers of successful and unsuccessful transactions between trustor and the group as the aggregated numbers of successful and unsuccessful transactions between trustor and members of that group. Based on Chernoff Bound theorem [17], the minimum number $M_{\text{min}}$ of transactions necessary in order to achieve a desired level of error with confidence may be determined as:

$$M_{\text{min}} \geq -\frac{1}{2\varepsilon^2} \ln(\delta/2)$$ (19)

Here $\varepsilon \in (0, 1)$ is the maximal level of error that can be accepted and $1 - \delta$ is the confidence measure. So for a trustor, if she has at least $M_{\text{min}}$ transactions with members of one group, she is confident about her current trust (i.e., stereotype) towards that group. Otherwise, she concludes that she does not have adequate information to determine the reliability of the specific stereotype, and hence should also not be sharing the same with others.

The above mechanism deals with mitigating the effect of inadvertently sharing misleading information by well behaved agents. Agents deliberately sharing misleading information, or agents with very different perspectives may be dealt with using conventional mechanisms, e.g., using a blacklist or a whitelist, etc., and we describe such a whitelist based strategy next.

5.2 Stereotypes sharing overlay network (SSON)

To evaluate trustworthiness of a target agent $a_y$, an inexperienced trustor $a_x$ who has few or no local knowledge needs to request other agents’ stereotypes. $a_x$ only requests the agents who are trustworthy in terms of providing correct stereotypes. That is, each agent $a_i$ in the system maintains a Trusted Stereotype
Provider list $\text{PROVIDER}(a_i)$ which stores the agents that this agent trusts in terms of providing correct local knowledge. A stereotype sharing overlay network (SSON), which is a virtual network on top of current network infrastructure (e.g., a P2P network) is thus constructed by connecting agents and their trusted stereotype providers. Fig. 4 depicts a SSON, which is represented by a directed graph. The graph nodes correspond to the agents. The directed edges represent the trust relationship in terms of providing correct stereotype between a pair of agents. For instance, agent $a_1$ trusts agent $a_2$ and the corresponding edge (trust relationship) is labeled with a trust score (i.e., 0.8 in this case). Trust score is determined and updated by the stereotype requester (denoted by $a_x$) after she evaluates correctness/usefulness of the information provided by the requestee. Note that since an agent may have very subjective perspective, even if requestee provides accurate stereotype, she may not be correct/useful to $a_x$. In this case, the requestee will be issued a low score although she is honest. In our work, the trust score falls into the range of $[0,1]$, where 1 represents completely trustworthy and 0 represents completely untrustworthy.

![Stereotypes sharing overlay network (SSON)](image)

Figure 4: A stereotypes sharing overlay network.

Initially, Trusted Stereotype Provider list $\text{PROVIDER}(a_i)$ is filled with $a_i$’s “familiar” agents (e.g., friends or colleagues in the real world, etc.). In case that no “familiar” agents are there, $a_i$ will choose stereotype providers randomly. After each request, the agent updates trust score of the corresponding stereotype provider (details will be described in Sec. 5.2.2). To collect correct stereotypes, $a_x$ would request from the top-K stereotype providers with highest trust scores in Trusted Stereotype Provider list, thus limiting communication overhead.

Notice that SSON is constructed to promote correct stereotypes sharing to help inexperienced agents estimate trustworthiness of an unknown service provider, but is not used to discover trustworthy transaction partner because it
has been reported in the literatures [27], [20] that the agents who provide high quality service may not necessarily report correct information and vice versa due to various reasons. In the scenario that the agents who provide correct stereotypes also act honestly in a transaction, SSON can be used to help promote successful transactions (i.e., select reliable agents who have high trust scores as the service providers). However, the goal of this work is to design mechanisms to estimate trustworthiness of an unknown service provider (i.e., no historical information is available), so discussion on relying SSON to derive agent’s trust like traditional trust mechanisms (e.g., feedbacks aggregation [12], [2], [22], web of trust [1] [11], etc.) is out of the scope of this paper.

5.2.1 Combining external stereotypes

After collecting other agents’ stereotypes, we discuss how to combine these external stereotypes to derive trust of the unknown target agent. We adopt a simple weight based strategy to combines the stereotypes. That is, the final trust score is derived as the weighted sum of the stereotypes. The weight for each stereotype depends on the trust (in terms of providing correct stereotype) of the corresponding stereotype provider. We denote the trust scores of the stereotype providers by $T = \{t_1, t_2, t_3, \ldots\}$. So the weight of stereotype provided by agent $a_i$ is calculated as $W_i = \frac{1}{\sum_j t_j}$.

There are several methods to combine stereotypes with weights. For instance, one can first calculate weighted sum of stereotypes and then derive trust score using the combined stereotype. Or one can first calculate trusts using individual stereotypes and then obtain the final trust by combining the trusts with corresponding weights. In Sec. 6 we will provide a running example illustrating how stereotypes are combined to derive trust.

5.2.2 Updating the stereotypes sharing overlay network

Since behaviors of the agents may change over time (e.g., honest stereotype provider may provide fake information or they may change their perspective on the system thus making the shared stereotypes not useful to the requester any more), it is essential to update trust scores of the stereotype providers such that the inaccurate/useless knowledge is not collected and less accurate/useful knowledge has less impact on the final decision.

It is effective if we update trust scores of the stereotype providers accordingly every time a transaction is completed. However, such strategy is inefficient because it incurs unnecessary computation overheads. We note that only trust scores of the stereotype providers whose local stereotypes produce contrary prediction on the outcome of the transaction need to be updated. So we adopt an update strategy, which is depicted by Algo. 1 to efficiently update trust scores of the stereotype providers. Note that the function MakingDecision($S_i$) returns the final decision on whether the encountered transaction is likely to be successful or not. Such decision is made based on the derived trust rating which relies on the stereotypes provided by agent $a_i$. 

16
Algorithm 1 Updating trust scores of stereotype providers (for trustor $a_x$)

for each stereotype provider $a_i \in PROVIDER(a_x)$, her trust score (in terms of providing correct stereotype) is $t_{x,i} = \frac{s_i + 1}{s_i + u_i + 2}$, where $s_i$ and $u_i$ represent times that $a_i$ provides correct and incorrect stereotypes respectively.

if The transaction $\Theta$ is unsuccessful then

for all agent $a_i \in PROVIDER(a_x)$ do

Decision $D \leftarrow$ MakingDecision($S_i$)

if $D$ is successful then

$t_{x,i} = \frac{s_i + 1}{s_i + u_i + 3}$

else

$t_{x,i} = \frac{s_i + 2}{s_i + u_i + 3}$

end if

end for

else

for all agent $a_i \in PROVIDER(a_x)$ do

$a_x$ evaluates correctness of $a_i$ with a certain probability

if $a_x$ evaluates correctness of $a_i$ then

Decision $D \leftarrow$ MakingDecision($S_i$)

if $D$ is successful then

$t_{x,i} = \frac{s_i + 2}{s_i + u_i + 3}$

else

$t_{x,i} = \frac{s_i + 1}{s_i + u_i + 3}$

end if

end if

end for

end if

After one transaction, if its outcome is unsuccessful, the stereotype requester $a_x$ evaluates correctness of the stereotypes provided by each stereotype provider. For stereotype provider $a_i$, $a_x$ estimates outcome of the transaction using stereotypes provided by $a_i$. If the transaction is estimated as successful, it means the corresponding stereotype is incorrect or useless to $a_i$ so trust score of $a_i$ must be lowered. Otherwise, her trust score should be increased. The trust score update follows Bayesian approach [12], which takes binary ratings as input and is based on computing trust scores by statistically updating beta probability density function (PDF). The posteriori trust score is computed by combining a priori trust score with new evidence. We denote the times that $a_i$ provides correct and incorrect stereotypes by $s_i$ and $u_i$ respectively (e.g., the original trust score $t_{x,i}$ is $\frac{s_i + 1}{s_i + u_i + 2}$). So $a_i$’s trust score is updated after one transaction using Eq. 20 (the provided stereotype is correct) and Eq. 21 (the

\[ \text{Since a bad transaction has greater impact on trustor compared to a good transaction, so we update trustor’s trusts in all stereotype provider, and do so to only a fraction of stereotype providers when the transaction is successful, which will be described later.} \]
provided stereotype is incorrect).

\[ t_{x,i} = \frac{s_i + 2}{s_i + u_i + 3} \quad (20) \]

\[ t_{x,i} = \frac{s_i + 1}{s_i + u_i + 3} \quad (21) \]

On the other hand, if the transaction is successful, \( a_x \) randomly evaluates correctness of a small fraction (e.g., 20\%) of the stereotypes. The process of update is similar to that described above. This can help to (1) avoid incurring too much unnecessary computation overheads because if the transaction is successful, it means combination of current external stereotypes is accurate enough to make a good prediction so no update is needed urgently and (2) proactively adjust weights of the third party knowledge thus making the combined knowledge more accurate (for the next prediction) while keeping the computation overheads low.

6 Evaluation

In this section, we conduct experiments to evaluate the performance of proposed StereoTrust models. We first discuss methodology in 6.1. In 6.2 and 6.3, we present our results that use Epinions dataset and synthetic dataset respectively.

6.1 Methodology

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 the target agent, she asks other agents across the network and aggregate feedbacks to derive target agent’s trust. Note that as trustor may not have experience with the probed agent, she cannot identify the dishonest agents, thus may suffer false feedbacks.

**EigenTrust.** EigenTrust [13] uses transitivity of trust and aggregates trust from peers by having them perform a distributed calculation to derive eigenvector of the trust matrix. Trustor first requests her 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 her friends’ friends, and so on, until the difference of two derived trusts in two subsequent iterations is smaller than a threshold (convergence). Pre-trusted 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, she asks her neighbors and her neighbors will ask their neighbors if they do not know target agent either. The trust graph is then formed by such trust relationships.

**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 the 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 she has previously interacted with into two groups: honest and dishonest (“honest agents” having more successful transactions 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.

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 5). 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 5), 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 9).

6.2 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.
6.2.1 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 relations. 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. Note that if there exist multiple such categories (i.e., stereotypes), in order to improve trust prediction accuracy, we only select the first three ones that have the highest information gains (see Section 2.4). 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.

6.2.2 Results

Figure 5 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 6 shows 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 5 and 6, 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 7 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
Figure 5: Comparison of StereoTrust model and ground truth using Epinions.com dataset.

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.

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>
<thead>
<tr>
<th></th>
<th>MAE</th>
<th>95% C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>StereoTrust (SOF)</td>
<td>0.1114</td>
<td>(0.1067,0.1161)</td>
</tr>
<tr>
<td>StereoTrust (SOP)</td>
<td>0.1177</td>
<td>(0.1136,0.1218)</td>
</tr>
<tr>
<td>d-StereoTrust (SOF)</td>
<td>0.0632</td>
<td>(0.0586,0.0678)</td>
</tr>
<tr>
<td>d-StereoTrust (SOP)</td>
<td>0.1299</td>
<td>(0.1245,0.1353)</td>
</tr>
<tr>
<td>dichotomy-only (SOF)</td>
<td>0.1365</td>
<td>(0.1307,0.1423)</td>
</tr>
<tr>
<td>dichotomy-only (SOP)</td>
<td>0.1750</td>
<td>(0.1690,0.1810)</td>
</tr>
<tr>
<td>group feedback aggrega</td>
<td>0.1452</td>
<td>(0.1386,0.1518)</td>
</tr>
</tbody>
</table>

6.2.3 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.

6.3 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.

6.3.1 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 has a value of $\lambda$, 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. Note that for a trustor, if there exist multiple such categories (i.e., stereotypes), in order to improve trust prediction accuracy, she only selects the first three ones that have the highest information gains (see Section 2.4).

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.

### 6.3.2 Results

We first demonstrate evaluation results without using stereotype sharing overlay network (SSON), i.e., the trustor has sufficient local knowledge. Figure 8 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 9 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 rep-
resent target agent’s trust. Similar to Epinions dataset, feedback derived trust (group feedback aggregation) is less accurate than that derived by SOF/SOP.

Figure 10 compares d-StereoTrust model (using SOF) with dichotomy-only (using SOF) and various existing algorithms (described in section 6.1). 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).

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.

We now present results of using SSON when no or few past interactions are available. To do so, we select 10 trustors with less than 5 interactions. When encountering a review, the inexperienced trustor requests trustworthy agents she knows for their stereotypes. The trustworthiness of stereotype provider is determined by accuracy of its past reports (refer to Alg. 1 for more detailed calculation). For stereotypes provided by each requested agent, trustor calculates corresponding trust values using StereoTrust model and then combines these trusts using trust scores of the stereotype providers as the weights.
Table 2: Mean Absolute Error and Coverage (for synthetic dataset)

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

Table 3 shows the performance of SSON. The average MAEs for honest agents, dishonest agents and all agents are 0.1277, 0.1211 and 0.1242 respectively. By comparing with results without SSON (Tab. 2), we notice that even if trustor does not have sufficient local knowledge, by requesting other agents’ stereotypes, it is still able to reasonably estimate trustworthiness of the potential interaction partner.

6.3.3 Discussion

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

![Figure 10: Comparison of all the algorithms using synthetic dataset.](image-url)
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.

When SSON is in use, the trustors without sufficient local knowledge can also predict trustworthiness of the unknown potential interaction partner by requesting other agents’ stereotypes. So the coverage of StereoTrust model becomes as high as 100% in this set of experiments. Since other agents may provide fake stereotypes maliciously, some of the collected stereotypes may not derive accurate trust, however, by updating trust scores of stereotype providers based on accuracy of their past reported stereotypes, the final aggregated stereotype information is still able to reasonably (comparing with other trust models) predict trustworthiness of the unknown agents.

7 Related Work

Past mutual interactions information can be used to predict an agent’s future behavior (e.g., [17]), 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 [1] and Jøsang et al [11] 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 [13] 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 [27]) 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 [19] 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 exist 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. [18] 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 [13]. 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 as well as machine learning techniques can be adopted to further improve StereoTrust’s performance. Some nascent attempts in this direction include our recent works [26, 23].

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 [6], 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.

8 Conclusion

We consider the problem of predicting trustworthiness of 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 her 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. Moreover, in case trustor does not have sufficient local knowledge, i.e., stereotypes cannot be formed, we propose a stereotype sharing overlay network in which inexperienced trustor can request stereotypes from other agents.

StereoTrust cannot 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. Another direction is to design incentive mechanisms to promote agents to share their accurate stereotype information.

9 Acknowledgement

We will like to thank Ee-Peng Lim for his inputs in the early phases of this work. The presented research is funded by A*Star SERC Grant number 072 134 0055.

References


