Analyzing Correlation between Trust and User Similarity in Online Communities

Cai-Nicolas Ziegler and Georg Lausen

Institut für Informatik, Universität Freiburg,
Georges-Köhler Allee, Gebäude 51,
79110 Freiburg i.Br., Germany

{cziegler,lausen}@informatik.uni-freiburg.de

Abstract. Past evidence has shown that generic approaches to recommender systems based upon collaborative filtering tend to poorly scale. Moreover, their fitness for scenarios supposing distributed data storage and decentralized control, like the Semantic Web, becomes largely limited for various reasons. We believe that computational trust models bear several favorable properties for social filtering, opening new opportunities by either replacing or supplementing current techniques. However, in order to provide meaningful results for recommender system applications, we expect notions of trust to clearly reflect user similarity. In this work, we therefore provide empirical results obtained from one real, operational community and verify latter hypothesis for the domain of book recommendations.

1 Introduction

Computational trust models [15,21,17] are becoming invaluable goods for today's networked worlds where uncertainty and anonymity prevail. According to Marsh [14], trust can render agents less vulnerable to others and may enhance collaboration significantly.

Recently, approaches incorporating trust models into recommender systems are gaining momentum [20,11,8], synthesizing recommendations based upon opinions from trusted peers. Most notably, *decentralized* recommender systems cannot rely upon generic collaborative filtering methods only, scaling poorly. These systems require novel approaches that allow some prefiltering and neighborhood formation, like, for instance, trust.

Trust therefore becomes supplementary or even surroagate filtering mechanism. However, in order to provide *meaningful* results, one should suppose trust to reflect user similarity to some extent. Clearly, recommendations only make sense when obtained from like-minded people having similar taste.

Hence, Abdul-Rahman and Hailes [2] claim that given some predefined domain and context, e.g., communities of people reading books, its members commence creating ties of friendship and trust primarily with persons resembling their own profile of interest. Reasons for latter phenomenon are manifold and

mostly sociologically motivated, like people's need for some sort of social affilia-tion. For instance, Pescovitz [24] describes endeavors to identify trust networks for crime prevention and security. Hereby, its advocates operate "on the as-sumption that birds of a feather tend to flock together [. . .]". However, though belief in correlation of trust and user similarity has been widely adopted and pre-supposed, thus constituting the foundations for trust-based recommender and rating systems, to our best knowledge, no endeavors have been made until now to provide "real-world" empirical evidence. We claim that latter correlation not only represents some desired, but even an *essential* and vital feature for reason-able application of trust to those systems. Profound empirical analysis therefore becomes indispensable, constituting our major contribution.

Hence, we want to investigate and analyze presence or absence of latter cor-relation, relying upon data mined from an online community focusing on books. Studies involve several hundreds of members telling which books they like and which other community members they trust, hence substantiating our results extensively. Our motivation mainly derives from incorporating trust models into *decentralized* recommender systems, exploiting trust not only for selecting small neighborhoods upon which to perform collaborative filtering, but also for intel-ligent prefiltering of relevant, similar peers.

In section 2, we briefly outline existing approaches dealing with the incor-poration of trust into reputation systems and online recommenders. Section 3 presents experiments we performed in order to investigate correlation between trust and similarity. Hereby, large parts of latter section are devoted to the con-ception and makeup of our novel approach to profile similarity computation, designed in order to render our experiments feasible. Suggestions for exploita-tion of correlation between trust and similarity are offered in section 4, while section 5 mentions open questions and possible future work.

2   Recommender Systems and Trust

Online recommender systems [26] intend to provide people with recommenda-tions of products they might appreciate, taking into account their past ratings profile and history of purchase or interest. Hereby, distinctions between three types of filtering systems are made [7], namely collaborative, content-based and economic. While content-based filtering, also dubbed item-to-item correlation [29], takes into account properties attributed to the nature of products them-selves, collaborative filtering relies upon building "neighborhoods of like-minded customers" [28] whose rating history may then serve to generate new recom-mendations. Economic filtering has seen little practical application until now and exerts marginal impact only.

Recent studies [31] have shown that people tend to prefer receiving recom-mendations from people they *know* and *trust*, i.e., friends and family-members, rather than from online recommender systems. Some researchers have therefore commenced to focus on computational trust models as appropriate means to supplement or replace current collaborative filtering approaches. Kautz et al. [9]

mine social network structures in order to render fruitful information exchange and collaboration feasible. Olsson [23] proposes an architecture combining trust, collaborative filtering and content-based filtering in one single framework, giving only vague information and insight. Another agent-based approach has been presented by Montaner et al. [20], who introduce so-called "opinion-based" filtering. Hereby, Montaner claims that trust should be *derived* from user similarity, implying that friends are exactly those people that resemble our very nature. However, Montaner's model only extends to the agent world and does not reflect evidence acquired from real-world social studies concerning the formation of trust. Similar agent-based systems have been devised by Kinateder [11,10] and Chen [5].

Apart from research in agent systems, online communities have also discovered opportunities through trust network leverage. For instance, Epinions (*http://www.epinions.com*) provides information filtering facilities based upon personalized "webs of trust" [8]. Guha tells that latter filtering approach has been greatly approved and appreciated by Epinion's members. However, justifications and causal analysis underpinning these findings, like indications of correlation between trust and interest similarity, have not been subject to Guha's work. All Consuming (*http://allconsuming.net*) represents another community combining ratings and trust networks. Unlike Epinions, All Consuming only poorly exploits synergies between social filtering and trust.

3   Analyzing Correlation between Trust and Similarity

Recent studies [31] have provided evidence that users tend to rely upon recommendations from friends and family members, i.e., people they trust, more than upon those from online systems. However, Sinha's experiments only included nineteen people, rendering his results fairly applicable. Furthermore, those studies did not investigate the *reasons* which made people stick to their friends' opinions rather than automated collaborative filtering. We believe that given an application domain, such as, for instance, the book-reading domain, people's trusted peers are considerably more similar to their sources of trust than arbitrary peers. More formally, let A denote the set of all community members and $\mathrm{trust}(x)$ the set of all users trusted by x:

$$\forall x \in A : \frac{\sum_{y \in \mathrm{trust}(x)} \mathrm{sim}(x,y)}{|\,\mathrm{trust}(x)|} \gg \frac{\sum_{z \in A \setminus \mathrm{trust}(x)} \mathrm{sim}(x,z)}{|A \setminus \mathrm{trust}(x)|} \tag{1}$$

For instance, given that agent x is interested in Sci-Fi and AI, chances that y, trusted by x, also likes these two topics are much higher than for peer z not explicitly trusted by x. Various social processes are involved, such as participation in those social groups that best reflect our own interests and desires. Some recommendation and reputation systems based upon trust have already been proposed [8,23], exploiting latter expected correlation between trust and interest similarity, but none have provided clear evidence that trust *does* correlate to profile similarity.

3.1   Model and Data Acquisition

Our study intends to close latter gap by analyzing rife user information collected from the All Consuming book-reading community. Hereby, we have opted for All Consuming for mainly two reasons. First, all information published on its site may be accessed without violation of copyright and without any other legal limitations. Second, All Consuming provides both, personal webs of trust that link users to peers they trust, as well as data about the books people have completed and are currently reading.

Information Model. Before delving into the details and makeup of application data our tools have been mining and collecting, we depict our underlying information infrastructure.

(a) **Set of agents** $A = \{a_1, a_2, \ldots, a_n\}$. Set A contains all agents part of the book-reading community.

(b) **Set of books** $B = \{b_1, b_2, \ldots, b_m\}$. All published books are comprised in set B, i.e., all those books that possess an International Standard Book Number. Latter ISBN consequently serves as the globally unique identifier for all $b_i \in B$.

(c) **Set of partial trust functions** $T = \{t_1, t_2, \ldots, t_n\}$. Every agent $a_i \in A$ has one partial trust function $t_i : A \rightarrow [0,1]^\perp$ that assigns continuous trust values to its peers. Functions $t_i \in A$ are partial since agents generally only rate small subsets of the overall community, hence rendering t_i sparse:

$$t_i(a_j) = \begin{cases} p, & \text{if } \mathrm{trust}(a_i, a_j) = p \\ \perp, & \text{if no trust rating for } a_j \text{ from } a_i \end{cases} \qquad (2)$$

We define high values for $t_i(a_j)$ to denote high trust from a_i in a_j, and low values near zero to express low trust, respectively.

(d) **Set of partial book rating functions** $R = \{r_1, r_2, \ldots, r_n\}$. In addition to functions $t_i \in T$, every $a_i \in A$ has one partial function $r_i : B \rightarrow [-1, +1]^\perp$ that expresses his liking or dislike of books $b_j \in B$. No person can read and rate every book published, so functions $r_i \in B$ are necessarily partial.

$$r_i(b_j) = \begin{cases} p, & \text{if } \mathrm{rates}(a_i, b_j) = p \\ \perp, & \text{if no book rating for } b_j \text{ from } a_i \end{cases} \qquad (3)$$

Intuitively, high positive values for $r_i(b_j)$ denote that a_i highly appreciates b_j, while low negative values near -1 express utter dislike, respectively.

(e) **Taxonomy C over set $D = \{d_1, d_2, \ldots, d_l\}$ of book categories.** Book category descriptors $d_k \in D$ represent topics and categories that books $b_j \in B$ may fall into. Hereby, topics can express broad or narrow categories.

Taxonomy C arranges all $d_k \in D$ in an acyclic graph by imposing partial subset order $\subseteq$ on D, similar to class hierarchies known from object-oriented languages. Hereby, *inner* topics $d_k \in D$ with respect to C are all topics having subtopics, i.e., an outdegree greater zero. On the other hand, *leaf* topics are topics with zero outdegree, i.e., most specific categories. Furthermore, taxonomy C has exactly one top element $\top$, which represents the most general topic and has zero indegree.

(f) **Book descriptor assignment function** $f : B \rightarrow 2^D$**.** Function f assigns a set $D_i \subset D$ of book topics to every book $b_i \in B$. Note that books may possess *several* descriptors, for classification into one single category generally entails loss of precision. Furthermore, all $d \in D_i$ are expected to represent *leaf* nodes with respect to taxonomy C.

The following section will now relate our formal environment model to an actual scenario, hereby making use of variable and function bindings introduced above.

Data Acquisition. All Consuming represents one of the few communities that allow members to express which other agents they trust as well as which items, in our case books, they appreciate. Hereby, users may import their list of trusted persons from other applications like FOAF [6]. Likewise, All Consuming also offers to automatically compile information about books its members have read from their personal weblog. Members may furthermore explicitly assert trust statements and indicate books they own, have read, like most, and so forth.

Trust assertions from user a_i to a_j in All Consuming are *boolean*, either denoting *full* trust, i.e., a_i explicitly states trust in a_j, or *no* trust, if a_i does not. Hence, our real-world scenario is less precise than our model, where we have defined $t_i : A \rightarrow [0,1]^\perp$ instead of $t_i : A \rightarrow \{0,1\}$. Moreover, book mentions in All Consuming seldom reflect "real" ratings, like dislike or liking. They rather indicate that agent a_i has *read* or *purchased* book b_j. These statements therefore count among *implicit* ratings, which nevertheless provide valuable information. Clearly, people tend to only buy and read books they expect to appreciate. In fact, numerous recommender systems are purely based upon implicit ratings [22] since user incentive to provide explicit ratings generally tends to be low [3]. Compared to our model presented in section 3.1, book rating information r_i for user a_i obtained from All Consuming is therefore more imprecise, mapping books b_j to values 0 or 1 instead of $[-1, +1]$. Hereby, we define $r_i(b_j) = 1$ to denote that a_i actually *has* mentioned b_j, and $r_i(b_j) = 0$ that a_i has not.

Our tools have mined data from about $2,074$ weblogs contributing to the All Consuming information base, and 527 users issuing 4.93 trust statements on average. These users have mentioned $6,592$ different books altogether. In order to obtain category descriptors $f(b_i)$ for all discovered books b_i, we have written several web extraction tools which have mined latter classification information from the Amazon online bookshop (*http://www.amazon.com*). For each book,

Amazon provides an average of about 4 classification topics. These topics represent leaf nodes relating to the huge Amazon book taxonomy, comprising $13,394$ categories after duplicate removal and data cleansing. We have extracted the taxonomy from the Amazon Associates pages via screen scraping tools written particularly for this purpose. Note that the Amazon book taxonomy induces a *tree* structure on our set of categories D, hence making each node d_k have at most one parent d_m. We adopt this model for our approach to user similarity computation and hence suppose taxonomy C to define a tree.

3.2   User Similarity Computation

In order to analyze correlation between trust and user similarity, we need mathematical models indicating how to *compute* latter similarity. Hereby, the book domain bears some notable differences to most other domains like videos, computer games, and DVDs. First, every published book is uniquely identified by its ISBN, which makes it easy to ensure interoperability and gather supplementary information from various other sources, e.g., mentioned category descriptors from Amazon for any given ISBN. Second, the set of published books is vast and much larger than for videos or DVDs. Consequently, profile overlap, i.e., the amount of books two given users $a_i, a_j \in A$ have both rated, is generally small. Common techniques used in collaborative filtering, such as computing Pearson's correlation coefficient [30,25], are therefore bound to fail within our context. Even more advanced techniques, like Sarwar's singular value decomposition [28], cannot reduce dimensionality satisfactorily for our book domain.

Profile Generation. We propose another, more reasonable approach which does not represent users by their respective *book*-rating vectors of dimensionality $|B|$, but by vectors of interest scores assigned to *topics* taken from the book categories taxonomy C. Our method is inspired by Middleton's work on the application of ontologies for content-based filtering [19,18] but goes much further.

Since $|C|$ is equal to the number of categories, user profile vectors shrink to size $|D|$, which tends to be significantly lower than $|B|$. Moreover, making use of profile vectors representing interest in *topics* rather than book *instances*, we can exploit the hierarchical structure of taxonomy C in order to generate overlap and make similarity computation more meaningful: for every leaf category $d_{j_k} \in f(b_j)$ of books b_j agent a_i has mentioned and thus implicitly rated, we also infer an interest score for all super-topics of topic d_{j_k} in user a_i's profile vector. However, interest score assigned for super-topics decreases with increasing distance from leaf d_{j_k}. We furthermore normalize profile vectors with respect to the amount of score assigned, according overall fix score s. Hence, suppose that $\boldsymbol{v_i} = (v_{i_1}, v_{i_2}, \dots, v_{i_{|D|}})^T$ represents the profile vector for user a_i, where v_{i_k} gives the score for topic $d_k \in D$. Then we require the following equation to hold:

$$\forall a_i \in A : \sum_{k=1}^{|D|} v_{i_k} = s \qquad (4)$$

We now formally define the profile generation algorithm for user a_i as follows: suppose that $B_i = \{b_j \in B \mid r_i(b_j) = 1\}$ constitutes the set of all books user a_i has mentioned and thus implicitly rated. Due to normalization, the score for each book $b_j \in B_i$ amounts to $s \, / \, |B_i|$, which is proportional to the number of distinct books a_i has mentioned. Consequently, for each topic descriptor $d_{j_k} \in f(b_j)$ categorizing book b_j, we obtain topic score $\mathrm{sc}(d_{j_k}) = s \, / \, (|B_i| \cdot |f(b_j)|)$. Topic score for b_j is hence distributed evenly among its topic descriptors.

Now suppose that $(p_0, p_1, \ldots, p_q)$ gives the path from top element $p_0 = \top$ to leaf node $p_q = d_{j_k}$ within our tree-structured taxonomy C for any given $d_{j_k} \in f(b_j)$. Hence, topic descriptor d_{j_k} has q super-topics. Score normalization and inference of fractional interest for super-topics imply that descriptor topic score $\mathrm{sc}(d_{j_k})$ may *not* become *fully* assigned to d_{j_k}, but in part to all its ancestors $p_{q-1}, \ldots p_0$, likewise. We therefore introduce another score function $\mathrm{sco}(p_m)$ that represents the eventual assignment of score to topics p_m along the taxonomy path leading from $p_q = d_{j_k}$ to p_0:

$$\sum_{m=0}^{q} \mathrm{sco}(p_m) = \mathrm{sc}(d_{j_k}) \tag{5}$$

Furthermore, we require that the interest score $\mathrm{sco}(p_m)$ accorded to p_m, which is super-topic to p_{m+1}, depends on the number of siblings $\mathrm{sib}(p_{m+1})$ of p_{m+1}. The less siblings p_{m+1} possesses, the more interest score is accorded to its super-topic p_m:

$$\forall m \in \{0, 1, \ldots, q-1\} : \mathrm{sco}(p_m) = \frac{\mathrm{sco}(p_{m+1})}{\mathrm{sib}(p_{m+1}) + 1} \tag{6}$$

We hereby assume that sub-topics have *equal shares* in their super-topic within taxonomy C. Clearly, this assumption may imply certain issues, e.g., when certain sub-taxonomies are much denser than others [27]. However, reasonable solutions to mitigate latter effect would require explicit annotation of the taxonomy telling semantic distances from sub-topics to super-topics, which is not the case for the Amazon book taxonomy and most other taxonomies.

Equations 5 and 6 describe conditions which have to hold for the computation of leaf node p_q's profile score $\mathrm{sco}(p_q)$ and the computation of scores for its taxonomy ancestors p_k, where $k \in \{0, 1, \ldots, q-1\}$. We hence derive the following recursive definition for $\mathrm{sco}(p_q)$:

$$\mathrm{sco}(p_q) := \frac{\mathrm{sc}(d_{j_k})}{g_q}, \tag{7}$$

where

$$g_0 := 1, \ g_1 := 1 + \frac{1}{\mathrm{sib}(p_q) + 1},$$

and $\forall n \in \{2, \ldots, q\}$

$$g_n := g_{n-1} + (g_{n-1} - g_{n-2}) \cdot \frac{1}{\mathrm{sib}(p_{q-n+1}) + 1}$$

Having computed $\mathrm{sco}(p_q)$, we may now apply Equation 5 to compute all other scores $\mathrm{sco}(p_k)$. These scores are then used to update profile vector $\boldsymbol{v_i}$ of user a_i, adding scores for the respective topics in $\boldsymbol{v_i}$. The procedure is repeated for every book mention $b_j \in B_i$ and every descriptor $d_{j_k} \in f(b_j)$.

Example 1 (Topic score assignment). Suppose the taxonomy given in Figure 1 which represents a tiny fragment from the original Amazon book taxonomy. Let user a_i have mentioned 4 books, namely *Matrix Analysis*, *Fermat's Enigma*, *Snow Crash*, and *Neuromancer*. For *Matrix Analysis*, 5 topic descriptors are given, one of them pointing to leaf topic *Algebra* within our small taxonomy.

Suppose that $s = 1000$ defines the overall accorded profile score. Then the score accorded to descriptor *Algebra* amounts to $s / (4 \cdot 5) = 50$. Ancestors of leaf *Algebra* are *Pure*, *Mathematics*, *Science*, and top element *Books*. Score 50 hence must be distributed among these topics according to Equation 5 and 6. Application of Equation 7 gives score 29.087 for topic *Algebra*. Likewise, applying Equation 6, we get 14.543 for topic *Pure*, 4.848 for *Mathematics*, 1.212 for *Science*, and 0.303 for top element *Books*. These values are then used to update the profile vector $\boldsymbol{v_i}$ of user a_i. Note that after elimination of numerical errors inferred by rounding, summation of latter scores yields exactly score 50.

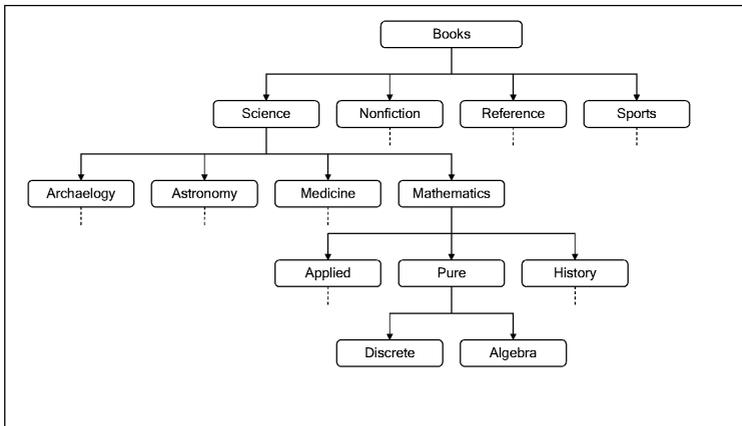**Fig. 1.** Small fragment from the Amazon book taxonomy

Profile Similarity Computation. The presented approach computes flat profile vectors $\boldsymbol{v_i} \in [0, s]^{|D|}$ for agents a_i, assigning score values between 0 and maximum score s to every topic d from the set of book categories D. However, one still needs to match these profile vectors against each other in order to come up with one single similarity metric value. Sarwar et al. [28] count nearest-neighbor techniques like Pearson's correlation coefficient [30,25] and cosine similarity, widely known from information retrieval, among the most popular approaches used

for measuring profile proximity. We opt for Pearson correlation instead of cosine similarity since Pearson's correlation coefficient also allows for detecting *negative* correlation. For two given profile vectors $\boldsymbol{v_i}, \boldsymbol{v_j} \in [0, s[^{|D|}$, Pearson correlation is defined as below:

$$\mathrm{PCorr}(a_i, a_j) = \frac{\sum_{k=0}^{|D|} (v_{i_k} - \overline{\boldsymbol{v_i}}) \cdot (v_{j_k} - \overline{\boldsymbol{v_j}})}{\sqrt{\sum_{k=0}^{|D|} (v_{i_k} - \overline{\boldsymbol{v_i}})^2 \cdot \sum_{k=0}^{|D|} (v_{j_k} - \overline{\boldsymbol{v_j}})^2}} \tag{8}$$

Hereby, $\overline{\boldsymbol{v_i}}$ and $\overline{\boldsymbol{v_j}}$ give mean values for vectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$. In our case, because of profile score normalization, both values are identical, i.e., $\overline{\boldsymbol{v_i}} = \overline{\boldsymbol{v_j}} = s\,/\,|D|$. Values for $\mathrm{PCorr}(a_i, a_j)$ range from -1 to $+1$, where negative values indicate negative correlation, and positive values positive correlation, respectively. Clearly, people who have read many books in common also have high similarity. For generic approaches to collaborative filtering, the opposite direction also holds, i.e., people who have *not* read many books in common have *low* similarity. Our approach, on the other hand, may compute high similarity values even for pairs of agents that have little or even no books in common. Clearly, quality hereby highly depends on the taxonomy's design and level of nesting. According to our scheme, the more score two profiles $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$ have accumulated in same branches, the higher their computed similarity:

Example 2. (Positive correlation) Suppose a_i has read only one single book b_m, bearing exactly one topic descriptor that classifies b_m into *Algebra*. Agent a_j has read another book b_n assigned to one of the leaf nodes[1] of *History*. Then $\mathrm{PCorr}(a_i, a_j)$ will still be reasonably high, for both profiles have significant overlap in categories *Mathematics* and *Science*.

On the other hand, negative correlation occurs when users have completely diverging interests. For instance, in our information base mined from All Consuming, we had one user reading books mainly from the genres of Sci-Fi, Fantasy, and AI. Latter person was rather negatively correlated to another one reading books about American history, politics, and conspiracy theories.

3.3   Experiment Setup and Analysis

We now proceed to describe the two experiments we have performed in order to analyze possible correlation between user similarity and trust. In both cases, experiments are run on data obtained from All Consuming, as has been described in section 3.1. Considering the slightly different composition of information the two experiments were based upon, we expected the first to define some upper bound for correlation analysis, and the second one some lower bound. Results obtained confirmed our assumption.

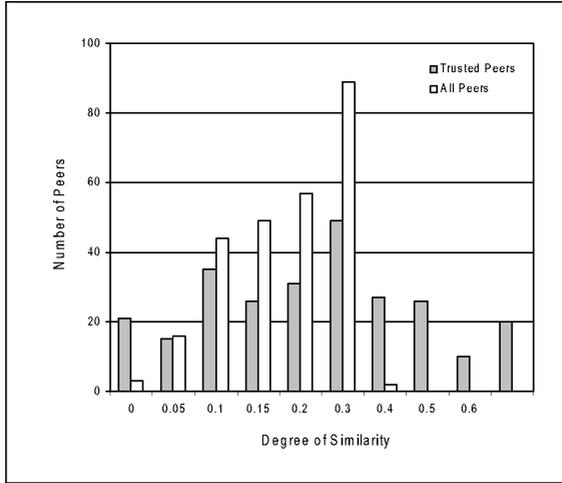[1] Leaf nodes of *History* are not shown in Figure 1.

Fig. 2. Results obtained from our upper-bound analysis

Upper-bound Analysis. Before running our first experiment, we applied data cleansing and duplicate removal to the All Consuming's active user base of 527 members. First, we pruned all users a_i having less than three books mentioned, removing them from user base A and defining $t_j(a_i) = 0$ for all other users $a_j \in A$. Next, we deleted all users a_i from our test base which did not issue any trust assertions. Interestingly, some users created *several* accounts. We discovered latter "duplicates" through searching account names for similarity patterns and via tracking identical or highly similar profiles in terms of books mentions. Eventually, we removed self-references, i.e., users trusting themselves.

Through data cleansing, 266 users were removed from our initial test set, leaving 261 users for our experiment to run upon. We denote the reduced set of users by A' and corresponding trust functions by $t'_i(a_j)$. For our first experiment, we proceeded as follows: for every single user $a_i \in A'$, we generated its profile vector and computed similarity with each profile of all *trusted* peers $a_j \in \{a \in A' \mid t'_i(a) = 1\}$. Then we took the average of these proximity measures and recorded latter value in some table. Next, we computed similarity of a_i's profile with the profiles of *all* agents, except a_i itself, from dataset A'. Again, we took the average of these proximity measures and stored the resulting value.

In 173 cases, users were more similar to their trusted peers than to the entirety of A'. The opposite held for only 88 users. On average, users had similarity score 0.247 with their trusted peers, while only 0.163 with *all* users of A'. In other words, users were more than 50% more similar to trusted agents than arbitrary peers. Histogram representations showing the distribution of similarity values for both cases of our first experiment are given in Figure 2.
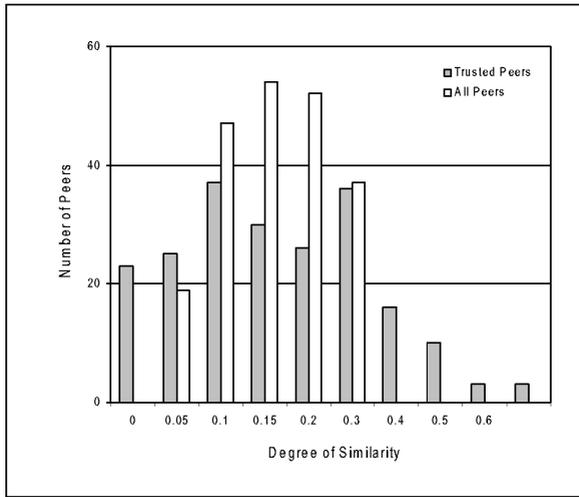
Fig. 3. Histogram representation of our lower-bound analysis

Lower-bound Analysis. The first experiment conducted underpins that peers tend to trust agents being significantly more similar than usual. However, we have to consider that All Consuming bears one feature that *proposes* friends to new users a_i. Hereby, All Consuming chooses users which have at least one book in common with a_i. Hence, we had to suppose that our first experiment was biased and too optimistic with respect to correlation between trust and similarity. Consequently, we pruned user set A' even further, eliminating trust statements whenever trusting user and trusted user had at least one book in common. We call latter user base A'', now reduced to 210 trusting users, and indicate its respective trust functions by $t_i''(a_i)$. Clearly, our approach to eliminate All Consuming's intrusion into the natural process of trust formation entailed the removal of many "real" trust relationships between agents a_i and a_j. These relationships had been established because of a_i actually knowing and trusting a_j, and not because All Consuming proposed a_j as an appropriate match to a_i. Proceeding for experiment two in exactly the same fashion as for its predecessor, we then expected results to be biased towards the other direction, i.e., unduly lowering correlation between trust and user similarity. Bear in mind that in A'', users have not one single book in common with their trusted peers.

Results obtained from the second experiment confirmed our expectations, being less indicative for an existing correlation between trust and user similarity. Nevertheless, similarity of users with trusted peers still significantly exceeded average similarity. In 112 cases, users were more similar to their trusted fellows than arbitrary peers. The opposite held for 98 users. Similarity between trusting users and trusted agents amounted to 0.164, while average similarity between any two arbitrary users only made 0.134. Hence, even for our lower-bound ex-

periment, users were still about 23% more similar to their trusted fellows than arbitrary agents.

We may conclude our experimental analysis noticing that without exact knowledge of how much noise All Consuming's friend recommender adds to our obtained results, we expect "true" correlation between trust and similarity to reside somewhere within our computed upper and lower bound. At any rate, sufficient evidence has been provided exposing that similarity may substantially increase when considering trusted peers opposed to arbitrary ones.

4   Exploiting Correlation between Trust and Similarity

Knowledge about positive correlation between trust and interest similarity may be exploited for diverse applications. In particular, we envision trust to play an important role for *decentralized* recommender systems. These filtering systems suppose distributed data and control and currently face various problems inherent to their very nature:

(a) **Credibility and attack-resistance.** The Semantic Web and other open systems lack dedicated mechanisms and facilities to verify user identity. Hence, these systems tend to encourage insincerity and fraudulent behavior. Moreover, penalization and banishment are hard to accomplish and facile to short-circuit. Collaborative filtering becomes particularly suceptive to attack, for malicious users simply have to create profiles replicating the victim's in order to obtain high similarity. Then they can lure the victim into buying items the purchase of which may provide some utility for the attacker.

(b) **Product-user matrix sparseness.** Communities often limit the number of ratable products, therefore avoiding product-user matrices from becoming overly sparse. Besides, Ringo [30] and other systems require users to rate items from *small product subsets* to generate user profiles with sufficient overlap. However, decentralized recommender system cannot suppose reduced item sets. Bear in mind that controlling product set contents and having users rate certain goods presupposes some central authority.

(c) **Computational complexity and scalability.** Centralized systems are able to control and limit the number of members. Depending on the community's size, large-scale server clusters ensure proper operativeness and scalability. In general, recommender systems imply heavy computations. For instance, collaborative filtering systems compute Pearson correlation for users a_i offline rather than on-the-fly. Recall that coefficients $\text{PCorr}(a_i, a_j)$ have to be computed for every other agent $a_j \in A$. Clearly, this approach does not work for large decentralized systems. Sensible prefiltering mechanisms which still ensure reasonable recall are needed.

Clearly, trust succeeds to address the credibility problem. Every agent builds its own neighborhood of trusted peers, relying upon direct trust statements

and those from trusted peers, likewise. For deriving trust, numerous metrics have been proposed during the last decade, among those [16], [1], [4], and [13]. However, we believe that local group trust metrics like Levien's Advogato [12] and Appleseed [33] best fit neighborhood formation in decentralized systems [32]. Unfortunately, trust cannot handle product-user matrix sparseness, nor substantially reduce dimensionality. Supplementary approaches are needed, e.g., taxonomy-based filtering techniques similar to the one proposed.

Increased computational complexity and loss of scalability are mitigated and may even be eliminated when supposing positive correlation between trust and user similarity. Note that our complexity issue itself does not require latter correlation: limiting collaborative filtering to selected peers part of agent a_i's trust neighborhood only entails complexity reduction, too. However, when supposing that trust does *not* reflect similarity, serious tradeoffs are implied, for scalability comes at the expense of recall. Mind that trust neighborhood A_{a_i} of agent a_i only represents one tiny fraction of the overall system A. Moreover, latter fraction not necessarily contains similar peers. Instead, trusted agents are on average no more similar than arbitrary ones. Recall, i.e., the proportion of agents a_j with $\text{sim}(a_i, a_j) \geq t$ found by the filtering process, degrades proportionally to $|A|/|A_t|$. On the other hand, when assuming that trust *does* correlate with similarity, respective degradation does not take place equally fast, thus ensuring reasonable recall.

Guha's approach [8] relies upon trust networks as only filtering mechanism, clearly exploiting latter correlation. Positive user feedback seems to justify his design decision. Nevertheless, we believe that trust should rather *supplement* than replace existing filtering techniques. For instance, ex-post application of collaborative filtering to computed trust neighborhoods A_{a_i} might boost precision significantly.

5   Discussion and Outlook

We have articulated our hypothesis that correlation between trust and user similarity exists when the community's trust network is tightly bound to some particular application. Empirical evidence has been provided based upon data obtained from the All Consuming book-readers' community. To our best knowledge, suchlike experiments have not been performed before, since communities incorporating explicit trust models are still very sparse.

We believe that our results will have substantial impact for ongoing research in recommender systems, where discovering user similarity plays an important role. Decentralized approaches will especially benefit from trust network leverage. Hereby, the outstanding feature of trust networks refers to sensible prefiltering of like-minded peers and credibility of recommendations. Arbitrary social networks, on the other hand, only allow for computation complexity reduction.

Though backing our experiments with information involving several hundreds of people, studies for distinct interest domains are required. We would also like to run our analysis on communities larger than All Consuming.

References

1. ABDUL-RAHMAN, A., AND HAILES, S. A distributed trust model. In *New Security Paradigms Workshop* (Cumbria, UK, September 1997), pp. 48–60.
2. ABDUL-RAHMAN, A., AND HAILES, S. Supporting trust in virtual communities. In *Proceedings of the 33rd Hawaii International Conference on System Sciences* (Maui, HW, USA, January 2000).
3. AVERY, C., AND ZECKHAUSER, R. Recommender systems for evaluating computer messages. *Communications of the ACM 40*, 3 (March 1997), 88–89.
4. BETH, T., BORCHERDING, M., AND KLEIN, B. Valuation of trust in open networks. In *Proceedings of the 1994 European Symposium on Research in Computer Security* (1994), pp. 3–18.
5. CHEN, M., AND SINGH, J. P. Computing and using reputations for internet ratings. In *Proceedings of the 3rd ACM Conference on Electronic Commerce* (Tampa, FL, USA, 2001), ACM Press, pp. 154–162.
6. GOLBECK, J., PARSIA, B., AND HENDLER, J. Trust networks on the semantic web. In *Proceedings of Cooperative Intelligent Agents* (Helsinki, Finland, August 2003).
7. GOLDBERG, D., NICHOLS, D., OKI, B., AND TERRY, D. Using collaborative filtering to weave an information tapestry. *Communications of the ACM 35*, 12 (1992), 61–70.
8. GUHA, R. Open rating systems. Tech. rep., Stanford Knowledge Systems Laboratory, Stanford, CA, USA, 2003.
9. KAUTZ, H., SELMAN, B., AND SHAH, M. Referral web: Combining social networks and collaborative filtering. *Communications of the ACM 40*, 3 (March 1997), 63–65.
10. KINATEDER, M., AND PEARSON, S. A privacy-enhanced peer-to-peer reputation system. In *Proceedings of the 4th International Conference on Electronic Commerce and Web Technologies* (Prague, Czech Republic, September 2003), vol. 2378 of *LNCS*, Springer-Verlag.
11. KINATEDER, M., AND ROTHERMEL, K. Architecture and algorithms for a distributed reputation system. In *Proceedings of the First International Conference on Trust Management* (April 2003), P. Nixon and S. Terzis, Eds., vol. 2692 of *LNCS*, Springer-Verlag, pp. 1–16.
12. LEVIEN, R. *Attack Resistant Trust Metrics*. PhD thesis, UC Berkeley, Berkeley, CA, USA, 2003.
13. LEVIEN, R., AND AIKEN, A. Attack-resistant trust metrics for public key certification. In *Proceedings of the 7th USENIX Security Symposium* (San Antonio, Texas, USA, January 1998).
14. MARSH, S. *Formalising Trust as a Computational Concept*. PhD thesis, Department of Mathematics and Computer Science, University of Stirling, Stirling, UK, 1994.
15. MARSH, S. Optimism and pessimism in trust. In *Proceedings of the Ibero-American Conference on Artificial Intelligence* (Caracas, Venezuela, 1994), J. Ramirez, Ed., McGraw-Hill Publishing.
16. MAURER, U. Modelling a public key infrastructure. In *Proceedings of the 1996 European Symposium on Research in Computer Security* (1996), E. Bertino, Ed., vol. 1146 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 325–350.
17. MCKNIGHT, H., AND CHERVANY, N. The meaning of trust. Tech. Rep. MISRC 96-04, Management Informations Systems Research Center, University of Minnesota, MN, USA, 1996.

18. MIDDLETON, S., ALANI, H., SHADBOLT, N., AND DE ROURE, D. Exploiting synergy between ontologies and recommender systems. In *Proceedings of the WWW2002 International Workshop on the Semantic Web* (Maui, HW, USA, May 2002), vol. 55 of *CEUR Workshop Proceedings*.

19. MIDDLETON, S., DE ROURE, D., AND SHADBOLT, N. Capturing knowledge of user preferences: Ontologies in recommender systems. In *Proceedings of the First International Conference on Knowledge Capture* (Victoria, British Columbia, Canada, October 2001).

20. MONTANER, M., LÓPEZ, B., AND DE LA ROSA, J. Opinion-based filtering through trust. In *Proceedings of the Sixth International Workshop on Cooperative Information Agents* (Madrid, Spain, September 2002), S. Ossowski and O. Shehory, Eds., vol. 2446 of *LNAI*, Springer-Verlag, pp. 164–178.

21. MUI, L., MOHTASHEMI, M., AND HALBERSTADT, A. A computational model of trust and reputation. In *Proceedings of the 35th Hawaii International Conference on System Sciences* (Big Island, HI, USA, January 2002), pp. 188–196.

22. NICHOLS, D. Implicit rating and filtering. In *Proceedings of the fifth DELOS Workshop on Filtering and Collaborative Filtering* (Budapest, Hungary, 1998), ERCIM, pp. 31–36.

23. OLSSON, T. Decentralized social filtering based on trust. In *Working Notes of the AAAI-98 Recommender Systems Workshop* (Madison, WI, USA, 1998).

24. PESCOVITZ, D. The best new technologies of 2003. *Business 2.0*, 11 (November 2003). Time Inc. Publishing.

25. RESNICK, P., IACOVOU, N., SUCHAK, M., BERGSTORM, P., AND RIEDL, J. GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work* (Chapel Hill, NC, USA, 1994), ACM, pp. 175–186.

26. RESNICK, P., AND VARIAN, H. Recommender systems. *Communications of the ACM 40*, 3 (1997), 56–58.

27. RESNIK, P. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence* (Montreal, Canada, 1995), pp. 448–453.

28. SARWAR, B., KARYPIS, G., KONSTAN, J., AND RIEDL, J. Application of dimensionality reduction in recommender systems - a case study. In *ACM WebKDD Workshop* (Boston, MA, USA, August 2000).

29. SCHAFER, B., KONSTAN, J., AND RIEDL, J. Recommender systems in e-commerce. In *Proceedings of the 1st ACM Conference on Electronic Commerce* (Denver, CO, USA, 1999), ACM Press, pp. 158–166.

30. SHARDANAND, U., AND MAES, P. Social information filtering: Algorithms for automating "word of mouth". In *Proceedings of the ACM CHI'95 Conference on Human Factors in Computing Systems* (1995), vol. 1, pp. 210–217.

31. SINHA, R., AND SWEARINGEN, K. Comparing recommendations made by online systems and friends. In *Proceedings of the DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries* (Dublin, Ireland, June 2001).

32. ZIEGLER, C.-N. Semantic web recommender systems. In *Proceedings of the Joint ICDE/EDBT Ph.D. Workshop 2004* (Heraklion, Greece, March 2004).

33. ZIEGLER, C.-N., AND LAUSEN, G. Spreading activation models for trust propagation. In *Proceedings of the IEEE International Conference on e-Technology, e-Commerce, and e-Service* (Taipei, Taiwan, March 2004), IEEE Computer Society Press.