STAR: Semiring Trust Inference for Trust-Aware Social Recommenders

Peixin Gao
Institute for Systems Research
University of Maryland
gaopei@umd.edu

Hui Miao
Dept. of Computer Science
University of Maryland
hui@cs.umd.edu

John S. Baras
Institute for Systems Research
University of Maryland
baras@umd.edu

Jennifer Golbeck
College of Information Studies
University of Maryland
jgolbeck@umd.edu

ABSTRACT

Social recommendation takes advantage of the influence of social relationships in decision making and the ready availability of social data through social networking systems. Trust relationships in particular can be exploited in such systems for rating prediction and recommendation, which has been shown to have the potential for improving the quality of the recommender and alleviating the issue of data sparsity, cold start, and adversarial attacks. An appropriate trust inference mechanism is necessary in extending the knowledge base of trust opinions and tackling the issue of limited trust information due to connection sparsity of social networks.

In this work, we offer a new solution to trust inference in social networks to provide a better knowledge base for trust-aware recommender systems. We propose using a semiring framework as a nonlinear way to combine trust evidences for inferring trust, where trust relationship is model as 2-D vector containing both trust and certainty information. The trust propagation and aggregation rules, as the building blocks of our trust inference scheme, are based upon the properties of trust relationships. In our approach, both trust and distrust (i.e., positive and negative trust) are considered, and opinion conflict resolution is supported. We evaluate the proposed approach on real-world datasets, and show that our trust inference framework has high accuracy, and is capable of handling trust relationship in large networks. The inferred trust relationships can enlarge the knowledge base for trust information and improve the quality of trust-aware recommendation.

1. INTRODUCTION

Recommender systems are continuously evolving to take advantage of new sources of information. Social sources in particular can be useful for building better recommendations. The idea of integrating signals from social networks to improve the performance of recommendation algorithms has been well accepted and has attracted an increasing amount of research [2, 28].

One type of social recommender system is the trust-aware recommender, where the social connections are annotated with trust relationship information in the recommendation algorithm. A series of trust-aware recommenders have been proposed, such as user-based approaches [12,16,25], and matrix factorization based methods [17,22,24]. With the help of the additional trust information, trust-aware recommenders have been shown useful for improving recommendation quality in terms of accuracy and user experience, as well as addressing challenges of classical recommender system approaches such as cold start and adversarial attacks [12, 23].

However, trust-aware recommender systems face several challenges: a) sparse social connections; b) inconsistency and conflicts in trust opinions; c) trust data availability.

To address these challenges, there has been a line of study regarding trust inference in trust-aware recommender systems, which can generally be categorized in two groups. One group conducts theoretical studies on trust metrics and their properties [3, 14, 18, 19, 30]. The other group proposes effective data-driven approaches for inferring trust, such as graph-theoretic models [4, 10–12, 25, 27] and machine learning approaches [20].

Graph-theoretic models generally have good performance in terms of efficiency and reasoning. However, as most work applies linear weighted averaging over trust opinions from neighbors, nonlinearity in human decision making and interaction between trust and distrust may not be fully captured. The path-based design is also sensitive to the connection sparsity in the trust network, since the trust path will not exist if some edges in between are missing. These defects restrain the accuracy of the models. On the other hand, machine learning methods can solve the data sparsity, sometimes with better label prediction accuracy. However, due to the nature of the formulation, machine learning methods are slower in speed and have weak interpretation on how trust is inferred.

It is ideal if there exist a method which addresses most challenges at the same time. In [9], we introduced a theoretical semiring structure for nonlinear trust fusion with distrust considered. It requires that both trust and certainty information are explicitly given. The model proposed

RecSys '16, September 15-19, 2016, Boston , MA, USA
 ISBN 978-1-4503-4035-9/16/09…$15.00
DOI: http://dx.doi.org/10.1145/2959100.2959148

in [9] is easy to interpret and runs fast. However,it cannot be applied in real world scenarios, due to lack of certainty models and inference algorithms on the overall network dealing with data sparsities. In order to apply it to trust evaluation for trust-aware recommendation, the original design needs revision and redefinition.

In this work, we propose a trust inference method for trust-aware recommendation. Our technology extends the semiring model proposed in [9]. The basic idea is to build up a nonlinear trust aggregation rule that can handle both trust and distrust information, and offer conflict resolution in the opinion combination process. Apart from the flexibility in modeling, this approach has the advantages of efficiency inherited from graph theoretic models and better accuracy due to an iterative component and optimization over parameters. The evaluated trust can enlarge the knowledge base for recommendation algorithms, and alleviate the issue of the data sparsity and cold start in recommender systems.

The contributions of our work are four-fold:

1. a novel trust metric for trust-aware recommender systems is introduced
2. we introduce two certainty models for semiring-based trust evaluation in our application, and discuss several properties of the evaluation framework
3. we introduce trust iteration, as well as partial reciprocity, in order to further improve the coverage and accuracy of our proposed trust metric
4. we evaluate our model using real-world dataset, and show that our approach can achieve the accuracy to about 95%

Our paper is organized as follows. In Sec. 2, we discuss some preliminaries and related work. Accordingly, we introduce STAR, our semiring nonlinear model for trust inference in a social recommender scenario in Sec. 3, with some extensions for better performance. We evaluate our proposed approach in Sec. 4, and summarize our work in Sec. 5.

2. PRELIMINARIES

2.1 Trust relationships in SNS

The wide application of trust has made it an umbrella term with multiple interpretations in different contexts. In [18], two definitions of trust, *reliability* and *decision* trust, are discussed. In social network scenario, the notion of decision trust can be applied, which links to the extent to which an agent (i.e. the truster) is willing to depend on another one (trustee) in a given situation for decision making, with a feeling of similarity, closeness or security [30].

2.1.1 Trust & distrust

In social network scenario, trust relationship is based on the social connection between truster and trustee. There are various ways to quantitatively represent trust [13,18,30,31]. The trust value domain $\mathcal{T}$ is application dependent. For instance, it can be a set of discrete labels $\{0,1\}$ or a continuous range like $[0,1]$. It can have both positive values and negative values, and can even be a multi-dimensional vector space. When $\mathcal{T}$ contains negative trust values, it can be used to differentiate unknown users (i.e. of trust value 0) from ones that are not trusted (e.g. negative trust values). The negative trust relationship is also called distrust. Although recent research [8, 13, 30] shows an emerging interest in modeling the notion of distrust, models that take into account both trust and distrust are still limited. Introducing distrust makes the trust model complex as the non-negativity of trust values no long exists, and the linearity of trust aggregation is hard to argue. This can be especially challenging for many trust inference approaches such as matrix factorization [28]. Additionally, the subjective property of trust makes it even more complicate as conflicts between trust opinions may exist.

2.1.2 Trust networks

We refer to a **Trust Network** as the graph based on the trust relationships in a social network. The trust network can be defined as a directed graph with trust weight on edges, $\mathcal{G}(V, E, t_e)$, where V is the set of users, E is the set of connections denoting the directed trust relationships, and $t_e : E \mapsto \mathcal{T}$ is a mapping from an edge to the trust value placed on the edge. We use $\langle v_i, v_j \rangle_{\text{trust}}$ to denote trust relationship when $e_{ij} \in E$ and $t_{ij} > 0$. For a user $v \in V$, we denote the 1-hop neighbors as **Neighbor Set** $\mathcal{N}_v$. The undirected situation can be seen as having two edges with equal weights on both directions for the pair of vertices.

In the defined trust network, a directed path of length k is a sequence of distinct nodes, $\{v_1, v_2 \cdots, v_{m+1}\}$, such that $(v_i, v_{i+1}) \in E, \forall i \in 1, 2, \cdots, m$. Between two nodes in the network, there might be multiple distinct paths.

2.1.3 Transitivity in trust

The transitivity of trust relationships is the foundation for most trust metrics, it allows the truster to acquire information about the trustee from friends and their friends ("word of mouth") [15]. For example, if node v_1 and v_3 are not directed connected, but if v_1 trusts v_2, and v_2 trusts v_3, then v_1 can use such trust evidence to infer its opinion about v_3 using transitivity.

Formally, transitivity in trust can be defined as:

$$\langle v_i, v_j \rangle_{\text{trust}} \wedge \langle v_j, v_k \rangle_{\text{trust}} \Rightarrow \langle v_i, v_k \rangle_{\text{trust}} \quad \forall v_i, v_j, v_k \in V \quad (1)$$

Based on the assumption of transitivity (may be partial), trust can propagate along the paths between two nodes in the network and their trust relationship can be inferred. When multiple paths exist, a combination scheme is needed to derive the trust value. Due to the subjective nature, conflicts may happen in such occasion and need to be resolved.

Transitivity becomes more complex when considering distrust relationships [7]. For example, the trust relation between node v_1 and v_3 is not obvious if v_1 distrusts v_2 and v_2 distrusts v_3. The conflict resolution is also much more difficult with distrust.

2.1.4 Reciprocity in trust

Trust reciprocity describes the extent of symmetry in directed trust relationships between two users (i.e. nodes) in the social network. Similar to Eq. 1, we can write the following expression for reciprocity.

$$\begin{aligned} &\langle v_i, v_j \rangle_{\text{trust}} \Rightarrow \langle v_j, v_i \rangle_{\text{trust}} && \text{(trust reciprocity)} \\ &\langle v_i, v_j \rangle_{\text{distrust}} \Rightarrow \langle v_j, v_i \rangle_{\text{distrust}} && \text{(distrust reciprocity)} \\ & && \forall v_i, v_j \in V \end{aligned} \quad (2)$$

As shown in [6,10], trust relationships are asymmetric in terms of values (especially magnitude) in social network scenarios, thus most trust evaluation approaches don't consider reciprocity. However, by relaxing the condition in Eq. 2 to only sign agreement of positive trust relationships, **partial**

reciprocity can be defined. As we will show in Sec. 4.1, the partial reciprocity exists in the Epinions trust network dataset. The partial reciprocity between nodes may be useful in inferring indirect trust under some circumstances when data is very sparse (e.g. social recommender).

2.2 Semirings

A semiring is an algebraic structure, consisting of a set $\mathbf{A}$ and two binary operations, addition ($\oplus$) and multiplication ($\otimes$), with several conditions over the operations.

Definition. A *semiring* is a tuple $\langle \mathbf{A}, \oplus, \otimes, \mathbf{0}, \mathbf{1} \rangle$ such that

- $\mathbf{A}$ is a nonempty (possibly infinite) set with two special elements $\mathbf{0}, \mathbf{1} \in \mathbf{A}$
- $\oplus$ is the additive operation, which is commutative and associative:
$$a \oplus b = b \oplus a$$
$$a \oplus (b \oplus c) = (a \oplus b) \oplus c \qquad \forall a, b, c \in A$$
with $\mathbf{0}$ as the unit element ($a \oplus \mathbf{0} = \ a = \mathbf{0} \oplus a$)
- $\otimes$ is the multiplicative operation, which is associative,
$$a \otimes (b \otimes c) = (a \otimes b) \otimes c \qquad \forall a, b, c \in A$$
with $\mathbf{1}$ as the unit element and $\mathbf{0}$ as absorbing element ($a \otimes \mathbf{1} = \ a = \mathbf{1} \otimes a, \ a \otimes \mathbf{0} = \ \mathbf{0} = \mathbf{0} \otimes a$)
- $\otimes$ distributes over $\oplus$,
$$a \otimes (b \oplus c) = \ (a \otimes b) \oplus (a \otimes c)$$

An example of a semiring is the set of nonnegative integers $\mathbb{N}$, with the usual addition ($+$) and multiplication ($\times$).

The operations used in a semiring structure can be seen as a generalization of addition and multiplication which have used in trust inference for social recommendation. These semiring operations, with careful design, are able to capture the nonlinearity in trust evaluation [29].

2.3 Semiring-based trust fusion

The sets $\mathbf{A}$ in most semiring models only have nonnegative elements (e.g. the range of $[0, 1]$) [5, 29]. Due to the complexity and asymmetry introduced by negativity, the semiring framework needs to be carefully modified. In [9], the possibility of applying semiring for nonlinear trust fusion in online decision making was discussed, and a semiring called a *Distrust Semiring* was introduced. The semiring structure was defined on a 2-dimensional trust opinion model, with both trust and distrust relationships being considered.

2.3.1 Trust opinion vectors

The 2-dimensional trust opinion vector $\tau_{ij} \in \mathcal{T} = [-1, 1] \times [0, 1]$ from truster i to trustee j is defined as

$$\tau_{ij} = (t_{ij}, c_{ij}) \tag{3}$$

where $t_{ij} \in [-1, 1]$ is the trust level representing how much i trusts (likes)/distrusts (dislikes) the opinions (tastes) of j. By making trust levels take values in the range of $[-1, 1]$, distrust is considered along with trust. $c_{ij} \in [0, 1]$ is the certainty level which shows how much i believes in the integrity of j. Note that certainty is orthogonal to trust value, it denotes the quality and accuracy of the trustee's opinion. While a high trust value may be because of similarity in taste or preference, a high certainty value may be due to direct connection with the truster or large number of connections (i.e. high degree). Certainty determines if the opinion will be considered, and opinions with a high certainty value are more useful in making trust decisions.

As both trust level and certainty level about the trustee are considered in the opinion vector, more complicated situations can be modeled and analyzed.

Based on Sec. 2.1.2 and the definition of trust opinion vector, the trust network can be derived.

Definition. A trust network based on the set $\mathcal{T}$ of trust opinions is a directed and weighted graph $\mathcal{G}(V, E, t_e)$, $t_e : E \mapsto \mathcal{T}$, where V is the set of users. E is the set of trust relationships. For $\forall e_{ij} = (v_i, v_j) \in E, \ v_i, v_j \in V, i \neq j$, t_e associates it with an opinion vector $\tau_{ij} = (t_{ij}, c_{ij}) \in \mathcal{T}$, indicating the trust and certainty that node v_i holds on v_j. Trust links are directed. $N_i = \{v_j | e_{ij} \in E\}$ is the neighbor set of node v_i.

2.3.2 Distrust semirings

A distrust semiring [9] is a 2-dimensional semring defined on the trust opinion set $\mathcal{T}$, such that

- $\mathbf{A} = \mathcal{T} = [-1, 1] \times [0, 1]$ the set of trust opinion vectors, with two special elements $\mathbf{0} = (0, 0), \mathbf{1} = (1, 1)$
- The additive operation $\oplus$ is defined as
$$(t_a, c_a) \oplus (t_b, c_b) = (t, c) \tag{4}$$
with $c = max\{c_a, c_b\}$, and
$$t = \begin{cases} t_a & c_a > c_b \\ t_b & c_b > c_a \\ sign(t_a + t_b) \cdot max\{|t_a|, |t_b|\} & c_a = c_b \end{cases} \tag{5}$$
- The multiplicative operation $\otimes$ is defined as
$$(t_a, c_a) \otimes (t_b, c_b) = (t, c) \tag{6}$$
where $c = c_a c_b$, and
$$t = \begin{cases} 0 & t_a < 0, \ t_b < 0 \\ t_a t_b & \text{otherwise} \end{cases} \tag{7}$$

For the trust network shown in Fig. 1, there are three paths from node v_1 to v_5. For the left path, when we multiply τ_{12} and τ_{25} using $\otimes$, we have $\tau_1 = \tau_{12} \otimes \tau_{25} = (0.36, 0.56)$. Similarly, the middle path has $\tau_2 = (0.12, 0.25)$ and the right one $\tau_3 = (-0.15, 0.56)$. When combining τ_1, τ_2, τ_3, we can use $\oplus$ defined in distrust semiring and have combined opinion $\tau = \tau_1 \oplus \tau_2 \oplus \tau_3 = (0.36, 0.56)$.

Based its definition in [9], distrust semiring is a promising algebra structure that can be used in designing trust metrics for social recommender systems. However, [9] only proposed the framework with some theoretical analysis, without experiments and evaluation with real datasets. Also, in the work, trust is represented by a 2-D vector with both trust and certainty information, with the assumption that information about both components are explicit and available. However, in social recommender setting, certainty information is generally missing. In order to use the 2-D representation of trust, a model for certainty based on social context is required yet has not been discussed.

3. STAR: SEMIRING TRUST INFERENCE FOR TRUST-AWARE RECOMMENDATION

In this section, we draw connections between distrust semiring and trust inference in trust-aware social recommender systems: a) we first show the distrust semiring definition

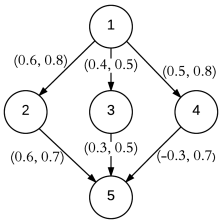


Figure 1: An example trust network

(Eq.4-7) is consistent with the properties for trust propagation and aggregation in Sec. 3.1; b) next in Sec. 3.2, we propose practical certainty models and parameter tuning framework in order to use distrust semiring in practice; c) we then propose the overall **S**emiring trust inference for **T**rust-**A**ware social **R**ecommenders (STAR) framework in Sec. 3.3; d) furthermore, to address the data sparsity and improve accuracy and coverage, we introduce iterative trust evaluation in semiring trust in Sec. 3.4.

3.1 Distrust semiring and trust properties

Trust information diminishes and becomes noisy in the process of propagation, and increases when aggregating neighborhood values. These are basic properties [19] of trust. Distrust semiring essentially defines an algebraic way to calculate trust, thus we first need to show whether the basic operations, addition used in trust aggregation and multiplication used in trust propagation, satisfies the commonsense.

By defining a partial ordering, we show that the semiring framework is intuitive and consistent with requirements for trust propagation and aggregation.

Definition. A partial order relation $\preceq$ can be defined upon two 2-D opinion vectors $\tau_a = (t_a, c_a)$ and $\tau_b = (t_b, c_b)$:

$$\tau_a \preceq \tau_b \tag{8}$$

if and only if

$$c_a \leq c_b, \text{ or } \quad |t_a| \leq |t_b| \text{ and } c_a = c_b \qquad \forall\ \tau_a, \tau_b \in \mathcal{T}$$

Theorem 3.1. The distrust semiring structure satisfies the following two conditions,

1. the multiplication operation is non-increasing:

$$\forall \tau_a, \tau_b \in \mathcal{T}, \quad \tau_a \otimes \tau_b \preceq \tau_a \ \wedge\ \tau_a \otimes \tau_b \preceq \tau_b \tag{9}$$

2. the addition operation is non-decreasing

$$\forall \tau_a, \tau_b \in \mathcal{T}, \quad \tau_a \preceq \tau_a \oplus \tau_b \ \wedge\ \tau_b \preceq \tau_a \oplus \tau_b \tag{10}$$

Proof. Let $\tau_a \otimes \tau_b = (t, c)$. As $c_a,\ c_b\ \in [0, 1]$, and $t_a,\ t_b\ \in [-1, 1]$, based on the definition of the multiplication operation, it is easy to see that $c = c_a c_b \leq c_a,\ c = c_a c_b \leq c_b$. For the trust value, if both t_a and t_b are negative, then $t = 0 < |t_a|, |t_b|$. Otherwise, $|t| = |t_a t_b| = |t_a||t_b| \preceq |t_a|$, and $|t| = |t_a t_b| \leq |t_b|$. Thus Eq. 9 holds, and the non-increasing property of the multiplication operation is proved.

The non-decreasing property of trust aggregation can be shown in a similar way. In aggregation, $\tau = (t, c) = \tau_a \oplus \tau_b$. based on definition, $c = \max\{c_a, c_b\} \geq c_a, c_b$. Thus the first condition is satisfied, and the non-decreasing property of the additive operation is proved. □

The non-increasing property of the multiplication operation is in accordance with the requirement for trust propagation process, whereas the non-decreasing property of the addition operation connects to the trust aggregation process. Thus the two operations for a distrust semiring can be applied to modeling trust propagation and aggregation respectively.

3.2 Certainty models

In the context of trust-aware recommenders, though trust data is available, the certainty information is generally implicit and contained in user interactions. Without certainty data, it is infeasible to apply the 2-D semiring model for trust inference. In practice, certainty may be derived by sentiment analysis or other natural language processing pipelines. However, NLP toolboxes are not light-weight and require rich text data in the SNS. In this paper, we consider the situation when only SNS connections and trust data are available. We propose two ways to model certainty. One is based on length of the trust path, the other one is degree-oriented.

3.2.1 Path-based certainty models

Based on the fact that neighbors which are reachable via a longer trust path carry less valuable trust information [12], we come up with a path-based certainty model. We model the certainty value of user v_s about v_t as a function of hops (i.e. the length of the shortest path) between the two.

$$c_{st} = g(\text{dist}_{s,t}) = \alpha^{\text{dist}_{s,t}} \tag{11}$$

where $\alpha \in (0, 1]$ is a hyperparameter and can be seen as the decay factor, and $\text{dist}_{s,t}$ represents the shortest trust path length between v_s and v_t. When $\alpha = 1$, the decay disappears and nodes of all distances are considered equally. Such definition is equivalent to introducing a 1-hop decay of magnitude α at each hop. Instead of calculating c_{st}, we consider the decay at node v_i in the middle:

$$c_i = g(v_i) = \alpha \qquad \text{(the decay factor)} \tag{12}$$

then along path_{st} from v_s to v_t, $c_{st} = \prod_{v_i \in \text{path}_{st}} c_i$. In this simple model, partial order relation between two paths is stable for $\forall \alpha \in (0, 1)$.

3.2.2 Degree-based certainty models

Another way to model certainty in social recommender setting is based on node degree, with the hypothesis that nodes of higher degree are more reliable and their trust opinions have more certainty. The certainty function of a node v_i can be accordingly denoted as

$$c_i = g(v_i) = g(d_i) \tag{13}$$

where d_i is the degree of v_i.

We consider two realizations for $g(d_i)$, a linear model and an exponential model. The linear model can be written as

$$g(d_i) = \min(\beta + \gamma d_i, 1) \tag{14}$$

which is determined by a cut-off degree value Δ, when $d_i > \Delta$, $g(d_i) = 1$.

The exponential model can be described by

$$g(d_i) = 1 - \eta^{d_i} \tag{15}$$

The coefficients β, γ, and η are tunable. Given the trust metric, an optimization problem over the parameters can be formulated accordingly to maximize its performance. The optimization formulation, due to the complexity of network structure and the semiring model we use, is non-convex and hard to solve. However, as there's only very few optimization variables (i.e. model parameters), we can simplify the problem by conducting parameter search schemes, such as greedy search, random search or even annealing approach, and measure its effect on system performance.

3.3 Inferring trust via semiring operators

With the certainty model defined above, the trust between two arbitrary users in the trust network can be inferred via the trust metric developed in this work. The trust metric consists of two components, i.e. trust propagation and aggregation, which can be defined using the distrust-semiring. In STAR, trust aggregation is conducted along different paths first before trust propagation, in order to reduce the noises caused by longer paths.

3.3.1 Trust propagation

In the trust metric, the trust value between two nodes v_s and v_t, who have no direct connection, can be estimated via the multiplicative operation $\otimes$ of trust values on edges along the path between the two nodes. Along each path path_{st},

$$(t_{st}, c_{st}) = \prod_{\otimes, e_{ij} \in \text{path}_{st}} (t_{ij}, c_{ij}) \tag{16}$$

Considering the decay of influence along the path, a maximum hop value λ can be introduced in order to stop early and accelerate the calculation. As shown in Sec. 4, 4-hop semiring calculation already reaches about 95% accuracy.

When multiple paths between two nodes exist, trust opinions reached along all paths should be aggregated together.

3.3.2 Trust aggregation

The trust aggregation component in the trust metric is to combine trust information from different sources (i.e. paths). It can be defined upon a semiring operation as follows:

$$(t, c) = \sum_{\oplus, \{a | \text{path}_a \in P\}} (t_a, c_a) \tag{17}$$

where P is the set of trust paths considered in aggregation, and each (t_a, c_a) with $\text{path}_a \in P$ is the trust vector along that trust path a.

3.3.3 Overall trust inference framework

A trust metric can be developed based on the trust propagation and aggregation rules defined above. Given a pair of users v_s (truster) and v_t (trustee), the proposed trust inference method is a function $f : V \times V \mapsto \mathcal{T}$, such that

$$(t_{st}, c_{st}) = f(v_s, v_t) = \sum_{\oplus, v_j \in N_s} (t_{sj}, c_{sj}) \otimes (t_{jt}, c_{jt}) \tag{18}$$

where N_s is the neighbor set of v_s. In order to save computation resources, a threshold for trust value (σ_t) and certainty value (σ_c) can be introduced. When below the thresholds, the trust opinion will not be considered in aggregation, i.e.

$$(t_{st}, c_{st}) = \sum_{\oplus, v_j \in N_s, (\sigma_t, \sigma_c) \preceq \tau_{sj} \otimes \tau_{jt}} (t_{sj}, c_{sj}) \otimes (t_{jt}, c_{jt}) \tag{19}$$

The way that the trust metric is applied to trust inference is shown in Alg. 1. The algorithm is to evaluate the trust opinion (t_{st}, c_{st}) of v_s (truster) about v_t (trustee), with λ the maximum hop number, σ_t and σ_c the lower bounds for trust and certainty value respectively. One can interpret the algorithm as follows: v_s asks her neighbors for their trust opinions about v_t. Each neighbor $v_i \in N_s$ provides her opinion about v_t (i.e. t_{it} and c_{it}), either directly or estimated using the trust inference algorithm recursively. At each hop forward, λ, the current hop reached, will decrease by 1 until reaching 0. Then v_s aggregates all the evidence and reach (t_{st}, c_{st}) about v_t.

In such a trust inference framework, both trust and distrust (i.e. negative trust) are taken into consideration for trust inference. As paths above the thresholds are all integrated into the calculation, trust information are fully exploited in this approach for better coverage. At each node, the trust information along different paths is first aggregated together and then propagation to the node as a unified trust opinion. Such scheme can reduce the noises caused by long paths. The possible conflicts among opinions are handled in a non-trivial way with the introduction of certainty value and nonlinear addition operation ($\oplus$).

Algorithm 1 STAR algorithm, $f_{star}(v_s, v_t, \lambda, \sigma_t, \sigma_c)$

```
Mark v_s as visited
if t_st exists then
  return (t_st, g(v_t))
end if
t_st ← 0
c_st ← 0
if λ = 0 then
  return (0, 0)
end if
for each v_i ∈ N_s, the neighbor set of node v_s do
  c_si = g(v_i)
  if (v_i visited) or (|t_si| < σ_t) or (c_si < σ_c) then
    continue
  end if
  (t_it, c_it) ← f_star(v_i, v_t, λ − 1, σ_t/|t_si|, σ_c/c_si)
  if (t_si < 0 and t_it < 0) or (t_it = 0) then
    continue
  end if
  (t_st, c_st) ← (t_st, c_st) ⊕ ((t_si, c_si) ⊗ (t_it, c_it))
end for
return (t_st, c_st)
```

3.4 Iterative trust evaluation

Alg. 1 is built on the trust paths between two users. In real world data, the trust data often is sparse, thus the coverage of STAR is limited by the amount of trust data present. In order to address the sparsity, we propose collective method from network analysis. Iterative method is a type of collective approaches in classification problem [21]. Such method is used when the data contains interconnection and correlation between objects, such as webpages and SNS, and has been shown to be very effective over network data [26]. The basic idea is treating the independent inference as a joint inference problem, and use an iterative approach to predict labels; the new label predicted in the previous iterations is used in the following iterations.

In the trust inference problem, trust relationships among users are intercorrelated, we introduce iterative trust evaluation in Alg. 2 to improve the performance. The basic idea of the design is to iteratively evaluate the trust opinions associated with the edges in the test dataset, conditioned on the both ground truth and current predictions. Initially, the knowledge base is the training dataset, and the result set is empty. While running, the "diff" in Alg. 2 is used to measure the difference of results between two iterations. Here the Euclidean norm is used in calculating the difference. The iteration ends when convergence on local predictions is reached or maximum number of iterations has finished.

3.5 Discussions

3.5.1 Exploiting partial reciprocity in trust

As discussed in Sec. 2.1.4, partial reciprocity can be added into the trust inference algorithm to improve both coverage and accuracy. We extend the trust inference algorithm

Algorithm 2 Iterative trust evaluation

```
Let E: set of edges to evaluate
knowledge_base ← known trust relationships
result_set ← ∅
Let K: max number of iterations
for i : 1 to K do
  diff ← 0
  for edge e_j ∈ E do
    knowledge_base ← knowledge_base \ (t_j, c_j)
    calculate trust metric f_star(e_j)
    if ((t_j, c_j) ∉ result_set) then
      diff ← diff + ||f_star(e_j)||
    else
      diff ← diff + ||f_star(e_j) − (t_j, c_j)||
      result_set ← result_set \ (t_j, c_j)
    end if
    (t_j, c_j) ← f_star(e_j)
    result_set ← result_set ∪ (t_j, c_j)
    knowledge_base ← knowledge_base ∪ (t_j, c_j)
  end for
  if (diff < ε) then
    return result_set
  end if
end for
return result_set
```

(Alg. 1) by introducing partial reciprocity in a careful way. Apart from considering trust evidences from neighbors, the truster also treat the direct trust opinion about herself from the trustee as a source of information, as a reflection of partial reciprocity. In order to reduce the error due to asymmetry in trust relationships, we only consider positive reciprocity (i.e. the reciprocity in positive trust relationships), and the certainty value in a trust opinion reached using reciprocity has a smaller magnitude compared the one reached through transitivity-based propagation.

3.5.2 Optimistic vs pessimistic semiring definitions

In Sec. 3.3, the trust metric takes an optimistic definition for trust aggregation, as the binary operation $\oplus$ takes the larger magnitude of the two and its sign as the combined trust value when the certainty levels are the same. We use this to define trust aggregation based on the hypothesis that extreme opinions weigh more than neutral ones [11, 12]. In a pessimistic model, when $c_a = c_b$, the addition operation for trust aggregation (Eq. 5) can be modified as

$$t = sign(t_a + t_b) \cdot \min\{|t_a|, |t_b|\} \tag{20}$$

Such definition is "pessimistic", as the aggregated trust value conservatively picks the smaller magnitude between the two incoming trust values. With such model, trust aggregation is not non-decreasing any more, which is against the basic properties of trust as mentioned in Sec. 3.1.

4. EVALUATION

In order to verify the model and evaluate the performance of our semiring-based approach, we conduct experiments on a real world dataset.

We use the Epinions trust network dataset for evaluation [1]. The dataset contains both direct trust connections (with edge weight of $+1$) and distrust ones (with edge weight of -1), and form the trust network $\mathcal{G}(V, E, t_e)$ where $t_e : E \mapsto \mathcal{T} = \{-1, 1\}$. To the best of our knowledge, it is the largest dataset available online that contains both explicit trust and distrust information marked in a binary ($+1$ and -1) format. The dataset has 91.053 users with 841,372 edges, 717,667 trust relationships and 123,705 distrust relationships. Notice that the method works on continuous trust value domains as well. However, large-scale public datasets are not available.

4.1 Transitivity and partial reciprocity in data

In order to evaluate our trust metric, we first verify the transitivity and partial reciprocity of trust relationships using the Epinions dataset.

4.1.1 Transitivity

Inspired by previous works, we investigate the transitivity phenomenon. For transitivity, we count the number of triangles represented by triplets (i, j, k) such that

$$v_i, v_j, v_k \in V,\ e_{ij}, e_{jk}, e_{ik} \in E \tag{21}$$

$$\text{and} \qquad t_{ij}t_{jk}t_{ik} > 0 \qquad \text{(structural balance)} \tag{22}$$

Based on this definition, among all 11033232 triangles that satisfy Eq. 21, 10229847 (92.7%) are transitive. We also look at the case when v_i distrusts v_j, v_j distrusts v_k (i.e. $t_{ij} < 0$ and $t_{jk} < 0$), what the relationship between v_i and v_k is. It turns out that $\approx 50\%$ situations have $t_{ik} < 0$ with another half having $t_{ik} > 0$, which endorses the setting "the enemy of your enemy is actually unknown" in our trust metric.

4.1.2 Partial reciprocity

We also evaluate the partial reciprocity of trust relationships in Epinions dataset. We consider partial reciprocity of trust relationships as symmetry on signs, i.e. we consider pairs (v_i, v_j) as having partial reciprocal trust relationships when both e_{ij} and e_{ji} exists and $t_{ij}t_{ji} > 0$.

Based on the experimental results, we notice that partial reciprocity commonly exists between nodes in the network. Among all 259,751 pairs of nodes having bi-directional relationships, 254,345 ($\approx 98\%$) are partially reciprocal relationships (both directions are of the same sign, either positive or negative), and 98% of reciprocal ones are of positive connections, which corresponds to the concept of partial reciprocity that we discussed in Sec. 2.1.4.

However, as both trust and distrust relationships are single-valued, the symmetry on magnitude of trust relationships is unable to be evaluated.

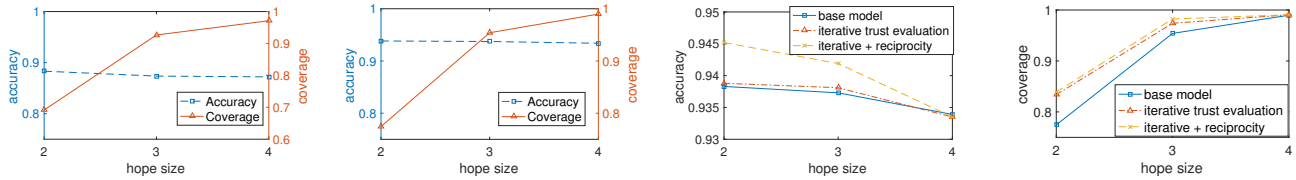
4.2 Experimental design

In Epinions dataset, all trust values in the training dataset are in the set $\mathcal{T} = \{-1, 1\}$. Based on the definition of our trust metric model, though certainty value of each predicted edge varies in $[0, 1]$, the set for predicted trust values will be $\mathcal{T}_p = \mathcal{T} \cup \{0\} = \{-1, 0, 1\}$, where the value of $t_{st} = 0$ represents the case when not enough information is available for predicting the trust relationship associated with the edge e_{st}. Unlike linear approach such as [14], no rounding is needed for predicting discrete trust values.

To evaluate the performance of our trust evaluation approach, we measure and compare *accuracy* and *coverage*. Accuracy is the fraction of correctly predicted trust relationships among all nonzero ones in the test data:

$$\text{accuracy} = \frac{\|\{e_{ij} \in S_{\text{test}} \mid t_{ij} = t'_{ij}\}\|}{\|\{e_{ij} \in S_{\text{test}} \mid t_{ij} \neq 0\}\|} \tag{23}$$

where S_{test} is the test edge set, t_{ij} is the trust value reached using our trust inference algorithm, and t'_{ij} is the ground



(a) Large test dataset (b) Small test dataset (c) Accuracy (d) Coverage

Figure 2: Evaluation results of the proposed semiring trust inference framework

truth trust value. The denominator in Eq. 23 only take edges that have nonzero predictions.

Coverage is defined as the number of edges predicted over the number of edges predictable in test data:

$$\text{coverage} = \frac{\|\{e_{ij} \in S_{\text{test}} \mid t_{ij} \neq 0\}\|}{\|\{e_{ij} \in S_{\text{test}} \mid \exists k, l \in V \text{ s.t. } e_{ik}, e_{lj} \in S_{\text{train}}\}\|} \quad (24)$$

where an edge is predictable if both of its vertices are present in the training set.

4.2.1 Test data partitioning

We partition Epinions dataset into two parts to evaluate STAR. The first part contains all the known trust relationships, i.e. the trust network before inference, while the second part includes a set of edges serving as the ground truth of our evaluation. We evaluate STAR using the known trust information in the first part to predict the trust relationships in the second part. This is a practice used in previous trust inference work and link prediction literatures. On the partition strategy, we examine the time-ordered partition and the random partition. The evaluation results are consistent between the two strategies. All numbers reported here are under the time-ordered partition strategy.

We conduct experiments on datasets of two sizes, we refer the test dataset generated with percentage $\rho = 0.5\%$ as the small dataset, while the one using $\rho = 5\%$ as the large test dataset. The experimental results for the two datasets are shown in Fig. 2(a) and Fig. 2(b) respectively. From the results, we see that the performance of our approach works better on the smaller dataset, in terms of both accuracy and coverage. The major reason is that more trust information is known for the smaller test dataset, which means the trust network used for prediction is more connected and has more trust evidence for prediction. Thus, the performance is positively correlated with the known/test data ratio.

4.2.2 Experimental results

Varying trust path length λ: As discussed in previous literature [10, 25], the longer the trust path in graphical trust evaluation models, the more noise may be introduced. While the introduction of distant friends improves the coverage of the social recommendation algorithm, it affects its accuracy. As the average shortest path length in Epinions dataset is about 4, in the experiment, we set the maximum hop length λ as 2, 3 and 4 respectively, and evaluate the performance of our trust metric under different settings.

In Fig. 2(a) and Fig. 2(b) we can see that, the coverage of our approach is better with paths of more hops considered, as more nodes are reachable and used in trust evaluation. However, though the variation is subtler compared to coverage change, the accuracy result is just slightly decreasing with an increasing maximum hop length parameter.

Table 1: Alg.2 using random-ordered test data

random set #	1	2	3	4	5
Accuracy	0.9314	0.9314	0.9328	0.9314	0.9319
Coverage	0.9895	0.9899	0.9895	0.9899	0.9895

Iterative trust evaluation: Apart from data size and maximum hop length, we also compare the base STAR model (Alg. 1) with the model applying collective methods (Alg. 2). As shown in Fig. 2(c)(d), the iterative trust inference method improves the performance in both coverage and accuracy.

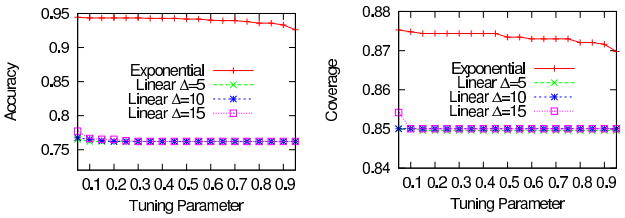
When applying iterative approaches in classification problem [21], the ordering of value updates in the iterative trust evaluation may affect the predictive accuracy and convergence rate. Here, in order to investigate the stability in trust iteration over test data, we randomize the order of the node pairs for prediction, and compare the experimental results. We list the results for the 4-hop case in Table 1. According to the results, the application of iterative approach for the trust metric in social recommender system setting is fairly robust to randomized orderings of the test dataset.

Exploiting partial reciprocity: From Fig. 2(c) and Fig. 2(d), by getting trust information based on partial reciprocity for trust evaluation, the coverage and accuracy of the trust inference algorithm are even better. By applying collective methods and partial reciprocity, more trust evidence can be used in the trust metric.

Parameter tuning in certainty: As discussed in Sec. 3.2, we proposed two tunable models for certainty based on degree. Here we vary the values of the parameters and investigate the influence of parameter tuning towards model performance. In the linear model (Eq. 14), there are two parameters, β and γ. β is in the range of $[0, 1]$, and γ is dependent on β and the cut-off degree. The exponential model described in Eq. 15 has one parameter $\eta \in [0, 1]$. In the experiment, we fix maximum degree value used in linear model to be 5, 10 and 15, and let β and η both vary from 0 to 1 with step change to be 0.05. The results are shown in Fig. 3, where Δ represents maximum degree value. From the plots we can see that the linear model leads to lower accuracy and coverage, while the exponential model doesn't change very much with varying parameter values. Using exponential model for certainty with a relatively small η around 0.2 would be a good setting for trust inference.

4.3 Comparison with other approaches

Base on the experimental results discussed in Sec. 4.2, the performance of our method STAR can be as small as 5.8% error rate and as good as 98.3% coverage rate for reachable pairs, As a fair comparison, the graph-theoretic linear approach based on matrix operations [14] has an optimal prediction error rate of 6.4%. The machine learning approach introduced in [20] can reach an accuracy about 0.934



(a) Accuracy (b) Coverage

Figure 3: Performance with parameter tuning

(i.e. $\approx$ 6.7% error rate). For [7] which used probabilistic confidence models for trust inference, it achieved an accuracy of 89% (11% error rate) using Epinions dataset. From comparison, we can see an improvement on accuracy can be obtained using the STAR approach, not to mention its computation efficiency and interpretability.

4.4 Discussions

Because the trust (and distrust) relationships in the dataset are binary (+1 and −1), a lot of nuance and variance in the trust relationships are lost. This prevents us from conducting some evaluations. For example, since the trust values are not continuous in $[-1, 1]$, the deviation of evaluated trust value from ground-truth difficult to measure and analyzed.

The high accuracy of STAR on this dataset shows its flexibility and power in SNS trust-aware recommendation.

5. CONCLUSIONS

In this work, we propose STAR, a novel trust inference method for trust-aware social recommenders. It not only has high accuracy comparing with machine learning methods, but also has the efficiency from graph-theortical models. Our method is based on a 2-D semiring framework reflecting trust propagation and aggregation properties over the trust network. The nonlinearity and inconsistency in human's opinion formulation process are captured in the trust inference algorithm. To address the data sparsity, we introduce collective semiring methods and propose partial reciprocity. In order to validate the model and evaluate the performance of our approach, we conduct a series of experiments using the Epinions dataset. The experimental results show that the approach proposed in this work has advantages in both accuracy and coverage. The trust relationships inferred via our approach can be used to improve the quality of trust-aware recommendations.

6. REFERENCES

[1] Epinions trust network dataset – KONECT, Oct. 2014.

[2] C. C. Aggarwal. Social and trust-centric recommender systems. In *Recommender Systems*. Springer, 2016.

[3] R. Andersen, C. Borgs, J. Chayes, U. Feige, A. Flaxman, A. Kalai, V. Mirrokni, and M. Tennenholtz. Trust-based recommendation systems: an axiomatic approach. In *WWW 2008*.

[4] P. Avesani, P. Massa, and R. Tiella. A trust-enhanced recommender system application: Moleskiing. In *ACM SAC*, 2005.

[5] S. Bistarelli. *Semirings for soft constraint solving and programming*, volume 2962. Springer Science & Business Media, 2004.

[6] C. Castelfranchi and R. Falcone. *Trust theory: A socio-cognitive and computational model*, volume 18. John Wiley & Sons, 2010.

[7] T. DuBois, J. Golbeck, and A. Srinivasan. Predicting trust and distrust in social networks. In *Privacy, Security, Risk and Trust (PASSAT) and IEEE SocialCom*, 2011.

[8] R. Forsati, I. Barjasteh, F. Masrour, A.-H. Esfahanian, and H. Radha. Pushtrust: An efficient recommendation algorithm by leveraging trust and distrust relations. In *ACM RecSys*, 2015.

[9] P. Gao, J. S. Baras, and J. Golbeck. Semiring-based trust evaluation for information fusion in social network services. In *IEEE Fusion*, 2015.

[10] J. Golbeck. *Computing and applying trust in web-based social networks*. PhD thesis, University of Maryland College Park, 2005.

[11] J. Golbeck. Personalizing applications through integration of inferred trust values in semantic web-based social networks. In *Semantic Network Analysis Workshop at ISWC*, 2005.

[12] J. Golbeck. *Generating predictive movie recommendations from trust in social networks*. Springer, 2006.

[13] J. Golbeck. *Computing with social trust*. Springer, 2008.

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

[15] C. Haydar, A. Roussanaly, and A. Boyer. Local trust versus global trust networks in subjective logic. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies*.

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

[17] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *ACM RecSys*, 2010.

[18] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision support systems*, 43(2):618–644, 2007.

[19] A. Jøsang, S. Marsh, and S. Pope. Exploring different types of trust propagation. In *Trust management*, pages 179–192. Springer, 2006.

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

[21] B. London and L. Getoor. Collective classification of network data. *Data Classification: Algorithms and Applications*, 399, 2014.

[22] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *ACM SIGIR*, 2009.

[23] H. Ma, M. R. Lyu, and I. King. Learning to recommend with trust and distrust relationships. In *ACM RecSys*, 2009.

[24] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *ACM SIGIR*, 2008.

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

[26] J. Neville and D. Jensen. Iterative classification in relational data. In *Proc. AAAI Workshop on Learning Statistical Models from Relational Data*, pages 13–20, 2000.

[27] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In *The Semantic Web-ISWC 2003*, pages 351–368. Springer, 2003.

[28] J. Tang, X. Hu, and H. Liu. Social recommendation: a review. *Social Network Analysis and Mining*, 3(4):1113–1133, 2013.

[29] G. Theodorakopoulos and J. S. Baras. Trust evaluation in ad-hoc networks. In *3rd ACM workshop on Wireless security*, 2004.

[30] P. Victor, C. Cornelis, and M. De Cock. *Trust networks for recommender systems*, volume 4. Springer, 2011.

[31] C.-N. Ziegler and G. Lausen. Propagation models for trust and distrust in social networks. *Information Systems Frontiers*, 7(4-5):337–358, 2005.