Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems

School of Information Systems

7-2011

Trust network inference for online rating data using generative models

Freddy Tat Chua CHUA

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

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

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

Citation

Trust Network Inference for Online Rating Data Using Generative Models

Freddy Chong Tat Chua
School of Information Systems
Singapore Management University
freddy.chua.2009@smu.edu.sg

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

ABSTRACT

In an online rating system, raters assign ratings to objects contributed by other users. In addition, raters can develop trust and distrust on object contributors depending on a few rating and trust related factors. Previous study has shown that ratings and trust links can influence each other but there has been a lack of a formal model to relate these factors together. In this paper, we therefore propose Trust Antecedent Factor (TAF) Model, a novel probabilistic model that generate ratings based on a number of rater's and contributor's factors. We demonstrate that parameters of the model can be learnt by Collapsed Gibbs Sampling. We then apply the model to predict trust and distrust between raters and review contributors using a real data-set. Our experiments have shown that the proposed model is capable of predicting both trust and distrust in a unified way. The model can also determine user factors which otherwise cannot be observed from the rating and trust data.

Categories and Subject Descriptors

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

General Terms

Algorithms, Measurement, Experimentation

Keywords

probability, statistics, trust prediction, social network

1. INTRODUCTION

1.1 Motivation

The popularity of Web 2.0 has transformed many users from passive consumers to active contributors. Web users today devote much more time and effort creating content objects including photos, videos, blogs, and reviews so as to share them with other users. While this brings about unprecedented quantity and richness of user generated content,

finding interesting and good quality content objects in Web 2.0 has also become increasingly challenging. To address the content quality issue, many Web 2.0 sites including Youtube, Amazon, and Flickr allow users to rate content objects contributed by other users. For example, Youtube users may give 5-star ratings to videos they like, or 1-star ratings otherwise. Ratings on the same video therefore can be used to determine its overall quality. Beyond ratings, some sites also introduce Web of Trust which allows users to express whom they trust (or like) or distrust (or dislike). A trust link from a user to another can therefore be labeled trust or distrust. In this way, users can choose to view content from users they like and are familiar with, blocking or ignoring content from disliked users.

Both ratings and Web of Trust are vital information for personalized/social search and recommendation applications. Ratings are exploited in collaborative filtering where inter-user similarities determined by common rating patterns are used to predict personalized ratings that have not yet been given [11]. Collaborative filtering has also been extended to incorporate trust links among users such that the ratings given by trusted users are given more weights in computing personalized ratings than the non-trusted ones [19]. Experiments have shown that the predicted ratings by collaborative filtering using trust are more accurate than not using trust when the ratings on the objects do not converge [7].

Just as when researchers attempt to predict ratings using trust information, it is equally interesting to predict trust using rating data. Trust prediction refers to predicting the existence of trust link between a given user pair. Unlike rating prediction which is used in content object recommendation, trust prediction is useful in friendship recommendation. In any Web 2.0 site, getting all users to express their trust (or distrust) links explicitly so as to construct a web of trust can be non-trivial due to (a) a lack of diligence on the users' part, or (b) unwillingness of users to share trust links publicly. In order to get users better connected by trust links so as derive a more tightly knitted user network, trust prediction is necessary.

Trust prediction solution approaches can be unsupervised and supervised. A survey of methods using these two approaches will be given in Section 2. The unsupervised approach first assigns each user pair to be predicted a score based on some criteria (e.g., number of common neighbors) so as to rank them from most likely trust pair to least likely trust pair. The top ranked trust pairs are thus returned as the prediction results. The supervised approach defines for each user pair a set of features and trains some classifier us-

ing known trust pairs. The unseen user pairs can then be classified by the trained classifier.

While several trust prediction methods have been developed based on the two approaches, most of them suffer from some common pitfalls listed below.

- Most of the existing works make very simple assumptions about trust formation and ignore well studied trust antecedent framework developed in the management and social science domain [21]. By not considering the antecedent factors of trust links, trust prediction methods will not be able to identify the right features and their dependencies in trust prediction. This may then affect the prediction accuracy and restrict the applicability of these methods to specific Web 2.0 sites. Furthermore, one would not be able to explain trust links using these factors.

- Most existing methods consider prediction of trust only or distrust only, but not both. When trust links are to be predicted, these methods treat distrust and no-trust under the same category and assign them the same label although they are conceptually different. Guha and Kumar pointed out that distrust does not propagate transitively like trust. Furthermore, they also found out that one-step distrust propagation patterns can be used for distrust prediction well [10].

- We know that not all trust links can be observed in the data as many users may not have expressed them. Hence, we have an incompletely labeled data-set issue. Most trust prediction methods based on supervised approach however ignore this by assuming all user pairs with unobserved trust links do not have trust links.

- Most existing methods are not developed based on formal probabilistic framework. As a consequence, it is not easy to associate confidence with the prediction scores for a given user pair.

1.2 Objectives and Contributions

This paper proposes to predict both trust and distrust in Web 2.0 sites that support both online rating and trust expressions. We model a Web 2.0 site as a network with raters, contributors and content objects as shown in Figure 1. The network allows raters to trust/distrust contributors, raters to rate objects from contributors, and contributors to upload objects. There are non-raters trusting other users. For example, a user may express trust on another user after reading the latter's contributed content. Such cases are not easily tracked since most sites do not track viewing activities. We therefore exclude such users from taking trustor roles. We also exclude non-contributors from being trusted as such users may have gained trust from another user not based on rating activities.

Given the rating and trust network in Figure 1, we aim to model the antecedent factors of trust as latent (or unobserved) variables and develop probabilistic graphical models [1] for predicting trust and distrust links. Our proposed model, **Trust Antecedent Factor (TAF) Model** overcome the pitfalls of existing methods mentioned in Section 1.1 by adopting the **Trust Antecedent Framework** well studied in the management and social science discipline [21].
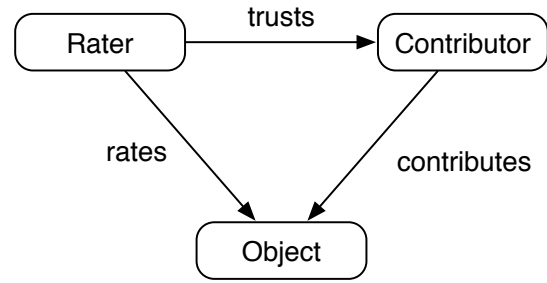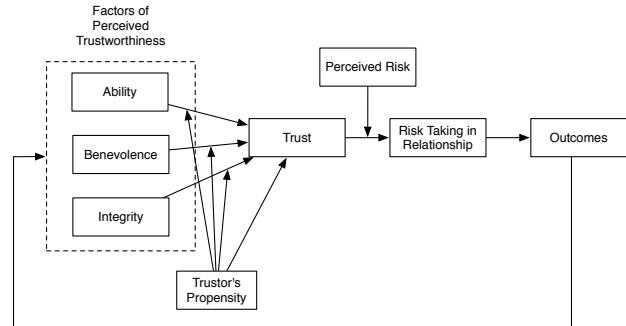

Figure 1: Rating and Trust Data

Figure 2: Trust Antecedent Framework

The framework essentially outlines trustee and trustor factors that influence the formation of trust links. Refer to Figure 2 for an overview of the Trust Antecedent Framework from Mayer [21]. The trustee factors are:

- *Ability*: Trustee has skills and competence to deliver desired outcome for the trustor;

- *Benevolence*: Trustee wants to do good with the trustor; and

- *Integrity*: Trustee adheres to a set of good moral principles.

The only trustor factor is *trust propensity* that refers to how easy a trustor trusts someone. The Trust Antecedent Framework says that a person trusts another person if the latter has high ability, benevolence and integrity and the former has the propensity to trust. In this paper, we will attempt to model the ability and propensity factors as latent variables of contributors and raters respectively in our probabilistic model, and trust links as semi-latent variables dependent on them. Both the benevolence and integrity factors are left out as it is not obvious that they can be characterized in this context. In our TAF model, ratings are represented as observable variables. Our proposed model also introduces other rater variables to model some rating related factors. Both trust and distrust are modeled simultaneously by representing the trust value of {0} as distrust and {1} as trust.

In Figure 2, the effects of trusting someone leads to events that gives feedback loop to the trustee factors, we will ignore them in this paper. The reason is because in online networks, most of the content is free for anyone to view, thus, making it difficult to assess the cost and risk involve

in trusting someone. Social networks that involve monetary transactions will need to account for such risks. However, such data is hard to obtain for our research.

In the following, we summarize our contributions:

- We introduce a novel probabilistic graphical model for trust prediction incorporating the well known trust antecedent framework. The proposed model incorporates important factors about raters and contributors, and uses them to derive trust probability capturing the confidence of trust. The model nicely bridges the qualitative research in management and social science with quantitative statistical learning.

- Our proposed model TAF Model uses simple assumptions to perform the task of inferring the latent trust network. To the best of our knowledge, this is the first probabilistic model that infers trust links from explicit user interactions.

- We tested our proposed model on a publicly available trust and rating data-set from Epinions. This data set was also used by Massa and Avesani [18]. We evaluated our proposed model on the Epinions data-set using the naive model as baseline for comparison and showed that we can perform trust prediction from users to users. More generally, we are able to uncover the latent trust network from the explicit user interactions.

Because the amount of data for inference and prediction is often inconsistent for different users, our model allows us to be confident about users' preferences and behavior when we observe more data and give us less confidence when we have less data for prediction. We can measure such confidence by evaluating the variance of probabilistic distributions.

2. RELATED WORK

A recent work prior to ours is Jamali's TrustWalker [12]. TrustWalker augments the collaborative filtering objective using an additional trust network. TrustWalker predicts user ratings given a set of ratings and the trust network.

Kenegis analyzed the Slashdot social network with dual polarity edges (i.e., trust and distrust edges) [13]. Kenegis used traditional social network analysis measures such as centrality measures, clustering coefficient, distance measures and similarity measures in signed social networks to predict links between users.

Massa proposed the notions of a global trust and a local trust metrics[18]. A global trust score is assigned to each user based on the user's contribution to the community as a whole while the local trust metric accounts for personal controversial opinions.

Golbeck introduced TidalTrust that performs a breath first search in the trust network from the source to all other users. The trust value between two users is then computed based on the aggregated ratings of these users [8]. Kuter and Golbeck also attempted the usage of Bayesian Network for trust inference in Sunny [14]. Kuter has showed that Sunny is able to give more accurate estimates of trust prediction as compared to TidalTrust. Yutaka also proposed a trust measure claiming that trust and ratings have bidirectional effects [20].

The trust prediction can also be seen as an instance of the link prediction problem in social networks. Liben-Nowell gave a comprehensive coverage of various link prediction methods [16]. These link prediction methods take advantage of the topological structure of social networks.

Yue Lu proposed a model based approach for review quality prediction [17]. The model is then formulated as a regression problem and solved using gradient descent methods. While their work focused on exploiting trust relationships for inferring review quality, our work focus on review ratings for inferring trust relationships.

Within the data mining and computer science community, the work by Leskovec is most similar to our work here [15]. Leskovec draws inspiration from the social science discipline based on the theory of balance and status. The social theory is given a more quantitative explanation using a logistic regression model. Leskovec also advocates that there are latent links between pairs of users which is not explicitly expressed.

All of these related works have not considered the dual polarity of trust networks and its dependency on ratings. We present a more principled and detailed version of trust inference using Bayesian Networks. Such models have shown significant impact in the area of text clustering [2] and is easily extensible by adding more latent variables with parametric distributions [4].

3. PRELIMINARIES

As our proposed trust prediction probabilistic graphical model requires the use of distribution functions with hyperparameters to model variable parameters, we first briefly introduce the concept of hyper-parameters and how they can be learnt. We then introduce the common variable parameters to be used in our proposed model and relate them to parameter learning.

3.1 Parameter Learning

Suppose we have a coin that can land either heads denoted by $\{1\}$ or tails denoted by $\{0\}$. We let $x_i \in \{0, 1\}$ be the random variable to denote the ith outcome of tossing the coin and X represent the set $\{x_0, x_1, \ldots, x_i, \ldots, x_{I-1}\}$. We use a variable parameter f to represent our prior belief of the coin landing heads. We further assume that f follows a symmetric Beta Distribution with θ as hyper-parameter. Formally, the probability of any f value ($P(f|\theta)$), the expected value of f ($E(f|\theta)$) and the probability of getting a toss outcome x_i given f ($P(x_i|f)$) are expressed as follows:

$$f \sim Beta(\theta)$$
$$P(f|\theta) \propto f^{\theta-1}(1-f)^{\theta-1}$$
$$P(x_i|f) = f^{x_i}(1-f)^{1-x_i}$$
$$P(x_i|\theta) = \frac{\theta}{2\theta}$$

In probabilistic graphical modeling, we use graphical plate notation to describe a set of probabilistic equations. For the above coin tossing example, the graphical plate notation is shown in Figure 3. In the figure, the left side shows the expanded graphical model where each event x_i is dependent on the parameter f and parameter f is dependent on its own hyper-parameters θ. The right side of Figure 3 is the compact equivalent of the expanded graphical model where the rectangle with symbol I surrounding the shaded variable x, compresses X. The shaded circles represent observable variables and unshaded circles represent unobservable or latent

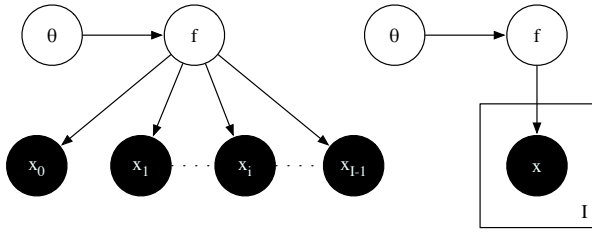variables. Inferring the values of unobservable variables is the main objective of parameter learning.

Figure 3: Coin Toss Plate Notation

Suppose we throw the coin I times and observe the outcomes X. Let H be the number of heads observed and T be the number of tails observed. After I throws, we can make a guess on the $I + 1^{th}$ outcome as follows,

$$P(x_{I+1}|x_1,\ldots,x_I,\theta) = \int P(x_{I+1},f|x_1,\ldots,x_I,\theta)\ df$$

$$= \int P(x_{I+1}|f)P(f|x_1,\ldots,x_I,\theta)\ df$$

$$\propto \int P(x_{I+1}|f)P(f|\theta)\prod_{i=0}^{I-1}P(x_i|f)\ df$$

$$\propto \int f\ f^{\theta+H-1}(1-f)^{\theta+T-1}\ df$$

$$\propto \frac{H+\theta}{H+T+2\theta}$$

From the above equations, we show how the posterior belief can be derived from initial prior belief and observable data. Conveniently, the posterior belief has the same distribution family as the prior belief because the Beta Distribution is a conjugate prior to Bernoulli random variables. For a more detailed explanation of conjugate priors, readers may refer to [6].

3.2 Parameters in Our Proposed Model

In our proposed model, we hypothesize on the existence of unobservable directed trust links from raters to contributors. These directed links can be hidden because raters do not always update their Web of Trust. We claim that these links are present because the raters form an unexpressed opinion about the contributors when rating the contributors' objects. Therefore, these links possess two attributes; *trust value label* and *observability*.

From the trust antecedent framework, we derive contributor's ability and rater's propensity as two latent parameters to be learnt for each contributor and rater.

- The contributor's **ability** to attract trusts from raters. We denote the ability of contributor j by a_j.

- The rater's **propensity** to form trust or distrust links. The propensity of rater i is denoted by y_i.

We need another latent parameter to represent the tendency of a rater to express his or her trust links towards contributors. We call this the rater's **expressiveness** and denote the expressiveness of rater i by e_i. The expressiveness parameter e_i consists of two components, one for trust and another for distrust denoted by $e_{i|t=1}$ and $e_{i|t=0}$ respectively.

Finally, we introduce a latent parameter to describe a rater's tendency to give high or low ratings. We call this the rater's **stringency** in ratings and denote the stringency of rater i by b_i. Again, the stringency parameter has two components, one for trust and another for distrust denoted by $b_{i|t=1}$ and $b_{i|t=0}$ respectively.

Each of the above parameters is modeled as a distribution parameter. The expected ability a_j values of 1 and 0 represent full ability and no ability respectively. Similarly, the expected propensity y_i values of 1 and 0 represent full propensity to trust and no propensity respectively. The expected expressiveness $e_{i|t=1}$ (or $e_{i|t=0}$) values of 1 and 0 represent full expressiveness and non-expressiveness for trust links that are labeled with trust (or distrust) respectively. Finally, the expected stringency $b_{i|t=1}$ (or $b_{i|t=0}$) values of 1 and 0 represent highest stringency and least stringency involving trust links labeled with trust (or distrust) respectively.

Similar to the coin toss example in Section 3.1, we place prior distributions on the parameters to model their dependency with the observable ratings. These parameters are used to construct different probabilistic graphical models as described in the next section.

4. TRUST ANTECEDENT FACTOR MODEL

We begin with the introduction of our model, Trust Antecedent Factor (**TAF**) Model. TAF formulates the generative process of observed trust links and ratings based on the following dependency assumptions [22]:

1. *Contributor j's ability a_j and rater i's propensity y_i influence the rater's decision to trust or distrust the contributor $t_{i,j} \in \{0,1\}$. This causal dependency is directly extracted from the trust antecedent framework. Here, $t_{i,j} = 1$ represents trust, and $t_{i,j} = 0$ represents distrust from rater i to contributor j.*

2. *The rater-contributor trust link $t_{i,j}$ and rater's expressiveness e_i influence the **observability** of trust link denoted by $o_{i,j} \in \{0,1\}$. The value of observability $o_{i,j}$ equals 1 in the event $t_{i,j}$ is observable, and 0 otherwise. The observability variable is introduced to address the issue of unexpressed trust link issue. Instead of treating unexpressed trust and distrust the same, the observability variable allows us to model situations where a rater decides not to express them. We expect raters should be sensitive about whether to express trust links depending on whether the trust links are trust or distrust, and whether the raters are outspoken (or expressiveness). In particular, people are known to be careful in stating who they distrust in public.*

3. *The rater-contributor trust link $t_{i,j}$ and rater's stringency b_i influence the ratings a rater gives to a contributor's object k denoted by $r_{i,j,k} \in \{0,1,\cdots,m\}$. We believe that rating a familiar contributor's objects is different from rating those of a stranger. A stringent rater will also give smaller rating values.*

We now describe the generative process adopted by TAF model. Let $\mathbf{I}$ be the total number of raters, $\mathbf{J}$ denote the total number of contributors, and $\mathbf{K}_j$ be the total number of objects contributed by j. The generative process of TAF follows the steps below:

1. For each rater $i \in \mathbf{I}$, sample distribution parameters: propensity y_i, using Beta distribution with symmetric hyper-parameters

$$y_i \sim Beta(\tau)$$

2. For each contributor $j \in \mathbf{J}$, sample distribution parameters: ability a_j using Beta distribution with symmetric hyper-parameters

$$a_j \sim Beta(\alpha)$$

3. For each rater i and contributor j,

   (a) Rater generates trust for contributor $t_{i,j}$ based on rater's propensity y_i and contributor's ability a_j.

   $$P(t_{i,j} = t|y_i, a_j) = y_i^t \, a_j^t (1 - y_i)^{1-t} (1 - a_j)^{1-t}$$

   (b) Rater sample distribution parameters: expressiveness $e_{i|t}$ and stringency $b_{i|t}$, using Beta distribution with symmetric hyper-parameters

   $$b_{i|t} \sim Beta(\beta)$$
   $$e_{i|t} \sim Beta(\epsilon)$$

   (c) Rater generates the observability of link $o_{i,j}$ based on rater's expressiveness $e_{i|t}$ and trust of the link $t_{i,j}$.

   $$P(o_{i,j} = o|t_{i,j} = t, e_{i|t}) = e_{i|t}^o (1 - e_{i|t})^{1-o}$$

   (d) For each object $k \in \mathbf{K}_j$, Rater generates rating $r_{i,j,k}$ for k based on

      i. rater's stringency $b_{i|t}$ and,
      ii. trust $t_{i,j}$ towards the contributor j of the object k,

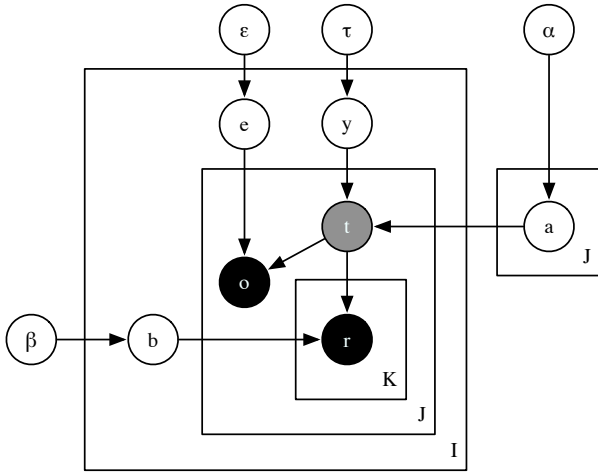 $$P(r_{i,j,k} = r|t_{i,j} = t, b_{i|t}) = \binom{m}{r} b_{i|t}^r (1 - b_{i|t})^{m-r}$$

Figure 4: Plate Notation for TAF Model

The above generative process can be represented in graphical plate notations as shown in Figure 4. The directed arrows in Figure 4 denote dependency assumptions. Variables in shaded circles (i.e., $o_{i,j}$'s and $r_{i,j,k}$'s) denote completely observed variables while those in grey circles (i.e., $t_{i,j}$) represent partially observable variables. The remaining variables are non-observable.

Note that the generative process requires a few hyper-parameters τ, ϵ, α, and β for the parametric probability distributions. Our modeling goal is to learn these hyper-parameters based on the observed data. After learning the parameters from observed data, we will be able to infer the values of future unobserved data. These hyper-parameters may be set as constants at the beginning of the learning process.

The trust link and observability are modeled as Bernoulli events since they have binary values. The rating has **m** states and we can choose to model the rating as a set of m multinomial variables. Multinomial distribution however assumes that each state is independent and identically distributed. This assumption unfortunately does not hold in practice. Instead, we divide an object into m parts, a rater will give a rating of m if the rater likes all m parts, a rater gives a rating of $m - 1$ if the rater likes all but one part, conversely, a rater gives a rating of zero if the rater does not like any parts. Hence, if we model each part as a Bernoulli event, then the rating distribution can be generalized to a Binomial distribution.

To further justify the use of binomial distribution for ratings, we will like to explain the relationship between binomial, poisson and gaussian distributions. The binomial distribution requires factorial calculations which may be infeasible when the number of states approaches infinity, i.e. $m \to \infty$. Poisson distribution is a discrete approximation for the binomial distribution when $m \to \infty$ and the probability of each state approaches zero. The gaussian distribution is a continuous approximation of binomial distribution when $m \to \infty$ but the probability of each state occurring is near to 0.5. Since object ratings are usually in the discrete range of $[0, 4]$, it is more appropriate to use binomial distributions.

Due to the presence of partially observed variable $t_{i,j}$, the joint probability cannot be factorized into separable components and the parameters cannot be solved analytically. To learn the parameters, we will use Collapsed Gibbs Sampling which has been used for parameter learning in topic models [9, 2].

4.1 Inference using Collapsed Gibbs Sampling

The key idea of Gibbs Sampling is to break up the Bayesian network into manageable chunks and sample each unobserved variable separately from the rest [3]. Each time we sample the unobserved variable, we make assumptions that the rest of the network is correctly sampled. Repeating the sampling process for multiple iterations will improve the accuracy of the initial assumptions made. Collapsed Gibbs Sampling improves the efficiency over Gibbs Sampling by marginalizing the parameters out of the joint distribution. We sample whenever we encounter an unobserved link $t_{i,j}$. The Gibbs sampling inference procedure can be written in this form $P(t_{i,j}|t_{i,-j}, o, r, \beta, \epsilon, \tau, \alpha)$ where $t_{i,-j}$ refers to the set of all t variables except for $t_{i,j}$,

$$
\begin{aligned}
&P(t_{i,j} = t|t_{i,-j}, o, r, \beta, \epsilon, \tau, \alpha) \\
&\propto P(o_{i,j}|t_{i,j} = t, o_{i,-j}, \epsilon) P(t_{i,j} = t|t_{i,-j}, \tau, \alpha) \\
&\quad P(r_{i,j}|t_{i,j} = t, r_{i,-j}, \beta)
\end{aligned}
$$

The three components are as follows,

$$P(o_{i,j}|t_{i,j} = t, o_{i,-j}, \epsilon) = \frac{n(o_i^0|t) + \epsilon}{n(o_i^0|t) + n(o_i^1|t) + 2\epsilon}$$

where $n(o_i^0|t)[n(o_i^1|t)]$ is the number of times i choose to hide [show] the trust given that the trust i to j is t. Due to the condition on $o_{i,-j}$, $n(o_i^0|t)$ and $n(o_i^1|t)$ must exclude the counts between i and j.

$$P(t_{i,j} = t|t_{i,-j}, \tau, \alpha)$$
$$= \left[\frac{n(t_i^t) + \tau}{n(t_i^0) + n(t_i^1) + 2\tau}\right] \left[\frac{n(t_j^t) + \alpha}{n(t_j^0) + n(t_j^1) + 2\alpha}\right]$$

where $n(t_i^t)$ is the number of links with value t from i and $n(t_j^t)$ is the number of links with value t to j. Due to $t_{i,-j}$, $n(t_i^t)$ and $n(t_j^t)$ must exclude the counts between i and j.

$$P(r_{i,j}|t_{i,j} = t, r_{i,-j}, \beta)$$
$$= \left[\Gamma\left(m.n(r_i^r|t) + 2\beta\right) \Gamma\left(\sum_{r=0}^{m} r.\left[n(r_i^r|t) + n(r_{i,j}^r|t)\right] + \beta\right) \right.$$
$$\left. \Gamma\left(\sum_{r=0}^{m}(m-r).\left[n(r_i^r|t) + n(r_{i,j}^r|t)\right] + \beta\right) \right]$$
$$\left[\Gamma\left(\sum_{r=0}^{m} r.n(r_i^r|t) + \beta\right) \Gamma\left(\sum_{r=0}^{m}(m-r).n(r_i^r|t) + \beta\right) \right.$$
$$\left. \Gamma\left(m.\left[n(r_i^r|t) + n(r_{i,j}^r|t)\right] + 2\beta\right) \right]^{-1}$$

where $n(r_i^r|t)$ are the ratings i has rated with value r given trust t, $n(r_{i,j}^r|t)$ are the ratings i has rated on j objects with values r given that the trust is t.

4.2 Parameter Learning

After inference on unobserved variables $t_{i,j}$'s, we update the parameters y, a, e, b. The derivation of parameter learning equations for y, a and e follows that of Section 3.1 and we leave them out due to space constraint. Instead, we state the equations for y, a, e and show the derivation for b. The updated posterior distributions of y, a and e parameters are:

$$P(y_i|t_i, \tau, \alpha) \sim Beta\left(\tau + n(t_i^1), \tau + n(t_i^0)\right)$$
$$P(a_j|t_j, \tau, \alpha) \sim Beta\left(\alpha + n(t_j^1), \alpha + n(t_j^0)\right)$$
$$P(e_{i|t}|t_i, o_i, \epsilon) \sim Beta\left(\epsilon + n(o_i^1|t), \epsilon + n(o_i^0|t)\right)$$

The updated posterior distribution of b_i's is derived as follows.

$$P(b_{i|t}|r_i, t_i, \beta) \propto P(b_{i|t}|\beta) \prod_{j,k} P(r_{i,j,k} = r|t_{i,j}, b_{i|t})$$
$$\propto b_{i|t}^{\beta-1}(1 - b_{i|t})^{\beta-1} \prod_{j,k} \prod_r \binom{m}{r} b_{i|t}^r (1 - b_{i|t})^{m-r}$$
$$\sim Beta\left(\sum_{r=0}^{m} r \ n(r_i^r|t) + \beta, \sum_{r=0}^{m}(m-r) \ n(r_i^r|t) + \beta\right)$$

4.3 Trust Prediction

After learning the parameters, we predict the trust between i and j, $t_{i,j}$ using its Markov Blanket [22]. The Markov Blanket is the minimum set of variables required

to infer the value of a variable. This minimum set is given by the variable's children, parents and children's parents. Hence, probability of $t_{i,j}$ given its Markov Blanket is:

$$P(t_{i,j} = t|o_{i,j} = o, r_{i,j}, y_i, e_i, b_i, a_j)$$
$$\propto P(o_{i,j} = o|t_{i,j} = t, e_i)P(t_{i,j} = t|y_i, a_j)P(r_{i,j}|t_{i,j} = t, b_i)$$

If we are not able to observe the ratings between i and j, we can use the law of total probability as follows,

$$P(t_{i,j}|o_{i,j}, y_i, e_i, b_i, a_j)$$
$$\propto \prod_k \sum_r P(t_{i,j} = t, r_{i,j,k} = r|o_{i,j}, y_i, e_i, b_i, a_j)$$
$$\propto P(o_{i,j}|t_{i,j}, e_i)P(t_{i,j}|y_i, a_j)$$

Algorithm 1 illustrates the relation of these three sections.

Algorithm 1 Inference and Learning

Randomly assign $\{0, 1\}$ to $t_{i,j} \ \forall \ i, j$
Count $n(t_i), n(t_j), n(o_i|t), n(r_i^r|t) \ \forall \ i, j$
while not converge **do**
  for $i \leftarrow 1$ to I **do**
    for $j \leftarrow 1$ to J **do**
      Subtract from $n(t_i), n(t_j), n(o_i|t), n(r_i^r|t)$ to exclude the results of $t_{i,j}$ from previous iterations.

      $t_{i,j} \leftarrow$ Perform Inference. (Section 4.1)

      Add to $n(t_i), n(t_j), n(o_i|t), n(r_i^r|t)$ to include the results of current iteration.
    end for
  end for
end while
Update the Parameters (Section 4.2).
Perform the Prediction (Section 4.3).

---

5. EXPERIMENTAL EVALUATION

In this section, we evaluate the the performance of our model using a real data set from Epinions [18]. We run two sets of experiments. The first set simulates the existence of ground truth by artificially hiding the true value of links. In the second set of experiments, we apply our model on the entire data-set with all available ratings and trust link labels. We examine the distribution of network properties before and after learning. We then show that the distribution after learning is a better reflection of the network.

5.1 Data Set

We used the Epinions data set from trustlet.org. Epinions is an online website that provides reviews of products. The reviews are meant to help other users acquire knowledge and opinions about product. These reviews are written by users and can be rated by other users. Users can choose to read and/or rate subsets of reviews by filtering. The Epinions system filters the reviews based on the users' Web of Trust. To apply our proposed models to the data set, we consider the users who write reviews as contributors and users who rate objects as raters.

To keep the computation manageable, we extract a subset of the Epinions network data set as follows. We first initialize a rater set and a contributor set to be empty. The rater with the largest number of rated contributors is first added

to the rater set. The rated contributors by this rater are added to the contributor set. From the set of contributors, we further add their raters to the rater set. The process repeats until we have exhausted all raters and contributors reachable from the initial rater. Finally, we have 85,132 raters, 75,294 contributors, and 4,042,345 rater-contributor pairs (i.e., the rater has rated some object(s) by contributor). Among these pairs, 467,047 have observable trust links while 3,575,298 have unobservable trust links. Among the observable trust links, 421,756 have trust label and 45,291 have distrust label. This data subset represents about 65% of all users, 89% of all rater-contributor pairs, and 70% of all trust links in the original Epinions data-set.

5.2 Trust Prediction Experiments

In these experiments, we evaluate trust prediction accuracy of our proposed model. We compare our model against a naive baseline model. We divide the constructed data set into five folds. In five-fold validation, we hide the trust link labels in one of the folds and predict the labels for these links using four other folds as training set. We measure the performance using precision-recall curves for distrust links prediction as the distrust link label is a smaller class.

5.2.1 Naive Baseline Model

As shown in Figure 5, the baseline model is a stripped down version of TAF. The generative process is as follows:

1. For each rater $i \in \mathbf{I}$, sample the trust propensity y_i using Beta distribution with symmetric hyper-parameters,

$$y_i \sim Beta(\tau)$$

2. For each rater i and contributor j,

   (a) Rater i generates trust for contributor $t_{i,j}$ based on rater's propensity y_i,

   $$P(t_{i,j} = t|y_i) = y_i^t(1-y_i)^{1-t}$$

   (b) Rater i sample stringency using Beta distribution with symmetric hyper-parameters,

   $$b_{i|t} \sim Beta(\beta)$$

   (c) For each object $k \in \mathbf{K_j}$, Rater generates rating $r_{i,j,k}$ for k based on

      i. rater's stringency $b_{i|t}$ and,

      ii. trust $t_{i,j}$ towards the contributor j of the object k,

   $$P(r_{i,j,k} = r|t_{i,j} = t, b_{i|t}) = \binom{m}{r} b_{i|t}^r (1-b_{i|t})^{m-r}$$

For inference,

$$P(t_{i,j} = t|t_{i,-j}, r, \tau, \beta) = P(t_{i,j} = t|t_{i,-j}, \tau)P(r_{i,j}|t_{i,j}, r_{i,-j}, \beta)$$

where

$$P(t_{i,j} = t|t_{i,-j}, \tau) = \frac{n(t_i^t) + \tau}{n(t_i^0) + n(t_i^1) + 2\tau}$$

The $P(r_{i,j}|t_{i,j}, r_{i,-j}$ is similar as before so we will not show in detail here. And for prediction,

$$P(t_{i,j}|r, y, b) = P(t_{i,j}|y_i)P(r_{i,j}|t_{i,j}, b_i)$$

Due to a lack of ground truth in our data, we do not explicitly show trust prediction when ratings are not available. But one may observe that the naive model only takes into account of rater's attributes and ignore contributor's attributes. When there is a lack of observable ratings between rater and contributor, the trust prediction of naive model will not work based on rater's attribute alone.
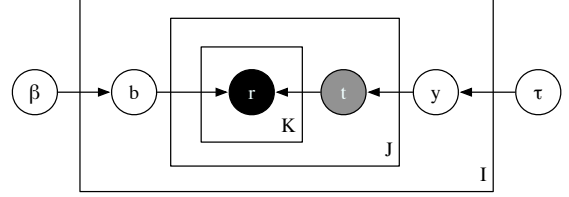

Figure 5: Naive Baseline Model

5.2.2 Simulating Missing Trust Links and Performance Metric

The objective of our model is to infer the unobservable trust links in the network. However, due to the lack of ground truth for unobservable trust links, we have to hide existing trust links to simulate missing ones. In this way, we are able to compare our prediction results with the trust labels of the artificially hidden links.

Since our trust links have two attributes, observability and trust label, we run two experiments. First experiment treats the trust label as missing information and observability as known information. Second experiment treats both trust label and observability as missing information.

We divide all the rater-contributor pairs into five data sets using simple round robin for pairs belonging to the same rater. Using five fold cross validation, we perform training on four sets and test on the remaining set. This process is performed for a total of five times by using different folds as test data. We present the precision-recall curve by taking average of the five-fold validation. The precision-recall curve measures the predictive performance for distrust links. We have chosen to leave out the prediction of trust label results because trust links with trust label represent a larger class making it easier to predict. Moreover, because of the skewed distribution in data, precision recall curves is a more suitable measure than receiver operating characteristic curves [5].

We define the precision, recall and F1 values of our prediction as follows,

$$Recall = \frac{TP}{TP + FN} \qquad Precision = \frac{TP}{TP + FP}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

where TP, FN and FP are numbers of true positives, false negatives and false positives respectively. We compute the F1 by taking the predicted distrust to be links where trust probability is less than $\frac{1}{2}$.

5.2.3 Results of Prediction Experiments

We show the results of distrust link prediction for TAF and Naive models in Figure 6 and Table 1. Under the condition that observability of trust links is known, the results
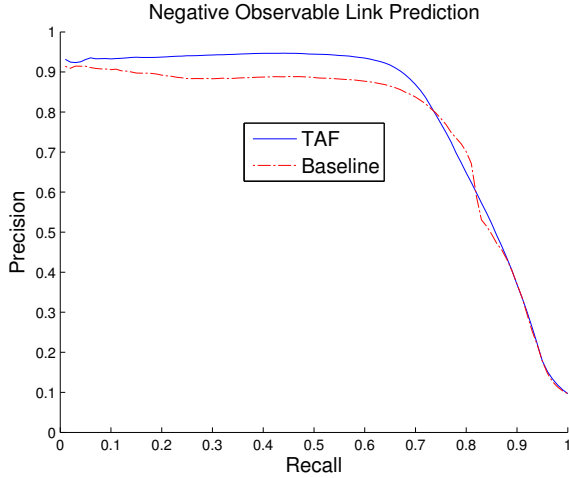
(a) Observable
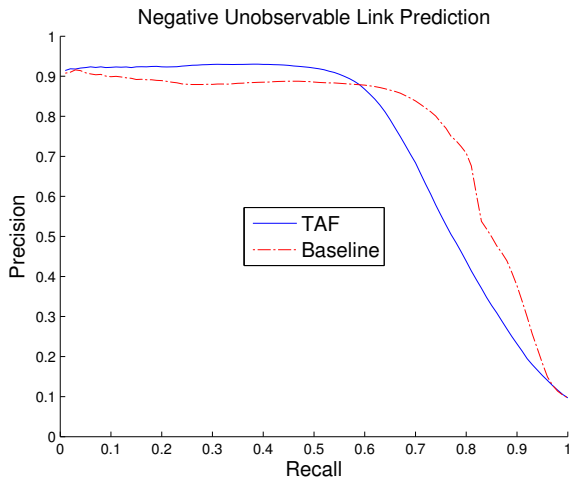
(b) Unobservable

Figure 6: Prediction for Distrust Links

clearly show that TAF is better than the Naive model as measured by AUC of PR curves and F1.

Under the condition that observability of trust links is unknown, TAF is still better than Naive model by AUC of PR curves. Nevertheless, the AUC result of TAF is poorer than that under the previous observability condition. This is expected because the true observability is unknown. The naive model surprisingly performs well under F1. We believe the use of 0.5 trust probability as the threshold classify trust and distrust is not ideal, and should be tuned further. We shall investigate this further in the future work.

Table 1: Area under PR Curve

	TAF	Baseline
AUC (observable)	0.804	0.708
AUC (unobservable)	0.742	0.708
F1 (observable)	0.780	0.739
F1 (unobservable)	0.711	0.739

5.3 Results of Data Modeling Experiments

In the second set of experiments, we perform trust inference on the entire data set as a single instance. We examine the distribution of trust and distrust links for each rater and contributor. Figure 7(a) shows the trust ratio distribution of contributors before inference. The trust ratio of a contributor j is computed as follows:

$$\frac{number\ of\ observable\ trust\ links\ to\ j}{number\ of\ observable\ links\ to\ j} \qquad (1)$$

Based on this definition, the distribution excludes trust links that are unobserved. We also deliberately exclude contributors with fewer than five observable links as they are insignificant cases. After inference, we are able to give a measure of our uncertainty about the unobservable trust for each link. Hence, the trust distribution of each contributor j after inference is given by,

$$\frac{1}{n}\sum_i^n E(t_{i,j}) \quad n\ is\ the\ number\ of\ links\ to\ j \qquad (2)$$

where $E(t_{i,j})$ represents the expected trust rater i has for contributor j. Since our trust $t_{i,j}$ follows a bernoulli distribution, the expected value is also the probability $P(t_{i,j}=1)$.

We present the trust distribution in Figure 7(b). The distribution as shown is much smoother than the distribution in Figure 7(a). This indicates that we recover much of the missing information in the unobservable links.
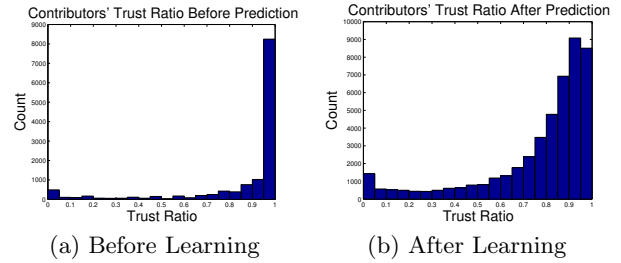

(a) Before Learning   (b) After Learning

Figure 7: Distribution of Contributors' Trust Ratio

Figure 8 shows the distribution of raters who express their trust links. The expressiveness of a rater i is calculated as follows,

$$expressiveness = \frac{number\ of\ observable\ links\ from\ i}{total\ number\ of\ links\ from\ i}$$

The distribution result suggests that most raters do not express their trust links. The number of raters who actively update their trust network is a minority. Therefore, our TAF model increases the number of trust links to generate a more complete network for most users. In other words, we generate estimated missing values for sparse data.

We will now examine the distribution of trust links for raters. Figure 9(a) shows the distribution of trust ratio for each rater before learning. The definition of rater's trust ratio is similar to Equation 1 except that the link count is rater specific. After learning, the trust ratio of a rater is computed as:

$$\frac{1}{n}\sum_j^n E(t_{i,j}) \quad n\ is\ the\ number\ of\ links\ to\ j \qquad (3)$$

Equation 3 sums over j unlike the earlier Equation 2 which sums over i. The distribution of trust ratios is shown in Figure 9(b). Similar to contributors' trust ratios, the after learning distribution is much smoother than that before learning. This again demonstrates the TAF model's ability to recover the missing trust links.
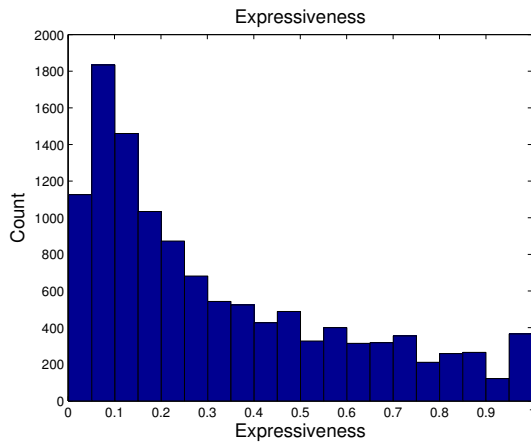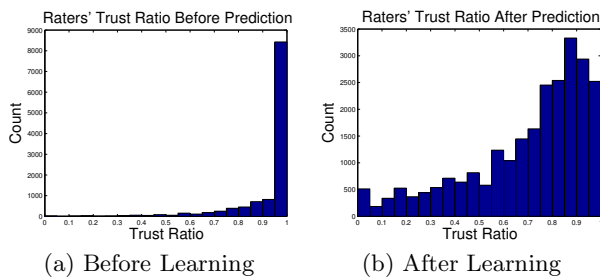

Figure 8: Distribution of Expressiveness

(a) Before Learning   (b) After Learning

Figure 9: Distribution of Raters' Trust Ratio

6. CONCLUSION

We have introduced the Trust Antecedent Factor (TAF) model that captures the relationships between trust and ratings in online rating and trust networks. We have only considered ratings as observable interaction between users and trust as partially observable variables. Our proposed model derives other trust related variables and their dependency relationships from a well established trust antecedent framework. They can predict both trust and distrust in a unified way, and are capable of modeling the observability of trust links. We have evaluated the model using real data-set and showed that the model can predict artificially hidden observable trust links with good accuracy compared with Naive model and reveal the unobservable trust links.

For the future work, we can extend the TAF model to handle data-sets with other types of user interaction data such as message activities, clicks on user's contributed objects and time spent viewing contributed objects. Exploring other parametric distribution functions for modeling latent variables can also be an interesting future research.

7. REFERENCES

[1] C. M. Bishop. *Latent variable models*. MIT Press, 1999.

[2] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine learning Research*, 2003.

[3] G. Casella and E. I. George. Explaining the gibbs sampler. *The American Statistician*, 46(3):167–174, 1992.

[4] J. Chang, J. Boyd-Graber, and D. M. Blei. Connections between the lines: augmenting social networks with text. In *KDD*, 2009.

[5] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML*, 2006.

[6] P. Diaconis and D. Ylvisaker. Conjugate priors for exponential families. *The Annals of Statistics*, 7(2):269–281, 1979.

[7] J. Golbeck. Generating predictive movie recommendations from trust in social networks. In *iTrust*, 2006.

[8] J. A. Golbeck. *Computing and applying trust in web-based social networks*. PhD thesis, 2005.

[9] T. L. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101, 2004.

[10] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *WWW*, 2004.

[11] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM TOIS*, 22(1), 2004.

[12] M. Jamali and M. Ester. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *KDD*, 2009.

[13] J. Kunegis, A. Lommatzsch, and C. Bauckhage. The slashdot zoo: mining a social network with negative edges. In *WWW*, 2009.

[14] U. Kuter and J. Golbeck. Sunny: a new algorithm for trust inference in social networks using probabilistic confidence models. In *AAAI*, 2007.

[15] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW*, 2010.

[16] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[17] Y. Lu, P. Tsaparas, A. Ntoulas, and L. Polanyi. Exploiting social context for review quality prediction. In *WWW*, 2010.

[18] P. Massa and P. Avesani. Controversial users demand local trust metrics: an experimental study on epinions.com community. In *AAAI*, 2005.

[19] P. Massa and P. Avesani. Trust-aware recommender systems. In *ACM RecSys*, 2007.

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

[21] R. C. Mayer, J. H. Davis, and D. F. Schoorman. An integrative model of organizational trust. *The Academy of Management Review*, 20(3), 1995.

[22] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., 1988.