Bilattice-Based Aggregation Operators for Gradual Trust and Distrust

P. Victor and C. Cornelis

Dept of Appl Math and CS, Ghent University, Gent, Belgium
E-mail:{Patricia.Victor,Chris.Cornelis}@UGent.be

M. De Cock*[†] and E. Herrera-Viedma[‡]

[†]*Institute of Technology, University of Washington, Tacoma, USA*
[‡]*Dept of CS and AI, University of Granada, Granada, Spain*
E-mail: mdecock@u.washington.edu, viedma@decsai.ugr.es

Trust and distrust are two increasingly important metrics in social networks. Since many of these networks are very large, it is only natural that not all users know each other. To this aim, propagation and aggregation operators are often used to estimate (dis)trust relations for users that are not directly connected through the network. In this paper, we introduce bilattice-based aggregation approaches and show that they can be used to accurately predict trust and distrust predictions for the social networking site CouchSurfing.org.

Keywords: Trust network; Distrust; Aggregation operator; Bilattice.

1. Introduction

Social networks in which the users explicitly express their opinion as trust and distrust statements are called *trust networks*. An interesting example is CouchSurfing[a], a large worldwide hospitality exchange network. Its members can use the social network to find or offer free accommodation (= a 'couch'). After a couch experience, users can evaluate their guest or host. One of the questions assesses the trust relationship between the parties involved; these statements give rise to a large trust network among the CouchSurfers. Forming your own opinion on the users might have been easy when the network was still rather small, but nowadays the site contains over one million users, making it increasingly difficult to find the hosts/guests

*On leave from Ghent University

[a] See `www.couchsurfing.org`

that you would get along with well, let alone the ones that are trustworthy.

In such large networks, it is very unlikely that all users know each other directly. This means that, if a user a wants to form a trust opinion about an unknown user x, a has to inquire about x with one of its own trust relations, who in turn might consult a trust connection, etc., until a user connected to x is reached. The process of predicting the trust score along the thus constructed path from a to x is called *trust propagation*. Since it often happens that a has not one, but several trust connections that it can consult for an opinion, we also require a mechanism for combining trust scores originating from different sources. This process is called *trust aggregation*.

Since trust and distrust are very often a gradual phenomenon, fuzzy relations are the pre-eminent tools for modeling trust networks. For our purposes, a trust network is a triplet (A, E, R) where (A, E) is a directed graph with the users as nodes, and directed trust links as edges. R is an $E \to [0, 1]^2$ mapping that associates to each couple (a, b) of connected agents in E a trust score $R(a, b) = (t, d)$ in $[0, 1]^2$, in which t is called the trust degree of a in b and d is called the distrust degree. We model the set of trust scores as a bilattice[1] structure (see Ref. 2 for more details); hence, in other words, the available trust information is modeled as a bilattice-fuzzy relation in the set of users that associates a trust score with each ordered pair of users.

We define the knowledge defect of a trust score (t, d) as $kd(t, d) = |1-t-d|$. We call trust scores (t, d) with $t+d \leq 1$ consistent, and all others inconsistent. We assume that every user is consistent, i.e., issues consistent trust scores. However, modeling inconsistent information is still needed if we want to accurately represent the result of a trust score aggregation process.

In Ref. 2, we proposed four operators for propagation in the trust score space, each reflecting a different strategy of dealing with the available trust information. One example is $P_1((t_1, d_1), (t_2, d_2)) = (T(t_1, t_2), T(t_1, d_2))$, with T a t-norm. P_1 reflects the basic strategy of taking over information only from trusted sources, and is at the basis of all other propagation operators. Note that consistent inputs yield a consistent propagated score; like this, we can ensure that all inputs (either direct trust scores or the result of propagations) for the aggregation process are consistent.

Most gradual trust-enhanced applications use a simple average aggregation that only takes into account trust. In this paper, however, we introduce several aggregation operators for both trust and distrust (Sect. 2). In Sect. 3, we investigate their applicability on a large real-world data set and show that they can be used to achieve accurate trust recommendations.

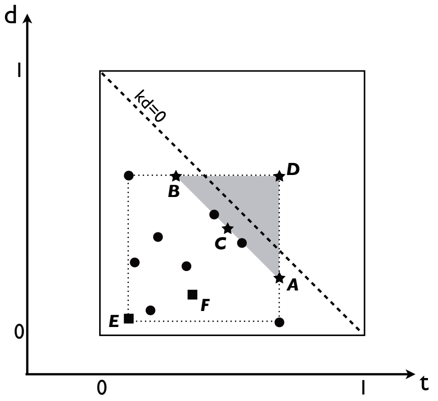

Fig. 1. Example of eight trust scores (dots) to be aggregated in $\mathcal{BL}^{\square}$.

2. Trust score aggregation operators

We first postulate three desirable properties and then set out to define a number of aggregation operators. We look for a trust score aggregation operator $A : ([0,1]^2)^n \rightarrow [0,1]^2$ $(n \geq 1)$ satisfying the following characteristics.

Definition 2.1 (Boundary preservation). *We say that a trust score aggregation operator A satisfies the trust (1), distrust (2) and knowledge (3) boundaries iff $A((t_1, d_1), \ldots, (t_n, d_n)) = (p, q)$, with*

(1) $\min(t_1, \ldots, t_n) \leq p \leq \max(t_1, \ldots, t_n)$
(2) $\min(d_1, \ldots, d_n) \leq q \leq \max(d_1, \ldots, d_n)$
(3) $p + q \geq \max(t_1 + d_1, \ldots, t_n + d_n)$

The first two conditions ensure that the aggregated trust score reflects a consensus about the (dis)trust estimation, it should not contain more (dis)trust than the maximum (dis)trust value among the aggregates (and analogously for the minimum). The last condition ensures that the knowledge contained in the aggregated trust score does not decrease when aggregating additional opinions. In other words, the aggregated trust score will contain at least as much knowledge as the most knowledgeable aggregate.

Fig. 1 shows an example in which eight trust scores, represented by dots, have to be aggregated. All of them are consistent, since they reside under the $kd = 0$ line (representing (t, d)'s with perfect knowledge, i.e., $t+d = 1$). The trust and distrust boundaries yield a limited number of possible aggregation results, marked out by the dotted lines. By also imposing the knowledge condition, however, only part of the possibilities remain, depicted by the gray area. Each of the trust scores marked by stars make sense as aggregated score: B is the most optimistic choice (maximum trust degree t for the lowest possible knowledge level), C the most pessimistic one (maximum distrust degree d), D the moderating approach (average of the most knowledgeable scores) and A the most extreme, knowledge maximizing, option: maximum t and d, often resulting in an inconsistent trust estimation.

Definition 2.2 (TMAX). *The trust maximizing trust score aggregation operator* $TMAX$ *is defined as* $TMAX\left((t_1, d_1), \cdots, (t_n, d_n)\right) = \left(\max\left(t_1, \cdots, t_n\right), \max\left(t_1 + d_1, \cdots, t_n + d_n\right) - \max\left(t_1, \cdots, t_n\right)\right)$.

Definition 2.3 (DMAX). *The distrust maximizing trust score aggregation operator* $DMAX$ *is defined as* $DMAX\left((t_1, d_1), \cdots, (t_n, d_n)\right) = \left(\max\left(t_1 + d_1, \cdots, t_n + d_n\right) - \max\left(d_1, \cdots, d_n\right), \max\left(d_1, \cdots, d_n\right)\right)$.

Definition 2.4 (KAV). *The knowledge preference averaging trust score aggregation operator is defined as* $KAV\left((t_1, d_1), \cdots, (t_n, d_n)\right) = (p, q)$*, such that* $p = \left(\sum_{i=1}^{n} w_i \cdot t_i\right) / \left(\sum_{i=1}^{n} w_i\right)$ *and* $q = \left(\sum_{i=1}^{n} w_i \cdot d_i\right) / \left(\sum_{i=1}^{n} w_i\right)$*, with* $w_i = 1$ *if* $t_i + d_i = \max\left(t_1 + d_1, \cdots, t_n + d_n\right)$*, and 0 otherwise.*

Definition 2.5 (KMAX). *The knowledge maximizing trust score aggregation operator* $KMAX$ *is defined as* $KMAX\left((t_1, d_1), \cdots, (t_n, d_n)\right) = \left(\max\left(t_1, \cdots, t_n\right), \max\left(d_1, \cdots, d_n\right)\right)$.

Proposition 2.1. *TMAX, DMAX, KAV and KMAX satisfy the trust, distrust and knowledge boundaries.*

3. Results and discussion

To measure the performance of the operators, we conduct experiments on a large data set from CouchSurfing with gradual trust and knowledge statements. Due to space restrictions, we omit the details about the translation of the trust and knowledge levels to trust scores. We take into account propagation paths of exactly length 2 (with P_1 and $T = \min$), so that every argument for the aggregation results from a propagation process (with the same path length). We use the leave one out method to compute the accuracy of the trust predictions. This method consists of hiding a trust relation (t_{r_i}, d_{r_i}) (the real value) and then trying to predict its hidden real value; the prediction is denoted by (t_{p_i}, d_{p_i}). The accuracy can then be measured by the trust mean average error $\text{MAE} = \left(\sum_{i=1}^{m} |t_{r_i} - t_{p_i}| + |d_{r_i} - d_{p_i}|\right)/m$ $(\in [0, 2])$, with m the number of leave one out experiments to perform.

We compare the operators with two baseline strategies (squares in Fig. 1): KMIN which computes $(t, d) = (\min(t_1, \cdots, t_n), \min(d_1, \cdots, d_n))$, and 'Fixed values', a strategy that always yields $(0.431, 0.206)$, which represents the average trust and distrust degree in the translated data set. Remark that these operators do not always reside in the gray area of Fig. 1.

Fig. 2 shows that TMAX and KAV achieve slightly better results than DMAX and KMAX. The baselines clearly produce the highest errors. Hence, at first glance, it looks as if good trust score aggregation operators must indeed fulfill all boundary conditions (gray area), and that there is no clear winner among TMAX, DMAX, KMAX and KAV.

However, as Fig. 3 illustrates, the split-up of the results according to the number n of aggregates gives us a completer picture of an operator's performance. Notice that all operators perform more or less equally for small n, but that these classes are exactly the ones that are overrepresented in our experiment. The bars illustrate e.g. that in more than 500 000 leave one out experiments there was only one propagation path, or in almost 300 000 exactly two, as opposed to about 1000 experiments which have to aggregate between 50 and 75 trust scores.

On average, one can see that it becomes more difficult to produce accurate predictions as the number of inputs starts to increase, and that there is clearly a performance difference between the operators: DMAX's and KMAX's MAE is often almost twice as high as TMAX's. Obviously, these two operators are too extreme for the CouchSurfing application. TMAX and KAV adapt themselves much better to changing aggregation conditions. Note that TMAX achieves somewhat lower errors in cases with more than 75 inputs, which can be explained by the fact that the average trust degree t_{r_i} for $n > 75$ is significantly higher than for $n \leq 75$ (viz. 0.594 vs. 0.423).

Fig. 2. Overall MAE performance of aggregation strategies; MAE $\in [0, 2]$.

Aggregation operator		Fig. 1	MAE
Trust maximizing	TMAX	A	0.316
Distrust maximizing	DMAX	B	0.325
Knowledge preference averaging	KAV	C	0.318
Knowledge maximizing	KMAX	D	0.322
Knowledge minimizing	KMIN	E	0.389
Fixed values		F	0.340

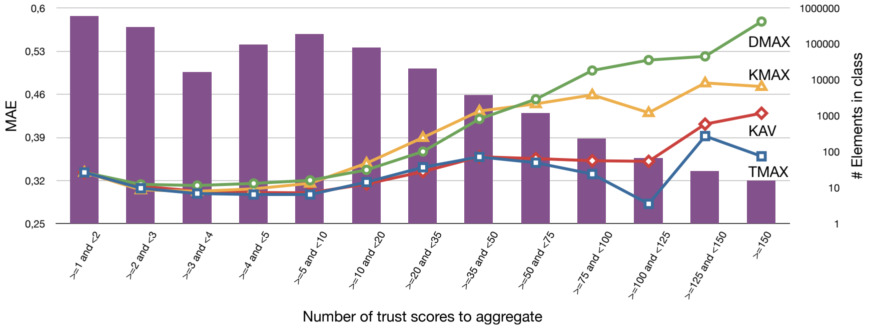

Fig. 3. MAE vs. the number of aggregates n.

4. Conclusions and future work

In this paper, we have built upon previous work[2] in which trust scores are drawn from a bilattice, and introduced four new trust score aggregation operators. Experiments on a large data set showed that trust maximizing and knowledge preference averaging operators perform much better than distrust maximizing or maximum operators. Obviously, the reported performances do not only depend on the choice of aggregation operator, but also on the combination with propagation, which inherently introduces errors in the computation too. Hence, a first step in our future research is the investigation of the synergy between the two operator types and their separate influence on the accuracy. In a next step, we plan to further refine the operators. E.g., only the most knowledgeable agents take part in the knowledge preference aggregation strategy, even if the difference with some of the other arguments is almost negligible. Mitigating its behavior by incorporating knowledge defects might improve the predictions.

Acknowledgements

The authors thank CouchSurfing for making the data available. Chris Cornelis thanks the Research Foundation—Flanders for funding his research.

References

1. M. Ginsberg, Multi-valued logics: A uniform approach to reasoning in artificial intelligence, *Comput Intel* **4**, 265–316 (1988).
2. P. Victor, C. Cornelis, M. De Cock and P. Pinheiro da Silva, Gradual trust and distrust in recommender systems, *Fuzzy Set Syst* **160**, 1367–1382 (2009).