Does Social Contact Matter? Modelling the Hidden Web of Trust Underlying Twitter*

Mozhgan Tavakolifard
Computer and Information Science Department
Norwegian University of Science and Technology
mozhgan@idi.ntnu.no

Kevin C. Almeroth
Dept of Comp Sci
UC-Santa Barbara
almeroth@cs.ucsb.edu

Jon Atle Gulla
Computer and Information Science Department
Norwegian University of Science and Technology
jag@idi.ntnu.no

ABSTRACT

Social recommender systems aim to alleviate the information overload problem on social network sites. The social network structure is often an important input to these recommender systems. Typically, this structure cannot be inferred directly from declared relationships among users. The goal of our work is to extract an underlying hidden and sparse network which more strongly represents the actual interactions among users. We study how to leverage Twitter activities like micro-blogging and the network structure to find a simple, efficient, but accurate method to infer and expand this hidden network. We measure and compare the performance of several different modeling strategies using a crawled data set from Twitter. Our results reveal that the structural similarity in the network generated by users' retweeting behavior outweighs the other discussed methods.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentations**]: Miscellaneous

Keywords

Recommender systems, Social Networks, Similarity, Trust, Twitter

1. INTRODUCTION

Social recommender systems have emerged as a promising solution for the information overload problem on social network sites. Through a recommender system, a user can be provided with more proactive and personalized information services. A key challenge is what information to use to provide the recommendations. The structure of a user's social network is an important input to the recommendation algorithm. Our focus, therefore, is on techniques to determine a user's social network.

A key problem, however, is that the most online social networks do not distinguish between high and low quality relationships [9]. Users interact with very few other people in their networks. A study of social interactions within Twitter reveals that the driver of usage is a sparse and *hidden* network of connections underlying the "declared" set of friends and followers [9]. This hidden network is often referred to as the "web-of-trust" [1, 2, 13, 7, 4]. We believe that using the more powerful web-of-trust is a better input into a recommendation algorithm than a user's entire social network.

Our goal in this work is to determine a user's web-of-trust hidden within the larger social network. To understand our goal, we first provide a reminder that the term "trust" is not defined to mean human trust, i.e., that two people can fully rely on each other, but rather, the idea that an interaction between two trusted parties will have an expected outcome. Given the important distinctions, we described the concept of trust more fully later in the paper.

The process of building a user's web-of-trust would be simple if "trust values" were explicit and identifiable. In practice, however, this is not the case. Therefore, the challenge is to determine a user's implicit trust values. We believe one way to determine these values is to study the interactions between users, for example, how often they interact and the kind of interaction that takes place.

In this work, the focus of our study is on Twitter, and identifying the hidden web-of-trust for Twitter users. Our goal is to design efficient methods to filter and discover the implicit trust relationships among Twitter users based on the set of activities and relationships that exist in Twitter.

Our approach has two main steps: inference (filtering) and propagation (discovery). In the first step, the existing relationships among Twitter users are filtered using users' behaviors as indications of trust (i.e., retweeting and adding tweets to the 'favorite' list). The webs-of-trust developed in this step is often too sparse to be helpful in practice since users usually have relationships and interactions with only a very small fraction of the total population. Thus, very often there will be no link to an intended new user. In the second step, propagation, the webs-of-trust are expanded by prediction of new potential links using four different propagation methods.

It is impossible to compare the estimated web-of-trust, generated in the first step, with a user's actual web-of-trust, as the actual trust relationships cannot be known. Therefore, we focus our evaluation on the second step. We compare the webs-of-trust determined in the second step using different metrics and also study the efficiency of each propagation method. We used a crawled data set from Twitter with more than 20,000 users as a data set. The main ad-

*This work is partially funded by Telenor Research and Future Studies.

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

vantages of our solution are efficiency, acceptable accuracy, and scalability. Furthermore, we use implicit information as opposed to trust modeling based on explicit trust ratings.

While our long-term goal is to identify the actual web-of-trust, we believe this work is a key first step. Future work will entail including the methods presented in this paper with methods for content analysis since users are not passive consumers of content, they are often content producers as well.

The remainder of this paper is organized as follows. Section 2 provides background information related to our work. The relevant work is explored in Section 3. We present the details of our proposal in Section 4. We describe an analysis and evaluation of our proposal in Section 5. Finally, Section 6 provides concluding remarks and future research directions.

2. BACKGROUND

In this section, we introduce the necessary background for creation of an implicit web-of-trust on Twitter. Twitter, as an online social network, is an information sharing system where users choose to follow other notable users to gain updates on news and statuses. Relationship links on Twitter are directional, meaning that each user has followers and followees. Twitter allows users to post and exchange 140-character-long messages, which are also known as "tweets". Twitter users usually use hashtags (#) to identify certain topics. Hashtags are assigned to a tweet in its own body text. Tweets can be published by sending e-mails, SMS text-messages, and directly from smart phones using a wide array of Web-based services. Furthermore, Tweets can be repeated throughout the network by "re-tweeting". A retweeted message usually starts with "RT @username", where the @ sign represents a reference to the one who originally posted the messages. The strength of Twitter as a medium for information diffusion is based in large part on its speed of retweets. Retweeting is often used as an indication that the original information was of high value or significant interest [3]. Users can also add tweets to their "favorite" list.

The simplicity of Twitter and its real-time message streams are its most powerful features. These real-time message streams have greatly expanded the usage of social network sites, from political campaigning to education, and from emergency news reporting to marketing and public relations. In particular, Twitter is an ideal environment for the dissemination of breaking-news directly from the news source and/or geographical location of the events. It has made interesting inroads into novel domains such as emergency response and recovery under crisis situation (*e.g.,*, Twitter-based early warning systems [16], help during a large-scale fire emergency[1], updates during riots in Kenya[2], and live traffic updates to track commuting delays[3]).

The most basic definition of "trust" covers various structurally connected phenomena. It is crucial to distinguish at least between two kinds and meanings of trust: (a) trust as a *psychological attitude* of Person A towards Person B relative to some possible desirable behavior, and (b) trust as the *decision* and the *act of relying on* another person.

Our study only deals with the first type of trust. Motivated by the idea of using implicit user behaviors as indications of trust, we adopt the following as our trust definition in online social networks: "trust in a person is a commitment to an action based on a belief that the future actions of that person will lead to a good outcome" [7]. The action and commitment in this definition do not have to be significant. For example, in the case of Twitter, we could say User A trusts User B regarding the semantics of Twitter messages (tweets) if she chooses to read the messages that User B posts, retweets, or identifies as a favorite (commits to an action) based on her belief that User B will not waste her time.

The main intuition and motivation for our work is that trust between two users may result in certain typical behaviors. These behaviors are not only an expression of trust, but could facilitate the development of other trust relationships. Although, such behavioral expressions are not guaranteed expressions of trust, the more often they occur, the stronger the indication. Several properties of trust follow from this definition [7], namely:

- **Transitivity**: the primary property of trust is transitivity. Trust is not perfectly transitive in the mathematical sense, that is, if User A highly trusts User B, and User B highly trusts User C, it does not always follow that User A will highly trust User C. There is, however, a notion that trust can be passed between people.

- **Composability**: There is a greater belief in the validity of information if it comes from several people. Therefore, if we look at trust recommendations as evidence to support the belief component of trust, then trust values from many sources can be combined to form a single opinion.

- **Asymmetry and personalization**: Trust is not necessarily identical in both directions, because individuals have different experiences, psychological backgrounds, and histories.

An online social network can be modeled as a graph with users as the nodes and the relationships among users as the links. A web-of-trust is a directed weighted graph with the same nodes as the online social network graph. The weight on the link from User A to User B indicates the strength of the trust attitude User A has towards User B.

3. RELATED WORK

The majority of the related work relies on explicit trust ratings. The various approaches to trust inference can be summarized as: statistical patterns and clustering techniques [1], trust ontology [2], PageRank [13], Semantic Web-based approaches [7], Bayesian networks [7], and game theory and social network measures [4]. An important characteristic of Twitter is its real-time nature. Hence, the efficiency of such algorithms is very important, but has not received much attention in the field. In this section, the existing work is compared according to the information source and the algorithm that is used for trust prediction. We describe only the most relevant subset of work on trust modeling for social networks.

Sibel et al. present measures of trust based in social networks [1]. The basis of their approach is an assumption that trust results in communication behavior patterns that are statistically different from communication between random

[1]http://factoryjoe.com/blog/2007/10/22/twitter-hashtags-for-emergency-coordination-and-disaster-relief/
[2]http://www.economist.com/node/10608764/
[3]http://lifehacker.com/355453/track-commuting-delays-via-twitter-with-commuter-feed/

members of a network. The proposed measure of who-trusts-whom in the network relies on detecting statistically significant patterns of the trust-like behavior and they validated these measures on Twitter network data.

Anantharam et al. developed a general ontology of trust that is independent of specific domain and discussed how concepts in their ontology can be used in the context of Twitter as an application scenario [2]. They define two types of trust called referral trust where one user sends another user's tweet and functional trust where one user follows another user.

Noordhuis et al. applied PageRank (the Google's method for measuring the relative quality of a URL) to Twitter's social graph of users and their followers to determine users of importance [13].

Golbeck introduces an approach to integrate trust with annotations in Semantic Web systems [7]. In another paper, Kuter and Golbeck propose to model the trust network as a Bayesian network . Therefore, they also use explicit trust information.

Ziegler and Lausen introduce Appleseed, a local group trust metric based on spreading activation models, designed for computing subjective neighborhoods of most trustworthy peers on the network [21]. The basic intuition of Appleseed is motivated by *spreading activation models* from Cognitive Science.

Buskens proposes a game-theoretic solution [4]. Problematic social situations can be described as trust games with two players and two periods of play. A Trust Game is a one-sided Prisoner's Dilemma Game.

There are several other projects that present trust modeling between a user and a statement on social networks. For example, Richardson et al. use social networks with trust to calculate the belief a user may have in a statement [15]. Current social network systems on the Web, however, primarily focus on trust values between one user and another, and thus, their aggregation function is not applicable in these systems.

In this work, we mainly focused on behavioral features and methods to compute the social ties individuals have and the strength of these ties without the help of textual features. We used the definition of trust in social networks from the work by Goldbeck [7] and three behaviors as the indication of trust. The retweet behavior is also mentioned in the work by Sibel et al. [1] and Anantharam et al. [2]. Therefore, we confirm and complement their results in this way. Furthermore, the weighted-transitivity formula is inspired by the work proposed by Golbeck [7]. In the next section, we present the details of trust modeling on Twitter.

4. INFERENCE AND EXPANSION OF THE HIDDEN WEB OF TRUST ON TWITTER

A link between two users in the web-of-trust is created as a result of two steps. In the first step, *inference*, the hidden web-of-trust is extracted from the Twitter network (i.e., follower-following network) using the implicit information about users' behaviors. This step can be thought as an abductive reasoning approach, which is a backward reasoning from consequences to hypotheses. In this case, the hypothesis is a link in the web-of-trust and the consequences are, for example, the retweeting and favorite actions. Apparently, there might be other hypotheses as well, such as the user might be interested in the topic. Thus, such behavioral expressions of trust should be more viewed as noisy indicators. The more often they occur, the more likely that a link exists on the web-of-trust [1].

The goal of the second step, *propagation*, is to make the generated web-of-trust generated less sparse. We applied and compared four different methods for prediction of non-exiting links. The details of these two steps are provided in the following sub-sections.

4.1 Inference

This step corresponds to filtering social relationships among users to identify the stronger relationships. Our baseline is the original social network, that is the network of users based on follower-following relationships. As discussed in Section 2, this network can be represented as a directed graph with users as nodes and social relations as links. We consider two behaviors as an implicit expression of a stronger relationship on Twitter: retweeting, and the behavior of making tweets favorites. There are also other possible indicating behaviors (e.g., direct conversation between users called *mentions* in Twitter) that we did not consider. We chose our indicating behaviors intuitively; and in our opinion, *direct conversations* are not necessarily an indication of a trust relationship between two persons. In the following, we describe how each behavior can be considered as an indication of trust.

- **Retweet**: This indication of trust is based on the propagation of information. If User B retweets messages from User A often, then we assume that User B must be implicitly trustworthy to User A. The motivation behind this idea is that we observed people only retweets from a small number of users and only a subset of a user's followers actually retweet.

- **Favorite**: If User A often makes some of user B tweets as his/her own favorites, we consider this to implicitly indicate that User A trusts User B.

The Retweet and Favorite webs-of-trust are directed and weighed graphs with nodes as users and links as implicit trust relationships among them. The weights are in the range (0, 1] and the higher weight mean the stronger relationship. For the web-of-trust generated from retweeting behavior, the weight of a trust link from User A towards User B may be calculated in two different ways: (a) as the proportion of the messages User A has retweeted from User B to the total messages that User A has retweeted so far, and (b) as the proportion of the messages User A has retweeted from User B to the total tweet messages of User B. The first measure captures how much of User A's retweeting effort is spent retweeting messages from User B, while the second measure captures the fraction of User B's messages that User A considers worthy of retweeting. We have tried both in our experiments, and they yield similar results. For this reason, we only report the results of (a). In extremely heterogeneous networks, these two measures could capture different aspects of trust, however in homogeneous networks they behave similarly. A similar calculation is used for assigning the weights to trust links in the Favorite web-of-trust.

The webs-of-trust resulting from these inference methods are often too sparse to be helpful in practice (e.g., to be used by a recommender system) [17]. The primary reason is users typically have relationships or interactions with only a very

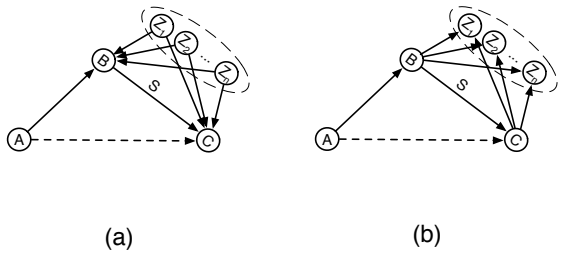

Figure 2: Trust propagation through structural similarity: (a) similar trustees and (b) similar trusters.

small fraction of the total community members. Therefore, very often there will be no link to a targeted user. Trust propagation methods can predict some of the missing links in the trust web to make it more dense.

4.2 Propagation

We apply four different methods for expansion of the two webs-of-trust (Retweet and Favorite) generated in the first step. Three of them are based on the *transitivity* feature of trust and the fourth is based on the idea of *similarity* between users based on the structure of the social network as a predictor of trust. By trust transitivity, we expect that people who a user trusts highly will tend to agree with the user more about the trustworthiness of others than people who are less trusted. For example, users are more likely to trust the "taste" of people they are following in Twitter. That is, if User A trusts User B who trusts User C, then User A will probably trust User C, as shown in Figure 1(a). On the other hand, this approach is helpful, provided that a complete transitive trust path exists between the truster and the trustee.

There are other possibilities for trust propagation such as "co-citation," "trust coupling," and "transpose/reciprocity" shown respectively in Figure 1(b, c, and d). Co-citation propagation refers to situation where User A and User B have a common trusted third User D. In this situation, User A has some level of trust towards User C. Likewise, User A and User C trust the same person(s) (see Figure 1(c)), they might trust one another as User A's trust in User B could propagate to User C and the same conclusion can be induced from User C's side as well. Another track for the propagation of trust is that a direct relation from User A to User C may hold (see Figure 1(d)). In other words, if User A trusts User C, User C may intuitively begin to develop a sense of trust in return. This is called transpose or reciprocated trust. Since transpose trust results from explicit trust, we did not consider it in this work.

We propose an alternative approach based on similarity in the network structure. One can exploit the like-mindedness of individuals based on collaborative filtering to infer trust to yet unknown parties. For instance, if one knows that, with respect to a specific property, two parties are trusted alike by a large number of different trusters, one can assume that they are similar, as shown in Figure 2(a). Likewise, if two persons trust a large number of overlapping users, they can be assumed to be similar, as shown in Figure 2(b).

More specifically, if User A has a level of trust towards User B (i.e., there is a direct trust link from User A to User B in the web-of-trust) who is similar to User C (i.e., they are *similar trustees*), then User A can infer some trust towards User C. Two trustees are similar if they are both similarly trusted by other Users Z_1, Z_2, ..., Z_n, as shown in Figure 2(a). The main advantage of this method is trust links can be predicted even between two users who are not reachable from each other on a transitive path. Another advantage is the other trust propagation possibilities, trust coupling and co-citations, can also be modeled through this approach. We provide the details of each method's formulation in the following sub-sections.

4.2.1 Trust Propagation through Transitivity

The simplest and most intuitive form of trust propagation is trust transitivity, which is widely discussed in the literature [5, 8, 12, 14]. That is, if User A trusts User B who trusts User C, then User A will also trusts User C, which we call it "simple-transitivity". This method has the time complexity of O(m*d) and the space complexity of O(m), where m is the number of links and d is the average degree of the nodes in the graph.

It is important to consider the number of users like User B that form a transitive path between User A and User C. The higher the number of these users, the stronger the predicted trust relationship between User A and User C. In an improved method called "weighted-transitivity", we assume User A trusts User C provided that there exists at least a given number of other users that connects them in a transitive path $A \rightarrow B \rightarrow C$. For instance, the average number of paths of length two between users in the graph can be used as the threshold. The weighted-transitivity method predicts a new trust link and assigns its weight as T_{AC} by multiplying the weights of the links in the transitive path (T_{AB} and T_{BC}), as shown in Figure 1. The intuition behind this multiplication is that we assume User B as a referral who sends her opinion of trust (T_{BC}) about User C to User A. It is important to include User A's trust in User B (T_{AB}) in the formulation; therefore, the recommended trust by User B (T_{BC}) is multiplied by Users B's trustability (T_{AB}). As a consequence, If User B is not much trusted by User A, then her recommended trust value will be de-emphasized as well.

$$T_{AC} = T_{AB} * T_{BC} \tag{1}$$

Another formula proposed by Golbeck [7] is:

$$T_{AC} = \frac{\sum T_{AB} * T_{BC}}{\sum T_{AB}}$$

We call this method "golbeck-transitivity". The main difference in her formula is that the trust value is not de-emphasized by the reliability of the referral. For example, the inferred trust value will be equal to T_{BC} in the case of one existing transitive path between User A and User C. Both weighted-transitivity and golbeck-transitivity methods have time complexity of O(m*d) and space complexity of O(m).

In this paper, we consider the paths of length two as the transitive paths. Theoretically, the transitive path between User A and User C can be of any length. However, previous work has addressed this issue and shown that, as expected, shorter paths lead to more accurate information [7]. In our formulation, the predicted value between two nodes over a very long path will be very small. Therefore, we considered

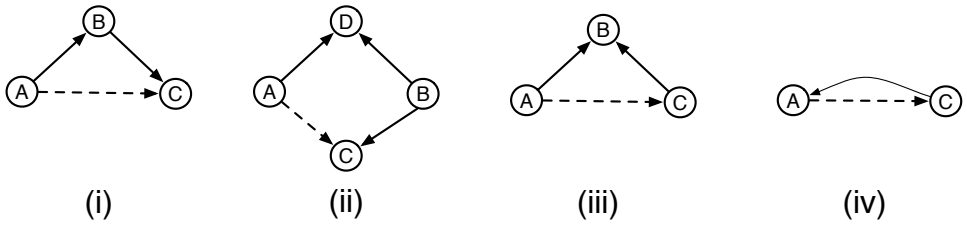

Figure 1: Trust propagation possibilities: (i) transitivity, (ii) co-citation (iii) trust coupling, (iv) transpose.

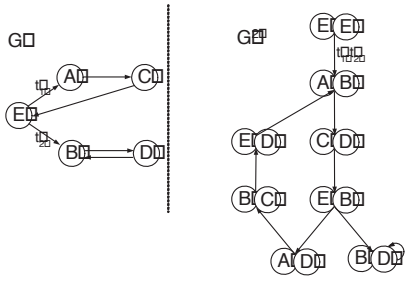

Figure 3: Similarity measurement.

only the paths of length two in our study. It is intuitive since a user expect that neighbors who are connected more closely will give more accurate information than those who are further away in the network.

4.2.2 Trust Propagation Through Structural Similarity

This method addresses the way in which the level of trust in cooperative relations depends on the similarity of nodes in the network structure. We measure similarity between two users with respect to trusting other users or being trusted by other users. The intuition behind our algorithm is that, *similar* users are related to *similar* users. More precisely, Users A and B are similar if they are related to Users C and D, respectively, and C and D are themselves similar. The base case is that each user is similar to itself. If we call the web-of-trust G, we can form a node-pair graph G^2 in which each node represents an ordered pair of nodes of G as depicted in Figure 3. A node (A, B) of G^2 points to a node (C, D) if, in G, A points to C and B points to D. Similarity scores are symmetric, so for clarity we draw (A, B) and (B, A) as a single node A, B (with the union of their associated links) [10].

SimRank is an iterative fixed-point algorithm that computes similarity scores for node-pairs in G^2 [10]. The similarity score for a node v of G^2 gives a measure of similarity between the two nodes of G represented by v. Scores can be thought of as flowing from a node to its neighbors. Each iteration propagates scores one step forward along the direction of the links until the system stabilizes (i.e., scores converge). Since nodes of G^2 represents pairs in G, similarity is propagated from pair to pair. Under this computation, two nodes are similar if they are linked by similar nodes.

For each iteration, k, an iterative similarity functions $sim_k(*,*)$ is introduced. The iterative computation is started with $sim_0(*,*)$ defined as

$$sim_0(A,B) = \begin{cases} 1, & if\ A = B \\ 0, & if\ A \neq B \end{cases} \quad (2)$$

On the $(k+1)$-th iteration, $sim_{k+1}(*,*)$ is defined in special cases as

$$\begin{array}{c} sim_{k+1}(A,B) = 1, \quad if\ A = B \\ sim_{k+1}(A,B) = 0, \quad if\ I(A) = \emptyset\ or\ I(B) = \emptyset \\ sim_{k+1}(A,B) = 0, \quad if\ O(A) = \emptyset\ or\ O(B) = \emptyset \end{array} \quad (3)$$

$I(A)$ is the set of in-neighbors of A while $O(A)$ specifies the set of A's out-neighbors. Individual in-neighbors are denoted as $I_i(A)$, for $1 \leq i \leq |I(A)|$, and individual out-neighbors are denoted as $O_i(A)$, for $1 \leq i \leq |O(A)|$. $sim_{k+1}(*,*)$ is computed from $sim_k(*,*)$, in the general case, as follows:

$$sim_{k+1}(A,B) = \frac{w}{|I(A)||I(B)|} \sum_{A' \in I(A)} \sum_{B' \in I(B)} sim_k(A', B') \quad (4)$$

where $I(X)$ denotes the set of nodes linking to X (in-neighbors); if $I(A)$ or $I(B)$ is empty, then $sim_{k+1}(A,B) = 0$ by definition. For a node pair with $A = B$ we simply let $sim_{k+1}(A,B) = 1$. w is a constant between in the range (0, 1) and can be thought of either as a confidence level or a decay factor.

Consider a simple scenario where User X has two relations with Users M and N, so we conclude some similarity between M and N. The similarity of X with itself is 1, but we do not want to conclude that $sim(M,N) = sim(X,X) = 1$. Rather, we let $sim(M,N) = w \times sim(X,X)$ meaning that we are less confident about the similarity between Users M and N than we are between User X and itself. This formula is alternately computed in iterations until the resulting similarity values converge. The structural similarity method has time complexity of $O(n^3)$ and space complexity of $O(n^2)$. We enhanced the algorithm to achieve the time complexity of $O(m * n)$ and space complexity of $O(n + m)$ by using the following heuristics [20].

I) The similarity score can be seen as a random walker defined on a node-pair graph G^2 depicted in Figure 3(b). The walker may wander into an enclosed sub-section of the entire graph that has no out-link so that it will get stuck in the small sub-graph with no possibility of returning to the outside. This scenario is associated with the fact that the graph is not strongly connected. A technique termed "teleportation" is used to make the graph irreducible and solve this problem.

II) We represent similarity equations in matrix form and employ a sparse storage scheme.

III) The similarity matrix often contains a very large fraction of non-zeros entries whose values are almost zero after several iterations. These small similarity values require a

Method	Time complexity	Space complexity
simple-transitivity	O(m*d)	O(m)
weighted-transitivity	O(m*d)	O(m)
golbeck-transitivity	O(m*d)	O(m)
structural-similarity	O(m*n)	O(n+m)

Table 1: Time and space complexities.

significant amount of storage space for little practical information. We devised a pruning technique to eliminate these near-zero similarity values by setting a threshold for each iteration. This operation will also decrease the redundant similarity computations and space-per-iteration.

IV) For the similarity computation to be I/O-efficient, the adjacency matrix needs to be pre-ordered, which requires offline pre-computation to minimize the bandwidth at query time. Therefore, A reordering technique is used, which not only speeds the convergence rate, but achieves I/O efficiency as well.

The details of these techniques are beyond the scope of this paper and are not presented because of space limitations. However, Yu et al. [20] present the details of implementation in their work. Although our proposal is based on the graph structure, the trust value between each two nodes can be calculated in a real time manner. In the next section, we describe our analysis and evaluation.

5. EVALUATION

Ideally, we would like to compare the Retweet and Favorite webs-of-trust with the real trust network on Twitter. This is not possible because the actual trust relationships are unknown. Therefore, we focus our evaluation on the second step using a crawled data set from Twitter with more than 20,000 users. The goal of the evaluation is to compare the *accuracy* and *efficiency* of the four propagation methods applied on the Retweet and Favorite graphs. Table 1 summarizes the efficiency of each method. In the table m is the number of links, n is the number of nodes, and d is the average degree of the nodes in the graph. The efficiency of all four algorithms are almost the same.

We compared the accuracy of each method by inductive reasoning. Our hypothesis is that any method found to correctly approximate the existing links and their weights in the Retweet and Favorite graphs with good accuracy will also approximate links over the whole network with enough accuracy. We used the leave-one-out technique, a machine learning evaluation technique, for the evaluation in this step. In the leave-one-out technique, we split the data set (i.e., the set of graph links) into m partitions of size 1, where m is the number of links in the graph. Each partition is used systematically to test exactly once where remaining partitions are used for training. Then, each propagation method is applied to predict the link in the the test partition. The main advantage of this technique is that the result does not depend on the way we split the data into a training set and a test set. The predicted link is compared with the actual link in the test partition using the following two metrics.

- **Coverage**: the percentage of currently available links that can be predicted using the propagation method.

- **Mean absolute error (MAE)**: The weight of the newly predicted link is compared against the original weight of the link. The average of the prediction error over all links is then calculated.

Algorithm 5.1: NODE SELECTION ALGORITHM(*seeds*)

1 Put seeds (starting nodes of the crawl) into a queue
2 Select a node from the queue
3 Crawl the node
4 Add the neighbors of the crawled node into the queue
5 Go to Step 2 or terminate if stop conditions are met

5.1 Data Gathering

As a basis for evaluating our proposal, we first needed data to evaluate. The main challenges in data gathering are the factors that may introduce biases towards high degree nodes, which may further contaminate or even skew the results. The main factors are: 1) choice of seeds as the starting point of the crawl, 2) node selection algorithm that decides which node to crawl in each step, and 3) size of the crawled subgraph, which is subject to real world resource constraints such as network bandwidth, time, machines, and the rate limits enforced by online social networks providers.

Node/link coverage (the number of nodes/links seen by the crawler versus the number of nodes/links in the graph) is not sensitive to the number of seeds or to the degree of seeds. The reason is that it has been widely documented that social networks have the properties of small world networks, where lots of nodes are tightly coupled together within a few hops of each other [18]. The small world effect on online social networks makes the choice of seeds less critical and crawling a small portion of the network is sufficient to reveal most nodes/links according to the work by Ye et al. [19]. It is a strong sign of the small world phenomenon. Kwak et al., in their thorough study on the Twitter social network, show that Twitter also has a short average path length, although, the relationships are directed among users [11]. It is also important to note that all links are included for the crawled portion of the network; thus, the result of study will not be affected by missing links.

The crawling of the social graph starts from an initial node and proceeds iteratively. In every operation, we visit a node and discover all its neighbors. The process for crawling a social graph and gathering a partial data set can be outlined as follows [19]: There are several widely used node selection algorithms [19], e.g., the *BFS (Breadth First Search)* algorithm, which simply selects the first item in the queue; the *Greedy* algorithm, which selects the node with the largest degree in the queue; or the *Random Walk* algorithm, which selects a node in the queue with probability proportional to its degree. Therefore, the probability of moving from a Node u to its neighbor (the transition probability) is $\frac{1}{u_{degree}}$.

However, these algorithms lead to samples that not only are biased towards high degree nodes, but also do not have provable statistical properties. We used an algorithm, the *Metropolis-Hastings Random Walk (MHRW)*, for our crawling that is claimed to be not biased towards the nodes with higher degrees [6]. This algorithm obtains a uniformly distributed random sample of nodes by appropriately modifying the probabilities of transition from the current crawled

Algorithm 5.2: NODE SELECTION ALGORITHM(*seeds*)

```
queue ← seeds
while stopping criterion not met
do { u ← queue.GET()
     CRAWL(u)
     while true
     do { Select node v uniformly at random from
          neighbors of u
          Generate uniformly at random 0 ≤ p ≤ 1
          if p ≤ u_degree / v_degree
          then { queue.ADD(v)
                 return
```

node to its neighbors. Pseudocode 5.2 shows the process. In every iteration of MHRW, at the current node, u, the algorithm randomly selects a neighbor, v, and moves there with probability $min(1, \frac{u_{degree}}{v_{degree}})$. It always accepts the move towards a node of smaller degree, and reject some of the moves towards higher degree nodes. As a result, the bias of RW towards high degree nodes is eliminated.

In the rest of this section, we give the details about our crawling procedure on Twitter. The Twitter API was used to gather the data. We selected the first twenty most active users among places where users have most tweeted [4] as the seeds.The seeds were gathered in GMT: +1, +9, -8, -7, -6, -5, and 0, corresponding to places where the tweet counts were greater than one million. The public time-line command, an API function provided by Twitter, was used to sample the most active users. Twitter continually posts a series of twenty most recent status updates. The status updates in the time-line data set are presumably a random snapshot of currently active users. Samples were made by retrieving the public time-line and extracting the set of users associated with the statuses in the time-line; then, details of these users were collected. Over a period of one month, we crawled Twitter information streams of more than 20,000 users. Together, there are 144,962 followers-followees relationships, 23,280 retweeted messages, and 50,713 favorited messages. The results of the evaluation are presented in the following.

5.2 Results

We applied the four different methods for propagation: simple-transitivity, weighted-transitivity, golbeck-transitivity, and structural-similarity on the Retweet and Favorite webs-of-trust. The results were compared according to the three different metrics: triadic closure, coverage, and MAE. We used the average number of paths of length two in each graph as the threshold value for the weighted-transitivity method. As shown in Figure 4, this is the best choice since the coverage decrease sharply when we increase the threshold value in Favorite web-of-trust. The Retweet web-of-trust showed the same pattern.

We start by discussion of the triadic closure for the Retweet and Favorite webs-of-trust. Triadic closure is the percentage of all the paths of length two in the graph that would eventually close because of transitivity. In other words, the third closing link exists in the graph. As Table 2 shows, the third closing link exists for approximately 12% of all paths of length two in both graphs. This metric shows that only 12% of the transitive paths resulted in a closing link in the current graph based on existing relationships. This result indicates that transitivity alone might not be strong enough to predict a trust relationship.

[4]http://www.socialnetworkingsandiego.com/social-networking/twitter-as-a-marketing-media/

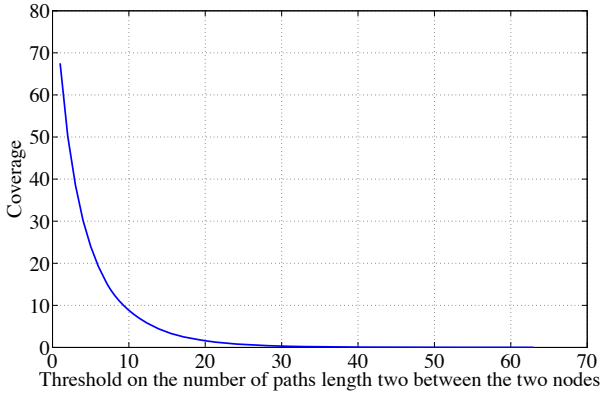

Figure 4: Coverage vs. threshold in Favorite web-of-trust.

	Retweet	Favorite
Triadic Closure (%)	12.26	12.16

Table 2: The number of triadic closures.

Table 3 shows the values for the first metric, coverage, which is the percentage of existing links in the Retweet and Favorite webs-of-trust that are predicted using each propagation method. As the results shows, the coverage for weighted-transitivity and golbeck-transitivity methods are less than the simple-transitivity method because of the constraints on the number of transitive paths between two connected nodes. The structural-similarity method should gives 100% coverage for each graph, as this method is able to measure similarity between each two nodes on the graph. However, we use several heuristics to increase the efficiency of this method. Consequently, some nonzero similarity measures, whose values are almost zero, are eliminated. Although, we can see in the results that the heuristics have not had a very adverse effect on the results since the coverage results are 99.9%.

Next, we describe the MAE results from Table 4. This metric is not applicable to the simple-transitivity method, as it does not calculate the weights on the predicted links. The weighted-transitivity method gives less error than the golbeck-transitivity method. The coverage is the same for both methods. This result shows that considering the reliability of the recommender (i.e., the trust of truster in the recommender) is an important issue. The structural-similarity method on the graph generated by users' retweeting behavior gives fewer errors in comparison with the weighted-transitivity and golbeck-transitivity methods. Moreover, the average error is less on the Retweet graph than on the Favorite graph.

Method/Metric	Coverage (%)	
	Retweet	Favorite
simple-transitivity	15.67	16.39
weighted-transitivity	4.86	5.53
golbeck-transitivity	4.86	5.53
structural-similarity	99.96	99.98

Table 3: The coverage of predicted links.

Method/Metric	MAE (%)	
	Retweet	Favorite
simple-transitivity	N/A	N/A
weighted-transitivity	14.87	15.82
golbeck-transitivity	22.04	23.43
structural-similarity	3.69	8.97

Table 4: The errors of predicted links.

Overall, the results show that the weighted-transitivity method does not give a better result than the simple-transitivity method on any of the graphs. The coverage is smaller and the error is larger. The golbeck-transitivity method also gave worse results with the same coverage and larger average error. However, the structural-similarity method gave the best results both in terms of coverage and average error. The average error for the Favorite graph is almost twice as large as the average error for the Retweet graph. As discussed earlier, this method has also a good time complexity of O(m*n) and a space complexity of O(n+m).

6. CONCLUSION AND FUTURE WORK

In this paper, we discussed how users' activities in online social networks can be used to discover a stronger and more influential network among users. Given a data set consisting of more than 20,000 user, we generated two different trust webs and applied four different prediction/propagation methods for expansion of these networks. We found that structural similarity is a better propagation method on a web-of-trust generated by a user's retweet behavior. In addition, this web-of-trust can be expanded efficiently to create new trust predictions using the graph's structural properties. The main advantages of our approach are accuracy, efficiency, scalability, and the use of implicit information.

This work is a key first step towards the creation of a web-of-trust that can be used by social recommender systems. As a future work, we aim to show how the incorporation of such information can enhance the performance of social recommender systems. Moreover, we may be able to improve the methods proposed in this paper with simple semantic analysis. Efficient algorithms for statistically analyzing the tweets along different dimensions can enhance our algorithms. Future work could also include further investigation of contextual information, such as the semantics of tweets (e.g., topics and hashtags) and demographic information about users (e.g., location and age) and their impact on inference of the web-of-trust.

7. REFERENCES

[1] S. Adali et al. Measuring behavioral trust in social networks. In *the IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 150–152, 2010.

[2] P. Anantharam, C.A. Henson, K. Thirunarayan, and A.P. Sheth. Trust model for semantic sensor and social networks: A preliminary report. In *the IEEE National Aerospace and Electronics Conference (NAECON)*, pages 1–5, 2010.

[3] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1–10, 2010.

[4] V. Buskens. The social structure of trust. *Social Networks*, 20(3):265–289, 1998.

[5] L. Ding, P. Kolari, S. Ganjugunte, T. Finin, and A. Joshi. Modeling and Evaluating Trust Network Inference. Technical report, Maryland Univ Baltimore, 2005.

[6] M. Gjoka, M. Kurant, C.T. Butts, and A. Markopoulou. Walking in facebook: A case study of unbiased sampling of OSNs. In *IEEE INFOCOM*, pages 1–9, 2010.

[7] J. Golbeck. Combining provenance with trust in social networks for semantic web content filtering. *Provenance and Annotation of Data*, pages 101–108, 2006.

[8] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *the International Conference on World Wide Web (WWW)*, pages 403–412, 2004.

[9] B.A. Huberman, D.M. Romero, and F. Wu. Social networks that matter: Twitter under the microscope. *First Monday*, 14(1):8, 2009.

[10] G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. In *the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 538–543, 2002.

[11] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *the International Conference on World Wide Web (WWW)*, pages 591–600, 2010.

[12] R. Morselli, B. Bhattacharjee, J. Katz, and M. Marsh. Exploiting approximate transitivity of trust. In *International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, pages 515–524, 2007.

[13] P. Noordhuis, M. Heijkoop, and A. Lazovik. Mining twitter in the cloud: A case study. In *the IEEE International Conference on Cloud Computing (CLOUD)*, pages 107–114, 2010.

[14] D. Quercia, S. Hailes, and L. Capra. Lightweight Distributed Trust Propagation. In *the IEEE International Conference on Data Mining (ICDM)*, pages 282–291, 2007.

[15] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. *The Semantic Web-ISWC*, pages 351–368, 2003.

[16] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA, 2010.

[17] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. Methods and metrics for cold-start recommendations. In *the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260, 2002.

[18] D.J. Watts. *Small worlds: the dynamics of networks between order and randomness*. Princeton Univ Pr, 2003.

[19] S. Ye, J. Lang, and F. Wu. Crawling online social graphs. In *the International Asia-Pacific Web Conference (APWEB)*, pages 236–242, 2010.

[20] W. Yu, X. Lin, and J. Le. A space and time efficient algorithm for simrank computation. In *the International Asia-Pacific Web Conference (APWEB)*, pages 164–170, 2010.

[21] C.N. Ziegler and G. Lausen. Propagation models for trust and distrust in social networks. *Information Systems Frontiers*, 7(4):337–358, 2005.