Inferring Trust based on Similarity with TILLIT

Mozhgan Tavakolifard, Peter Herrmann, and Svein J. Knapskog

Abstract A network of people having established trust relations and a model for propagation of related trust scores are fundamental building blocks in many of todays most successful e-commerce and recommendation systems. However, the web of trust is often too sparse to predict trust values between non-familiar people with high accuracy. Trust inferences are transitive associations among users in the context of an underlying social network and may provide additional information to alleviate the consequences of the sparsity and possible cold-start problems. Such approaches are helpful, provided that a complete trust path exists between the two users. An alternative approach to the problem is advocated in this paper. Based on collaborative filtering one can exploit the like-mindedness resp. similarity of individuals to infer trust to yet unknown parties which increases the trust relations in the web. For instance, if one knows that with respect to a specific property, two parties are trusted alike by a large number of different trusters, one can assume that they are similar. Thus, if one has a certain degree of trust to the one party, one can safely assume a very similar trustworthiness of the other one. In an attempt to provide high quality recommendations and proper initial trust values even when no complete trust propagation path or user profile exists, we propose TILLIT — a model based on combination of trust inferences and user similarity. The similarity is derived from the structure of the trust graph and users' trust behavior as opposed to

Mozhgan Tavakolifard
Centre for Quantifiable Quality of Service in Communication Systems (Q2S), Norwegian University of Science and Technology (NTNU), Trondheim, Norway
e-mail: mozhgan@q2s.ntnu.no

Peter Herrmann
Department of Telematics (ITEM), Norwegian University of Science and Technology (NTNU), Trondheim, Norway
e-mail: herrmann@item.ntnu.no

Svein J. Knapskog
Centre for Quantifiable Quality of Service in Communication Systems (Q2S), Norwegian University of Science and Technology (NTNU), Trondheim, Norway
e-mail: knapskog@q2s.ntnu.no

other collaborative-filtering based approaches which use ratings of items or user's profile. We describe an algorithm realizing the approach based on a combination of trust inferences and user similarity, and validate the algorithm using a real large-scale data-set.

1 Introduction

Many online communities are only successful if sufficient mutual trust between their members exists. Users want to know whom to trust and how much to trust in the competence and benevolence of other community members in a specific application domain. The process of building trust is hereby performed in two different ways. First, one can establish trust (or distrust) by gaining direct experience with another party. Of course, every positive event increases the assumed trustworthiness of the trustee while every negative one reduces it. Second, one can gain trust based on recommendations of third parties. If, e.g., Alice has high trust in Bob's ability to assess the trustworthiness of other people, Bob has similar trust in Claire's recommendations, and Claire considers David trustable based on her personal experience with him, then Alice gains also trust in David even if she has no or very limited knowledge of him at all. This form of propagated trust is called trust transitivity.

Based on the two forms of trust, a so-called web of trust between community members is created which is often used in recommender systems helping users of e-commerce applications to get an idea about the trustworthiness of their mostly personally unknown cooperation partners. Unfortunately, however, these webs of trust are often too sparse to be helpful in practice since — at least in large online communities — a user has experience with only a very small fraction of the other community members. Thus, very often there will be no trust relation to an intended new partner of an e-commerce transaction at all [14].

As a model to increase the number of trust relations, we propose the method TILLIT[1] (Trust Inference Links based on Like-minded Interaction Transitions). It enables to derive trust not only from direct experience and by transitive propagation but also from the similarity between users and vice versa. In particular, two users are considered similar if they either built akin trust relations to other users or if they are trusted very similarly by others. This can be used to propagate already known trust to new trust relations encompassing people similar to those of the yet known relationships. Thus, the web of trust can be augmented significantly.

In our model, we measure similarity based on the existing web of trust in a community using an iterative fixed-point algorithm on node-pair graphs introduced later in this paper. As a method to describe the values of trust as well as its propagation we apply the TNA-SL model [12] which is based on the Subjective Logic [10]. Our approach, however, would also work with other methods like [1, 7].

[1] "Tillit" is the Norwegian word for trust.

In comparison with other approaches based on similarity, our work has the following differences:

- It intends to alleviate the sparsity problem in the web of trust matrix itself instead of the matrix of users rating items in the system. Since users have usually few items rated in common, the classic recommender system techniques are often ineffective and are not able to compute a user similarity weight for many of the users. Instead, exploiting the web of trust, it is possible to propagate trust better and to infer additional trust information about other users.
- It calculates the similarity from the structure of the web of trust and trust relations (the trust graph structure and trust values) instead of user-item ratings.
- It proposes methods to convert trust values to similarity measures and vice versa based on the TNA-SL model.

We conducted experiments on a large real dataset showing how our proposed solution increases the coverage (number of trust relations that are predictable) while not reducing the accuracy (the error of predictions). This is especially true for users who have provided few ratings.

The rest of this paper is organized as follows: In section 2, we briefly explain the TNA-SL model as the background of our work. Our proposed model for trust inference is described in section 3. Next in section 4, we present the evaluation plan and results. Section 5 provides an overview of the related research. Finally, discussion and conclusion are given in section 6.

2 Trust Network Analysis with Subjective Logic

Our model is mainly based on TNA-SL [12], a model for trust network analysis. TNA-SL uses the Subjective Logic [10] which enables to represent a specific belief calculus. There trust is expressed by a belief metric called opinion. An opinion is denoted by $\omega_B^A = (b,d,u,a)$ expressing the belief of a relying party A in the trustworthiness of another party B. The parameters b and d represent the belief resp. disbelief in B's trustworthiness while d expresses the uncertainty of A about to trust B or not. The three parameters are all probability values between 0 and 1 and fulfill the constraint $b+d+u=1$. The parameter a is called the base rate, and determines how uncertainty shall contribute to the opinion's probability expectation value which is calculated as $E(\omega_x^A) = b+au$. The opinion space can be mapped into the interior of an equal-sided triangle, where, the three parameters b, d, and u determine the position of the point in the triangle representing the opinion. Fig.1 illustrates an example where the opinion is $\omega_x = (0.7,0.1,0.2,0.5)$.

Based on TNA-SL, there are two different types of trust relations: *functional trust* (FT) and *referral trust* (RT). The former concerns A's direct trust in B performing a specific task; the latter concerns A's trust in B giving a recommendation about someone else doing a task or in other words is the trust in the ability to refer to a third party. As mentioned in the introduction, the simplest form of trust inference is

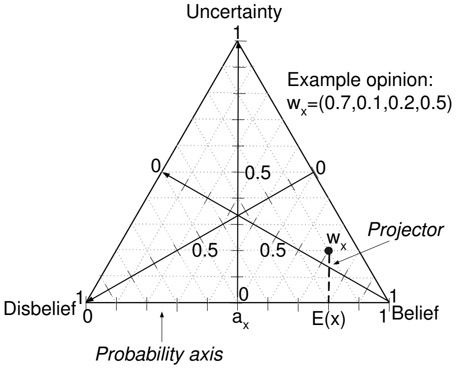

Fig. 1 Opinion triangle with an example opinion [10].

trust transitivity which is widely discussed in literature [3, 8, 19, 24, 27]. That is, if *A* trusts *B* who trusts *C*, then *A* will also trusts *C*. A valid transitive trust path requires that the last edge in the path represents functional trust and that all other edges in the path represents referral trust. Referral trust transitivity and parallel combination of trust paths are expressed as part of TNA-SL model (figure 2) [12].

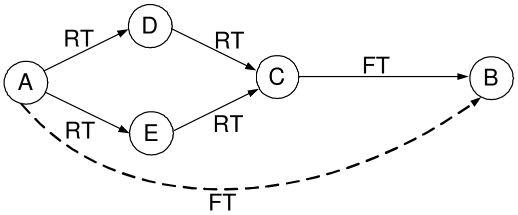

Fig. 2 Referral trust transitivity and parallel combination of trust paths.

The discounting operator (⊗) [11] is used to derive trust from transitive trust paths, and the consensus operator (⊕) allows to combine parallel transitive trust paths. The trust network in figure 2 can then be expressed as

$$FT_B^A = ((RT_D^A \otimes RT_C^D) \oplus (RT_E^A \otimes RT_C^E)) \otimes FT_B^C$$

While we consider TNA-SL and the Subjective Logic as a suitable fundament for our similarity model, it can be, as already mentioned, adapted to all trust management models enabling to combine referral and functional trust (e.g., [1, 7]).

3 The Proposed Model

Our model for the estimation how much trust *A* can place in *B* considers not only direct experience and recommendations but also similarities between agents with respect of trusting other agents or being trusted by other parties. The two kinds of similarities between trusters resp. trustees can be gradually expressed by triples very similar to the first three operands of the opinion quadruples such that we can use the consensus operator of the subjective logic for the trust value computation.

3.1 Similar Trustees

If A has functional trust in C who is similar to B (they are *similar trustees*), then A can infer its functional trust to B ([3], see figure 3(a)). Two trustees are similar if they are both similarly trusted by other agents Z_1, Z_2, ..., Z_n (figure 3(b)). This is an extension of TNA-SL in which it is not possible to infer any trust value of A towards B in a trust network.

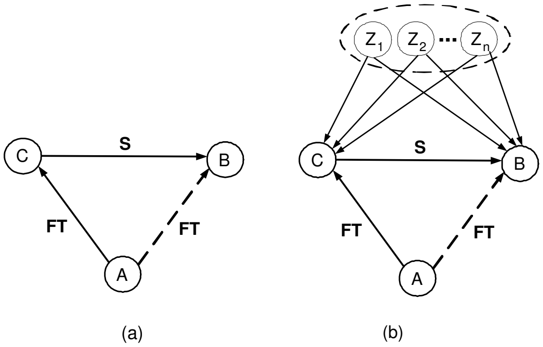

Fig. 3 (a) Similar trustees (b) Similarly trusted.

Similarly to Jøsang's way to define opinions, we use triples to describe similarity which enables us to consider uncertainty. In particular, the degree of similarity depends on the number n of agents Z_1, Z_2, ..., Z_n used for the computation reflecting that we are more certain about the similarity of two parties if they are trusted by a significant large number of other agents in an akin way.

Definition 1. The similarity opinion S_B^C from C towards B is the triple[2] (*similarity*, *non-similarity*, *uncertainty*). If $C = B$, the similarity opinion is defined to be $(1,0,0)$. Otherwise, it is calculated based on the measure $sim_{te}(C,B)$ of similarity between the two trustees C and B which is introduced in subsection 3.3:

$$S_B^C = \left(\frac{n \cdot sim_{te}(C,B)}{c+n}, \frac{n \cdot (1 - sim_{te}(C,B))}{c+n}, \frac{c}{c+n}\right) \tag{1}$$

c is a constant determining how fast uncertainty is replaced by assurance. As higher its value is, as more agents are needed to reduce the uncertainty value in favor of the

[2] This metric is inferred from a metric for the trust value computation [13] by Jøsang and Knapskog.

similarity and non-similarity values. The similarity opinion fulfills the constraints that the sum of all three values is equal to 1.

Our similarity opinion is a special form of referral trust. It reflects that the akin trust evaluations of B and C by several other trusters are a kind of recommendation by these agents to A to treat B and C similarly. Thus, we see the discounting operator $\otimes$ as the correct mechanism to combine the similarity opinion between B and C with the functional trust of A in C in order to infer the functional trust of A in B:

$$FT_B^A = S_B^C \otimes FT_C^A \tag{2}$$

As higher the similarity between B and C is, as closer the trust of A to B will equal to that between A and C. As lower this similarity is, as more uncertain A will be about whether to trust B or not.

3.2 Similar Trusters

If C has functional trust to B and A is similar to C (they are *similar trusters*), then A can also infer functional trust towards B ([3], see figure 4(a)). We call C and A similar trusters if they have alike trust in several other agents Z_1, Z_2, ..., Z_n. In this case, if C has functional trust to a new agent B, then A can infer a functional trust to B (figure 4(b)). Again using TNA-SL alone, there is no way to infer a new trust value.

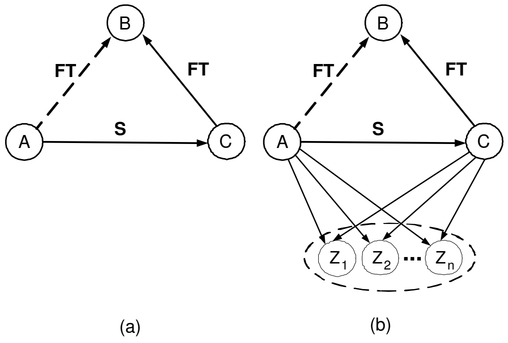

Fig. 4 (a) Similar trusters (b) Similarly trusting.

Like (1), the similarity opinion S_C^A from A to C is calculated using the measure of similarity $sim_{tr}(C,A)$ between trusters which is also introduced in subsection 3.3:

$$S_C^A = (\frac{n \cdot sim_{tr}(C,A)}{c+n}, \frac{n \cdot (1 - sim_{tr}(C,A))}{c+n}, \frac{c}{c+n}) \tag{3}$$

This similarity opinion is discounted by the functional trust FT_B^C from C to B to form the new trust value.

$$FT_B^A = S_C^A \otimes FT_B^C \tag{4}$$

3.3 Similarity Calculation

In order to measure similarities, we model trusters, trustees, and trust relationships as a graph with nodes representing trusters and trustees and edges representing trust relations. The intuition behind our algorithm is that, *similar* trustees are related to *similar* trusters. More precisely, trusters A and B are similar if they are related to trustees C and D, respectively, and C and D are themselves similar. The base case is that each node is similar to itself. If we call this graph G, then we can form a node-pair graph G^2 in which each node represents an ordered pair of nodes of G as depicted in figure 5. A node (A,B) of G^2 points to a node (C,D) if, in G, A points to C and B points to D. Similarity scores are symmetric, so for clarity we draw (A,B) and (B,A) as a single node A,B (with the union of their associated edges) [9].

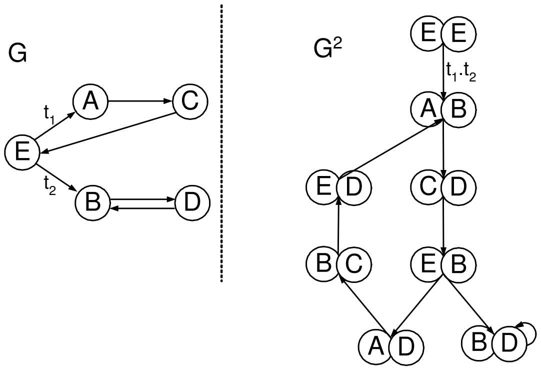

Fig. 5 Similarity measurement.

We propose an iterative fixed-point algorithm on G^2 to compute similarity scores[3] for node-pairs in G^2. The similarity score for a node υ of G^2 gives a measure of similarity between the two nodes of G represented by υ. Scores can be thought of as flowing from a node to its neighbors. Each iteration propagates scores one step forward along the direction of the edges, until the system stabilizes (i.e., scores converge). Since nodes of G^2 represents pairs in G, similarity is propagated from pair to pair. Under this computation, two trustees are similar if they are trusted by similar trusters.

For each iteration k, iterative similarity functions $sim_{te,k}(*,*)$ for trustees and $sim_{tr,k}(*,*)$ for trusters are introduced. The iterative computation is started with $sim_{0,*}(*,*)$ defined as

[3] An alternative approach to measure this similarity is to model an agent's mental structure as an ontology and using various methods proposed in our previous work [25, 26]

$$sim_{0,*}(A,B) = \begin{cases} 1, & if A = B \\ 0, & if A \neq B \end{cases} \tag{5}$$

On the $(k+1)$-th iteration, $sim_{*,k+1}(*,*)$ is defined in special cases as

$$\begin{aligned} sim_{*,k+1}(A,B) = 1, \quad & if A = B \\ sim_{te,k+1}(A,B) = 0, \quad & if\ I(A) = \emptyset\ or\ I(B) = \emptyset \\ sim_{te,k+1}(A,B) = 0, \quad & if\ O(A) = \emptyset\ or\ O(B) = \emptyset \end{aligned} \tag{6}$$

$I(A)$ is the set of in-neighbors of A while $O(A)$ specifies the set of A's out-neighbors. Individual in-neighbors are denoted as $I_i(A)$, for $1 \leq i \leq |I(A)|$, and individual out-neighbors are denoted as $O_i(A)$, for $1 \leq i \leq |O(A)|$. $sim_{te,k+1}(*,*)$ is computed from $sim_{tr,k}(*,*)$ in the general case as follows:

$$sim_{te,k+1}(A,B) = \frac{\sum_{i=1}^{n}\sum_{j=i}^{n} sim_{tr,k}\left(I_i(A), I_j(B)\right) \cdot (1 - \text{distance}(I_i(A), I_j(B), A, B))}{\sum_{i=1}^{n}\sum_{j=i}^{n} sim_{tr,k}\left(I_i(A), I_j(B)\right)} \tag{7}$$

and $sim_{tr,k+1}(*,*)$ is computed from $sim_{te,k}(*,*)$ in the general case as:

$$sim_{tr,k+1}(A,B) = \frac{\sum_{i=1}^{n}\sum_{j=i}^{n} sim_{te,k}\left(O_i(A), O_j(B)\right) \cdot (1 - \text{distance}(A, B, O_i(A), O_j(B)))}{\sum_{i=1}^{n}\sum_{j=i}^{n} sim_{te,k}\left(O_i(A), O_j(B)\right)} \tag{8}$$

Formulas (7) and (8) are alternately computed in iterations until the resulting similarity values sim_{tr} and sim_{te} converge. The corresponding algorithm is sketched as the procedure *CalculateSimilarity* in figure 4.1.

The *distance* function is used to compare trust relations. $distance(A,B,C,D)$ expresses the difference between the trust from A, B to C, D. It averages the Euclidean distances between the trust values of A and C resp. B and D on the opinion triangle (see figure 1):

$$\begin{aligned} \text{distance}(A,A,C,D) &= \sqrt{(b_{AC} + \tfrac{1}{2}u_{AC} - b_{AD} - \tfrac{1}{2}u_{AD})^2 + \tfrac{3}{4}(u_{AC} - u_{AD})^2} \\ \text{distance}(A,B,C,C) &= \sqrt{(b_{AC} + \tfrac{1}{2}u_{AC} - b_{BC} - \tfrac{1}{2}u_{BC})^2 + \tfrac{3}{4}(u_{AC} - u_{BC})^2} \\ \text{distance}(A,B,C,D) &= \begin{cases} \tfrac{1}{2}(\sqrt{(b_{AC} + \tfrac{1}{2}u_{AC} - b_{BD} - \tfrac{1}{2}u_{BD})^2 + \tfrac{3}{4}(u_{AC} - u_{BD})^2} \\ +\sqrt{(b_{AD} + \tfrac{1}{2}u_{AD} - b_{BC} - \tfrac{1}{2}u_{BC})^2 + \tfrac{3}{4}(u_{AD} - u_{BC})^2}) \end{cases} \end{aligned} \tag{9}$$

For the sake of simplicity, all base rate values (a_{AD}, a_{AC}, a_{BD}, a_{BC}) are assumed to be $\frac{1}{2}$. The factor $\frac{3}{2}$ is used for the vertical axis to adapt the measures. Otherwise, the opinion triangle would be compressed and the distance between the points (0,1,0) and (0,0,1) would not be equal to one. Figure 6 illustrates the *distance* function graphically.

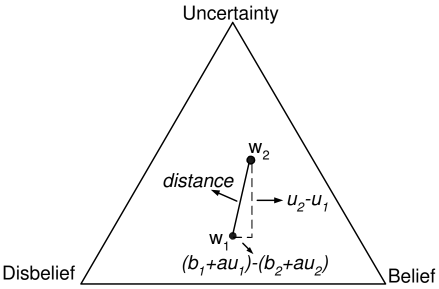

Fig. 6 The distance between opinions.

4 Evaluation

We chose a publicly available dataset taken from a real system known as Advogato [2]. Advogato (`http://advogato.org`) is an online community site dedicated to free software development. On Advogato a user can certify another user as “Master”, “Journeyer”, “Apprentice” or “Observer”, based on the perceived level of involvement in the free software community. The Advogato social network is an example of a real-world, directed, weighted, large social network. There are indeed other web communities using the same software powering Advogato.org and they also have reached similar trust levels and use the same certifications system, but we do not use them for our analysis in this paper, mainly because:

- Our model is based on user-user trust matrix and not the user-item rating matrix.
- They are much smaller than the Advogato dataset.

4.1 Dataset

Precise rules for giving out trust statements are specified on the Advogato site. *Masters* are supposed to be principal authors of an “important” free software project, excellent programmers who work full time on free software. *Journeyers* contribute significantly, but not necessarily full-time. *Apprentices* contribute in some way, but are still acquiring the skills needed to make more significant contributions. *Observers* are users without trust certification, and this is the default. It is also the level at which a user certifies another user to remove previously expressed trust certifications.

The Advogato dataset is a directed, weighted graph with 11934 nodes and 57610 trust relations. There are 18053 Master judgments, 23091 for Journeyer, 10708 for Apprentice and 5758 for Observers. Figure 7 illustrates the allocation of ratings that correspond to each user. In our tests, we apply our model to 3 different datasets and the results are averaged. Each 3000 users built a trust graph of approximately 4000 relations.

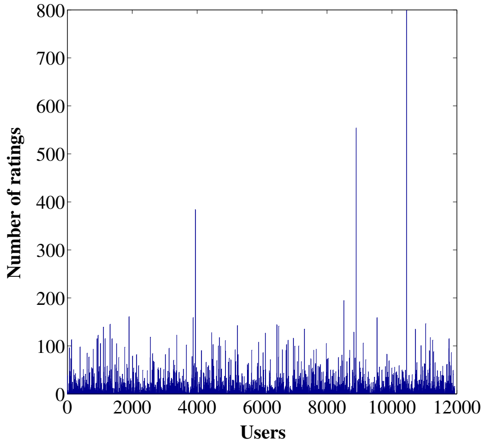

Fig. 7 Users' rating activity.

For the purpose of this paper, we consider these certifications as trust statements. Trust statements are directed and not necessarily symmetric. By aggregating the trust statements expressed by all the members of the community it is possible to build the entire trust network. A trust network is hence a directed, weighted graph. Arbitrarily, we map the textual labels Observer, Apprentice, Journeyer and Master respectively to rating values 0, 1, 2, 3. which have to be yet converted to subjective logic opinions. In general, with n-level rating values (in our case $n = 3$) in which the number of ratings of level i is described by function $f(i)$, we can use the following conversion method in which c is a constant:

$$b = \frac{\sum_{i=1}^{n} i \cdot f(i)}{c + n \cdot \sum_{i=0}^{n} f(i)}, \quad d = \frac{\sum_{i=0}^{n-1} (n-i) \cdot f(i)}{c + n \cdot \sum_{i=0}^{n} f(i)}, \quad u = \frac{c}{c + n \cdot \sum_{i=0}^{n} f(i)} \tag{10}$$

In this formula, the highest rating value 3 is mapped to three positive valuations, while 2 corresponds to two positive valuations and a negative one, etc.

Algorithm 4.1: EVALUATION($users, trust\_graph$)

```
procedure CALCULATESIMILARITY($users, trust\_graph$)
  repeat
    for each $i, j \in users$
      do if $i = j$
      then $similarity\_matrix[i, j] \leftarrow (1,0,0)$
      else
        if $i < j$
          then $neighbors \leftarrow$ common in-neighbors of i and j
        comment: similarity of trustees
          else $neighbors \leftarrow$ common out-neighbors of i and j
        comment: similarity of trusters
        if $number\_of\_neighbors == 0$
          then $sim \leftarrow 0$
          else $sim \leftarrow$ GETSIMILARITY($neighbors$)
        comment: According to (7) and (8)
        $similarity\_matrix[i, j] \leftarrow$ GETOPINION($sim, number\_of\_neighbors$)
        comment: According to (1)
  until $converge$
  return ($similarity\_matrix$)

procedure PREDICTTRUSTEDGE($(i, j), trust\_graph$)
  $opinion \leftarrow (0,0,1)$
  for each $k \in users - \{i, j\}$
    do
      $similarity\_trustee(k, j) \leftarrow similarity\_matrix[min(k, j), max(k, j)]$
      $similarity\_truster(i, k) \leftarrow similarity\_matrix[max(i, k), min(i, k)]$
      $predicted\_opinion\_te \leftarrow trust\_opinion(i, k) \otimes similarity\_trustee(k, j)$
      $predicted\_opinion\_tr \leftarrow trust\_opinion(k, j) \otimes similarity\_truster(i, k)$
      $opinion \leftarrow (opinion \oplus predicted\_opinion\_te \oplus predicted\_opinion\_tr)$
  return ($opinion$)

procedure DOEVALUATION($trust\_graph, predicted\_trust\_graph$)
  $coverage \leftarrow$ number of predicted edges in predicted_trust_graph
  $fcpe \leftarrow$ fraction of correctly predicted edges
  $mae \leftarrow$ mean absolute error of predicted values
  $rmse \leftarrow$ root mean squared error of predicted values
  output ($coverage, fcpe, mae, rmse$)

main
  global $similarity\_matrix \leftarrow$ CALCULATESIMILARITY($users, trust\_graph$)
  for each $edge \in trust\_graph$
    do
      $predicted\_edge \leftarrow$ PREDICTTRUSTEDGE($edge, trust\_graph - edge$)
      $predicted\_trust\_graph \leftarrow predicted\_trust\_graph \cup predicted\_edge$
  DOEVALUATION($trust\_graph, predicted\_trust\_graph$)
```

4.2 Plan

We use the leave-one-out technique [4] (a machine learning evaluation technique) to show the performance of our approach. Leave one out involves hiding one trust edge and then trying to predict it. The predicted trust edge is then compared with the real edge (using the distance function) and the difference is the prediction error. This procedure is repeated for all edges in the trust graph. The real and the predicted values are then compared in several ways: the coverage, which refers to the fraction of edges for which, after being hidden, the algorithm is able to produce a predicted edge, FCPE which is the fraction of correctly predicted edges, MAE (mean absolute error) which is average of the prediction error over all edges, and RMSE (root mean squared error) which is the root mean of the average of the squared prediction error. RMSE tends to emphasize large errors.

The evaluation can be described in pseudo-code as in algorithm 4.1. First, the similarity matrix is calculated by calling the procedure *CalculateSimilarity* from the main procedure. Since similarity is symmetric, the similarity of trustees is stored in the lower triangle of the similarity matrix and the similarity of trusters in the upper triangle. Next, for each edge in the real trust graph, an equivalent trust edge is calculated by calling procedure *PredictTrustEdge*. This procedure takes the real trust graph without that edge as an input. The predicted edges form the predicted trust graph. Finally, the real and predicted trust graph are compared according to the four metrics (coverage, FCPE, MAE, and RSME) by calling procedure *DoEvaluation*.

4.3 Results Summary

Figure 8 depicts the similarity measures among the first 150 users. For each two users, their similarity as trustees is in the lower triangle of the similarity matrix and their similarity as trusters is in the upper triangle of the similarity matrix.

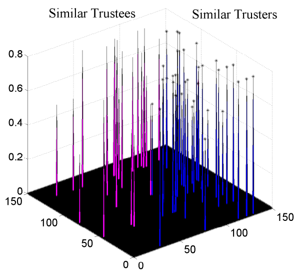

Fig. 8 Similarity measures among the first 150 users

In table 1 we present the final results of the evaluation. We start by commenting the column “coverage”. The coverage becomes an important issue on a very sparse dataset that contains a large portion of cold start users since many trust values become hardly predictable [17]. Our baseline is a method called “Random” which randomly generates trust edges. Results ($coverage \approx 0.6$) indicate that our model is able to predicate approximately one edge from each two existing edges. The second important result is the fraction of correctly predicted edges (FCPE) which is 0.8. It shows that from each 10 predicted edge 8 edges are predicted correctly. Further, prediction errors (MAE and RMSE) computed are small in comparison with the Random method ($MAE \approx 0.14$ & $RMSE \approx 0.18$).

Table 1 Final evaluation results

Metric	Dataset1	Dataset2	Dataset3	Average	Random
Coverage	0.5783	0.5678	0.6520	0.5994	1
FCPE	0.8169	0.8299	0.8227	0.8232	0.3068
MAE	0.1389	0.1427	0.1409	0.1408	0.4570
RMSE	0.1823	0.1828	0.1864	0.1838	0.5036

Figure 9 shows the sparsity of the trust graph before and after prediction for the first dataset. The sparseness has been decreased significantly. All-in-all, the results of the evaluation lead to the expectation that the method TILLIT will increase the coverage of trust relationships significantly, and that the accuracy of the predicted additional will be fairly high as well.

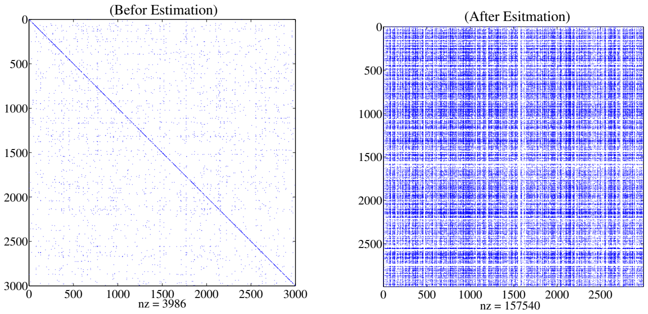

Fig. 9 Sparsity of the trust graph before and after prediction for the first dataset

5 Related Research

Most popular approaches proposed to deal with the sparsity problem include dimensionality reduction of the user-item matrix, application of associative retrieval techniques in the bipartite graph of items and users, item-based similarity instead of user-based similarity, and content-boosted collaborative filtering (see [21]). The dimensionality reduction approach addresses the sparsity problem by removing unrepresentative or insignificant users or items so as to condense the user-item matrix. We briefly explain those which are based on similarity measurement and thus more closely resemble our work. These approaches can be categorized in two groups: rating-based similarity and profile-based similarity.

In [22, 23], Pitsilis and Marshall explain how similarity can benefit from special characteristics of trust such as the ability to propagate along chains of trusted users; in this way similarity can support transitivity. In their model they use ordinary measures of similarity taken from collaborative filtering to form the potential trust between the users which would be propagated in a similar way to the word-of-mouth scheme through a trust graph. Finally, by transforming the value back into similarity measure terms, it could be made appropriate for use in collaborative filtering algorithms. More specifically, for each pair of users they first calculate how similar they are, applying Pearsons correlation coefficient formula over the user-item ratings, and then they calculate the indirect trust between them. Next, this trust value is converted to a similarity metric using their formula.

Massa et al. present in [16, 18] evidence that, by incorporating trust, recommender systems can be more effective than systems based on traditional techniques like collaborative filtering. They show how the similarity measure, on average, is computable only against a very small portion of the user base and is, in most cases, a noisy and unreliable value because computed on few items rated in common by two users. Instead, trust-aware techniques can produce a trust score for a very high number of other users; the trust score of a user estimates the relevance of that users' preferences. In this paper, similarity is measured using Pearsons correlation coefficient on user-item ratings.

A number of techniques for performing collaborative filtering from the point of view of a trust-management problem are outlined in [15]. In this work authors propose a variation of k-nearest neighbor collaborative filtering algorithm for trusted k-nearest recommenders. This algorithm allows users to learn who and how much to trust one another by evaluating the utility of the rating information they receive. They mainly address the problem of learning how much to trust rating information that is received from other users in a recommender system.

A model for computing trust-based reputation for communities of strangers is proposed in [5]. The model uses the concept of knots, which are sets of members having high levels of trust in each other. Different knots typically represent different view points and preferences. The assumption underlying this knot-aware reputation model is that use of relatively small, but carefully selected, subsets of the overall community's reputation data yields better results than those represented by the full dataset.

In [20], O'Donovan and Smyth argue that profile similarity on its own may not be sufficient, that other factors might also have an important role to play. Specifically they introduce the notion of trust in reference to the degree to which one might trust a specific profile when it comes to make a specific rating prediction. They develop two different trust models, one that operates at level of the profile and one at level of the items within a profile. In both of these models trust is estimated by monitoring the accuracy of a profile at making predictions over an extended period of time. Trust then is the percentage of correct predictions that a profile has made in general (profile-level trust) or with respect to a particular item (item-level trust).

In [28], the authors experimentally prove that there exists a significant correlation between the trust expressed by the users and their profile similarity based on the recommendations they made in the system. This correlation is further studied as survey-based experiments in [6].

In this paper we provide an alternative approach to deal with the sparsity problem. We measure similarity based on the users' trust relationships, i.e. trust graph structure and trust values (in contrast to the other approaches which have used user-item ratings or profile similarity), and propose novel formulas to convert it to subjective logic opinions. The consideration of these similarities leads to extra information accessible for trust inferences.

6 Discussion and Conclusion

In order to overcome sparseness of the web of trust, we consider users' similarity as a factor to derive trust connectivity and trust values. The main idea is that we account two persons similar if either a fair number of others have akin trust in them or if they themselves trust several other people alike. In the first case, every person who has trust in one of them can infer similar trust to the other one, at least as an estimated starting value. In the second case, a person may infer the trust value of a third party from other trusters similar to her.

We consider a similarity-based recommendation system for singers and songs as a good application example for our model. Normally, in systems like iTunes only the most popular songs or other songs of artists, of whom one already has bought songs, are advertised without any guarantee that one likes these songs as well. Using our approach, it is possible to find other customers who have an akin taste about music as the customer Alice reading the advertisements. Songs rated positively by these customers but not bought yet by Alice can be advertised to her since she will like them probably as well. This will make Alice more receptive to the advertisements.

In the future, we aim to evaluate the accuracy of a whole recommender system that employs our proposed model. Furthermore, we assess the possibility of modeling some of other trust propagation methods using our approach. An example is transposition resp. reciprocity [8] assuming that A's trust in B causes B to develop also some level of trust towards A. Another propagation method is Coupling, in which A's trust in C propagates to B because C and B trust people in common [8]. This propagation rule is depicted in figure 10. According to this rule we can use the similarity between trusters to propagate the trust in one trustee to another.

Moreover, one can use similarity in a complete different way. Trust is very specific and nobody trusting Bob as a good car mechanic will automatically trust him

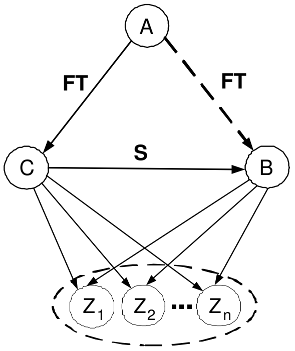

Fig. 10 Coupling: a trust propagation method.

also in undertaking heart surgeries. But probably, he will be capable in repairing motorcycles. Thus, there is a large similarity between the domains of repairing cars and motorcycles but a very low one between both of these and medical surgery. We think to use trust relations in one domain to infer ones in similar domains and consider ontologies describing the degrees of similarity between the domains as a useful means. All-in-all, we are convinced, that the various forms of similarity are good vehicles to tackle the major problem of too sparse webs of trust in online communities.

References

1. A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *Proceedings of the 33rd Hawaii International Conference, Volume 6*, Maui, Hawaii, 2000. IEEE Computer Society Press.
2. http://www.trustlet.org/wiki/Advogato_dataset.
3. L. Ding, P. Kolari, S. Ganjugunte, T. Finin, and A. Joshi. Modeling and Evaluating Trust Network Inference. Technical report, MARYLAND UNIV BALTIMORE DEPT OF COMPUTER SCIENCE AND ELECTRICAL ENGINEERING, 2005.
4. K. Fukunaga and D.M. Hummels. Leave-one-out procedures for nonparametric error estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):421–423, 1989.
5. N. Gal-Oz, E. Gudes, and D. Hendler. A Robust and Knot-Aware Trust-Based Reputation Model. In *Proceedings of IFIPTM 2008 - Joint iTrust and PST Conferences on Privacy, Trust Management and Security*, pages 167–182. Springer, 2008.
6. J. Golbeck. Trust and nuanced profile similarity in online social networks. *Journal of Artificial Intelligence Research*, 2006.
7. T. Grandison and M. Sloman. Specifying and analysing trust for internet applications. In *Proceedings of the 2nd IFIP Conference on E-Commerce, E-Business & E-Government (I3E)*, pages 145–157, Lisbon, 2002. Kluwer Academic Publisher.
8. R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM Press New York, NY, USA, 2004.
9. G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 538–543. ACM Press New York, NY, USA, 2002.

10. A. Jøsang. A Logic for Uncertain Probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(3):279–311, 2001.
11. A. Jøsang. The consensus operator for combining beliefs. *Artificial Intelligence*, 141(1-2):157–170, 2002.
12. A. Jøsang, R. Hayward, and S. Pope. Trust network analysis with subjective logic. In *Proceedings of the 29th Australasian Computer Science Conference-Volume 48*, pages 85–94. Australian Computer Society, 2006.
13. A. Jøsang and S. J. Knapskog. A metric for trusted systems. In *Proceedings of the 21st National Security Conference*. NSA, 1998.
14. Y.A. Kim, M.T. Le, H.W. Lauw, E.P. Lim, H. Liu, and J. Srivastava. Building a web of trust without explicit trust ratings. In *Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on*, pages 531–536, 2008.
15. N. Lathia, S. Hailes, and L. Capra. Trust-Based Collaborative Filtering. In *Proceedings of IFIPTM 2008 - Joint iTrust and PST Conferences on Privacy, Trust Management and Security*, pages 119–134. Springer, 2008.
16. P. Massa and P. Avesani. Trust-Aware Collaborative Filtering for Recommender Systems. *LECTURE NOTES IN COMPUTER SCIENCE*, pages 492–508, 2004.
17. P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24. ACM Press New York, NY, USA, 2007.
18. P. Massa and B. Bhattacharjee. Using Trust in Recommender Systems: An Experimental Analysis. In *Trust Management: Second International Conference, ITrust 2004, Oxford, UK, March 29-April 1, 2004: Proceedings*. Springer, 2004.
19. R. Morselli, B. Bhattacharjee, J. Katz, and M. Marsh. Exploiting approximate transitivity of trust. In *Broadband Communications, Networks and Systems, 2007. BROADNETS 2007. Fourth International Conference on*, pages 515–524, 2007.
20. J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces*, pages 167–174. ACM New York, NY, USA, 2005.
21. M. Papagelis, D. Plexousakis, and T. Kutsuras. Alleviating the sparsity problem of collaborative filtering using trust inferences. In *Proceedings of iTrust*, pages 224–239. Springer, 2005.
22. G. Pitsilis and L. Marshall. Model of Trust Derivation from Evidence for Use in Recommendation Systems. Technical report, Newcastle Universty, School of Computing Science, 2004.
23. G. Pitsilis and L.F. Marshall. Modeling Trust for Recommender Systems using Similarity Metrics. In *Proceedings of IFIPTM 2008 - Joint iTrust and PST Conferences on Privacy, Trust Management and Security*, pages 103–118. Springer, 2008.
24. D. Quercia, S. Hailes, and L. Capra. Lightweight Distributed Trust Propagation. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 282–291, 2007.
25. M. Tavakolifard, S. Knapskog, and P. Herrmann. Cross-Situation Trust Reasoning. In *Proceedings of The Workshop on Web Personalization, Reputation and Recommender Systems (WPRRS08)*. IEEE Computer Society Press, 2008.
26. M. Tavakolifard, S. Knapskog, and P. Herrmann. Trust Transferability Among Similar Contexts. In *Proceedings of The 4th ACM International Workshop on QoS and Security for Wireless and Mobile Networks (Q2SWinet 2008)*. ACM, 2008.
27. Y. Yang, ACT Canberra, L. Brown, S. Wales, ACT ADFA, E. Lewis, and V.A. Melbourne. W3 Trust Model: Evaluating Trust and Transitivity of Trust of Online Services. In *International Conference on Internet Computing*, pages 354–362, 2002.
28. C.N. Ziegler and J. Golbeck. Investigating interactions of trust and interest similarity. *Decision Support Systems*, 43(2):460–475, 2007.