International Journal on Applications in Engineering and Technology
Volume 4: Issue 2: May 2018, pp 71 – 75 www.aetsjournal.com ISSN (Online) : 2455 - 0523

WEB CONTENT TRUST RATING PREDICTION USING EVIDENCE THEORY

S.ARUNA, M.VIJAYANANTHKUMAR , R.VETRIVENTHAN, SENTHIL KUMARAN

Abstract— Social networks are a popular movement on the web. Trust can be used effectively on the Semantic Web as annotations to social relationships. In this paper, we present a two level approach to integrating trust, provenance, and annotations in Semantic Web systems. We describe an algorithm for inferring trust relationships using provenance information and trust annotations in Semantic Web-based social networks. Then, we present two applications that combine the computed trust values with the provenance of other annotations to personalize websites. The FilmTrust system uses trust to compute personalized recommended movie ratings and to order reviews. An open source intelligence portal, Profiles In Terror, also has a beta system that integrates social networks with trust annotations. We believe that these two systems illustrate a unique way of using trust annotations and provenance to process information on the Semantic Web.

Keywords— FOAF , FilmTrust , trust rating .

I. INTRODUCTION

Tracking the provenance of Semantic Web metadata can be very useful for filtering and aggregation, especially when the trustworthiness of the statements is at issue. In this paper, we will present an entirely Semantic Web-based system of using social networks, annotations, provenance, and trust to control the way users see information.

Social Networks have become a popular movement on the web as a whole, and especially on the Semantic Web. The Friend of a Friend (FOAF) vocabulary is an OWL format for representing personal and social network information, and data using FOAF makes up a significant percentage of all data on the Semantic Web. Within these social networks, users can take advantage of other ontologies for annotating additional information about their social connections. This may include the type of relationship (e.g. ”sibling”, ”significant other”, or ”long lost friend”), or how much they trust the person that they know. Annotations about trust are particularly useful, as they can be applied in two ways. First, using the annotations about trust and the provenance of those statements, we can compute personalized recommendations for how much one user (the source) should trust another unknown user (the sink) based on the paths that connect them in the social network and the trust values along those paths. Once those values can be computed, there is a second application of the trust values. In a system where users have made statements and we have the provenance information, we can filter the statements based on how much the individual user trusts the person who made the annotation. This allows for a common knowledge base that is personalized for each user according to who they trust.

In this paper, we will present a description of social networks and an algorithm for inferring trust relationships within them. Then, we will describe two systems where trust is used to filter, aggregate, and sort information: FilmTrust, a movie recommender system, and Profiles in Terror, a portal collecting open source intelligence on terrorist activities.

II. SOCIAL NETWORKS AND TRUST ON THE SEMANTIC WEB

Social networks on the Semantic Web are generally created using the FOAF vocabulary . There are over 10,000,000 people with FOAF files on the web, describing their personal information and their social connections . There are several ontologies that extend FOAF, including the FOAF Relationship Module and the FOAF Trust Module. These ontologies provide a vocabulary for users to annotate their social relationships in the network. In this research, we are particularly interested in trust annotations.

Using the FOAF Trust Module, users can assign trust ratings on a scale from 1 (low trust) to 10 (high trust).There are currently around 3,000 known users with trust relationships included in their FOAF profile. These statements about trust are annotations of relationships. There are interesting steps that can be taken once that information is aggregated. We can choose a specific user, and look at all of the trust ratings assigned to that person. With that information, we can get an idea of the average opinion about the person’s trustworthiness. Trust, however, is a

S.Aruna , Master of Computer Applications , Meenaakshi Ramasamy Engineering College , Thathanur , Ariyalur Dt , Tamil Nadu .

M.Vijayananthkumar , Assistant professor/MCA , Meenaakshi Ramasamy Engineering College , Thathanur , Ariyalur Dt , Tamil Nadu .

Mr.R.Vetriventhan , BE , MTech , MISTE , Head of the department , Department of MCA , Meenaakshi Ramasamy Engineering College , Thathanur , Ariyalur Dt , Tamil Nadu .

Mr.Senthil Kumaran , ME Phd , Managing Director , Meenaakshi Ramasamy Engineering College , Thathanur , Ariyalur Dt , Tamil Nadu.

subjective concept. Consider the simple example of asking whether the President is trustworthy. Some people believe very strongly that he is, and others believe very strongly that he is not. In this case, the average trust rating is not helpful to either group. However, since we have provenance information about the annotations, we can significantly improve on the average case. If someone (the source) wants to know how much to trust another person (the sink), we can look at the provenance information for the trust assertions, and combine that with the source's directly assigned trust ratings, producing a result that weights ratings from trusted people more highly

1) Background and Related Work

We present an algorithm for inferring trust relationships in social networks, but this problem has been approached in several ways before. Here, we highlight some of the major contributions from the literature and compare and contrast them with our approach.

There are several algorithms that output trust inferences, but none of them produce values within the same scale that users assign ratings. For example, many rely on eigenvector based approaches that produce a ranking of the trust worthiness, but the rankings do not translate to trust values in the same scale.

Raph Levin's Advogato project also calculates a global reputation for individuals in the network, but from the perspective of designated seeds (authoritative nodes). His metric composes certifications between members to determine the trust level of a person, and thus their membership within a group. While the perspective used for making trust calculations is still global in the Advogato algorithm, it is much closer to the methods used in this research. Instead of using a set of global seeds, we let any individual be the starting point for calculations, so each calculated trust rating is given with respect to that person's view of the network.

Richardson et. al. use social networks with trust to calculate the belief a user may have in a statement. This is done by finding paths (either through enumeration or probabilistic methods) from the source to any node which represents an opinion of the statement in question, concatenating trust values along the paths to come up with the recommended belief in the statement for that path, and aggregating those values to come up with a final trust value for the statement. Current social network systems on the Web, however, primarily focus on trust values between one user to another, and thus their aggregation function is not applicable in these systems.

2) Issues for Inferring rust

Then two individuals are directly connected in the network, they can have trust ratings for one another. Two people who are not directly connected do not have that trust information available by default. However, the paths connecting them in the network contain information that can be used to infer how much they may trust one another.

For example, consider that Alice trusts Bob, and Bob trust Charlie. Although Alice does not know Charlie, she knows and trusts Bob who, in turn, has information about how trustworthy he believes Charlie is. Alice can use in- formation from Bob and her own knowledge about Bob's trustworthiness to infer how much she may trust Charlie. This is illustrated in Figure 1.

To accurately infer trust relationships within a social network, it is important to understand the properties of trust networks. Certainly, trust inferences will not be as accurate as a direct rating. There are two questions that arise which will help refine the algorithm for inferring trust: how will the trust values for intermeidate people affect the accuracy of the inferred value, and how will the length of the path affect it.

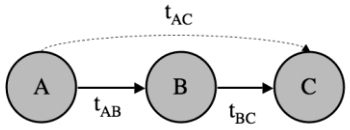

Figure 1: An illustration of direct trust values between nodes A and B (tAB), and between nodes B and C (tBC). Using a trust inference algorithm, it is possible to compute a value to recommend how much A may trust C (tAC).

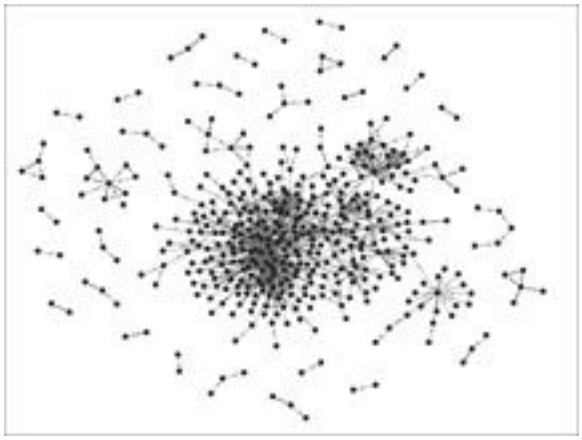

Figure 2: This figure illustrates the social network in the FilmTrust website. There is a large central cluster of about 450 connected users, with small, independent groups of users scattered around the edges.).

We expect that people who the user trusts highly will tend to agree with the user more about the trustworthiness of others than people who are less trusted. To make this comparison, we can select triangles in the network. Given nodes n_i, n_j, and n_k

, where there is a triangle such that we have trust values t_{ij}, t_{ik}, and t_{kj}, we can get a measure of how trust of an intermediate person can affect accuracy. Call Δ the difference between the known trust value from n_i to n_k (t_{ik}) and the value from n_j to n_k (t_{ik}). Grouping using several real networks. The first network is part of the Trust Project, a Semantic Web-based network with trust values and approximately 2,000 users. The FilmTrust network , see Figure 2, is a network of approximately 700 users oriented around a movie rating and review website. We will use FilmTrust for several examples in this paper. Details of the analysis can be found in the referenced work, but we present an overview of the analysis here.

TABLE 1: MINIMUM Δ FOR PATHS OF VARIOUS LENGTHS CONTAINING THE SPECIFIED TRUST RATING.

Trust Value	Path Length			
	2	3	4	5
10	0.953	1.52	1.92	2.44
9	1.054	1.588	1.969	2.51
8	1.251	1.698	2.048	2.52
7	1.5	1.958	2.287	2.79
6	1.702	2.076	2.369	2.92

To see the relationship between path length and trust, we performed an experiment. We selected a node, n_i, and then selected an adjacent node, n_j. This gave us a known trust value t_{ij}. We then ignored the edge from n_i to n_j and looked for paths of varying lengths through the network that connected the two nodes. Using the trust values along the path, and the expected error for those trust values, as determined by the analysis of the correlation of trust and similarity determined in. Call this measure of error Δ. This comparison is repeated for all neighbors of n_i, and for all n_i in the network.

For each path length, Table 1 shows the minimum average Δ ,(Δ) These are grouped according to the minimum trust value along that path.

In Figure 3, the effect of path length can be compared to the effects of trust ratings. For example, consider the Δ for trust values of 7 on paths of length 2. This is approximately the same as the Δ for trust values of 10 on paths of length 3 (both are close to 1.5). The Δ for trust values of 7 on paths of length 3 is about the same as the Δ for trust values of 9 on paths of length 4. A precise rule cannot be derived from these values because there is not a perfect linear relationship, and also because the points in Figure 3 are only the minimum Δ among paths with the given trust rating.

3) TidalTrust: An Algorithm for Inferring Trust

The effects of trust ratings and path length described in the previous section guided the development of TidalTrust, an algorithm for inferring trust in networks with continuous rating systems. The following guidelines can be extracted from the analysis of the previous sections: 1. For a fixed trust rating, shorter paths have a lower Δ. 2. For a fixed path length, higher trust ratings have a lower Δ. This sec- tion describes how these features are used in the TidalTrust algorithm.

4) Incorporating Path Length

The analysis in the previous section indicates that a limit on the depth of the search should lead to more accurate results, since the Δ increases as depth increases. If accuracy decreases as path length increases, as the earlier analysis suggests, then shorter paths are more desirable. However, the tradeoff is that fewer nodes will be reachable if a limit is imposed on the path depth. To balance these factors, the path length can vary from one computation to another. In- stead of a fixed depth, the shortest path length required to connect the source to the sink becomes the depth. This preserves the benefits of a shorter path length without limiting the number of inferences that can be made.

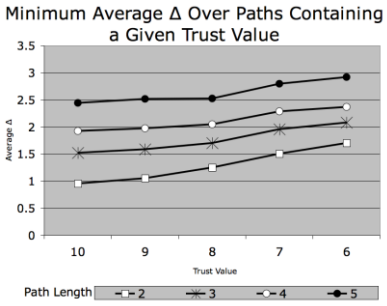

Figure 3: Minimum Δ from all paths of a fixed length containing a given trust value. This relationship will be integrated into the algorithms for inferring trust presented in the next section.

5) Incorporating Trust Values

The previous results also indicate that the most accurate information will come from the highest trusted neighbors. As such, we may want the algorithm to limit the information it receives so that it comes from only the most trusted neighbors, essentially giving no weight to the information from neighbors with low trust. If the algorithm were to take information only from neighbors with the highest trusted neighbor, each node would look at its neighbors, select those with the highest trust rating, and average their results. However, since

different nodes will have different maximum values, some may restrict themselves to returning information only from neighbors rated 10, while others may have a maximum assigned value of 6 and be returning information from neighbors with that lower rating. Since this mixes in various levels of trust, it is not an ideal approach. On the other end of possibilities, the source may find the maximum value it has assigned, and limit every node to returning information only from nodes with that rating or higher. However, if the source has assigned a high maximum rating, it is often the case that there is no path with that high rating to the sink. The inferences that are made may be quite accurate, but the number of cases where no inference is made will increase. To address this problem, we define a variable max that represents the largest trust value that can be used as a minimum threshold such that a path can be found from source to sink.

6) Full Algorithm for Inferring Trust

Incorporating the elements presented in the previous sections, the final Tidal Trust algorithm can be assembled. The name was chosen because calculations sweep forward from weights to calculate a weighted average rating. Because the inferred trust values reflect how much the user should trust the opinions of the person rating the movie, the weighted average of movie ratings should reflect the user's opinion. If the user has an opinion that is different from the average, the rating calculated from trusted friends who should have similar opinions should reflect that difference. Similarly, if a movie has multiple reviews, they are sorted according to the inferred trust rating of the author. This presents the reviews authored by the most trusted people first to assist the user in finding information that will be most relevant.

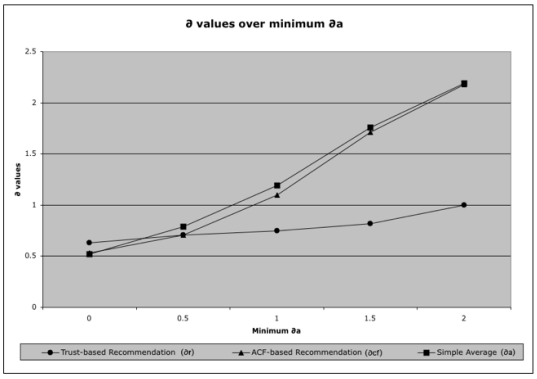

Figure 4: A user's view of the page for "A Clockwork Orange," where the recommended rat- ing matches the user's rating, even though δa is very high (δa = 2.5).).

7) Site Personalization: Movie Ratings

One of the features of the Film Trust site that uses the social network is the "Recommended Rating" feature. As figure 4 shows, users will see this in addition to the average rating given to a particular movie.

The trust values are used in conjunction with the Tidal Trust algorithm to present personalized views of movie pages. When the user chooses a film, they are presented with basic film data, the average rating of the movie, a personalized recommended rating , and the reviews written by users. The personalized recommended rating is computed by first selecting a set of people who rated the movie. The selection process considers trust and path length; details on how this set of people are chosen are provided in [5]. Using the trust values (direct or inferred) for each person in the set who rated the movie as a weight, and computing the weighted average rating. For the set of selected nodes S, the recommended rating r from node s to movie m is the average of the movie ratings from nodes in S weighted by the trust value t from s to each node:

$$r_{sm} = \frac{\sum_{i \in S} t_{si} r_{im}}{\sum_{i \in S} t_{si}} \qquad (2)$$

This average is rounded to the nearest half star, and that value becomes the "Recommended Rating" that is personalized for each user.

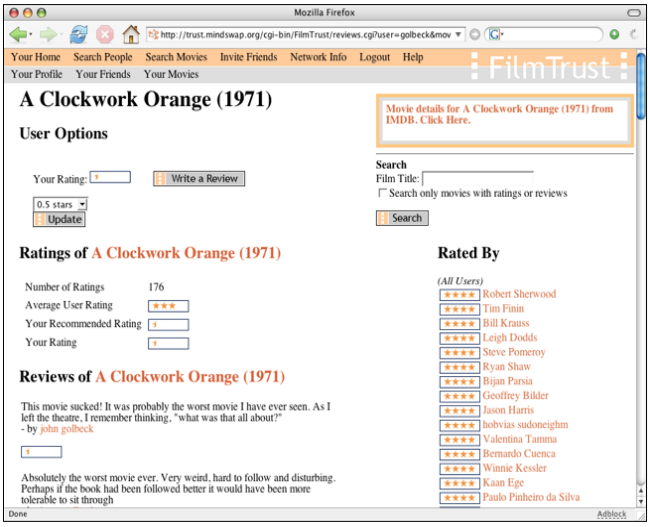

Figure 5: The increase in δ as the minimum δa is increased. Notice that the ACF-based recommendation (δcf) closely follows the average (δa). The more accurate Trust-based recommendation (δr) significantly outperforms both other methods.

As a simple example, consider the following: Alice trusts Bob 9 Alice trusts Chuck 3 Bob rates the movie "Jaws" with 4 stars Chuck rates the movie "Jaws" with 2 stars Then Alice's recommended rating for "Jaws" is calculated as follows:

$$\frac{t_{Alice->Bob}\, r_{Bob->Jaws} + t_{Alice->Chuck}\, r_{Chuck->Jaws}}{t_{Alice->Bob} + t_{Alice->Chuck}} = \frac{(9 \cdot 4 + 3 \cdot 2)}{9+3} = \frac{42}{12} = 3.5$$

For each movie the user has rated, the recommended rating can be compared to the actual rating that the user as signed. In this analysis, we also compare the user's rating with the average rating for the movie, and with a recommended rating generated by an automatic collaborative filtering (ACF) algorithm. There are many ACF algorithms, and one that has been well tested, and which is used here, is the classic user-to-user nearest neighbor prediction algorithm based on Pearson Correlation. If the trust-based method of calculating ratings is best, the difference between the personalized rating and the user's actual rating should be significantly smaller than the difference between the actual rating and the average rating.

Figure 6: A sample page from the PIT portal illustrating provenance information for a statement, as well as probabilities.

On first analysis, it did not appear that that the personalized ratings from the social network offered any benefit over the average. The difference between the actual rating and the recommended rating (call this δr) was not statistically different than the difference between the user's actual rating and the average rating (call this δa). The difference between a user's actual rating of a film and the ACF calculated rating (δcf) also was not better than δa in the general case. A close look at the data suggested why. Most of the time, the majority of users actual ratings are close to the average. This is most likely due to the fact that the users in the FilmTrust system had all rated the AFI Top 50 movies, which received disproportionately high ratings. A random sampling of movies showed that about 50% of all ratings were within the range of the mean +/- a half star (the smallest possible increment). For users who gave these near mean rating, a personalized rating could not offer much benefit over the average.

III. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a two level approach to integrating trust, provenance, and annotations in Semantic Web systems. First, we presented an algorithm for computing personalized trust recommendations using the provenance of existing trust annotations in social networks. Then, we introduced two applications that combine the computed trust values with the provenance of other annotations to personalize websites. In Film Trust, the trust values were used to compute personalized recommended movie ratings and to order reviews. Profiles In Terror also has a beta system that integrates social networks with trust annotations and provenance information for the intelligence information that is part of the site. We believe that these two systems illustrate a unique way of using trust annotations and provenance to process information on the Semantic Web.

IV. ACKNOWLEDGMENTS

This work, conducted at the Maryland Information and Network Dynamics Laboratory Semantic Web Agents Project, was funded by Fujitsu Laboratories of America College Park, Lockheed Martin Advanced Technology Laboratory, NTT Corp., Kevric Corp., SAIC, the National Science Foundation, the National Geospatial Intelligence Agency, DARPA, US Army Research Laboratory, NIST, and other DoD sources.

References

[1] T. Beth, M. Borcherding, and B. Klein. Valuation of trust in open networks. Proceedings of ESORICS 94., 1994.
[2] I. Davis and E. V. Jr. Relationship: A vocabulary for describing relationships between people. 2004.
[3] J. P. Delgrande and T. Schaub. Expressing preferences in default logic. Artif. Intell., 123(1-2):41–87, 2000.
[4] J. Golbeck. Computing and Applying Trust in Web-based Social Networks. Ph.D. Dissertation, University of Maryland, College Park, 2005.
[5] J. Golbeck. Filmtrust: Movie recommendations using trust in web-based social networks. Proceedings of the Consumer Communication and Networking Conference, 2006.
[6] J. Golbeck. Generating Predictive Movie Recommendations from Trust in Social Networks. Proceedings of The Fourth International Conference on Trust Management, 2006.
[7] J. Herlocker, J. A. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. Proceedings of the 2000 ACM conference on Computer supported cooperative work, 2000.
[8] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. Proceedings of the 12th International World Wide Web Conference, May 20-24, 2004.
[9] R. Levin and A. Aiken. Attack resistant trust metrics for public key certification. 7th USENIX Security Symposium, 1998.
[10] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web.