Start Trusting Strangers? Bootstrapping and Prediction of Trust⋆

Florian Skopik, Daniel Schall, and Schahram Dustdar

Distributed Systems Group, Vienna University of Technology
Argentinierstr. 8/184-1, A-1040 Vienna, Austria
{skopik,schall,dustdar}@infosys.tuwien.ac.at

Abstract. Web-based environments typically span interactions between humans and software services. The management and automatic calculation of trust are among the key challenges of the future service-oriented Web. Trust management systems in large-scale systems, for example, social networks or service-oriented environments determine trust between actors by either collecting manual feedback ratings or by mining their interactions. However, most systems do not support bootstrapping of trust. In this paper we propose techniques and algorithms enabling the prediction of trust even when only few or no ratings have been collected or interactions captured. We introduce the concepts of *mirroring* and *teleportation* of trust facilitating the evolution of cooperation between various actors. We assume a user-centric environment, where actors express their opinions, interests and expertises by selecting and tagging resources. We take this information to construct tagging profiles, whose similarities are utilized to predict potential trust relations. Most existing similarity approaches split the three-dimensional relations between users, resources, and tags, to create and compare general tagging profiles directly. Instead, our algorithms consider (i) the understandings and interests of actors in tailored subsets of resources and (ii) the similarity of resources from a certain actor-group's point of view.

1   Introduction

Trust and reputation systems are essential for the success of large-scale Web systems. In such systems usually information, provided by users or obtained during their interactions, is collected to detect beneficial social connections, mostly leading to trust between their members. While many trust-, recommendation- and reputation systems have been described, including there underlying models and modes of operation [1–3], one particular problem has been mostly neglected: How to put the system into operation, i.e., how to bootstrap trust between users to let them benefit from using the system; even if there is not yet much, or even no, collected data available.

Our prior work [4] describes, an environment comprising humans and services, in which interactions spanning both kinds of entities are monitored. We strongly believe that trust can only be based on the success and outcome of previous interactions [4, 5]. Without having knowledge of prior (observed) interactions, we argue that trust between users cannot be determined in a reliable manner. Therefore, we propose an approach for *trust prediction* that aims at compensating the issue of bootstrapping trust. We consider influencing factors stimulating the evolution of trust. In various environments, such as collaborative systems, trust is highly connected to interest similarities and capabilities of the actors. For instance, if one actor, such as a human or service, has the capabilities to perform or support a collaborative activity reliably, securely and dependably, it may be sensed more trustworthy than other actors. Moreover, we argue, that if actors have interests or competencies similar to well-known trusted actors, they may enjoy initial trust to some extend.

The contributions of this paper are as follows. First, we introduce our concepts to trust prediction, and model the application environment. Second, we present our approach for creating and comparing tagging profiles based on clustering, and a novel method for trust prediction using similarity measurements. Third, we evaluate our algorithms using real world data sets from the tagging community `citeulike`[1], and show a reference implementation of our approach.

The remainder of the paper is organized as follows. Sect. 2 is about the motivation and concepts of trust prediction. In Sect. 3 we model the tagging environment, and describe our approach in Sect. 4. The results of the evaluation are depicted in Sect. 5. In Sect. 6 we introduce the architecture of a framework utilizing our new approach. Related work is listed in Sect. 7. We conclude and show our future plans in Sect. 8.

2 Towards Prediction of Trust

Trust between entities can be managed in a graph model (for example, see [6]). The graph is defined as $G = (V, E)$ composed of the set V of vertices defining entities trusting each other and the set E of directed edges denoting trust relations between entities. This model is known as the *Web of Trust*.

In Fig. 1 four different scenarios are depicted, which show concepts for trust determination in a web of trust. We assume a situation where trust from entitiy a to entitiy c is to be determined. The first case in Fig. 1(a) visualizes the optimal case, in which a trust relation from a to c can be inferred directly, e.g., based on previous interactions [4]. In the second case in Fig. 1(b), no direct trust relation could be determined, however trust can be propagated if we assume transitivity of trust relations [2], enabling b to recommend c to a. The third case in Fig. 1(c) depicts, that there is neither a direct nor a propagated trust relation from a to c. However, unrelated third party entities d and e may provide a weaker, but acceptable, notion of trust in c through the means of reputation. For our work

---

[1] http://www.citeulike.org

the fourth use case in Fig. 1(d) is the most interesting one, which demonstrates the limitations of the web of trust. If no one interacted with c in the past and no one has established trust to c, our trust prediction approach needs to be applied.
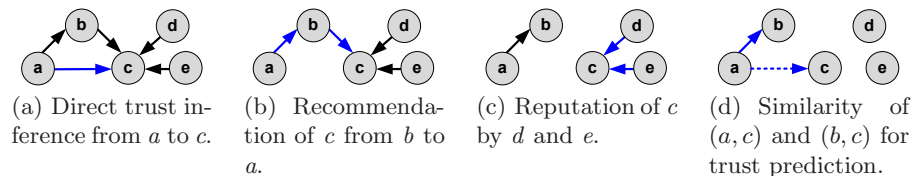

(a) Direct trust inference from a to c.

(b) Recommendation of c from b to a.

(c) Reputation of c by d and e.

(d) Similarity of (a, c) and (b, c) for trust prediction.

Fig. 1. trust(a,c)=?: The need for trust prediction in a web of trust

We distinguish the following both modes of trust prediction:

– **Trust mirroring.** Depending on the environment, interest and competency similarities of people can be interpreted directly as an indicator for future trust. This is especially true in environments where actors have the same or similar roles (e.g., online social platforms). There is strong evidence that actors "similar minded" tend to trust each other more than any random actors [7, 8]; e.g., movie recommendations of people with same interests are usually more trustworthy than the opinions of unknown persons. In Fig. 1(d), this means measuring the similarity of a's and c's interests, allows, at least to some extent, trust prediction (dashed line).
– **Trust teleportation.** As depicted by Fig. 1(d), we assume that a has established a trust relationship with b in the past, for example, based on b's capabilities to assist a in work activities. Therefore, others having similar interests and capabilities as b may become similarly trusted by a in the future. In contrast to mirroring, trust teleportation is applied in environments comprising actors with different roles. For example, a manager might trust a developer belonging to a certain group. Other members in the same group may benefit from the existing trust relationship by being recommended as trustworthy as well. We attempt to predict the amount of future trust from a to c by comparing b's and c's interests and capabilities.

Sophisticated profile similarity measurements are needed in both cases to realize our vision of trust prediction.

3 Tagging Environment

According to our concepts of trust prediction, we need models to manage the interests and competencies of humans, and features of resources, e.g., services, respectively. In contrast to traditional centralized approaches where, one instance such as the human resource department, manages a catalog of competencies, we follow a **dynamic self-managed user-centric approach**. We assume an environment where each actor tags different types of resources s/he is interested
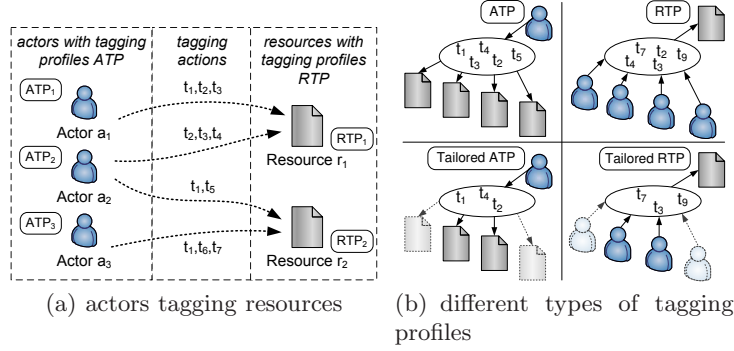
(a) actors tagging resources

(b) different types of tagging profiles

Fig. 2. Description of the tagging environment

in, such as bookmarks, scientific papers and Web services. Based on the kind of resource tagged and the tags assigned, we can infer the centers of interest, expressing to some extent their knowledge areas and capabilities; but from a community's view also the features or preferred usage of tagged resources. By utilizing this knowledge and applying our concepts of *trust mirroring* and *trust teleportation*, we think it is possible to predict trust relations potentially emerging in the future.

We model the environment as depicted in Fig. 2(a) which consists of:

– a set of actors A, having different interests reflected by actor-tagging-profiles (ATP). These profiles are derived from tags $T' \subseteq T$ used by $a_i \in A$ on a subset of resources $R' \subseteq R$.
– a set of resources R, having different properties (covering actor interests) reflected by resource-tagging-profiles (RTP). These profiles are derived from tags $T' \subseteq T$ used by a subset of actors $A' \in A$ on $r_j \in R$.
– a set of tagging actions $T = \{t_x\}$, where each t_x is created by an actor $a_i \in A$ for a resource $r_j \in R$.

3.1 Modes of Profile Similarity Measurement

We determine *tagging profiles* for both actors (ATP) and resources (RTP) (Fig. 2(b)). ATPs express independent from particular resources, which tags are frequently used by actors and therefore, their centers of interest. RTPs describe how a particular resource is understood in general, independent from particular actors. According to our motivating scenario depicted in Fig. 1(d), ATP similarities can be either interpreted as *trust mirroring* or *trust teleportation*. In contrast to that, RTP similarities are mostly only meaningful for *trust teleportation* (e.g., Actor a trusts a resource r_j, thus s/he might trust a very similar resource r_k as well.)

Compared to general profile similarity, and common profile mining approaches, e.g. in recommender systems [9], we do not only capture which actor uses which tags (ATP) or which resource is tagged with which tags (RTP). We rather consider how an actor tags particular subsets of resources. Using such *Tailored ATP*s

we can infer similarities of tag usage between actors $a_i, a_j \in A$, and therefore similarities in understanding, using, and apprehending the same specific resources $R' \subseteq R$. Furthermore, we capture how two resources $r_i, r_j \in R$ are tagged by the same group of actors $A' \subseteq A$. Such *Tailored RTP*s can be utilized to determine similarities between resources and how they are understood and used by particular groups of actors (Fig. 2(b)).

4   Similarity-based Trust Prediction

Similarities of actors' tag usage behavior can be directly calculated if an agreed restricted set of tags is used. There are several drawbacks in real-life tagging environments that allow the usage of an unrestricted set of tags. We identified two major influencing factors prohibiting the direct comparison of tagging actions. First, synonyms cause problems as they result in tags with (almost) the same meaning but being differently treated by computer systems, e.g., `football` v.s. `soccer`. Second, terms, especially combined ones, are often differently written and therefore not treated as equal, e.g., `social-network` v.s. `socialnetwork`.

Due to the described drawbacks of comparing tagging actions directly, we developed a new approach, which measures their similarity indirectly. This approach to similarity measurement and *trust prediction*, is depicted in Fig. 3. Three steps are performed: (i) *Clustering.* Identifying tagging actions, each consisting of an actor $a_i \in A$ tagging a resource $r_j \in R$ using tags $T' = \{t_x\}, T' \subseteq T$, and hierarchically clustering tags in global interest areas (*interests tree*). (ii) *Mapping.* Mapping of actor interests and resource properties to the created tree, to construct tagging profiles. (iii) *Predicting.* Calculating similarities of ATPs and RTPs, and applying trust prediction to determine potential trust relations.

4.1   Hierarchical Clustering of Global Interest Areas

The advantage of clustering related tags is twofold: (i) we are able to identify widely used synonyms and equal, but differently written, tags (including singular/plural forms), and (ii) we are able to identify tags with similar meanings or
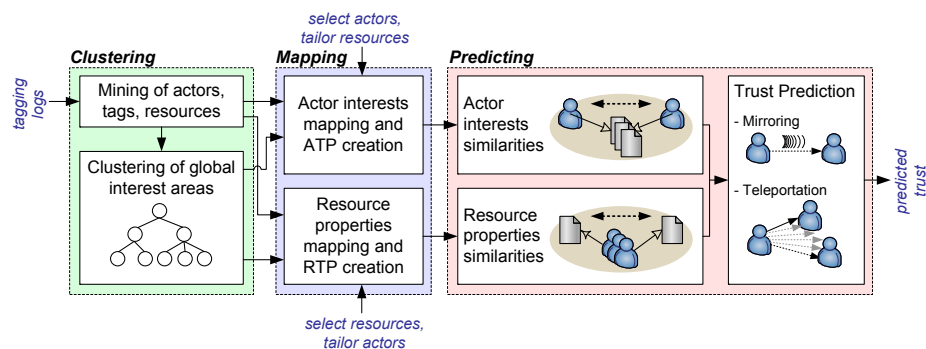

Fig. 3. An approach to trust prediction based on clustering and similarity

tags mostly used in combination. To this end, we build from the captured tagging actions a global interests tree by applying hierarchical clustering. This interests tree reflects which tags are generally applied to resources in combination, and therefore, their relatedness.

The utilized concepts are well-known from the area of information retrieval (see for instance [10]), however, while they are normally used to determine the similarities of documents based on given terms, we apply them in the opposite way. This means we determine term, i.e., tag, similarities based on given tag sets, forming kinds of documents.

The tag frequency vector $\boldsymbol{t_x}^2$ (1) describes the frequencies f the resources $R = \{r_1, r_2 \ldots r_j\}$, are tagged with tag $t_x \in T$ globally, i.e., by all actors A.

$$\boldsymbol{t_x} = \langle f(r_1), f(r_2) \ldots f(r_j) \rangle \quad . \tag{1}$$

The tag frequency matrix tfm (2), built from tag frequency vectors, describes the frequencies the resources R are tagged with tags $T = \{t_1, t_2 \ldots t_x\}$.

$$tfm = \langle \boldsymbol{t_1}, \boldsymbol{t_2} \ldots \boldsymbol{t_x} \rangle_{|R| \times |T|} \quad . \tag{2}$$

The popular tf^*idf model [10] introduces tag weighting based on the relative distinctiveness of tags (3). Each entry $tf(t_x, r_j)$ in tfm is weighted by the log of the total number of resources $|R|$, divided by the amount $n_{t_x} = |\{r_j \in R \mid tf(t_x, r_j) > 0\}|$ of resources the tag t_x has been applied to.

$$tf^*idf(t_x, r_j) = tf(t_x, r_j) \cdot \log \frac{|R|}{n_{t_x}} \quad . \tag{3}$$

Finally, the cosine similarity, a popular measure to determine the similarity of two vectors in a vector space model, is applied (4).

$$\text{sim}(\boldsymbol{t_x}, \boldsymbol{t_y}) = \cos(\boldsymbol{t_x}, \boldsymbol{t_y}) = \frac{\boldsymbol{t_x} \cdot \boldsymbol{t_y}}{||\boldsymbol{t_x}|| \cdot ||\boldsymbol{t_y}||} \quad . \tag{4}$$

We perform hierarchical clustering to the available tag vectors. This clustering approach starts by putting each tag vector $\boldsymbol{t_x}$ into a single cluster, and compares cluster similarities successively. Tag clusters are then merged bottom-up when the similarity measurement result exceeds predefined thresholds. The output of clustering is a hierarchical tree structure, i.e., a dendrogram, reflecting global interest areas and their similarity (Fig. 4). The details of the algorithm are shown in Sect. 6.

The approach can be further refined by applying the concept of latent semantic indexing (LSI) [11]. However, very common in information retrieval, this method demands for carefully selected configuration parameters not to distort the similarity measurement in our case. Our approach applies hierarchical clustering, which means tag clusters are merged based on varying similarity thresholds. Thus, we do not necessarily need a further level of fuzziness introduced by LSI.
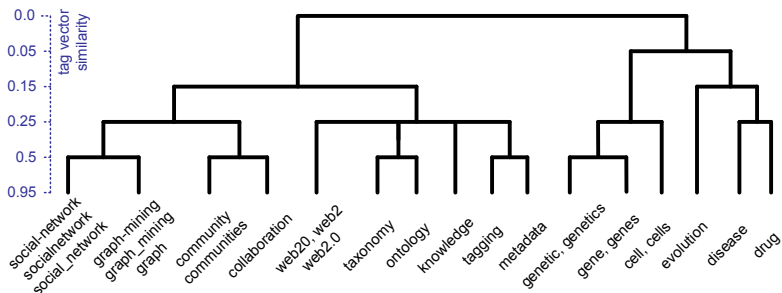
---

[2] bold printed symbols denote vectors

Fig. 4. A small part of the `citeulike` global interests tree

4.2 Tagging Profile Creation

As mentioned earlier, we create tagging profiles for both actors and resources. While ATPs describe the interests of actors, RTPs reflect features and properties of resources. The performed steps to create either kind of tagging profile are almost identical. Therefore we show exemplary the construction of ATPs in Fig. 5. For RTPs the transposed tagging matrices are used.

The upper part of the left picture (Fig. 5(a)) depicts the tree of global interests, created in the previous step. The lower part describes tagging matrices of three actors, e.g., actor a_1 tags resource r_{11} with tag t_1. In Fig. 5(b), these tagging activities are weighted and mapped to the *bottom* clusters of the interests tree (here: level 2). For this purpose, the impact w of each tag t_x on a_i's ATP is calculated by (5), assuming that the sum runs over all resources $R_{a_i} \subseteq R$ that are tagged by a_i with tag $t_x \in T_{r_j}$. Therefore, the more tags are assigned to one resource $r_j \in R_{a_i}$, the less impact one tag t_x has on the description of the resource. The assigned weights to each cluster build the ATP vectors $\boldsymbol{p_{a_i}}$ (see Fig. 5(b)).

$$w(a_i, t_x) = \sum_{\forall r_j \in R_{a_i}} \frac{1}{|T_{r_j}|} \ . \tag{5}$$

In the next steps the ATP vectors are aggregated and propagated to the upper levels, by simply building the average of all weights assigned to child clusters. Hence, new ATP vectors on a higher and more abstract level are built. Finally, the root of the interests tree is reached according to Fig. 5(b).

For each actor either all tagged resources or a representative subset (e.g., the most frequently tagged resources) is used to create the ATP. Such a general ATP reflects an actor's general interests. The same can be applied to resources, where RTPs describe their general use. Instead, tailored ATPs reflect the actor's understanding and apprehension of a particular and carefully selected subset of resources. For instance, in the case of trust prediction in a collaborative environment, resources might be selected according to their importance in an ongoing task. According to Fig. 5, this means each actor tags exactly the same resources, i.e., $r_{x1} = r_{x2} = r_{x3} \ \forall x \in \{1, 2, 3\}$. On the other hand, tailored RTPs can be used for trustworthy replacing one resource with another one, on which a particular subset of actors have similar views.
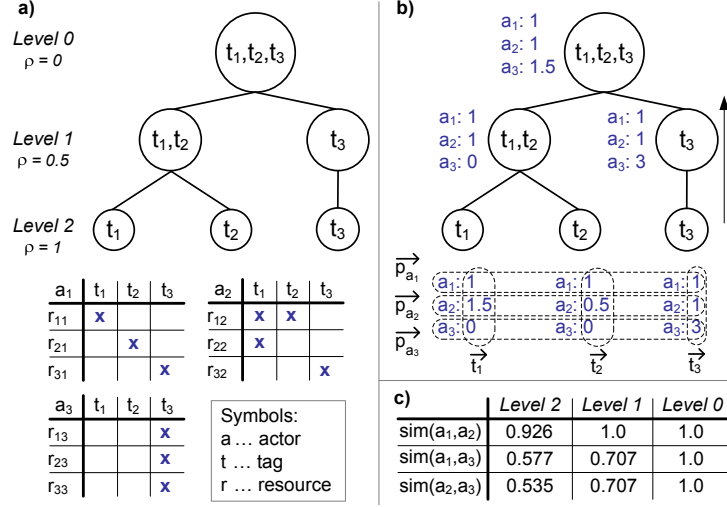
8

Fig. 5. An example for tag mapping and ATP comparison: a) interest tree and actor tagging actions. b) creating ATPs by mapping tagging actions to the tree. c) calculating ATP similarities on different tree levels

4.3 Trust Prediction

The similarity of two actors a_i and a_j is determined by the cosine of the angle between their ATP vectors $\boldsymbol{p_{a_i}}$ and $\boldsymbol{p_{a_j}}$ (cosine similarity). This similarity can be calculated for each level of the global interests tree, whereas the similarity increases when walking from the bottom level to the top level. Fig. 5(c) shows the similarities of ATP vectors on different levels for the given example.

However, the higher the level and the more tags are included in the same clusters, the more fuzzy is the distinction of tag usage and therefore the similarity measurement. Thus, we introduce the notion of reliability ρ (6) of a tagging profile similarity measurement.

$$\rho(\text{sim}(a_i, a_j)) = \frac{level}{numLevels} \ . \tag{6}$$

For mirrored trust τ_M (7), as defined in Sect. 2, only profile similarities and their reliability are used to predict a level of potential trust.

$$\tau_M(a_i, a_j) = \text{sim}(a_i, a_j) \cdot \rho(\text{sim}(a_i, a_j)) \ . \tag{7}$$

Teleported trust τ_T (8) means an existing directed trust relation $\tau(a_i, a_k)$ from actor a_i to a_k is teleported to a third actor a_j depending on the similarity of a_k and a_j. This teleportation operation $\otimes$ can be realized arithmetically or rule-based.

$$\tau_T(a_i, a_j) = \tau(a_i, a_k) \otimes (\text{sim}(a_k, a_j) \cdot \rho(\text{sim}(a_k, a_j))) \ . \tag{8}$$

5 Evaluation and Discussion

We evaluate and discuss our new tagging profile creation and similarity measurement approach using real-world data sets from the popular `citeulike`[3] community. `Citeulike` is a platform where users can register and tag scientific articles. But before we used this tagging data, we performed two refactoring operations: (i) removing tags reflecting so-called stop words, e.g., `of`, `the`, `in`, `on` etc., resulting from word groups which are sometimes separately saved; (ii) filtering of tags reflecting ambiguous high level concepts such as `system`, `paper`, `article`; (iii) deleting tags not related to the features or properties of resources, including `toread`, `bibtex-import`, `important`. These steps reduce the available 'who-tagged-what' data entries from 5.1 million to 4.4 million.

5.1 Interests Tree Creation

For the later following ATP and RTP creation, all actor or resource tags are mapped to the same global interest tree. Therefore, the tree must be broad enough to contain and cover the most common tags. Due to the huge size of the data set, we picked the 100 articles to which most distinct tags have been assigned, and use all tags which have been applied to at least five of these articles.

In `citeulike` users are free to add arbitrary self-defined tags, raising the problem of differently written tags reflecting the same content. For instance the tag `social-network` appears written as `socialnetwork`, `social_networks`, `social-networks` etc., all meaning the same. To realize their equality, we start by clustering tags with a comparably high similarity (≥ 0.95), and consider these clusters as our initial cluster set. As long as differently written, but equally meant tags are similarly frequently used and distributed among the resources, we can capture their potential equality, otherwise the impact of alternative tags is comparably low and negligible. Then, we compare tag clusters applying much lower similarities (≤ 0.50) to capture tags reflecting similar concepts.

Table 1 summarizes the tagging data properties used to construct the interests tree. This tree consists of six levels, starting with 760 clusters on the lowest one (see Fig. 4 in Sect. 4). The utilized algorithm is detailed in the next section.

Table 1. Data properties for constructing the global interests tree

Metric	Filtered data set	Interests tree
Number of articles	1020622	100
Number of articles recognized by more than 50 users	25	21
Number of distinct tags	287401	760
Number of distinct tags applied by more than 500 users	272	-
Number of distinct users	32449	-
Average number of tags per article	1.2	157
Average number of users per article	3.5	37

---

[3] `http://www.citeulike.org/faq/data.adp`

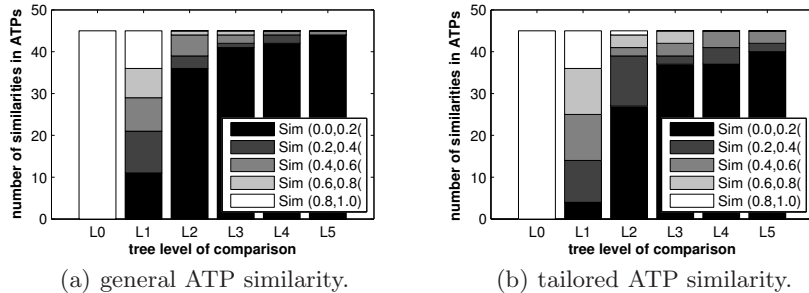(a) general ATP similarity. (b) tailored ATP similarity.

Fig. 6. ATP similarity in `citeulike` on different levels

5.2 Profile Mapping and Trust Prediction

We determine (i) for 10 highly active users the similarities of their general ATPs, and (ii) for 10 users in the area of the *social web* their tailored ATPs. For the first test we select the 10 articles which have been tagged with most distinct tags. Then, for each of these articles, we picked the user who applied most tags to it. Therefore, we get users, who tag highly recognized but different articles. We create the ATPs by retrieving all tags, that each of the selected users applied to his/her 10 most tagged articles (between 50 and 300 tags per ATP). We compare all ATPs with each other (in total 45 comparisons) on each level of the interests tree. The results are depicted in Fig. 6(a). As expected, level 5 comparisons result mostly in no similarity, only two ATPs have a similarity of 0.42 on this level. The amount of similar ATPs in different similarity classes increases when we compare them on higher levels of the interests tree. On level 0, of course, all ATPs are equal, because all tags are merged in the same cluster. These results show, that our approach of indirect similarity measurement provides distinguishable similarity results on different levels of the interests tree.

In a second experiment we measure similarities of tailored ATPs. We restrict the tags used for ATP creation to a subset of resources, and consider only tags assigned to articles in the field of the *social web*. We filter all articles, which are not linked to the `citeulike` groups *Semantic-Social-Networks*[4], *social_navigation*[5], and *Social Web*[6]. The ATP similarity results for the 10 most active users spanning these groups are depicted in Fig. 6(b). Obviously, due to the restricted tag set and a common understanding of tag usage, ATP similarities, especially on level 2 to 4, are significantly higher than in the general comparison before. Furthermore, we compare two sets of users, interested in computer science, but only members of one set participate in *social web* groups. Their general ATPs are largely similar on level 1 to 3, because all users assigned many general tags related to computer science. However, if we compare both groups' ATPs tailored to the *social web*, there is nearly no remarkable similarity before level 1. We conclude, that tailored profiles are a key to more precise trust prediction.

[4] `http://www.citeulike.org/groupfunc/328/home` (82 users, 694 articles)
[5] `http://www.citeulike.org/groupfunc/1252/home` (20 users, 507 articles)
[6] `http://www.citeulike.org/groupfunc/3764/home` (27 users, 444 articles)

6 Implementation

In this section we introduce the architectural components of the trust management framework. Our architecture has been implemented on top of Web service technology suitable for distributed, large-scale environments. Furthermore, we detail the clustering algorithm by showing the steps needed to create hierarchical, tag-based interests trees.

6.1 Reference Architecture

The architecture evolved from our previous efforts in the area of trust management in service-oriented systems (see [4] for details on the VieTE framework).

Our architecture consists of the following main building blocks:

- **Tagging and Social Network Web Services** facilitate the integration of existing systems and the usage of external data sources. Tagging and social networks, for example, interaction graphs, can be imported via Web services.
- **Data Provisioning** comprises a set of **Providers**. We separated these providers in resource-centric (e.g., *Tag*, *Resource*, *Actor*) and trust-centric blocks. Providers enable access to **Tagging Data** and **Social Network Data** using the messaging system JMS[7]. We use the WS-Resource Catalog (WS-RC) specification[8] to manage resources in the system.
- **Trust Prediction** components consist of **Management** support, responsible for the ATP/RTP creation and tailoring of tagging profiles, and **Measurement** support used for various algorithmic tasks such as trust prediction and similarity calculation.
- **Trust Prediction Web Service** enables access to predicted trust in a standardized manner. We currently support SOAP-based services but plan to enhance our system by adding RESTful services.
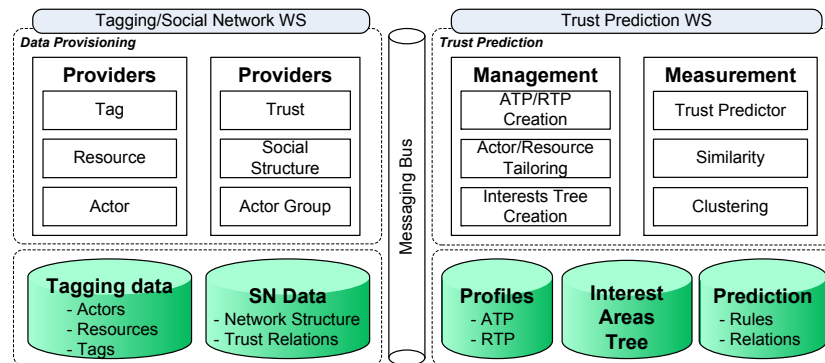

Fig. 7. Reference architecture enabling trust prediction in social network platforms

---

[7] http://java.sun.com/products/jms/
[8] http://schemas.xmlsoap.org/ws/2007/05/resourceCatalog/

6.2 Clustering Algorithm

We detail our clustering approach to interests tree creation as illustrated by Algorithm 1. The clustering starts by putting each tag vector t_x (see (1) in Sect. 4) into a single cluster, and comparing cluster similarities successively. After comparing each cluster with each other, all clusters having cosine similarities above a predefined threshold ϑ and have not been merged yet, are combined to single clusters. Then, ϑ is lowered and the algorithm compares again all available clusters. Finally, all tag vectors are merged in one single cluster, resulting in a tree structure, that reflects the global interests (Fig. 4 in Sect. 4). The function $getSimilarity()$ implements an average similarity measurement by comparing artificial tag vectors that are based on the averages of all vectors within respective clusters.

Algorithm 1 Hierarchical clustering of global interest areas

/* create tag frequency matrix */
$\langle A, R, T \rangle = retrieveTaggingDataFromDB()$
$tfm = \emptyset$
for each $t_x \in T$ **do**
    $t_x = createTagFrequencyVector(t_x, \langle A, R, T \rangle)$
    $addToTagFrequencyMatrix(tfm, t_x)$
end for
/* weight single tag entries */
for each $t_x \in T$ **do**
    for each $r_j \in R$ **do**
        $tf(t_x, r_j) = extractValue(tfm, t_x, r_j)$
        $updateValue(tfm, tf(t_x, r_j) * idf(t_x, r_j))$
    end for
end for
/* perform hierarchical clustering */
$\vartheta[] = \{0.95, 0.5, 0.25, 0.15, 0.05, 0.0\}$
$Cluster[][1] = createClusterForEachTag(tfm)$
for $i = 1 \rightarrow |\vartheta[]|$ **do**
    for $u = 1 \rightarrow |Cluster[][i]|$ **do**
        $C_u = Cluster[u][i]$
        if $\neg isMerged(C_u)$ **then**
            $C_{sim}[] = \{C_u\}$
            for $v = u + 1 \rightarrow |Cluster[][i]|$ **do**
                $C_v = Cluster[v][i]$
                if $\neg isMerged(C_v)$ **and** $getSimilarity(C_u, C_v) \geq \vartheta[i]$ **then**
                    $addToClusterArray(C_{sim}[], C_v)$
                end if
            end for
            $C_m = mergeClusters(C_{sim}[])$
            $addToClusterArray(Cluster[][i + 1], C_m)$
        end if
    end for
end for

---

7 Related Work

Recently, trust in collaborative environments and service-oriented systems has become a very important research area. Several EU-funded projects such as COIN[9] focus on, for example, trusted collaboration in networked enterprises. Some surveys of trust related to computer science have been performed [2, 3, 6], which outline common concepts of trust, clarify the terminology and show the most popular trust models. Trust management systems for service-oriented-environments [12, 13] as well as for mixed systems [14], comprising humans and services, such as the VieTE framework [4], are a focus of current research. VieTE aims at collecting interaction data in collaborations of humans and services, and facilitating the emergence of trust among collaboration participants. For bootstrapping such systems we introduced two concepts of trust prediction. Both concepts model the inference of trust based on interest similarities as studied in [7, 8]. Other approaches to trust prediction do not necessarily address the cold-start problem. They focus more on the forecast of trust evolvement based on earlier determined trust relations [15], or on the prediction of non-existing trust relations applying transitive trust propagation [16].

Tagging and its meaning has been widely studied in [17]. Several approaches have been introduced, dealing with the construction of hierarchical structures of tags [18, 19], generating user profiles based on collaborative tagging [9, 20], and collaborative filtering in general [21].

Determining profile similarities has not been addressed well in previous works. Therefore, we applied the concepts of tailored tagging profiles, and indirect similarity measurement. Our approach uses various mathematical methods from the domain of information retrieval, including term-frequency and inverse document frequency metrics [10], measuring similarities, and hierarchical clustering [22].

8 Conclusion and Future Work

In this paper we introduced concepts for trust prediction, i.e., *trust mirroring* and *trust teleportation* which address the cold-start problem and facilitate bootstrapping trust management systems. As these concepts are based on profile similarities, we described a novel approach to compare interests and capabilities of entities within Web-based environments. The application of this approach has been evaluated with real data sets, gathered from a community which has similar characteristics as our proposed tagging environment. We found out that our approach of indirect tagging profile similarity measurement provides adequate results for trust prediction.

Our future plans are twofold. First, we plan to apply the presented bootstrapping mechanisms in our VieTE [4] trust management system for service-oriented environments, and study their influences on trust determination and improvements from the users' point of view. Second, we will test the extended version of VieTE in real cross-enterprise collaboration scenarios of the COIN project.

[9] http://www.coin-ip.eu

14

References

1. Grandison, T., Sloman, M.: A survey of trust in internet applications. IEEE Communications Surveys and Tutorials **3**(4) (2000)
2. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems **43**(2) (2007) 618–644
3. Ruohomaa, S., Kutvonen, L.: Trust management survey. In: iTrust. Volume 3477 of LNCS., Springer (2005) 77–92
4. Skopik, F., Schall, D., Dustdar, S.: The cycle of trust in mixed service-oriented systems. In: Euromicro SEAA. (2009)
5. Skopik, F., Truong, H.L., Dustdar, S.: Trust and reputation mining in professional virtual communities. In: ICWE. (2009) 76–90
6. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. J. Web Sem. **5**(2) (2007) 58–71
7. Ziegler, C.N., Golbeck, J.: Investigating interactions of trust and interest similarity. Decision Support Systems **43**(2) (2007) 460–475
8. Matsuo, Y., Yamamoto, H.: Community gravity: Measuring bidirectional effects by trust and rating on online social networks. In: WWW. (2009) 751–760
9. Shepitsen, A., Gemmell, J., Mobasher, B., Burke, R.: Personalized recommendation in social tagging systems using hierarchical clustering. In: RecSys, ACM (2008) 259–266
10. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management **24**(5) (1988) 513–523
11. Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R.: Indexing by latent semantic analysis. Journal of the American society for information science **41**(6) (1990) 391–407
12. Conner, W., Iyengar, A., Mikalsen, T., Rouvellou, I., Nahrstedt, K.: A trust management framework for service-oriented environments. In: WWW. (2009)
13. Malik, Z., Bouguettaya, A.: Reputation bootstrapping for trust establishment among web services. IEEE Internet Computing **13**(1) (2009) 40–47
14. Schall, D., Truong, H.L., Dustdar, S.: Unifying human and software services in web-scale collaborations. IEEE Internet Computing **12**(3) (2008) 62–68
15. Chang, E., Dillon, T.S., Hussain, F.K.: Trust and reputation for service-oriented environments: technologies for building business intelligence and consumer confidence, Wiley (2006)
16. Massa, P., Avesani, P.: Trust-aware collaborative filtering for recommender systems. In: CoopIS, DOA, ODBASE. (2004) 492–508
17. Golder, S.A., Huberman, B.A.: The structure of collaborative tagging systems. The Journal of Information Science (2006)
18. Heymann, P., Garcia-Molina, H.: Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical Report 2006-10, Computer Science Department (April 2006)
19. Eda, T., Yoshikawa, M., Yamamuro, M.: Locally expandable allocation of folksonomy tags in a directed acyclic graph. In: WISE. Volume 5175 of LNCS., Springer (2008) 151–162
20. Michlmayr, E., Cayzer, S.: Learning user profiles from tagging data and leveraging them for personal(ized) information access. In: Proceedings of the Workshop on Tagging and Metadata for Social Information Organization, WWW. (2007)
21. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1) (2004) 5–53
22. Romesburg, H.C.: Cluster Analysis for Researchers. Krieger Pub. Co. (2004)