Combining Provenance with Trust in Social Networks for Semantic Web Content Filtering

Jennifer Golbeck

University of Maryland, College Park, College Park MD 20742, USA
golbeck@cs.umd.edu
http://mindswap.org

Abstract. Social networks are a popular movement on the web. On the Semantic Web, it is simple to make trust 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 an application, FilmTrust, that combines the computed trust values with the provenance of other annotations to personalize the website. The FilmTrust system uses trust to compute personalized recommended movie ratings and to order reviews. We believe that the results obtained with FilmTrust illustrate the success that can be achieved using this method of combining trust and provenance on the Semantic Web.

1 Introduction

Social Networks have become a popular movement on the web as a whole, and the Semantic Web is rich with social network information. Friend of a Friend (FOAF) is an OWL-based vocabulary for representing personal and social network information; 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 can be 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.

L. Moreau and I. Foster (Eds.): IPAW 2006, LNCS 4145, pp. 101–108, 2006.

In this paper, we will present a description of social networks and an algorithm for inferring trust relationships within them. Then, we will describe FilmTrust, a movie recommender system, where trust is used to filter, aggregate, and sort information.

2 Social Networks and Trust on the Semantic Web

Social networks on the Semantic Web are usually created using the FOAF vocabulary [2]. There are over 10,000,000 people with FOAF files on the web, describing their personal information and their social connections [4]. There are several ontologies that extend FOAF, including the FOAF Relationship Module [3] and the FOAF Trust Module [4]. 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 profiles. Once that information is aggregated, we can make computations with trust values. We 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 subjective concept. Consider the simple example of asking weather 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.

In this work, we use the term "provenance" to refer to *who* made a particular statement. Since we have provenance information about the trust annotations in FOAF networks, 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 than those from untrusted people.

In this section, we present an algorithm for inferring trust relationships that combines provenance information with the user's direct trust ratings.

2.1 Background and Related Work

When two individuals are directly connected in the network, they can have trust ratings for one another. Two people who are not directly connected to 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 information from Bob and her own knowledge about Bob's trustworthiness to infer how much she may trust Charlie. This is illustrated in Figure 1.

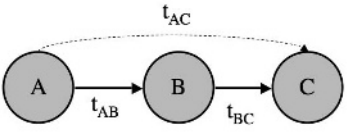

Fig. 1. An illustration of direct trust values between nodes A and B (t_{AB}), and between nodes B and C (t_{BC}). Using a trust inference algorithm, it is possible to compute a value to recommend how much A may trust C (t_{AC}).

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 intermediate people affect the accuracy of the inferred value, and how will the length of the path affect it.

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.

The EigenTrust algorithm [7] is used in peer-to-peer systems and calculates trust with a variation on the PageRank algorithm[9], used by Google for rating the relevance of web pages to a search. EigenTrust is designed for a peer-to-peer system while ours is designed for use in humans' social networks, and thus there are differences in the approaches to analyzing trust. In the EigenTrust formulation, trust is a measure of performance, and one would not expect a single peer's performance to differ much from one peer to another. Socially, though, two individuals can have dramatically different opinions about the trustworthiness of the same person. Our algorithms intentionally avoid using a global trust value for each individual to preserve the personal aspects that are foundations of social trust.

Raph Levin's Advogato project [8] 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.[10] 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.

2.2 Issues for Inferring Trust

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 the Δ values by the trust value for the intermediate node (t_{ij}) indicates on average how trust for the intermediate node affects the accuracy of the recommended value. Several studies [11],[4] have shown a strong correlation between trust and user similarity in several real-world networks.

It is also necessary to understand how the paths that connect the two individuals in the network affect the potential for accurately inferring trust relationships. The length of a path is determined by the number of edges the source must traverse before reaching the sink. For example, source-sink has length two. Does the length of a path affect the agreement between individuals? Specifically, should the source expect that neighbors who are connected more closely will give more accurate information than people who are further away in the network? Previous work[4],[6] has also addressed this issue and shown that, as expected, shorter paths lead to more accurate information. As with trust values, it will be important to consider the length of connecting paths when developing an algorithm for inferring trust.

2.3 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 error ($\overline{\Delta}$). 2. For a fixed path length, higher trust ratings have a lower $\overline{\Delta}$. This section describes how these features are used in the TidalTrust algorithm.

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 $\overline{\Delta}$ 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. Instead 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.

Incorporating Trust Values. The previous results also indicate that the most accurate information will come from the highest trusted neighbors. To incorporate this into the algorithm, we establish a minimum trust threshold, and only consider connections in the network with trust ratings at or above the threshhold. This value cannot be fixed before the search because we cannot predict what the highest trust value will be along the possible paths. If the value is set too high, some nodes may not have assigned values and no path will be found. If the threshold is too low, then paths with lower trust may be considered when it is not necessary. 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. max is computed while searching for paths to the sink by tracking trust values that have been seen.

Full Algorithm for Inferring Trust. Incorporating the elements presented in the previous sections, the final TidalTrust algorithm can be assembled. The name was chosen because calculations sweep forward from source to sink in the network, and then pull back from the sink to return the final value to the source.

$$t_{is} = \frac{\sum_{j \in adj(j) \mid t_{ij} \geq max} t_{ij} t_{js}}{\sum_{j \in adj(j) \mid t_{ij} \geq max} t_{ij}} \tag{1}$$

TidalTrust is a modified breadth-first search. The source's inferred trust rating for the sink ($t_{source,sink}$) is a weighted average if the source's neighbors' ratings of the sink (see Forumula 1).

The source node begins a search for the sink. It will poll each of its neighbors to obtain their rating of the sink. If the neighbor has a direct rating of the sink, that value is returned. If the neighbor does not have a direct rating for the sink, it queries all of its neighbors for their ratings, computes the weighted average as shown in Formula 1, and returns the result .

To improve the accuracy of the algorithm, path length and path strength considerations are included. At each node that is reached in the search, Each node that is reached performs this process, keeping track of the current depth from the source. Each node will also keep track of the strength of the path to it. Nodes adjacent to the source will record the source's rating assigned to them. Each of those nodes will poll their neighbors. The strength of the path to each neighbor is the minimum of the source's rating of the node and the node's rating of its neighbor. The neighbor records the maximum strength path leading to it. Once a path is found from the source to the sink, the depth is set at the maximum depth allowable. Since the search is proceeding in a Breadth First

Search fashion, the first path found will be at the minimum depth. The search will continue to find any other paths at the minimum depth. Once this search is complete, the trust threshold (max) is established by taking the maximum of the trust paths leading to the sink. With the max value established, each node can complete the calculations of a weighted average by taking information from nodes that they have rated at or above the max threshold.

The accuracy of this algorithm is addressed in depth in [4] and [6]. While the error will very from network to network, our experiments in two real world social networks show the results to be accurate to within about 10%.

2.4 Accuracy of TidalTrust

As presented above, TidalTrust strictly adheres to the observed characteristics of trust: shorter paths and higher trust values lead to better accuracy. However, there are some things that should be kept in mind. The most important is that networks are different. Depending on the subject (or lack thereof) about which trust is being expressed, the user community, and the design of the network, the effect of these properties of trust can vary. While we should still expect the general principles to be the same−shorter paths will be better than longer ones, and higher trusted people will agree with us more than less trusted people−the proportions of those relationships may differ from what was observed in the sample networks used in this research. A more extensive comparison and analysis of accuracy, including a comparison to a PKI algorithm[1], is available in [6] and [4].

Table 1. $\overline{\Delta}$ for TidalTrust and Simple Average recommendations in both the Trust Project and FilmTrust networks. Numbers are absolute error on a 1-10 scale.

	Algorithm	
Network	TidalTrust	Simple Average
Trust Project	1.09	1.43
FilmTrust	1.35	1.93

3 Using Trust to Personalize Content

While the computation of trust values is in and of itself a user of provenance and annotations together, the resulting trust values are widely applicable for personalizing content. If we have provenance information for annotations found on the semantic web, and a social network with trust values such that a user can compute the trustworthiness of the person who asserted statement, then the information presented to the user can be sorted, ranked, aggregated, and filtered according to trust.

FilmTrust, at http://trust.mindswap.org, is a website with a social network. Users can rate movies on a scale of 0.5 to 4 stars, and write reviews of films. While the users interact with a simple web interface, the data is all stored as Semantic

Web annotations. In the social network, users also rate the trustworthiness of their friends on a scale of 1-10 using the FOAF Trust Module.

The trust values are used in conjunction with the TidalTrust 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:

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

We tested the quality of these results in FilmTrust by comparing the trust-based rating with the known rating that a user gave to a movie. While the experimental details are beyond the scope of this paper, the results were encouraging. We have shown in [4] and [6] that in the FilmTrust system, recommended ratings produced with trust are significantly more accurate than the simple average ratings as well as recommended ratings generated using a Pearson correlation-based automated collaborative filtering algorithm when the user's' opinion of a movie is at least 1 star different from the average.

Trust values for users in the system are also used to order movie reviews. When there are multiple reviews for a movie, the reviews from the most trusted users are displayed first. Thus, information from people the users trust is displayed more prominently. A small user study[4] showed a strong user preference for this ordering, because the most relevant information for the user was easiest to see.

4 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 FilmTrust, the trust values were used to compute personalized recommended movie ratings and to order reviews. We believe the FilmTrust system offers promise for using trust systems for additional content filtering. We envision social networks with trust values being incorporated to more critical systems to judge statements. We are currently working to combine a social network with Profiles In Terror[1], an open source intelligence

[1] http://profilesinterror.mindswap.org

project. Intelligence professionals can assign trust based on how much they trust the information and analyses provided by other users. That, in turn, can be used with provenance about the statements to rate the quality of information in the system.

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. D. Brickley and L. Miller. Foaf vocabulary specification. *http://xmlns.com/foaf/0.1/*, 2005.
3. I. Davis and E. V. Jr. Relationship: A vocabulary for describing relationships between people. 2004.
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. 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.
8. R. Levin and A. Aiken. Attack resistant trust metrics for public key certification. *7th USENIX Security Symposium*, 1998.
9. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. *Technical Report 1998, Stanford University*, 1998.
10. M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. *Proceedings of the Second International Semantic Web Conference*, 2003.
11. C.-N. Ziegler and J. Golbeck. Investigating Correlations of Trust and Interest Similarity. *Decision Support Services*, 2006.