Understanding Graph-Based Trust Evaluation in Online Social Networks: Methodologies and Challenges

WENJUN JIANG, Hunan University
GUOJUN WANG, Guangzhou University
MD ZAKIRUL ALAM BHUIYAN and JIE WU, Temple University

Online Social Networks (OSNs) are becoming a popular method of meeting people and keeping in touch with friends. OSNs resort to trust evaluation models and algorithms to improve service quality and enhance user experiences. Much research has been done to evaluate trust and predict the trustworthiness of a target, usually from the view of a source. Graph-based approaches make up a major portion of the existing works, in which the trust value is calculated through a trusted graph (or trusted network, web of trust, or multiple trust chains). In this article, we focus on graph-based trust evaluation models in OSNs, particularly in the computer science literature. We first summarize the features of OSNs and the properties of trust. Then we comparatively review two categories of graph-simplification-based and graph-analogy-based approaches and discuss their individual problems and challenges. We also analyze the common challenges of all graph-based models. To provide an integrated view of trust evaluation, we conduct a brief review of its pre- and postprocesses (i.e., the preparation and validation of trust models, including information collection, performance evaluation, and related applications). Finally, we identify some open challenges that all trust models are facing.

Categories and Subject Descriptors: A.1 [**Introductory and Survey**]; C.2.4 [**Computer-Communication Networks**]: Distributed Systems

General Terms: Design, Reliability, Management

Additional Key Words and Phrases: Trusted graph, trust evaluation, simplification, analogy, online social networks (OSNs), trust models

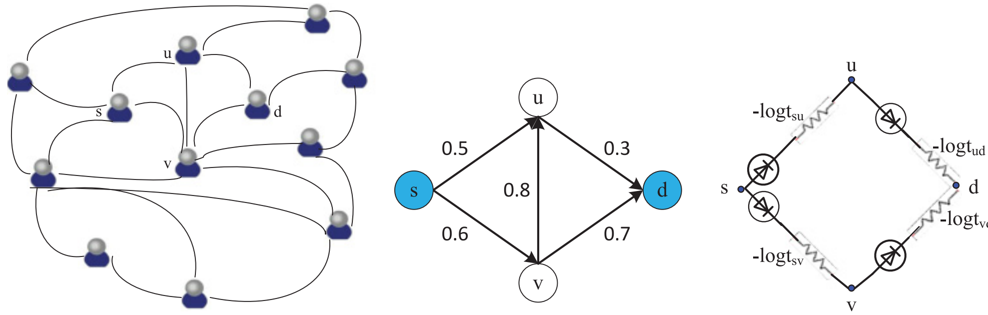
ACM Reference Format:
Wenjun Jiang, Guojun Wang, Md Zakirul Alam Bhuiyan, and Jie Wu. 2016. Understanding graph-based trust evaluation in online social networks: Methodologies and challenges. ACM Comput. Surv. 49, 1, Article 10 (May 2016), 35 pages.
DOI: http://dx.doi.org/10.1145/2906151

1. INTRODUCTION

Online Social Networks (OSNs) are popular tools for users to find new friends who share similar interests, maintain social relationships, and locate various User-Generated Content (UGC) [Mislove et al. 2007; Quan et al. 2011]. Today, increasing numbers of

This work is supported by NSFC grants 61502161, 61472451 and 61272151, the Chinese Fundamental Research Funds for the Central Universities 531107040845, ISTCP grant 2013DFB10070. This work is also supported in part by NSF grants ECCS 1231461, ECCS 1128209, and CNS 1138963.
Authors' addresses: W. Jiang, College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, P. R. China; email: jiangwenjun@hnu.edu.cn; G. Wang (Corresponding author), School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China; email: csgjwang@gmail.com; J. Wu and M. Z. A. Bhuiyan, Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122; email: jiewu@temple.edu, zakirulalam@gmail.com.

DOI: http://dx.doi.org/10.1145/2906151



(a) A social graph (b) A trusted graph (c) Resistive network

Fig. 1. An example of (a) a social graph, (b) a trusted graph, and (c) a resistive network.

people join OSNs for daily communications or even business activities. Many of those activities involve the process of forming opinions of trust about a particular user or product. Interactions among people through OSNs are rather complex because they involve interactions with others who may be strangers. The notion of trust is a central issue in OSNs. "To be trusting is to be fooled from time to time; to be suspicious is to live in constant torment" [Wu 2009]. People face trust issues in their daily lives when making decisions, and this become more serious in OSNs due to the lack of real-life interactions and mutual understanding. Without trust, our everyday social lives, which we take for granted, would not be possible [Good 1988]. Quantifying trust in OSNs is a notoriously difficult problem because of the complexity of OSNs and of trust itself. Studies in trust span multiple disciplines including economics [Huang 2007], sociology [Möllering 2001], political science [Newton 2001], psychology [Julian 1967; Cook et al. 2005], and computer science [Marsh 1994]. Meanwhile, researches in OSNs have been conducted from multiple aspects, including network structure [Yuan et al. 2010; Watts 1999], user behaviors [Zhu et al. 2012; Bakshy et al. 2012; Crandall et al. 2008], community detection [Ciglan et al. 2013; Sathik et al. 2011; Qi et al. 2012], and so on. Furthermore, the high popularity of OSNs has led to a new computing paradigm: social computing.

The trust mechanism (or trust evaluation) is a tool used to facilitate decision-making in diverse applications. Trust and trust-related issues have attracted significant attention in various networking environments, including OSNs [Sherchan et al. 2013], wireless communication networks [Yu et al. 2010; Jiang and Wu 2014], multiagent systems [Pinyol and Sabater-Mir 2013], and P2P networks [Kamvar et al. 2003; Singh and Liu 2003; Marti and Garcia-Molina 2006; Kamvar et al. 2003]. As a consequence, many trust models have been proposed (and are being reviewed, as in Jøsang et al. [2007], Sherchan et al. [2013], Pinyol and Sabater-Mir [2013], and Grandison and Sloman [2000]). Among these, the graph-based model is one of the most important branches. In this article, we particularly focus on the current progress and challenges of graph-based trust models in OSNs. We also give a brief introduction of the pre- and postprocesses of trust models. An OSN is usually represented by a *social graph* in which a node represents a user and an edge represents the connection or relationship between two users. Figure 1(a) shows an example of a social graph.

Concepts. Generally speaking, trust is "a measure of confidence that an entity or entities will behave in an expected manner" [Sherchan et al. 2013]. Both the source and target can be either a party or an entity. Marsh's work [Marsh 1994] is one of the first seeking to formalize trust in a computational model. However, there is no consensus on how trust should be defined. Therefore, *trust* itself has many definitions and categories

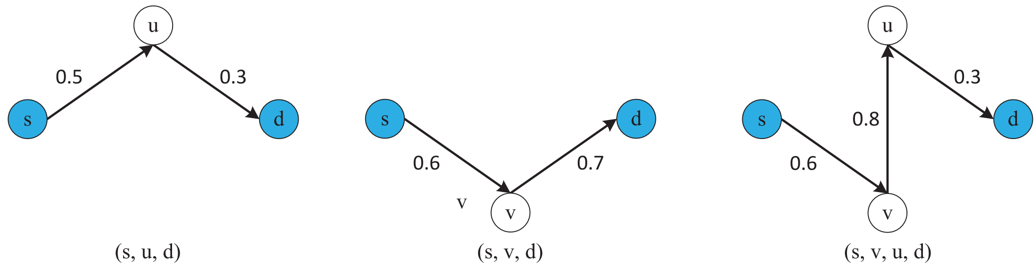


Fig. 2. The trusted paths of trusted graph in Figure 1(b), which can be used for simplification.

[Jøsang et al. 2007; Sherchan et al. 2013]. Under the computational model, Jøsang et al. [2007] define trust as "the subjective probability by which one user expects that another user performs a given action." In Golbeck [2005], trust in a person is defined as "a commitment to an action, based on a belief that the future actions of that person will lead to a good outcome." Trust has also been defined as "the extent to which one party is willing to participate in a given action with a given partner, considering the risks and incentives involved" [Ruohomaa and Kutvonen 2005]. Grandison and Sloman [2000] define trust as "the firm belief in the competence of an entity to act dependably, securely and reliably within a specified context." Although these definitions vary from each other, they all indicate a common nature of trust (which we discuss in detail later). Therefore, they are suitable to most of the existing graph-based models in OSNs.

In Figure 1(b), u and v have known d earlier and have an opinion of the trustworthiness of d; how, then, does s form her or his own opinion on d via u and v? This depends on the influence/recommendation among users and the personalities of users. A *trusted path* can be constructed through iterative recommendations; for example, path (s, u, d) representing s's trust of d via u's recommendation. Multiple parallel and sequential paths are overlapped to form a *trusted graph* from s to d. In this article, we call the trust models that work with a trusted graph *graph-based models*.

Trust built through direct contact is called *first-hand trust*, such as a direct link from s to u; that through a recommendation is called *second-hand trust*, such as the trust from s to d via a trusted path (s, u, d). In Figure 1(b), (s, u) and (u, d) are two edges of a sequential path (s, u, d); (s, u, d) and (s, v, d) are two parallel paths; (s, u, v, d) is overlapping with (s, u, d) and (s, v, d) (Figure 2). Usually, each edge has a weight value between 0 (no trust) and 1 (full trust) to quantify each direct trust. Different models may design different ranges or even different dimensions for trust values (e.g., some models consider both trust level and confidence [Wang and Wu 2011a, 2011b]). In addition, according to their role in trust evaluation, a user can be a source (or trustor, e.g., s in Figure 1(b)), a target (or trustee, e.g., d in Figure 1(b)), or a recommender (an intermediate user in a trusted path, e.g., u and v in Figure 1(b)).

Categories of Trust Models. From the network perspective, trust models can be divided into two types: those using a local approach that considers personal bias, and those using a global approach that considers all users' opinions [Ziegler and Lausen 2005]. Graph-based models usually take the local approach. They can be scalar metrics, which cope with the setting where a source s is interested in a single target d; d can be another user or a service provider. Some users have prior opinions about d. s wants to estimate the trustworthiness of d based on the aggregated opinions of other users whom he knows. In this setting, trust models analyze each user's trust opinions independently. Alternatively, in some other models, the trust of a group of nodes can be calculated at once. This type of trust model is called the *group trust metric* [Levien 2003].

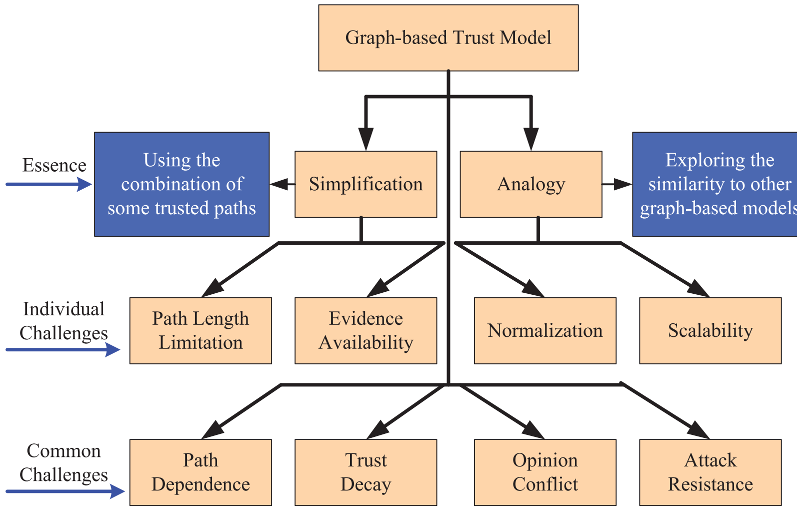


Fig. 3. The categories, essences, and challenges of graph-based trust models in OSNs: a taxonomy.

Based on how to cope with the trusted graph, graph-based trust models can be classified into two categories [Wang and Wu 2011a]:

—The *graph simplification-based* approach. As its name implies, this approach simplifies a trusted graph into multiple paths whose nodes or edges are disjoint with each other. It may also simplify a trusted graph into a directed series-parallel graph [Wang and Wu 2011a], which is an important concept in graph theory. For instance, given the trusted graph in Figure 1(b), this approach only uses one or several trusted paths in Figure 2 for trust evaluation. The essence of this approach is to use the combination of some trusted paths according to some predefined principles.
—The *graph analogy-based* approach. Different from the preceding approach, this approach does not remove any nodes or edges from trusted graphs. Instead of simplification, it emulates the trusted graph by using other graphs. An example analogy is shown in Figure 1(c), which uses the resistive network to emulate a trusted graph [Taherian et al. 2008]. The essence of this approach is to explore the similarity between graph-based trust models in OSNs and other graph-based models in other networking environments, such as network structure, diffusion pattern, and the like.

We use this classification to compare existing models in the following sections. Figure 3 illustrates the overview of this survey in which we identify the individual and common challenges of graph-based trust models. Specifically, graph simplification-based models face the challenges of *setting proper path length limitations* and *keeping evidence availability*. Graph analogy-based models face the challenges of *normalization* and *scalability*. Moreover, all graph-based models face four common challenges: *path dependence*, *trust decay*, *opinion conflict*, and *attack resistance*. Table I outlines the comparison of these representative models in detail, whereas Table II lists how they tackle these challenges.

In this article, we provide a comprehensive review of the methods and challenges of graph-based trust evaluation in OSNs. Our contributions are fourfold:

—We conduct a comparative study of graph-based trust models in OSNs. We survey the main techniques they use and the challenges they meet, as well as the most relevant areas. To be specific, we differentiate trust models in OSNs and classify trust evaluation techniques into two categories—graph simplification-based and graph analogy-based approaches—based on how they treat a trusted graph. Then, we compare representative trust models in each category and identify their individual challenges. Finally, we point out four common challenges.

—To better understand trust models in OSNs, we analyze the features of OSNs and properties of trust in OSNs, respectively. We also analyze two other closely related concepts of trust: recommendation and influence. In addition to graph-based trust models, we also provide an overview of trust models in literature and classify them into four categories in terms of methodologies.

—We discuss the pre- and postprocesses of trust models, including information collection, performance evaluation, and related applications. Finally, we identify three open challenges that all trust models meet.

—We conduct empirical and theoretical analysis on representative trust models. We also discuss each model's strengths and weaknesses.

The remainder of this article is organized as follows: Section 2 briefly summarizes the construction of OSNs and the properties and related concepts of trust. Section 3 reviews some existing works. Sections 4 and 5 survey the graph simplification- and graph analogy-based models and discuss their challenges. Section 6 states the common challenges that all graph-based models meet. Section 7 surveys the pre- and postprocesses of a trust model. Section 8 concludes this article and proposes some open challenges that may draw more attention in the future.

2. OVERVIEW OF OSNS AND TRUST

In this section, we introduce the background of OSNs and trust in terms of OSN construction, related concepts, and fundamental properties to better understand trust issues and trust evaluation in OSNs.

2.1. Construction of OSNs and Possible Trust Issues

OSNs are organized around users, and they are usually taken to represent a mapping from our real life to the cyber physical space. In general, OSNs have three main types of entities: users, their connections, and the information that users are generating and diffusing. Each entity has its own characteristics.

As the first kind of entity, online users can build connections with each other and can generate their own content, which leads to the emergence of the other two kinds of entities. For the second kind of entity, and similar to one's daily social life, connections among online users are usually topic-dependent and time-sensitive. This leads to the rich variety and dynamic evolution of OSNs. Moreover, the changing speed of OSNs is much faster than that of an offline social network. The third kind of entity (i.e., the information diffused among online users via their connections) is usually large in quantity and may cross time and space. In addition, information diffusion can be deemed as the central function of OSNs.

These features of users, their connections, and the information they diffuse lead to all kinds of trust issues in OSNs, such as "Can I trust the interaction partner who is a stranger to me?" "Is the information source trustful?" or "Can the service quality be guaranteed?" Those issues lead to a strong necessity for trust evaluation.

Generally, the scale of an OSN can be very large: Facebook has billions of users; more than 200 million users are using Tencent QQ at the same time; and Twitter has more than 100 million users. Figure 4 (by statista report[1]) shows the statistics of active user numbers in leading social networks. Due to its large scale, the efficiency

[1]http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users.

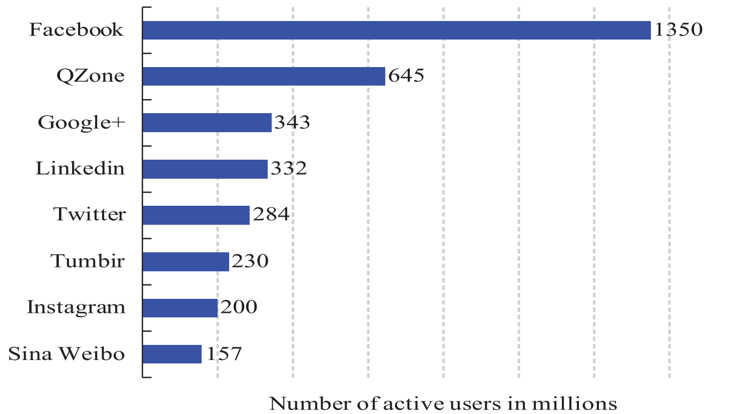


Fig. 4. Active users in leading social networks in November 2014.

and scalability of trust models in OSNs face challenges. Therefore, usually a small trusted graph is used as the basis for trust models. SWTrust in Jiang et al. [2014] is particularly proposed for trusted graph generation.

As pointed out in Sherchan et al. [2013], network structures can affect the evidence for trust and the resulting degree of trust. From a network structure view, OSNs appear to bear the characteristics of small-world networks [Watts 1999; Yuan et al. 2010]: higher clustering and shorter distances between any two nodes. Based on this, searching proper trust evidence in large OSNs can be complex.

2.2. Concepts and Categories of Trust

We have mentioned some key concepts in the introduction section. Here, we provide more formal definitions to better describe trust and trust evaluation.

First, we provide the definition of trust used in this survey. As mentioned, several definitions of trust have been presented, and they are suitable to graph-based trust models in OSNs. Hence, rather than present a new definition, we use the one from Jøsang et al. [2007].

Definition 2.1. *Trust*. "Trust is the subjective probability by which one user expects that another user performs a given action" [Jøsang et al. 2007].

Next, we describe formal definitions of *trustor*, *trustee*, *recommender*, *trusted path*, and *trusted graph*, which are commonly used concepts in graph-based trust models.

Definition 2.2. *Trustor*. A user who is trying to know the trustworthiness/trust degree of another user is the trustor.

Definition 2.3. *Trustee*. A user whose trustworthiness/trust degree is being evaluated is the trustee.

Definition 2.4. *Recommender*. An intermediate user who helps the trustor to evaluate the trustworthiness/trust degree of the trustee is the recommender.

Definition 2.5. *Trusted Path*. A path that consists of a trustor (the source), several recommenders, a trustee (the target), and trust relations among them is a trusted path from the trustor to the trustee.

Definition 2.6. *Trusted Graph*. All the trusted paths starting from a trustor and ending with a trustee form a trusted graph from the trustor to the trustee.

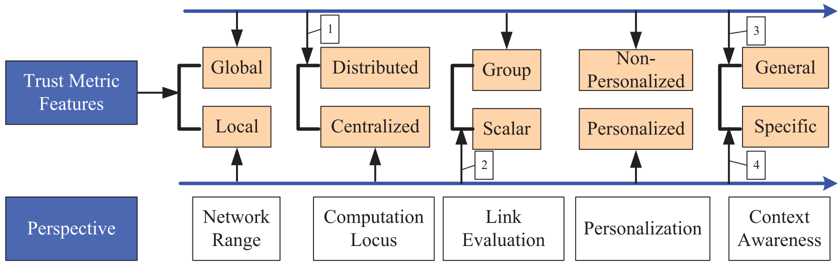


Fig. 5. The classification of trust (labels 1–4 indicate both categories are suitable).

We differentiate the trusts used in a trusted path into two types, as in Jøsang et al. [2006b]:

Definition 2.7. *Referral Trust*. Trust in the ability (of a recommender) to **recommend** a good service provider (i.e., trustee) represents referral trust.

Definition 2.8. *Functional Trust*. Trust in actually being a good service provider (i.e., trustee) represents functional trust.

From these two definitions, we can see that referral trust usually originates from a source (i.e., trustor) to a recommender, or from a recommender to another recommender; meanwhile, functional trust usually starts from a directly connected neighbor to the target (i.e., trustee).

Based on some prior work [Guha 2003; Levien 2003; Ziegler and Lausen 2005], and [Jøsang et al. 2007], we provide a classification of trust metric features as shown in Figure 5. It considers multiple aspects including network range (global or local), computation locus (distributed or centralized), link evaluation (group or scalar), personalization (personalized or not), and context awareness (general or specific). Next, we provide several new concepts based on this classification.

Definition 2.9. *Global Trust*. Metrics for measuring global trust consider the opinions of all users and all trust relations among them: It is calculated based on complete trust information in the network.

Definition 2.10. *Local Trust*. Metrics for measuring local trust consider the opinions of partial users, usually from the neighborhood of the trustor.

Definition 2.11. *Group Trust Metric*. Metrics for calculating the trustworthiness of a group of users simultaneously.

Definition 2.12. *Scalar Trust Metric*. Metrics for calculating the trustworthiness of each user independently.

Definition 2.13. *Personal Trust*. The trustworthiness of a user from the view of a particular user (e.g., the trustor) is personal trust.

Definition 2.14. *Specific Trust*. The trustworthiness of a user on some specific topic or topics is specific trust.

Definition 2.15. *General Trust*. The trustworthiness of a user without specifying any topic or topics is general trust.

In Figure 5, the arrows connecting multiple features represent their correlations. To be specific, for global trust, the computation can be done either in a distributed or centralized way, it evaluates groups of trust assertions at once, it is usually

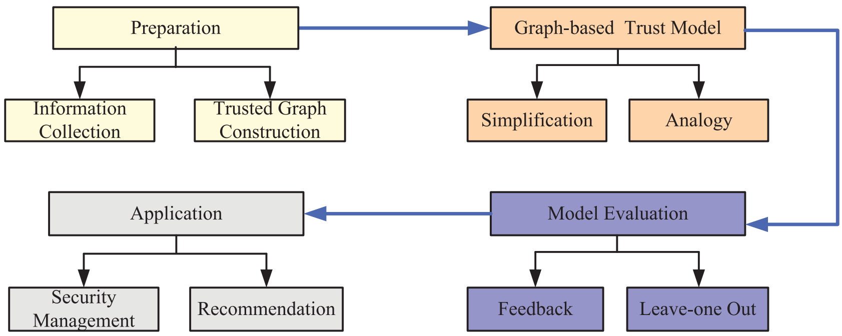


Fig. 6. The integrated process of trust evaluation in OSNs.

non-personalized, and it can be either general or follow some specific topics. Meanwhile, local trust is usually calculated in a centralized way, it evaluates each trust assertion independently, it is usually personalized, and it can be either general or follow some specific topics.

Generally speaking, global trust is used to represent overall reputation, whereas local trust indicates personal opinion. Although they use different network ranges for trust evaluation, both global and local trust can be predicted using graphs. Figure 6 shows the integrated process of graph-based trust evaluation in OSNs (it also applies in other networking environments). There are four basic steps: (i) Collecting information for trust, that is, evidence collection for evaluating a user's trust degree; (ii) constructing atrusted graph, that is, managing trust evidence using a graph; (iii) conducting trust evaluation, that is, designing algorithms to calculate trust degree; and (iv) applying the results to other applications (e.g., security management, recommendation system).

In this survey, the main scenario we consider is "a source s wants to know the trustworthiness of a target d." To this end, a trust model is trying to collect trust evidence and calculate the trust degree of d, from the view of s. The small-world characteristic of OSNs makes it possible to collect evidence for trust evaluation. Most of the representative models we review in this survey fall into the range of scalar/local/personal trust models, including TidaTrust [Golbeck 2005], MoleTrust [Massa et al. 2005], MeTrust [Wang and Wu 2011a], SWTrust [Jiang et al. 2014], RN-Trust [Taherian et al. 2008], FlowTrust [Wang and Wu 2011a], and GFTrust [Jiang et al. 2016]. A few of them are group-trust models, including Appleseed [Ziegler and Lausen 2005] and Advogato [Levien and Aiken 1998].

2.3. The "START" (Properties) of Trust in OSNs

Trust in OSNs has several common properties [Jøsang et al. 2007; Sherchan et al. 2013]. Here, we briefly introduce five typical and fundamental ones, which we call the "**START**" of trust: **s**ubjective, **t**opic-dependent, **a**symmetric, **r**isking betrayal, and **t**ime-sensitive. Note that the mentioned properties in this subsection are extracted from trust itself. In the next subsection, we describe some typical properties of trust from a computational view.

Trust is subjective since "trust in a person" is a kind of opinion from a specific user. In fact, the notion of trust has been widely studied in the fields of psychology and sociology [Marsh 1994]. This property indicates that trust is generally personalized: Different people may have different trust opinions about the same person. This subjective and personalized nature further leads to the difficulty of measuring it. In particular, how can we collect evidence to estimate people's trust in others, given that people usually do not express their trust opinions frankly due to concerns like worsening a relation with someone. It is even more challenging to consider how we can integrate incomplete and inaccurate evidence to make an accurate prediction.

Trust is topic-dependent because humans have different expertise in different fields. For instance, an expert A in computer science may not be good at music. Therefore, we may trust A when considering problems in computer science but not in those of music. The topic-dependence property is consistent with that of social connections since trust can be deemed as a special connection.

Trust is asymmetric, which indicates that the degree to which A trusts B is usually not equal to that of B trusting A. To some degree, we can even say that the asymmetric property of trust is caused by its subjectivity.

Trust risks betrayal, indicating that when we want to place trust in someone, usually there is a risk (some probability) that things may not happen as we expected. And if that happens, it may lead to some cost or loss to us, and we may never trust that person again. Therefore, trust is "hard to gain, easy to lose." According to the interaction experience, people will refine their initial trust opinions of the target and even those of the recommenders. This is exactly how trust evolves.

Similar to the topic-dependence property, trust is time-sensitive because trust is a type of human relation. As time passes, some old relations may weaken, while some new relations can be built. In addition, people's opinions may vary with time, which also leads to changes in trust.

2.4. The Properties of Computation Trust in OSNs

We described five typical properties of trust in the preceding subsection, which can be taken as the nature or feature of trust. In this subsection, we introduce two typical properties of computation trust in OSNs: *propagative* and *composable*, which can be deemed as the basis or the foundation of trust evaluation.

The propagative property of trust indicates that trust relations can be diffused along a chain. For example, s trusts u, and u trusts d; but s does not know d. However, s can derive some trust on d based on the degree s trusts u and u trusts d. This property is exactly why trust models can search iterative recommendation chains for trust evaluation. However, this property does not mirror "the strict transitive relation" in mathematics in which s trusts u and u trusts d will lead to the conclusion that s trusts d. Therefore, some work identifies the propagative property of trust as "weak-transitive" (e.g., Golbeck [2005] and Jøsang et al. [2006a]), whereas others identify "nontransitive" as one property of trust (e.g., Sherchan et al. [2013]).

The composable property of trust indicates that the trust values induced from multiple trusted paths can be integrated as one value. To some degree, the composable property is a special kind of propagative property. For instance, Sun et al. [2006a] use concatenation propagation and multipath propagation to differentiate them. Composing several trusted paths is quite difficult, especially when different paths provide contradictory information [Sherchan et al. 2013]. Using the composability property, Richardson et al. [2003] propose a composition function to combine trust evidence. Jøsang et al. [2006b], Jøsang [1999], and Jøsang et al. [2006a] integrate trust values using subjective logic. More recently, Liu et al. [2014] propose a trust model using three-valued subjective logic.

2.5. Closely Related Concepts

Before introducing graph-based trust models, we analyze two concepts that are closely related to trust: (social) influence and recommendation. They can help us understand how interactions happen and how trust propagates. Yedidia et al. [2001] provide a strict analysis of generalized belief propagation. Note that reputation, which is often taken as the average of global trust, is another closely related concept. Here, we do not introduce it since there are many other works on this topic (e.g., Houser and Wooders [2006], Sabater and Sierra [2002], Jøsang et al. [2007], Srinivasan et al. [2008], Mármol and Pérez [2010], and Noorian and Ulieru [2010]).

As we know, trust is the basis of almost all online interactions. Without trust, people cannot cooperate well with others, and the online system cannot run smoothly.

Social influence can be taken as the tool that makes user interactions happen. Rashotte [2007] defines it as "a change in an individual's thoughts, feelings, attitudes, or behaviors that result from interactions with another individual or a group." Suppose s wants to know the trustworthiness of d and s asks u for advice. If u suggests that s does something, and s decides to do it, we can say that s is influenced by u. Jiang et al. [2014] study the construction of a user's social influence from two aspects of the user—himself and his connections with friends—which can help lead to a better understanding of social influence.

Recommendation is the method that allows opinions to propagate. During the process of trust evaluation from s to d in Figure 1(b), the trust from s to u and from s to v can be taken as referral trust [Jøsang et al. 2006a], which is trust with respect to the ability to recommend a good target. u and v can be taken as recommenders. The final trust that s puts in d, through the referrals of u and v, is called recommendation trust [Wang and Wu 2011a], which is trust derived from others' experiences.

In summary, we can say that trust, influence, and recommendation are three key driving powers that make an OSN run. This also applies to real-world social networks.

3. RELATED WORK

We first review some existing surveys on trust and trust models. Then, we review works on trust evaluation in four categories in terms of their methodologies. Finally, we discuss some related works in the system and application communities.

3.1. Existing Surveys on Trust

Due to its importance in multiple aspects, trust has been widely studied and reviewed. Jøsang et al. [2007] propose a widely cited survey on the notions, categories, and applications of trust and reputation systems, particularly for online service provision. Sherchan et al. [2013] present an important review of trust in social networks in which they comprehensively examine trust definitions and measurements from multiple fields including sociology, psychology, and computer science. Noor et al. [2013] present a survey of trust models in cloud environments. Most recently, Cho et al. [2015] propose a comprehensive survey aiming to outline the foundations of trust models. Yu et al. [2012] study the effects of trust scheme in wireless sensor networks, mainly in terms of attack resistance, in which they categorize attack types and countermeasures. Cho et al. [2011] conduct a survey on trust schemes in mobile ad hoc networks (MANETs) in which they discuss multiple aspects including classifications and potential attacks. Marmol and Perez [2010] summarize common properties of trust/reputation models in distributed systems. Ruohomaa and Kutvonen [2005] provide an overview of the literature. They also list some example metrics in two phases of modeling trust and building a specific solution. Grandison and Sloman [2000] examine different trust definitions and provide one for trust in Internet applications.

There are more works on reputation, a concept that is closely related to trust. Jøsang et al. [2007], Marti and Garcia-Molina [2006], Sabater and Sierra [2005], Noorian and Ulieru [2010], and Yao et al. [2012a] provide a comprehensive review of a reputation system (a hybrid system of trust and reputation). For instance, Hoffman et al. [2009] focus on the characterization of reputation systems, particularly from the perspective of computer science.

3.2. Trust Models in Methodologies

From the view of sociology, the trust relationship is one of the most complex social relation. A reason for this lies in its dynamic properties (as mentioned in Section 2). Another reason is that a trust relation involves multiple factors: assumptions, expectations, behaviors, and environments. Therefore, it is very challenging to measure and predict trust. As a new method in the field of network and information security, trust evaluation has been studied in P2P networks, mobile ad hoc networks, multiagent systems, and semantic webs. The core of trust evaluation is the expression and measurement of trust relations and the construction of a trust evaluation scheme. Currently, researches on trust evaluation are conducted using tools from mathematics, statistics, or artificial intelligence. Specifically, many trust models are proposed based on fuzzy theory, subjective logic, machine learning, information entropy, game theory, graph theory, and so on. For instance, among existing path-based propagation methods are the Dempster-Shafer (D-S) combination rule [Jøsang and Pope 2012; Shafer 1976] from statistics, serial-parallel merge [Jøsang et al. 2006b] using subjective logic, triangular norms [Wang and Wu 2011a] as logical conjunctions from classic logic, and path concatenation [Richardson et al. 2003] from path algebra. In the following paragraphs, we review existing work briefly in four categories in terms of methodology:

—**D-S evidence theory and subjective logic-based approaches.** To deal with the subjective property of trust, this approach introduces the process of inferring uncertainty. The D-S theory [Dempster 1967; Shafer 1976] is a general framework for reasoning with uncertainty. It can combine evidence from different sources and get a final degree of belief (using the belief function). Jøsang et al. [2006b, 2008] improve the D-S evidence theory and propose a trust model with subjective logic (TNA-SL). As pointed out by the authors, in this model, "the confidence of a trust value is equivalent to the certainty of the corresponding opinion." Liu et al. [2014] propose using three-valued subjective logic for trust evaluation. They differentiate posteriori and priori uncertainty spaces to better describe and manage trust evidence. The former is introduced to store the evidence distorted from certain spaces as trust is propagated, and the latter is used to control the evidence size as trusts are combined.

—**Approaches using traditional mathematics tools (probability statistics, fuzzy logic, etc.).** This approach tries to "provide a sound mathematical model for trust evaluation" [Sherchan et al. 2013]. Probability models represent trust values as a probability and evaluate trust using probability functions (e.g., Sun et al. [2006b]). Taking a similar idea with PageRank [Page et al. 1999], each node in EigenTrust [Kamvar et al. 2003] is assigned a global trust value. Kuter and Golbeck [2007] propose an algorithm, SUNNY, to estimate trust by probabilistic confidence models. Commonly used probability tools in trust evaluation include the Beta model [Li and Wu 2010], the Bayesian model [Nielsen et al. 2007], and the hidden Markov model (HMM) [ElSalamouny et al. 2009, 2010]. Some trust models use Bayesian inference [Lee 2012], where probability can be updated or inferred with observations. Pearl [1999] presents a more general belief propagation algorithm for Bayesian networks to solve inference problems. A comprehensive survey of inference problems has been reported in Yedidia et al. [2002], which involves many fields such as statistical physics, computer vision, and artificial intelligence (AI). Another commonly used tool is fuzzy logic. It provides reasoning rules to deal with fuzzy metrics. Trust itself is fuzzy in some degree due to the involved uncertainties. Fuzzy logic is able to handle uncertainty and imprecision effectively and therefore seems ideally suited to reasoning about trust. REGRET in Sabater and Sierra [2002] and k-FuzzyTrust in Chen et al. [2014] fall into this category. In addition, Xia et al. [2011] build a subjective trust management model, AFStrust, which considers multiple factors including direct trust, recommendation trust, incentive function, and active degree and treats those factors based on Analytic Hierarchy Process (AHP) theory and fuzzy logic rules. Lin et al. [2009b] propose a method of hierarchical fuzzy trust evaluation for P2P network. However, fuzzy logic systems meet two main issues: how to design proper reasoning rules, and how to reduce the number of involved rules and decrease their computational requirements.

—**AI and information theory-based approaches.** Works in this category usually model the trust evaluation task as a learning problem or a decision support system. They seek to apply machine learning methods to overcome the generalizability issues in Bayesian trust models. Liu et al. [2014] describe how to build robust reputation systems using machine learning techniques and define a framework for translating a trust modeling problem into a learning problem. Peng et al. [2010] try to improve the security routing in MANETs by employing a dynamic trust mechanism, which considers multiple constraints and applies the idea of collaborative filtering. Huynh [2009] provides a mechanism to capture the trust evaluation process of users, which can be replicated by computers. A user can specify two things: (i) Given the information about the target, how he selects a trust model; and (ii) how he configures the model. Liu et al. [2013a] propose a machine learning-based trust framework especially for large-scale open systems. However, the above-mentioned approaches are often model-centric. That is, they put more focus on the model itself, rather than on the data. They also overlook the importance of system adaptability, which is essential for service selection [Hauke et al. 2013]. This leads to the risk of unrealistic model assumptions. Hauke et al. [2013] point out several requirements for probabilistic trust models to improve their robustness using supervised learning. They also explore a real-world dataset to validate the effectiveness of supervised methods.

—**Graph-based approaches.** The main focus of this article is on graph-based trust models. Following Wang and Wu [2011a], we broadly classify them into two categories: simplification- or analogy-based. Sun et al. [2006b] give a theoretic framework on trust propagation by stating two axioms as possible guiding principles: "concatenation propagation of trust does not increase trust," and "multi-path propagation of trust does not reduce trust." Jøsang et al. [2006b, 2008] and Zuo et al. [2009] propose the methods of simplification. The basic idea is to reduce the trusted graph into serial/parallel trusted paths or node/edge disjoint multiple paths [Golbeck and Hendler 2006; Mui et al. 2002; Sun et al. 2006b]. More works using the simplification approach include TidalTrust [Golbeck 2005], MoleTrust [Massa et al. 2005], MeTrust [Wang and Wu 2011b], and SWTrust [Jiang et al. 2014]. TidalTrust, MoleTrust, and SWTrut are based on breadth-first search. TidalTrust selects the strongest shortest path, whereas MoleTrust uses hop count to control the length of the selected paths, and SWTrust limits the width of each hop. These approaches, however, may suffer from the information-loss problem. Some more interesting approaches are graph analogy-based. In Mahoney et al. [2005], a generalized reliability theory is applied to a trusted network with failure-prone elements. In RNTrust [Taherian et al. 2008], a trusted graph is transformed into a resistive network. Mislove et al. [2008] propose the Ostra scheme to bound the total amount of unwanted communication a user can produce. The intuition is that it is difficult for a user to create an arbitrarily large number of trust relationships. Tran et al. [2009] further propose SumUp to detect sybils using the technique of adaptive vote flow aggregation. FlowTrust in Wang and Wu [2011a] relates the amount of flow to trust, considers both trust and confidence, and converts trust propagation to a maximum flow, minimum cost problem. Jiang et al. [2016] address trust evaluation using a more general network flow model with leakage representing trust decay.

3.3. Researches on Recommendation and Influence

As mentioned, recommendation and influence are two closely related concepts of trust. In fact, the two concepts are also hot topics in both academic research and practical systems. Researches on recommendation try to predict a user's opinion on a specific item to recommend proper items to the user. *Collaborative filtering* [Terveen and Hill 2001] is a non-trust-based technique used by some recommendation systems that predicts a user's interests by considering many others' preferences in that user's community. Some efforts have been made to combine trust-based and collaborative filtering approaches [Hang et al. 2013; Resnick et al. 2000; Jiang et al. 2014].

In recommendation systems, a user's opinion (called rating) is usually represented as a numeric value. Andersen et al. [2008] use a finite set, $\{+, -, 0\}$, to represent positive, negative, and neutral ratings, respectively. In our previous work in Jiang et al. [2014], an opinion is measured by fluid temperature, which can easily be updated based on the volume and temperature of the new fluid.

In addition, people can be associated with both an "innate opinion" and an "expressed opinion" [Goel et al. 2010] for an item. The former is formed by the user himself and is independent of his social interactions; the latter can be shaped and influenced by others [Das et al. 2013]. In our work [Jiang et al. 2014], these two types of opinions can be treated as initial and mixed fluids. The work in Zhu et al. [2012] finds that a person's opinion can be significantly impacted by others' opinions. Bakshy et al. [2012] conduct several experiments, and the results validate that stronger ties are more influential and weak ties are more effective for novel information propagation. Wang et al. [2014] propose a model to evaluate social influence. That study provides a fine-grained framework in which users can define some specific features. The maximization of influence has been modeled as an optimization problem [Kempe et al. 2003] with various extensions [Chen et al. 2009, 2010, 2011; Maghami and Sukthankar 2013; Zhang et al. 2013].

3.4. Trust-Related Systems and Applications

Trust-related systems and applications can be generally classified into two categories: experimental or commercial. Among experimental systems, FilmTrust [Golbeck 2005] uses TidalTrust to rate films. In some sites, such as Advogato [Levien and Aiken 1998], users can rank others based on their skills in software development. To guard against spam, each user is assigned a trust quota (flow), which can be redistributed to other users subject to the trust capacity constraint of each user. The objective is to distribute the quota to as many trustful users as possible (and to cut off malicious users) through the use of a modified max flow. Appleseed [Ziegler and Lausen 2005] provides a couple of quota distribution rules to neighbors. Among commercial systems, eBay [Resnick et al. 2000; Resnick and Zeckhauser 2002] uses voting networks with votes of $+$, $-$, or 0. Amazon (www.amazon.com) adopts a rating from 1 to 5. Multiple-level ratings have been widely used in other fields, such as restaurant ratings in Zagat (www.zagat.com) and Yelp (www.yelp.com). The Epinions [Massa and Bhattacharjee 2004] website provides reviews of products by real people. In Epinions, users can add other users into their webs of trust if they find their reviews helpful. This approach can relieve the sparsity problem that has hampered the collaborative filtering approach. Rating aggregations are usually governed by a set of axioms [Altman and Tennenholtz 2005; Andersen et al. 2008].

In large-scale distributed systems (e.g., P2P systems), a malicious user may pretend to have multiple identities to make profits or mislead other users; we call these *sybil attacks*. Leveraging the observation that sybil nodes tend to be poorly connected to non-sybil nodes [Viswanath et al. 2011], a series of works have been proposed to detect sybils, including SybilGuard [Yu et al. 2006], SybilLimit [Yu et al. 2008], and SybilInfer

[Danezis and Mittal 2009]. Yu et al. propose SybilGuard [Yu et al. 2006] and SybilLimit [Yu et al. 2008]. Both enable an honest node s (the verifier) to decide whether or not to accept another node d (the suspect). In this sense, it is very similar to a trust model, which tries to help s decide the trustworthiness of d. All three schemes employ random walk approaches. Their algorithmic differences are present in that [Danezis and Mittal 2009; Viswanath et al. 2011] SybilGuard uses a single instance of a very long random route, SybilLimit employs multiple instances of short random walks to sample nodes from the honest set, and SybilInfer takes Bayesian inference on the results of the random walks. Viswanath et al. [2011] provide a deep analysis and comparison of those schemes that proves that they are all detecting local communities and suggests that they can be taken as "implicitly ordering or *ranking* nodes in the network."

Some follow-up work includes SumUp [Tran et al. 2009] and Iolaus [Molavi Kakhki et al. 2013]. SumUp [Tran et al. 2009] detects sybils using the technique of adaptive vote flow aggregation, which creates a voting envelope with appropriate link capacities around the collector. Iolaus [Molavi Kakhki et al. 2013] leverages the underlying social network of online content rating systems as a defense against sybil attacks and the "buying" of ratings from users. The former attack is handled by weighing ratings, and the latter uses relative ratings to mitigate the effect of "bought" ratings.

Website ranking is similar to a global trust and reputation management system. Ranking is usually distributed link-based. PageRank [Page et al. 1999] used in Google and RankDex [Li 1998] used in Baidu maintain one ranking number for each site (similar to a trust degree for each user). Ranking numbers are iteratively calculated through number exchanges with neighbors. HITS [Kleinberg 1999] in Ask.com maintains two ranking numbers for each site: hubs and authorities, which again are calculated iteratively through neighbor exchanges. CLEVER [Clever 2014] in IBM uses the notion of community in ranking decisions. TrustRank [Gyöngyi et al. 2004] relies on initial trusted seeds to carefully rank other sites to mitigate spam.

4. GRAPH SIMPLIFICATION-BASED APPROACH

We review some of the major works using the graph simplification-based approach, including TidaTrust [Golbeck 2005], MoleTrust [Massa et al. 2005], MeTrust [Wang and Wu 2011a], and SWTrust [Jiang et al. 2014]. We first describe their basic ideas using examples. Next, we discuss the challenges they meet. Finally, we present the main findings of empirical studies, and we analyze their time complexity and scalability.

4.1. Representative Models

—**TidalTrust.** Golbeck [2005] proposes TidalTrust. Given two people in a network, TidalTrust generates a recommendation about the trust degree that one person can put on the other based on trusted paths. The trusted paths are explored by taking breadth-first search from the trustor to the trustee. Note that only the shortest strongest trusted paths are used in TidalTrust, the pros and cons of which will be discussed later. The calculation of trust from s to d is as follows:

$$t_{sd} = \frac{\sum_{j \in Ns,\ t_{sj} \geq max} t_{sj} t_{jd}}{\sum_{j \in N_s,\ t_{sj} \geq max} t_{sj}}, \quad (1)$$

where N_s is the neighbor set of s, and max is the threshold of being trustful (i.e., j is taken as trustful only if $t_{sj} \geq max$). In fact, the name "TidalTrust" was chosen because of the similarity between the calculation method and a tidal stream: Calculations sweep forward from the trustor to the trustee, and then pull back from the trustee to return the final value to the trustor. Taking Figure 1(b) as an example, TidalTrust will take the trusted path (s, v, d), and it will gain a result of $t_{sd} = 0.7$.

—**MoleTrust.** Massa et al. [2005] propose MoleTrust, which has two steps. The first step is to delete cycles by sorting users based on their shortest distances from the trustor s. Then, it considers all users up to a maximum-depth, which is given as an input. It is worth noting that the maximum depth is independent of any specific user. Therefore, the trusted graph they use is actually a reduced *Directed Acyclic Graph* (DAG). MoleTrust first calculates the trust degrees of users who are one step away from s, then two steps, three steps, and so on. In addition, the trust degree of a user who is k steps away from the source only depends on those of users who are k-1 steps away. In this way, each user's information is used only once. The calculation is done with the weighted average of all the trustful incoming neighbors' trust toward the trustee d using Equation (2), as follows:

$$t_d = \frac{\sum_{i \in N_d^+} t_i t_{id}}{\sum_{i \in N_d^+} t_i}, \tag{2}$$

where N_d^+ is the trustful incoming neighbors of d, and t_i is the trust degree of i. Taking Figure 1(b), for instance, we have $t_u = 0.5$, $t_v = 0.6$, and $t_d = \frac{0.5*0.3+0.6*0.7}{0.5+0.6} \approx 0.52$.

—**MeTrust.** Wang and Wu [2011a] propose a trust evaluation system, MeTrust, using multitrusted paths with multidimensional evidence. They conduct trust computation at three layers. The node layer considers multidimensional trust; each user can assign a different weight for each dimension. The path layer applies the Frank t-norm to control the rate of trust decay during combination. The graph layer consists of three algorithms—GraphReduce, GraphAdjust, and WeightedAverage—to simplify trusted graphs. Since MeTrust considers many possible scenarios and variable settings can be flexibly selected, we prefer to take MeTrust as a comprehensive framework rather than a specific model or algorithm.

—**SWTrust.** In graph-based trust models, it is usually assumed that a small trusted graph already exists. Our previous work in Jiang and Wang [2011] and Jiang et al. [2014] proposes a framework, SWTrust, to preprocess an OSN and generate a trusted graph. SWTrust takes the basis of the small-world network characteristics of OSNs and the theory of "weak ties" [Granovetter 1983]. It uses information on users' active domains to construct the trust value, which is more objective compared to explicit trust ratings. Neighbors of a user are divided into three categories: local neighbors, longer ties, and longest ties, according to their social distance from the user. Then, it uses width-adjustable breadth-first search to discover trusted paths, where we uniformly select next-hop neighbors from three categories in each search step. This work is the first that "focuses on generating small trusted graphs for large OSNs, and explores the stable and objective information (such as users' domain) for inferring trust" [Jiang and Wang 2011; Jiang et al. 2014]. In addition to generating trusted graphs, SWTrust also implements eight trust prediction strategies by combining three factors of propagation functions (*Min* and *Multiply*), aggregation functions (*Max* and *Weighted Average*), and whether only shortest paths are taken or not.

In addition, our previous work, Jiang et al. [2013, 2015], proposes the RATE algorithm to select proper recommenders to infer trust. In that work, we study how to measure the quality of recommenders and how to select the proper amount of recommenders. Liu et al. [2013b] propose MFPB-HOSTP, a "Multiple Foreseen Path-Based Heuristic algorithm" to find trust paths.

4.2. Challenge: Path Length Limitation & Evidence Availability

Graph simplification-based approaches use both the propagative and composable properties of computation trust. For a trusted path, propagation works in this way: If s

Table I. Comparison of Existing Graph-Based Trust Models in OSNs

Model*	Cat.	Computation Model	Trust Value	Dimension	Trust Information	Test data set
TidalTrust	S	linear model	discrete, [1, 10]	1	trust	FilmTrust
MoleTrust	S	linear model	continuous, [1, 5]	1	trust	Epinions
MeTrust	S	linear model	continuous, [0, 1]	2	confidence, trust	-
SWTrust	S	linear model	continuous, [0, 1]	1	trust	Epinions
RATE	S	linear model	continuous, [0, 1]	4	trust, influence, uncertainty, cost	Epinions
MFPB-HOSTP	S	linear model	continuous, [0, 1]	3	trust, intimacy, role impact	Enron email*
RN-Trust	A	resitive network	continuous, [0, 1]	1	trust	-
Appleseed	A	spreading activation	continuous, [0,in(s)]	1	trust	-
Advogato	A	network flow	discrete, 4 levels	1	trust	Advogato
FlowTrust	A	network flow	continuous, [0, 1]	2	confidence, trust	-
GFTrust	A	network flow	continuous, [0, 1]	1	trust	Epinions; Advogato

1. 2nd column, Cat., category; S/A, simplification/analogy.
2. Source of models: TidalTrust [Golbeck 2005]; MoleTrust [Massa et al. 2005]; MeTrust [Wang and Wu 2011a]; SWTrust [Jiang et al. 2014]; RN-Trust [Taherian et al. 2008]; Appleseed [Ziegler and Lausen 2005]; Advogato [Levien and Aiken 1998]; FlowTrust [Wang and Wu 2011a]; GFTrust [Jiang et al. 2016].
3. Enron email [Goldstein et al. 2006].

trusts u, and u trusts v, then s can derive some trust toward v. Then, it faces the challenge of setting a proper limitation of path length; a smaller limitation may lead to fewer paths, whereas a larger one may cause inaccurate prediction. For multiple trusted paths in a trusted graph, how to combine the available evidence is the main challenge.

For this point, Golbeck [2005] conducts some experiments and finds that a shorter path can predict trust with a higher accuracy. Hence, in the proposed TidalTrust algorithm, she only uses the "shortest and strongest" paths for trust inference. The approach has its two sides. On one side, it can filter out most of the noisy evidence; on the other side, some useful information may be neglected. Jøsang et al. [2006a] point out that "trust can be diluted through the propagation process," in which a longer trust referral chain leads to weaker predicted trust. Lesani and Montazeri [2009] present a different view. They suggest that the information inferred from a highly trustful long chain may be much more precise than that from a low trustful short chain. Their work indicates the balance of "trust availability" and "path reliability." Cho et al. [2012] take a further step and try to identify the optimal path length and generate the most accurate trust based on "a tradeoff between trust availability and path reliability over trusted space." Kim and Song [2011] study these problems comprehensively. They compare four trust prediction strategies: "weighted mean aggregation among shortest paths," "min-max aggregation among shortest paths," "weighted mean aggregation among all paths," and "min-max aggregation among all paths." Among those four, the "weighted mean aggregation among all paths" performs best. This finding indicates that more trust evidence may help trust prediction.

However, to the best of our knowledge, there is still no conclusion about *the best trusted path length* and *the most proper number of paths*, even in a specific context. The challenge is still open and worth further attention. Nevertheless, the problem is context-dependent, and the key is to find a balance between path length [Kim and Song 2011] and evidence availability [Cho et al. 2012].

4.3. Empirical Studies and Analysis

We conducted comparative experiments in Jiang and Wang [2011] and Jiang et al. [2014], with respect to TidalTrust, MoleTrust, and SWTrust. Here, we only report the main findings as follows:

—SWTrust is more robust against vicious nodes when using stable and objective information to infer trust. It can weaken the effect of vicious nodes because the information cannot be changed at will.
—In most cases, MoleTrust and SWTrust are more accurate than TidalTrust. This is because there is usually a single shortest strongest path that TidalTrust uses, and the opinion from multiple paths is usually better than that of a single path because it avoids being subjective and one-sided.
—SWTrust is more comprehensive for considering *trust conflict* in dealing with controversial users. Introducing the factor of *trust conflict* can increase accuracy because it can weaken the negative effect of one-sidedness, especially when using the *Max* function to do aggregation.

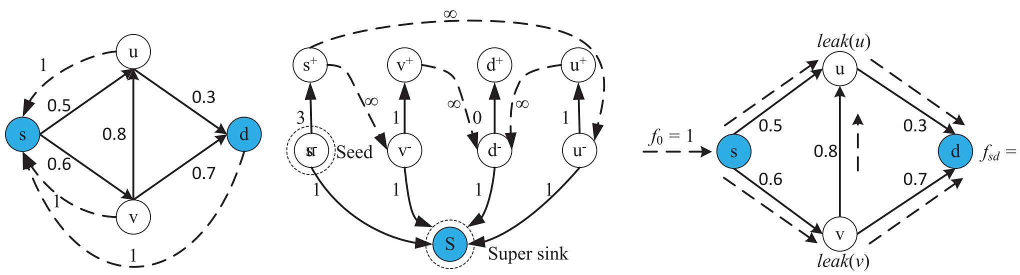
Time complexity and scalability. The main operation in TidalTrust, MoleTrust, and SWTrust is breadth-first search, for which the complexity is $O(|V|+|E|)$, where $|V|$ is the number of nodes and $|E|$ is the number of edges. Moreover, all three methods take some strategy to reduce complexity. TidalTrust takes only the shortest and strongest paths. MoleTrust limits the hops from source: for example, MoleTrust1 only considers the direct neighbors of the source, MoleTrust2 only considers the neighbors of the source's neighbors (2-hop distance from source), and so on. SWTrust restricts the width in each hop, with a parameter (say, $w = 3, 6, 9\ldots$) representing how many neighbors will be selected. As to scalability, since only small subsets of relatively constant size (e.g., the lengths of trusted paths or the width of next hops are limited) are visited, the graph simplification-based trust metrics will scale well to any social network size.

5. GRAPH ANALOGY-BASED APPROACH

In this section, we review some of the major works using a graph analogy-based approach, including RN-Trust [Taherian et al. 2008], Appleseed [Ziegler and Lausen 2005], Advogato [Levien and Aiken 1998], FlowTrust [Wang and Wu 2011a], and GFTrust [Jiang et al. 2016]. We first describe their basic ideas with examples. Next, we discuss the challenges they meet. Finally, we present the main findings of empirical studies, and we analyze their features, complexity, and scalability.

5.1. Representative Models

—**RN-Trust.** Inspired by the similarity between trust propagation and electric flows in which"the less resistance there is, the more the electric current that can pass," Taherian et al. [2008] propose RN-Trust. In this work, a trusted graph is transformed into a resistive network using the equation $r = -logt$, where r represents the resistance and t represents the trust degree. To gain the final trust from s to d,



(a) Appleseed (b) Advogato (c) GFTrust

Fig. 7. The graphs after transformation by (a) Appleseed, (b) Advogato, and (c) GFTrust.

RN-Trust first computes the equivalent resistance value, R_{sd}^{eq}. Then, the trust value can be inferred by $t_{sd} = 10^{R_{sd}^{eq}}$.

Figure 1(c) shows an example of a resistive network using two trusted paths (s, u, d) and (s, v, d) in Figure 2. In this example, RN-Trust will calculate resistances as follows. For edges, it has: $r_{su} = -logt_{su} = -log0.5 = 0.3$, $r_{ud} = -log0.3 \approx 0.52$, $r_{sv} = -log0.6 \approx 0.22$, $r_{vd} = -log0.7 \approx 0.15$. For each path, it has $r_{(s,u,d)} = -log0.5 - log0.3 \approx 0.82$, and $r_{(s,v,d)} = -log0.6 - log0.7 \approx 0.37$. Then, the two paths are parallel and $R_{sd}^{eq} = \frac{0.82*0.37}{0.82+0.37} \approx 0.25$. Finally, $t_{sd} = 10^{R_{sd}^{eq}} = 1.77$.

—**Appleseed.** Ziegler and Lausen [2005] propose a novel local group trust evaluation method, Appleseed. In this model, an initial amount of energy $in(s)$ (indicating the trust) is injected into the source s and then propagates into successor nodes. The more energy a node receives, the more trustworthy it is. The authors borrow the idea from spreading activation models and make some adaptations to handle more complex cases. Particularly, they use a global spreading factor δ to handle trust decay along trusted paths; they define an edge weight normalization function to make the result more reasonable, as follows:

$$e_{x\to y} = \delta * in(x) * \frac{t_{xy}}{\sum_{j\in N_x} t_{xj}}, \tag{3}$$

where t_{xj} is the trust degree from x to j, and N_x is the neighbor set of x. For node x, considering the total energy it receives and the decay factor, the amount of energy it can keep is calculated as follows:

$$in(x) = \sum_{x\in N_p} e_{p\to x} = \delta * \sum_{x\in N_p} \left(in(p) * \frac{t_{px}}{\sum_{j\in N_p} t_{pj}} \right). \tag{4}$$

It is worth noting that Appleseed makes use of backward propagation of trust to the source: When computing the metric, additional "virtual" edges (x, s) from every node x (note: here $x \neq s$) to the trustor s are created; moreover, $t_{xs} = 1$, indicating full trust (see Figure 7(a)). Next, the trust degree of x is updated as follows:

$$t_x = energy(x), or, t_x \leftarrow t_x + (1 - \delta) * in(x). \tag{5}$$

Finally, the algorithm terminates after k rounds, such that the change of trust values is not larger than some fixed accuracy threshold $T_c > 0$.

Again, we take Figure 1(b) as an example. Suppose the initial energy $in(s) = 5$ and the decay factor $\delta = 0.85$. s will keep $energy(s) = (1 - 0.85) * 5 = 0.75$ energy, and the amount of 4.25 is distributed to its successors. Then, $e_{s\to u} = 4.25 * \frac{0.5}{0.5+0.6} \approx 1.93$, $e_{s\to v} = 4.25 * \frac{0.6}{0.5+0.6} \approx 2.32$. Since v has only one incoming edge, we have

$in(v) = e_{s\to v} = 2.32$, and $energy(v) = (1 - 0.85) * 2.32 \approx 0.35$, remaining 1.97 will be distributed to s, u, and d. Similarly, $e_{v\to s} = 1.97 * \frac{1}{0.8+0.7+1} = 0.788$, $e_{v\to u} = 1.97 * \frac{0.8}{0.8+0.7+1} = 0.63$, $e_{v\to d} = 1.97 * \frac{0.7}{0.8+0.7+1} = 0.552$. Now, the energies in the two incoming edges of u are already known; we have $in(u) = e_{s\to u} + e_{v\to u} = 2.56$, and $energy(u) = (1 - 0.85) * 2.56 \approx 0.384$, and the remaining 2.176 will be distributed to s and d. That is, $e_{u\to s} = 2.176 * \frac{1}{0.3+1} \approx 1.674$, $e_{u\to d} = 2.176 * \frac{0.3}{0.3+1} \approx 0.502$. Finally, we have $in(d) = e_{u\to d} + e_{v\to d} = 0.502 + 0.552 = 1.054$, and $energy(d) = (1 - 0.85) * 1.054 \approx 0.158$. The remaining 0.896 will be distributed to s, and we update $energy(s) = 0.75 + 0.788 + 1.674 + 0.896 = 4.108$. Up to now, a round of calculations have been done. Then, multiple rounds will be done until it reaches the termination condition. At last, whether a node is trustful or not is determined by its trust degree and a trust threshold.

Network flow theory has been used in many fields [Ahuja et al. 1993] and has been introduced into a trust evaluation system in recent years. We mention three representatives of network-flow based models: Advogato, FlowTrust, and GFTrust.

—**Advogato.** Levien and Aiken [1998] propose the Advogato maximum flow trust metric. It identifies and cuts out "bad" nodes as well as other nodes that certify the "bad" nodes, which are taken as the unreliable portion of the network. The computation of Advogato is conducted according to trusted "seeds." Before any computation, some nodes are taken as trusted seeds. Then, the algorithm conducts breath-first search on the graph and assigns each node a capacity according to the shortest distance from the node to the seed. The capacity of seed is a given input n, representing how many trustful nodes the system wants to find. The capacity of distance level $l + 1$ is calculated as: $C^{l+1} = \frac{C^l}{Average^l_{outdegree}}$, where $Average^l_{outdegree}$ is the average outdegree of trust edges $e \in E$ extending from l. In this way, nodes closer to the seed will have higher capacities, and vice versa. Moreover, nodes with the same distance level will have the same capacity. After assigning capacities, Advogato uses a transform algorithm to convert single-source/multiple-sink graphs into single-source/single-sink. To be specific, it introduces a super sink, and it splits each node x into x^- and x^+. The edge capacity is assigned 1 from x^- to the super sink (when $C_x \geq 1$), and the remaining capacity of x is assigned to the edge from x^- to x^+. The capacity of the edge from x^+ to y^- is set to ∞. Finally, Advogato conducts a maximum integer network flow algorithm to choose trustful nodes.
  Taking the trusted graph in Figure 1(b) for instance (neglecting the trust values on edges), Advogato will transform the graph into Figure 7(b). The trustor s is taken as the seed whose capacity is given as an input; here we suppose $C_s = 4$. Next, it assign capacities to 1-hop neighbors u and v. Since the outdegree of s is 2, then both neighbors get a capacity of $C_s/2 = 2$. Repeat the process and d will get a capacity of $C_d = C_u/1.5 = 1$, where the average outdegree is 1.5 for u and v. Finally, seed s will accept itself, u, v, and d as trustful peers.

—**FlowTrust.** Wang and Wu [2011a] present FlowTrust, in which they consider two dimensional factors: the *trust value* and the *confidence level*. From those two factors, four metrics are deduced, namely "maximum flow of trust value," "maximum flow of confidence level," "minimum cost of uncertainty with maximum flow of trust value," and "minimum cost of untrust with maximum flow of confidence level." FlowTrust first computes these four metrics using maximum flow and minimum cost flow algorithms. Then, some algorithms are proposed to normalize the four metrics, to make the result fall into the range of [0, 1]. It is worth noting that the normalization may impact the final results.

—**GFTrust.** Jiang et al. [2016] propose a trust evaluation scheme, GFTrust, using generalized flow in which flow leaks as it is sent in the network. The leakage is represented as the gain factor of an edge. For instance, a flow of 3 enters into an edge (v, u); suppose the gain factor $g(v, u) = 0.8$. Then, $3 * 0.8 = 2.4$ flow will go out of the edge. Jiang et al. [2016] first identify two challenges of trusted path dependence and trust decay during propagation. Then, they address path dependence using network flow and model trust decay with the leakage associated with each node. To construct a generalized network, each intermediate node x is split into x^+ and x^-; an intermediate edge (x^+, x^-) is added into the network, with the gain factor $g(x^+, x^-) = 1 - leak(x)$, and the capacity is set to 1. The gain factor of other edges is 1. Capacities of other edges are equal to their trust values on it. The total initial flow is set to 1, to save the process of normalization. Finally, the calculation of trust from s to d is converted to calculate the near optimal maximum generalized flow f_{sd}. It has two steps: searching the shortest path with breadth-first search and augmenting flow through the path. The second step contains two operations: augmenting a flow f through the selected path and calculating the residual capacity of each edge from d to s, as well as the residual flow that s can send out.

Taking the trusted graph in Figure 1(b) for instance, GFTrust will transform the graph into Figure 7(c). After splitting intermediate nodes and constructing generalized flow network, we let $p_1 = (s, u^+, u^-, d)$, $p_2 = (s, v^+, v^-, d)$, $p_3 = (s, v^+, v^-, u^+, u^-, d)$. Next, do the breadth-first search and find the first unused shortest trusted path; suppose it is p_2. Then, send flow from s to d through p_2. After that, record p_2 in a used path list. Suppose the leakage is 0.1 and the gain factor is $1 - 0.1 = 0.9$. Then, the process of calculating a feasible flow is as follows: (i) Find the shortest trusted path p_2 to send flow. It results in $f_1 = 0.54$, and the residual flow of s is $1 - 0.6 = 0.4$. (ii) Send the residual flow along the second shortest trusted path p_1, which results in $f_2 = 0.3$, with remaining $f(s) = 0.06$. (iii) Send the residual flow along p_3. Since edge (s, v^+) has no capacity, no more flow can be sent. (iv) The final flow/trust is $f_{sd} = f_1 + f_2 = 0.84$.

5.2. Challenges: Normalization and Scalability

Normalization. Graph analogy-based models may produce a result that is out of the range of trust. Then, proper normalization will have to be done. RN-Trust uses the reverse mapping function to gain a trust value from an equivalent resistance. However, it does not mention the range of trust and how to guarantee such range. Network flow-based approaches usually compute the maximum flow, and they will have to resort to normalization to get a reasonable result. For instance, Wang and Wu [2011a] use $maxT/maxP$ to normalize the maximum flow into a trust value, where $maxT$ represents the "maximum trust flow" and $maxP$ represents the "maximum number of edge-disjoint paths" from a source to a target. Advogato does not conduct any normalization since it does not care what the exact trust value of a node is, but rather the rank of each node. For a general graph analogy-based model, how to design a reasonable normalization algorithm is a big challenge. Appleseed conducts normalization for edge weight during energy diffusion. In this way, it keeps the conservation of total energy. GFTrust sets the initial flow to 1 to guarantee that the trust value falls within the range of $[0, 1]$. Hence, it avoids the normalization process.

Scalability. In general, when using a graph analogy-based model, the network scale should not be large because of the time and space complexity. Massa and Avesani [2007b] prefer the local approach for accurate evaluation of personal trust. However, if there is a need to compute all users' trust degrees, the local approach is more complex. In this case, it is more reasonable to distributively run each user's trust metric in his own machine. We believe that a better way is to combine the graph simplification- and graph analogy-based approaches: first, use simplification to generate a small trusted graph; then, conduct a graph analogy trust evaluation.

5.3. Empirical Studies and Analysis

We conducted comparative experiments in Jiang et al. [2016] with respect to GFTrust, SWTrust, and some common strategies including AveR-MaxT, AveR-WAveT, MaxR-MaxT, and MaxR-WAveT. Suppose there are multiple paths from s to an incoming neighbor of d; AveR will take the average path weight as the reliability, whereas MaxR will take the maximum. Meanwhile, MaxT takes the direct trust of d's neighbor who has the maximum reliability, and WAveT takes the weighted average value among all direct neighbors of d. The main findings are as follows:

—The experiments verify the incentive property of GFTrust. For the pairs of s and d: (i) When GFTrust gives a higher trust $\widehat{t_{sd}}$ than the direct trust t_{sd}, there are usually several short, trusted paths between them, the length of which are $L \leq 4$. (ii) To the contrary, when GFTrust gives a lower trust than the direct trust, it usually happens when there are no short, trusted paths, for which there are two subcases: (a) There are several long paths, which have many intermediate nodes and cause too much leakage; (b) there are not enough paths to send all the initial flow.
—The use of flow improves the metric of FScore, and the proper setting of the leakage reduces the Mean Average Error (MAE).

In the experiments, we do not compare GFTrust with other models because it is difficult to conduct a fair comparison: RNTrust cannot deal with many scenarios; Appleseed and Advogato are group trust metrics where nodes are classified as either trustful or vicious ones; and FlowTrust has two-dimensional information on trust and confidence. Thus, we only analyze their strengths and weaknesses, as follows:

—RNTrust is elegant for its clever mapping from a trusted graph to a resistive network. However, due to the computational complexity of equivalent resistance, the network scale should not be very large. Moreover, it will be very challenging if there are overlapping paths. What may be even worse is that it cannot guarantee that the resulting trust values fall into some specific range (e.g., [0,1]).
—Appleseed and Advogato are group trust metrics that evaluate the trustworthiness of a group of users simultaneously. Appleseed takes advantage of a partial trusted graph to get computational scalability. Therefore, it is more efficient and can be taken as a hybrid approach of the graph simplification and graph analogy models. A detailed comparison of them can be found in Ziegler and Lausen [2005].
—RNTrust, FlowTrust, and GFTrust are scalar/personal trust metrics that calculate the trustworthiness of each user independently.
—Appleseed, Advogato, FlowTrust, and GFTrust obey the rule of conservation (i.e., energy conservation for Appleseed, flow conservation for the other three).
—Similar to Appleseed, FlowTrust and GFTrust also resort to a small trusted graph. In addition, GFTrust avoids the normalization process.

Time complexity and scalability. Suppose V and E are the node set and edge set of a trusted graph, respectively. Also, suppose V' and E' are the node set and edge set of a whole trust network/social network, respectively. Appleseed, FlowTrust, and GFTrust work on a trusted graph. For RNTrust, the main work is calculating equivalent resistance in a circuit network. Using the method of mesh current analysis, the worst case complexity is $O(|V|^3)$ if it works on a trusted graph and $O(|V'|^3)$ if it works on a whole network. Appleseed considers all neighbors of all nodes in a trusted graph, for which the complexity is $O(k|V||E|)$ (k is the total number of rounds after the algorithm converges). The main operations in FlowTrust and GFTrust are calculating maximum flow based on a given trusted graph, for which the complexity is $O(|V||E|^2)$ using the Ford-Fulkerson method [Ford and Fulkerson 1962]. GFTrust refines the path selection process according to its specific trust evaluation task. In this way, it reduces the complexity to $O(|E|)$ in most cases. Meanwhile, in the worst cases, the complexity is $O(|V||E|^2)$. Advogato runs a maximum network flow algorithm in the whole network, for which the complexity is $O(|V'||E'|^2)$ using the Ford-Fulkerson method. As to scalability, all the mentioned graph analogy-based models except Advogato are based on a small trusted graph; therefore, they scale well to any social network size.

6. COMMON CHALLENGES

In the preceding two sections, we comparatively study the graph simplification- and graph analogy-based trust models and point out their individual challenges. Table I compares the mentioned representative models from multiple aspects of category, computation model, form (discrete or continuous) and range of trust value, information for trust, and test dataset.

In this section, we extract the common challenges that any graph-based approach may encounter and review the existing literature. The most important four challenges are path dependence, trust decay, opinion conflict, and attack resistance.

6.1. Path Dependence

In a trusted graph, some trusted paths may overlap with others. For example, in Figure 1(b), (s, u, v, d) overlaps with (s, u, d) and (s, v, d). It is challenging to treat overlapping trusted paths since some paths may share one or several edges. We call this the "path dependence" challenge [Jiang et al. 2016].

Researchers (e.g., Lin et al. [2009a], Golbeck [2005], and Jøsang et al. [2006b]) have taken some action to address the challenge of path dependence. However, there is still no universally accepted solution. Existing work may ignore or reuse some information on the shared edges. For example, TidalTrust [Golbeck 2005] uses only the shortest, strongest paths and neglects all others. Lin et al. [2009a] use only the shortest paths. Jøsang et al. [2006b] take each path (no matter wether it is overlapping with others or not) as an independent path, which reuses the information on the shared edges. Trust evidence is one of the key factors of trust evaluation. Thus, a comprehensive trust model is expected to treat trust evidence properly. Based on our experience, either evidence ignorance or reuse may lead to inaccurate results for trust evaluation. Taking Figure 1 for instance: There are two neighbors, u and v, who express different opinions toward the trustee d. If only the shortest and strongest path is used, that is (s, v, d), the opinion of u will be neglected and may lead to an optimistic result. In contrast, if all three paths are used, that is, (s, v, d), (s, u, d), and (s, v, u, d), the opinion of u will be considered twice; this may lead to a discouraging result.

6.2. Trust Decay

Due to the time sensitivity of trust, it may change (usually decay) with time. Moreover, trust may decay via iterative recommendations because people put more trust in friends than in strangers. We call the former *decay with time* and the latter *decay with space*. The two types of decay indicate that time should be an essential factor of a comprehensive trust model, and the length of a trusted path cannot be too long. In both cases, the pattern of decay may impact the final trust evaluation.

Some work has studied trust decay. For decay with time, Nguyen et al. [2012] exhibit the impact of mobility to trust decay in MANETs and present an analysis of the trust decay rate for some general networking and trust computation models. Peng et al. [2013] present a model to update trust according to certain rules (e.g., "trust should increase slowly, but drop fast" and "trust should fade with time." For decay with space, Wang and Wu [2011b] provide parameters for users to decide decay rate. Sun et al. [2006b] develop four axioms to serve as the principles for trust propagation, including "concatenation propagation of trust does not increase trust" and "multipath propagation of trust does not reduce trust." Jiang et al. [2016] try to solve path dependence and trust decay simultaneously using a modified network flow model.

Similar to our daily life, many factors are involved in the processes of trust propagation and people's opinion formulation in OSNs, including the personality of users, the time the information is being created and propagated, and the strength of connections. Therefore, the patterns of how trust decays and propagates is worth further study.

6.3. Opinion Conflict

Due to the subjectivity of trust, people may have different opinions toward the same target: Some may give high opinions, while others give low ones. We call this phenomenon *conflict of opinions* or *controversiality of the target* [Massa and Avesani 2007b]. Then, how to combine different opinions becomes a big challenge.

The existing work uses several ways to solve conflicts. The most common approaches include (i) taking those paths whose trust levels are above a predefined threshold or taking the most reliable paths [Golbeck 2005] and neglecting others, which may cause information loss; and (ii) taking the weighted average or the most reliable path to gain a final opinion [Massa et al. 2005; Wang and Wu 2011a; Jiang et al. 2014; Wang and Wu 2011a; Kim and Song 2011]. Although these models can predict trust with high accuracy in most cases, in actuality, conflicts have not been well studied.

Currently, there is limited work in conflict solving, and better solutions are expected. Research in other related fields can be introduced to solve this challenge. For instance, the research in sociology [Cialdini and Trost 1998] finds that people often conform because of a desire for security within a group. Andersen et al. [2008] propose several axioms for integrating trust-based recommendations from different friends in which three opinions (positive, negative, and neutral) are considered. Jiang et al. [2014, 2015] study how a user's opinion is influenced by trusted friends using fluid dynamics theory. Leskovec et al. [2010a] study both positive and negative relations in OSNs. They find that a user's attitude toward another user can be estimated from their social relationships; in particular, negative relations can be predicted by exploring positive relations, and vice versa. They further study how the interplay between positive and negative relationships affects the structure of OSNs in Leskovec et al. [2010b]. Their work may be useful for resolving conflicting opinions.

Based on several prior studies, we suggest that efforts in solving opinion conflict in trust evaluation can be made from two main aspects: One is to fully understand the personal bias and features of the trustor; the other is to deeply explore the fundamental principles regarding the formulation and evolution of people's opinions.

6.4. Attack Resistance

Trust evaluation is taken as a "soft security" mechanism compared to security schemes such as encoding. However, the trust system itself may be attacked either by malicious or selfish users. In OSNs, users may conduct several kinds of misbehavior, such as providing bad service [Jøsang et al. 2007], sybil attack [Douceur 2002], bad-mouthing [Sun et al. 2006a], on-off attack [Sun et al. 2006a], conflicting behavior attack [Sun et al. 2006a], and social spamming [Stringhini et al. 2010]. Then, how to make a trust model resistant to possible attacks is a more serious challenge.

A few works have assessed attack-resistant trust models. Levin discussed the attack resistance property of trust metrics in Levien [2003]. He comparatively analyzes the attack resistance properties of group trust and scalar metrics, and he finds that the

Table II. Comparison of Representative Models on Challenge-Tackling

	Common Challenges				Individual Challenges		
Model*	**C1**	**C2**	**C3**	**C4**	**I1**	**I2**	**I3**
TidalTrust	Shortest path; information loss	yes	no	no	yes	N/A	N/A
MoleTrust	DAG, information loss	no	yes	no	no	N/A	N/A
MeTrust	*GraphReduce*; Information loss	yes	no	no	no	N/A	N/A
SWTrust	Trusted paths by width and length restricted breadth-first search; Information loss	yes	yes	yes	yes	N/A	N/A
RN-Trust	No	yes	no	no	N/A	no	no
Appleseed	partial trusted graph exploration; Information loss	yes	no	yes	N/A	yes	yes
Advogato	Social network; yes	no	no	yes	N/A	no	no
FlowTrust	Trusted graph; yes	no	no	no	N/A	yes	yes
GFTrust	Trusted graph; yes	yes	no	yes	Yes	yes	yes

1. C1, C2, C3, C4 represent path dependence, trust decay, opinion conflict, and attack resistance, respectively; I1, I2, I3 represent path dependence and evidence availability, normalization and scalability, respectively.
2. Source of models: TidalTrust [Golbeck 2005]; MoleTrust [Massa et al. 2005]; MeTrust [Wang and Wu 2011a]; SWTrust [Jiang et al. 2014]; RN-Trust [Taherian et al. 2008]; Appleseed [Ziegler and Lausen 2005]; Advogato [Levien and Aiken 1998]; FlowTrust [Wang and Wu 2011a]; GFTrust [Jiang et al. 2016].
3. In the 2nd column, the first part is the information or algorithm used to solve path dependence; "information loss" is the side effect; "yes" means solving the challenge successfully. DAG, direct acyclic graph.

former is better. Sun et al. [2006b] discuss several attacks of trust models. Jiang et al. [2014] study two types of collusion and noncollusion attacks. Jiang et al. [2016] propose a flow-based model that provides social incentive compatibility and sybil tolerance. Hoffman et al. [2009] study attacks in reputation systems, where they identify possible components that are vulnerable to attacks and compare several defense mechanisms.

Although much work has been done, due to the open and dynamic nature of OSNs and the variety of online attacks, there is a lack of comprehensive work that can handle all possible attacks. Therefore, a combination of trust and security models may be expected. An attack-resistant trust model should be able to punish malicious behaviors and provide more incentive techniques to encourage user cooperation.

Table II shows a summary of representative models from the aspects of their challenge-handling abilities. We can see that there is a need for comprehensive trust models that can handle more (or even all) possible challenges.

7. PRE- AND POSTPROCESSES

In the preceding sections, we reviewed and discussed trust evaluation regarding the evaluation process. In this section, we look at the work that needs to be done before and after the model; that is, the preparation (i.e., collecting information for trust and constructing trusted graph), the validation of performance, and the applications in which a trust model can be incorporated (see Figure 6).

7.1. Preparation

Information Collection for Trust. As mentioned, trust is commonly taken as being subjective and personal; trust can also be specific or general. Information collected for forming trust can be either one- or multidimensional [Gefen 2002; Wang and Wu 2011a; Yao et al. 2013]. Trust can be represented by a scalar number. It can be continuous and normalized to the range [0, 1]; it can also be binary (0 or 1 in Epinions) or have discrete levels (e.g., [1, 5] in Amazon). In Theodorakopoulos and Baras [2006], trust is

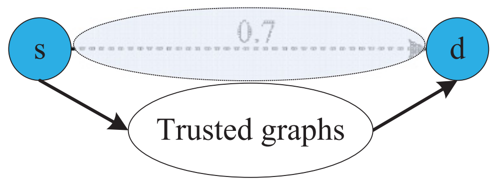


Fig. 8. The leave-one-out method.

measured by a tuple of rating values and confidence values from [0,1]: A total trust corresponds to the highest rating (which is 1) and full confidence (again 1). Li and Wu [2010] give a more general confidence measure that depends on both the frequency and duration of contact. Jøsang [1999] uses subjective logic and proposes using a triplet to represent trust, with belief (b), disbelief (d), and uncertainty (u), normalized such that $b + d + u = 1$.

TidalTrust [Golbeck 2005] takes user ratings as the information used for predicting trust. MoleTrust [Massa et al. 2005] uses the web of trust to indicate whether to trust or not. SWTrust [Jiang et al. 2014] uses a *domain* for inferring trust, which is objective and stable. MeTrust [Wang and Wu 2011a] takes multiple dimensional information into consideration. Gefen [2002] proposes a three-dimensional trust metric for e-commerce environments in which trust consists of “trustworthiness, dealing with integrity,” “benevolence,” and “ability in the unique case of online consumer trust.” RN-Trust [Taherian et al. 2008] assumes that trust information is already known. Advogato [Levien and Aiken 1998] decides the capacities of nodes according to their distance with seeds. FlowTrust [Wang and Wu 2011a] considers both confidence level and trust value. However, how to get the information is not mentioned.

Other information may also be incorporated into a trust evaluation system. Jiang et al. [2013, 2015] present the idea of evaluating trust by selecting proper recommenders; they identify four metrics to estimate the quality of recommenders, namely the “trustworthiness,” “influence,” “uncertainty,” and “cost.” Gefen and Pavlou [2012] study the boundaries of trust and risk. Liu et al. [2013b] propose the MFPB-HOSTP algorithm to select optimal trusted paths with multiple constraints. They consider three metrics of “trust,” “social intimacy degree,” and “role impact factor.” Yao et al. [2013] propose MATRI, in which they take “multiple aspects,” “transitive trust,” and “bias” into consideration. Table I summarizes the information used for evaluating trust.

Trusted Graph Construction. As mentioned, there is a need to generate a small trusted graph. SWTrust [Jiang et al. 2014] is proposed exactly for this aim. It takes advantage of weak ties by differentiating local neighbors from long contacts. Then it prefers long contacts to construct trusted chains to improve coverage and reduce cost. Yao et al. [2012b] also target the issue, proposing a two-stage method. One stage is path selection, the other stage is component induction. Zuo et al. [2009] propose a framework for trust evaluation using a set of trusted chains. They also provide the notion of a “base trusted chain set.” However, no algorithm is developed to identify the set.

7.2. Model Evaluation

Evaluation Method. There are two typical ways to evaluate the performance: (i) using feedback by simulation (due to a lack of enough real feedback), as in Caverleea et al. [2010]; and (ii) using the leave-one-out method, as in Massa and Avesani [2007a] and Jiang et al. [2014]. Figure 8 shows this method: Mask the direct edge (s, d) and calculate trust via a trusted graph. The performance can be evaluated by comparing the masked and calculated values to check how close the two values are. For instance, suppose we have a dataset with 100 pairs of trust relations (i.e., direct edges). For each pair (edge), the original edge (say, (s, d)), and trust value will be masked (i.e., assuming there is no such edge), and a trusted graph will be constructed by searching for proper trust chains from s to d. Based on the trusted graph, an algorithm can be implemented and a calculated trust value can be obtained and recorded. After that, edge (s, d) will be recovered, and the next edge will be treated in the same way. After all 100 pairs have been considered, we can measure the performance through calculating accuracy metrics (e.g., Precision, Recall, FScore). It is worth noting that not all test pairs have trust chains (other than direct relation) between them. Therefore, the *coverage* representing the ratio of predictable test pairs over the total number of test pairs is also an important metric.

Evaluation Metrics. The two most commonly used metrics are the *coverage* and *accuracy* of trust prediction. The former represents the ability of algorithms to provide a prediction; that is, the percentage of trust relationships that are predictable (at least one trusted path is available between two users). The latter represents "the ability of predicting whether a user will be trusted or not" [Jiang et al. 2014].

—*Coverage*. Let Γ be the amount of source/target pairs that are predictable and *Total* be the total number of test pairs. Then, $Coverage = \Gamma/Total$. The following metrics are defined for prediction accuracy.
—Mean Absolute Error (*MAE*) is calculated as $MAE = \sum(t_{sd} - \hat{t_{sd}})/\,\Gamma$, where t_{sd} and $\hat{t_{sd}}$ denote the real and predicted trust values, respectively.
—Root Mean Squared Error (*RMSE*)[2] is deemed an improvement over MAE [Massa and Avesani 2006]. It is calculated as $RMSE = \sqrt{\sum(t_{sd} - \hat{t_{sd}})^2/\,\Gamma}$.
The metrics of MAE and RMSE represent how close predictions are to real trust values; a smaller MAE or RMSE indicates a higher prediction accuracy.
—*Precision* represents the fraction of users who are predicted to be trusted and are really trusted. It can be calculated as $A_t \cap B_t/B_t$, where A_t is the number of source/target pairs in which the source trusts the target directly, and B_t is the number of source/target pairs in which the source trusts the target by the calculated trust.
—*Recall* is the fraction of users who are really trusted and are successfully predicted. It is calculated as $A_t \cap B_t/A_t$.
A higher Precision and Recall indicates a higher prediction accuracy.
—*FScore* is calculated as 2·Recall·Precision/(Recall+Precision).
The equation implies the purpose of the FScore metric: to combine the metrics of Recall and Precision and provide a joint measure.

Dataset. All OSNs (e.g., Facebook, Myspace, etc.) are potential testbeds for trust models in OSNs. However, most may not provide explicit trust information. Here, we introduce three commonly used real social network datasets with explicit trust: Advogato (www.advogato.org), Epinions.com (www.epinions.com), and FilmTrust (http://trust.mindswap.org/FilmTrust).

The website of Advogato is designed by Levin [Levien and Aiken 1998] for free software development and for testing the Advogato algorithm. Since its development, it has been widely used in testing other trust models. On Advogato, users can rank others with four choices: Observer, Apprentice, Journeyer, and Master, which can be assigned 0.4, 0.6, 0.8, and 1.0, respectively, to numerate the level of trust. The snapshots can be found at [Trustlet 2014].

Epinions is a good testbed that is widely used in the research of trust evaluation and trust-based recommendation [Massa and Avesani 2007a] because it has both information on user trust relationships and user-item ratings.[3] Users can review items and

[2]http://en.wikipedia.org/wiki/Root-mean-square_deviation.
[3]Unfortunately, eBay has discontinued allowing users to write reviews or rate other reviews, beginning in March 2014.

assign them numeric ratings in the range of [1, 5]. They can also build their own trust network by adding people whose reviews they think are valuable. One dataset from Epinions.com is published by Massa [Massa and Avesani 2006; Trustlet 2012]. [Tang 2012] published a dataset of Epinions with more information, including time stamps. The dataset of a very similar website, Ciao, is also provided in [Tang 2012]. Richardson et al. [2003] and Leskovec and Krevl [2014] also provide datasets for Epinions.

FilmTrust was developed by Golbeck [2005] for the testing and validation of trust models. It is a website on which users are encouraged to write reviews and provide ratings on movies or others' reviews. Each user of the website can derive personalized movie ratings based on his friends' ratings. Trust is also used to sort reviews.

7.3. Applications

Trust evaluation has many applications: In the survey of Grandison and Sloman [2000], the authors describe some influential examples of trust management, including information retrieval systems, medical information systems, and mobile code. In Jøsang et al. [2007], the authors introduce several real applications, including discussion fora (e.g., Slashdot), product review sites (e.g., Epinions), and expert sites (e.g., AllExperts (www.allexperts.com), Advogato [Levien 2003]). In this article, we introduce two of the most popular trust-based applications: trust-based/trust-incorporated security management and trust-based recommendation.

—**Trust-based/trust-incorporated security management.** Trust management is developed as an answer to the inadequacy of traditional authorization mechanisms [Blaze et al. 1999]. It is regarded as a "soft security" mechanism that uses collaborative methods to assess members' behaviors, identify those who behave well or not well, and provide corresponding measures [Jøsang et al. 2007]. Examples include the famous e-commerce websites (Taobao[4] in China, Amazon[5] in the US, etc.). Pretty Good Privacy (PGP) is the first system to use the term "Web of Trust." It is a program for data encryption and decryption, and it can enhance the security of data communication [Heinrich 2011]. Several important works have been proposed to detect and defend against sybil attack, such as SybilGuard [Yu et al. 2006], SybilLimit [Yu et al. 2008], SybilInfer [Danezis and Mittal 2009], SumUp [Tran et al. 2009], and so on. In addition, Wang et al. [2013] propose a trust-based framework (called ARTSense) and apply it in participatory sensing networks.

—**Trust-based recommendation.** One important application of trust is in a recommendation system, where users take advice from friends. A recommendation system is based on trust propagation [Amatriain 2012; Machanavajjhala et al. 2011; Yang et al. 2011a, 2011b] and is widely used in OSNs. Such a system includes two essential components: the *rating* (or opinion) of a user on an item and the *influence* of a user on another user when he recommends an item [Zhu et al. 2012; Yang et al. 2012; Bakshy et al. 2012]. Here, an item can represent different things according to the context, including a user, a view, or a real commercial product. Systems that support rating, ranking, and reputation include Amazon and eBay's recommendation systems [Houser and Wooders 2006], and Epinions' web of trust [Tan 2014].

Trust-based recommendation systems aim to produce recommendations for individuals based on the opinions of trusted friends [Ma et al. 2015]. This approach can help to solve the sparsity issue of other approaches and provide more suitable recommendations. The challenge of designing an appropriate trust-based recommendation model is that of efficiently capturing time-evolving ratings and influences from users

[4]http://www.taobao.com/.
[5]http://www.amazon.com/.

to items and from users to users, respectively. That is, we need to show how recommendations are propagated through influences among trusted users, which are highly personalized.

Almost all trust-based recommendation systems are working with a trusted graph. We briefly review some representative works in this area, again following the two categories of graph simplification- and graph analogy-based approaches.

—*Graph Simplification-based Recommendation*. Massa and Avesani [2007a] study recommendation systems and propose using trust relations to improve the quality of recommendation. They conduct experiments on the Epinions dataset, and the results show the advantage of their method over the traditional collaborative filtering (CF) approach, especially for cold-start items and users. In addition, Andersen et al. [2008] explore several axioms for trust-based recommendation. They discuss some typical recommendation systems and analyze which set of axioms are suitable for each system.

—*Graph Analogy-based Recommendation*. Jiang et al. [2014, 2015] propose a time-evolving rating prediction scheme in trust-based recommendation systems using fluid dynamic theory. A user is modeled as a container, the trust-based recommendation is modeled as fluid, and the trust relations serve as the pipe connecting containers. Several social and physical principles are proposed or examined in FluidRating, such as "first influence dominates," "stronger influence dominates," "mass conservation" and "energy conservation." A similar approach is used in influence diffusion. For instance, Ma et al. [2008] and Yang et al. [2007] use another physical phenomenon called heat diffusion. Because heat always flows from a position with a high temperature to a position with a low temperature, seeding users are given a high amount of heat that will be diffused to other users.

—**Trust visualization.** Another method of trust dissemination is trust visualization. Generally, this application evolves an overview analysis of the network status in terms of trust. The result can be reported to social service providers. However, it is beyond the scope of this article, since our focus is on personalized trust evaluation.

8. CONCLUSION AND OPEN CHALLENGES

To the best of our knowledge, our work provides the first review that focuses on graph-based trust evaluation in OSNs. We use two categories—graph simplification-based and graph analogy-based approaches—and we discuss their individual problems and challenges. Then we discuss the common challenges of all graph-based models. We also conduct a brief review of the preparation and validation of trust models, including information collection, performance evaluation, and trust-based applications. Finally, we would like to note open challenges for further research that is not restricted to graph-based approaches but may be encountered by any trust model. These are the privacy issue, the standard test bed, and distrust.

Privacy Preservation. The information used by existing trust models is explicitly expressed or at least is assumed to be. In fact, it is not easy to get trust information, since there are fewer people who give trust opinions than there are who engage in online interactions. When trust information can be gained, the privacy issue emerges. Currently, privacy-preserving schemes usually take advantage of encoding techniques. For example, Wang et al. [2013] propose ARTSense to solve the problem of "trust without identity" in participatory sensing networks, in which they use the blind signature technique to protect user privacy. How to reach a balance between trust evidence collection and privacy preservation is very challenging.

Feedback and Test Bed. To test the performance of different trust models, we need feedback from users and standard test beds. However, due to concerns for privacy and the fear of retaliation of negative ratings, there is less feedback than interactions.

As far as we know, there is no commonly accepted evaluation benchmark that would allow for a comparison of OSN trust models under a set of representative and common conditions. Hence, how to collect feedback and develop test beds is very meaningful but challenging.

Distrust. Whereas no trust means we have no idea about someone's trustworthiness, distrust indicates doubt about someone's trustworthiness. Some work (e.g., Guha et al. [2004] and Ziegler and Lausen [2005]) has considered incorporating distrust into trust models, whereas others prefer to deal with the issue separately. The reason is that distrust may not be propagative. Taking Figure 1(b) for instance: If s distrusts u, and u distrusts v, it is not likely that s will distrust v. The intuition behind this is "the enemy's enemy is my friend" [Antal et al. 2006]. In OSNs, distrust has to be carefully designed to avoid retaliatory negative feedback. From this point of view, dealing with distrust is even more challenging than dealing with trust. Tang et al. [2015] investigate a negative link prediction problem with only positive links and content-centric interactions in social media. It indicates that research on distrust can be closely related to trust. Therefore, although closely related in concept and application, the rules of evaluating distrust may be similar to but different from that of trust.

REFERENCES

R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and applications*. Prentice Hall (1993).

A. Altman and M. Tennenholtz. 2005. On the axiomatic foundations of ranking systems. In *Proceedings of International Joint Conference in Artificial Intelligence (IJCAI)*. 917–922.

X. Amatriain. 2012. Mining large streams of user data for personalized recommendations. *SIGKDD Explorations* 14, 2 (2012), 37–48.

R. Andersen, C. Borgs, J. Chayes, U. Feige, A. Flaxman, A. Kalai, V. Mirrokni, and M. Tennenholtz. 2008. Trust-based recommendation systems: An axiomatic approach. In *Proceedings of the 17th ACM International Conference on World Wide Web (WWW)* (2008), 199–208.

T. Antal, P. L. Krapivsky, and S. Redner. 2006. Social balance on networks: The dynamics of friendship and enmity. *Physica D: Nonlinear Phenomena* 224 (2006), 130–136.

E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic. 2012. The role of social networks in information diffusion. In *Proceedings of the 21st ACM International Conference on World Wide Web (WWW)*. 519–528.

M. Blaze, J. Feigenbaum, J. Ioannidis, and A. D. Keromytis. 1999. The role of trust management in distributed systems security. *Secure Internet Programming* (1999), 185–210.

J. Caverleea, L. Liu, and S. Web. 2010. The socialtrust framework for trusted social information management: Architecture and algorithms. *Information Sciences* 180(1) (January 2010), 95–112.

S. Chen, G. Wang, and W. Jia. 2014. k-FuzzyTrust: An efficient trust computation for large-scale mobile social networks using fuzzy implicit social graph. *Elsevier: Information Sciences*. DOI information:10.1016/j.ins.2014.09.058 (2014).

W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincon, X. Sun, Y. Wang, W. Wei, and Y. Yuan. 2011. Influence maximization in social networks when negative opinions may emerge and propagate. In *Proceedings of the 11th SIAM International Conference on Data Mining (SDM)*. 379–390.

W. Chen, C. Wang, and Y. Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proceedings of the 16th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 1029–1038.

W. Chen, Y. Wang, and S. Yang. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 199–208.

J. Cho, K. Chan, and S. Adali. 2015. A survey on trust modeling. *ACM Computing Surveys (CSUR)* 48, 2 (2015), 28.

J. Cho, A. Swami, and I. Chen. 2011. A survey on trust management for mobile ad hoc networks. *IEEE Communications Surveys Tutorials* 13, 4 (2011), 562–583.

J. Cho, A. Swami, and I. Chen. 2012. Modeling and analysis of trust management with trust chain optimization in mobile ad hoc networks. *Journal of Network and Computer Applications* 35(3) (2012), 1001–1012.

B. Cialdini and R. Trost. 1998. *Social Influence: Social Norms, Conformity and Compliance*. The handbook of social psychology, Vol. 1 and 2 (4th ed.). McGraw-Hill.

M. Ciglan, M. Laclavik, and K. Norvag. 2013. On community detection in real-world networks and the importance of degree assortativity. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1007–1015.

Clever. 2014. Homepage. Retrieved from http://www.research.ibm.com/topics/popups/innovate/hci/html/chow.html.

S. Cook, T. Yamagishi, C. Cheshire, R. Cooper, M. Matsuda, and R. Mashima. 2005. Trust building via risk taking: A cross-societal experiment. *Social Psychology Quarterly* 68, 2 (June 2005), 121–142.

D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. 2008. Feedback effects between similarity and social influence in online communities. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 160–168. http://doi.acm.org/10.1145/1401890.1401914

G. Danezis and P. Mittal. 2009. SybilInfer: Detecting sybil nodes using social networks. In *Proceedings of the 2009 Network and Distributed System Security Symposium (NDSS)*.

A. Das, S. Gollapudi, R. Panigrahy, and M. Salek. 2013. Debiasing social wisdom. In *Proceedings of the 19th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 500–508.

A. Dempster. 1967. Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics* 38, 2 (1967), 325–339.

J. Douceur. 2002. The sybil attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems(IPTPS'01)*. 251–260.

E. ElSalamouny, K. Krukow, and V. Sassone. 2009. An analysis of the exponential decay principle in probabilistic trust models. *Theoretical Computer Science* 410, 41 (2009), 4067–4084.

E. ElSalamouny, V. Sassone, and M. Nielsen. 2010. HMM-based trust model. *Formal Aspects in Security and Trust, Springer Berlin* (2010), 21–35.

L. Ford and R. Fulkerson. 1962. *Flows in Networks*. Princeton University Press, Princeton, NJ (1962).

D. Gefen. 2002. Reflections on the dimensions of trust and trustworthiness among online consumers. *SIGMIS Database* 33, 3 (2002), 38–53. http://doi.acm.org/10.1145/569905.569910

D. Gefen and P. A. Pavlou. 2012. The boundaries of trust and risk: The quadratic moderating role of institutional structures. *Information Systems Research* 23, 3-part-2 (2012), 940–959.

S. Goel, W. Mason, and D. J. Watts. 2010. Real and perceived attitude agreement in social networks. *Journal of Personality and Social Psychology* 99(4) (2010), 611–621.

J. Golbeck. 2005. Computing and applying trust in web-based social networks. PhD Dissertation, University of Maryland (2005).

J. Golbeck and J. Hendler. 2006. Inferring binary trust relationships in web-based social networks. *ACM Transactions on Internet Technology (TOIT)* 6, 4 (2006), 497–529.

J. Goldstein, A. Kwasinksi, P. Kingsbury, R. Sabin, and A. McDowell. 2006. Annotating subsets of the Enron email corpus. In *Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS)*.

D. Good. 1988. Individuals, interpersonal relations and trust. In *Trust: Making and Breaking Cooperative Relations*, D. Gambetta (Ed.). chapter 3 (1988), 31–48, NewYork: Basil Black.

T. Grandison and M. Sloman. 2000. A survey of trust in internet applications. *IEEE Communications Surveys and Tutorials* 3, 4 (Oct. 2000), 2–16. DOI:http://dx.doi.org/10.1109/COMST.2000.5340804

M. Granovetter. 1983. The strength of weak ties: A network theory revisited. *Sociological Theory* 1 (1983), 201–233.

R. Guha. 2003. *Open Rating Systems*. Technical Report. Stanford Knowledge Systems Laboratory, Stanford, CA.

R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. 2004. Propagation of trust and distrust. In *Proceedings of the 13th ACM International Conference on World Wide Web (WWW)*. 403–412.

Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. 2004. Combating web spam with trustrank. In *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB)*. 576–587.

C. Hang, Z. Zhang, and M. P. Singh. 2013. Shin: Generalized trust propagation with limited evidence. *Computer* 46, 3 (2013), 78–85.

S. Hauke, S. Biedermanny, M. Muhlhauser, and D. Heider. 2013. On the application of supervised machine learning to trustworthiness assessment. In *Proceedings of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2013), 525–534.

C. Heinrich. 2011. Pretty good privacy (PGP). In *Encyclopedia of Cryptography and Security (2nd Ed.)*, T. Van, C. A. Henk, and S. Jajodia (Eds.). Springer, 955–958.

K. Hoffman, D. Zage, and C. Nita-Rotaru. 2009. A survey of attack and defense techniques for reputation systems. *Computer Surveys* 42, 1 (2009), 1–31.

D. Houser and J. Wooders. 2006. Reputation in auctions: Theory and evidence from eBay. *Journal of Economics & Management Strategy* 15, 2 (2006), 353–369.

F. Huang. 2007. Building social trust: A human-capital approach. *Journal of Institutional and Theoretical Economics (JITE)* 163, 4 (2007), 552–573.

T. D. Huynh. 2009. A personalized framework for trust assessment. In *Proceedings of the 2009 ACM Symposium on Applied Computing (SAC)* (2009), 1302–1307.

W. Jiang and G. Wang. 2011. SWTrust: Generating trusted graph for trust evaluation in online social networks. In *Proceedings of the 2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 320–327.

W. Jiang, G. Wang, and J. Wu. 2014. Generating trusted graphs for trust evaluation in online social networks. *Future Generation Computer Systems* 31 (2014), 48–58.

W. Jiang and J. Wu. 2014. Trust models in wireless sensor networks and online social networks: A comparative study. In *Proceedings of the 11th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. 612–617.

W. Jiang, J. Wu, F. Li, G. Wang, and H. Zheng. 2016. Trust evaluation in online social networks using generalized flow. *IEEE Transactions on Computers* 65(3) (2016), 952–963.

W. Jiang, J. Wu, and G. Wang. 2013. RATE: Recommendation-aware trust evaluation in online social networks. In *Proceedings of the 12th IEEE International Symposium on Network Computing and Applications (NCA)*. 149–152.

W. Jiang, J. Wu, and G. Wang. 2015. On selecting recommenders for trust evaluation in online social networks. *ACM Transactions on Internet Technology (TOIT)* 15(4) (2015), Article 14.

W. Jiang, J. Wu, G. Wang, and H. Zheng. 2014. FluidRating: A time-evolving rating prediction in trust-based recommendation systems using fluid dynamics. In *Proceedings of the 33rd IEEE International Conference on Computer Communications (INFOCOM)* (2014), 1707–1715.

W. Jiang, J. Wu, G. Wang, and H. Zheng. 2016. Forming opinions via trusted friends: Time-evolving rating prediction using fluid dynamics. *IEEE Transactions on Computers* 65, 4 (2016), 1211–1224. DOI:10.1109/TC.2015.2444842

A. Jøsang. 1999. An algebra for assessing trust in certification chains. In *Proceedings of the Network and Distributed Systems Security Symposium (NDSS)*.

A. Jøsang, U. G. Center, and T. Bhuiyan. 2008. Optimal trust network analysis with subjective logic. In *Proceedings of the Second International Conference on Emerging Security Information, Systems and Technologies* (2008), 179–184.

A. Jøsang, E. Gray, and M. Kinateder. 2006a. Simplification and analysis of transitive trust networks. *Web Intelligence and Agent System* 4 (2) (2006), 139–161.

A. Jøsang, R. Hayward, and S. Pope. 2006b. Trust network analysis with subjective logic. In *Proceedings of the 29th Australasian Computer Science Conference (ACSC)*. 85–94.

A. Jøsang, R. Ismail, and C. Boyd. 2007. A survey of trust and reputation systems for online service provision. *Decision Support Systems* 43, 2 (2007), 618–644.

A. Jøsang and S. Pope. 2012. Dempster's rule as seen by little colored balls. *Computational Intelligence* 28, 4 (2012), 453–474.

B. R. Julian.1967. A new scale for the measurement of interpersonal trust. *Journal of Personality* 4 (1967), 651–665.

S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. 2003. The eigentrust algorithm for reputation management in P2P networks. In *Proceedings of the 12th ACM International Conference on World Wide Web (WWW)*. 640–651.

D. Kempe, J. Kleinberg, and É. Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 137–146.

Y. A. Kim and H. S. Song. 2011. Strategies for predicting local trust based on trust propagation in social networks. *Knowledge-Based Systems* 24(8) (2011), 1360–1371.

J. M. Kleinberg. 1999. Hubs, authorities, and communities. *Computer Surveys* 31, 4es, Article 5 (1999).

U. Kuter and J. Golbeck. 2007. Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models. In *Proceedings of the 22nd National Conference on Artificial Intelligence* (2007), 1377–1382.

P. M. Lee. 2012. *Bayesian Statistics: An Introduction*. John Wiley & Sons (2012).

M. Lesani and N. Montazeri. 2009. Fuzzy trust aggregation and personalized trust inference in virtual social networks. *Computational Intelligence* 25 (2) (2009), 51–83.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. 2010a. Predicting positive and negative links in online social networks. In *Proceedings of the 19th ACM International Conference on World Wide Web*. 641–650.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. 2010b. Signed networks in social media. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*. 1361–1370.

J. Leskovec and A. Krevl. 2014. SNAP: {Stanford} Large Network Dataset Collection. https://snap.stanford.edu/data/soc-Epinions1.html. (2014).

R. Levien. 2003. Attack resistant trust metrics. PhD thesis, UC Berkeley, Berkeley, CA. (2003).

R. Levien and A. Aiken. 1998. Attack-resistant trust metrics for public key certification. In *Proceedings of the 7th USENIX Security Symposium* (1998), 229–242.

F. Li and J. Wu. 2010. Uncertainty modeling and reduction in MANETs. *IEEE Transactions on Mobile Computing* 9, 7 (2010), 1035–1048.

Y. Li. 1998. Toward a qualitative search engine. *IEEE Internet Computing* 2, 4 (1998), 24–29.

C. Lin, N. Cao, S. Liuand S. Papadimitriou, J. Sun, and X. Yan. 2009a. Smallblue: Social network analysis for expertise search and collective intelligence. In *Proceedings of the 25th International Conference on Data Engineering (ICDE)*. 1483–1486.

H. Lin, X. Wu, and H. Lin. 2009b. Hierarchical fuzzy trust management for peer-to-peer network. In *ISECS International Colloquium on Computing, Communication, Control, and Management (CCCM)*, Vol. 4. 377–380.

G. Liu, Y. Wang, M. A. Orgun, and E. Lim. 2013b. Finding the optimal social trust path for the selection of trustworthy service providers in complex social networks. *IEEE Transactions on Services Computing* 6(2) (2013), 152–167.

G. Liu, Q. Yang, H. Wang, X. Lin, and M. Wittie. 2014. Assessment of multi-hop interpersonal trust in social networks by three-valued subjective logic. In *Proceedings of the 33rd IEEE International Conference on Computer Communications (INFOCOM)*. 1698–1706.

X. Liu, A. Datta, and E. Lim (Eds.). 2014. *Computational Trust Models and Machine Learning*. Chapman and Hall/CRC.

X. Liu, G. Tredan, and A. Datta. 2013a. A generic trust framework for large-scale open systems using machine learning. *Computational Intelligence* (2013).

H. Ma, H. Yang, M. R. Lyu, and I. King. 2008. Mining social networks using heat diffusion processes for marketing candidates selection. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*. 233–242.

T. Ma, J. Zhou, M. Tang, Y. Tian, A. Al-Dhelaan, M. Al-Rodhaan, and S. Lee. 2015. Social network and tag sources based augmenting collaborative recommender system. *IEICE Transactions on Information and Systems* E98-D, 4, 902–910.

A. Machanavajjhala, A. Korolova, and A. Sarma. 2011. Personalized social recommendations: Accurate or private. In *Proceedings of the 37th International Conference on Very Large Data Bases (VLDB)*. 440–450.

M. Maghami and G. Sukthankar. 2013. Hierarchical influence maximization for advertising in multi-agent markets. In *Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 21–27.

G. Mahoney, W. Myrvold, and G. C. Shoja. 2005. Generic reliability trust model. In *Proceedings of the 3rd Annual Conference on Privacy, Security and Trust (PST)* (2005), 113–120.

F. G. Marmol and G. M. Perez. 2010. Towards pre-standardization of trust and reputation models for distributed and heterogeneous systems. *Computer Standards and Interfaces* 32 (2010), 185–196.

S. Marsh. 1994. Formalising trust as a computational concept. PhD Thesis, Department of Mathematics and Computer Science, University of Stirling, Stirling, UK (1994).

S. Marti and H. Garcia-Molina. 2006. Taxonomy of trust: Categorizing P2P reputation systems. *Computer Networks* 50(4) (March 2006), 472–484.

P. Massa and P Avesani. 2006. Trust-aware bootstrapping of recommender systems. In *Proceedings of the ECAI Workshop on Recommender Systems* (2006), 29–33.

P. Massa and P. Avesani. 2007a. Trust-aware recommender systems. In *Proceedings of the 2007 ACM Conference on Recommender Systems (RecSys)*. 17–24.

P. Massa and P. Avesani. 2007b. Trust metrics on controversial users: Balancing between tyranny of the majority and echo chambers. *International Journal on Semantic Web and Information Systems* 3 (2007), 39–64.

P. Massa, P. Avesani, and R. Tiella. 2005. A trust-enhanced recommender system application: Moleskiing. In *Proceedings of the 2005 ACM Symposium on Applied Computing (SAC)*. 1589–1593.

P. Massa and B. Bhattacharjee. 2004. Using trust in recommender systems: An experimental analysis. In *Trust Management*. Vol. 2995. 221–235.

A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee. 2007. Measurement and analysis of online social networks. In *Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC)* (2007), 29–42.

A. Mislove, A. Post, P. Druschel, and K. Gummadi. 2008. Ostra: Leveraging trust to thwart unwanted communication. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. USENIX Association, Berkeley, CA, 15–30.

A. Molavi Kakhki, C. Kliman-Silver, and A. Mislove. 2013. Iolaus: Securing online content rating systems. In *Proceedings of the 22nd International Conference on World Wide Web (WWW)*. 919–930.

G. Möllering. 2001. The nature of trust: From Georg Simmel to a theory of expectation, interpretation and suspension. *Sociology* 35, 2 (2001), 403–420.

L. Mui, M. Mohtashemi, and A. Halberstadt. 2002. A computational model of trust and reputation for e-businesses. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS)*. 2431–2439.

K. Newton. 2001. Trust, social capital, civil society, and democracy. *International Political Science Review* 22, 2 (2001), 201–214.

D. Q. Nguyen, T. Kunz, and L. Lamont. 2012. Impact of mobility on trust decay rate. In *Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC)* (2012), 2852–2857.

M. Nielsen, K. Krukow, and V. Sassone. 2007. A Bayesian model for event-based trust. *Festschrift in Hounour of Gordon Plotkin. Electronic Notes in Theoretical Computer Science* (2007).

T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu. 2013. Trust management of services in cloud environments: Obstacles and solutions. *Computer Surveys* 46(1), 4 (2013), 12:1–12:30.

Z. Noorian and M. Ulieru. 2010. The state of the art in trust and reputation systems: A framework for comparison. *Journal of Theoretical and Applied Electronic Commerce Research* 5, 2 (Aug. 2010), 97–117.

L. Page, S. Brin, R. Motwani, and T. Winograd. 1999. *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report 1999-66. Stanford InfoLab.

J. Pearl. 1999. Reasoning with cause and effect. In *Proceedings of International Joint Conference in Artificial Intelligence (IJCAI)*. 1437–1449.

S. Peng, W. Jia, G. Wang, J. Wu, and M. Guo. 2010. Trusted routing based on dynamic trust mechanism in mobile ad-hoc networks. *IEICE Transactions on Information and Systems* 93-D (2010), 510–517.

S. Peng, A. Yang, H. Zhong, and Z. Feng. 2013. A dynamic trust updating model based on multiple constraints in wireless mesh networks. *International Conference on Information Science and Technology (ICIST)* (2013), 815–819.

I. Pinyol and J. Sabater-Mir. 2013. Computational trust and reputation models for open multi-agent systems: A review. *Artificial Intelligence Review* 40, 1 (2013), 1–25.

G. Qi, C. Aggarwal, and T. Huang. 2012. Community detection with edge content in social media networks. In *Proceedings of the IEEE 28th International Conference on Data Engineering (ICDE)*. 534–545.

H. Quan, J. Wu, and Y. Shi. 2011. Online social networks and social network services: A technical survey. *Handbook of Pervasive Communication*, M. Ilyas and H. Mouftah (Eds.). CRC Press.

L. Rashotte. 2007. Social influence. A. S. R. Manstead, M. Hewstone (Eds.), *The Blackwell Encyclopedia of Social Psychology*, Blackwell Publishing, 562–563.

P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. 2000. Reputation systems. *Communications of the ACM* 43, 12 (2000), 45–48.

P. Resnick and R. Zeckhauser. 2002. Trust among strangers in internet transactions: Empirical analysis of eBay's reputation system. In *The Economics of the Internet and E-Commerce*, M. R. Baye (Ed.). Vol. 11. JAI Press Inc., 127–157.

M. Richardson, R. Agrawal, and P. Domingos. 2003. Trust management for the semantic web. In *Proceedings of the 2nd International Semantic Web Conference (ISWC)*. 351–368.

S. Ruohomaa and L. Kutvonen. 2005. Trust management survey. In *Proceedings of the 3rd International Conference on Trust Management* (2005), 77–92.

J. Sabater and C. Sierra. 2002. Reputation and social network analysis in multi-agent systems. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS).* (2002), 475–482.

J. Sabater and C. Sierra. 2005. Review on computational trust and reputation models. *Artificial Intelligence Review* 24(1) (2005), 33–60.

M. M. Sathik, K. S. Kannan, and A. Rasheed. 2011. Comparative analysis of community discovery methods in social networks. *International Journal of Computer Applications* 14(8) (2011), 27–31.

G. Shafer. 1976. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton.

W. Sherchan, S. Nepal, and C. Paris. 2013. A survey of trust in social networks. *Computer Surveys* 45, 4 (2013), 47:1–47:33.

A. Singh and L. Liu. 2003. TrustMe: Anonymous management of trust relationships in decentralized P2P systems. In *Proceedings of the 3rd IEEE International Conference on P2P Computing*. 142–149.

A. Srinivasan, J. Teitelbaum, H. Liang, J. Wu, and M. Cardei. 2008. Reputation and trust-based systems for ad hoc and sensor networks. *Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks*, A. Boukerche (Ed.), Wiley, ISBN: 978-0-470-38358-2 (2008).

G. Stringhini, C. Kruegel, and G. Vigna. 2010. Detecting spammers on social networks. In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*. 1–9.

Y. Sun, W. Yu, Z. Han, and K. Liu. 2006b. Information theoretic framework of trust modeling and evaluation for ad hoc networks. *IEEE Journal on Selected Areas in Communications* 24, 2 (2006), 305–317.

Y. L. Sun, Z. Han, W. Yu, and K. J. R. Liu. 2006a. A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks. In *Proceedings of IEEE INFOCOM* (2006), 1–13.

M. Taherian, M. Amini, and R. Jalili. 2008. Trust inference in web-based social networks using resistive networks. In *Proceedings of the 3rd International Conference on Internet and Web Applications and Services (ICIW)*. 233–238.

J. Tang. 2012. Product Review Datasets: Epinions and Ciao. http://www.public.asu.edu/~jtang20/datasetcode/truststudy.htm.

J. Tang, H. gao, H. Liu, and A. Das Sarma. 2012. eTrust: Understanding trust evolution in an online world. In *Proceedings of the 18th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 253–261.

J. Tang, S. Chang, C. Aggarwal, and H. Liu. 2015. Negative link prediction in social media. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM)*. 87–96.

L. Terveen and W. Hill. 2001. Beyond recommender systems: Helping people help each other. In *HCI in the New Millennium*. 487–509.

G. Theodorakopoulos and J. S. Baras. 2006. On trust models and trust evaluation metrics for ad hoc networks. *IEEE Journal on Selected Areas in Communications* 24 (2006), 318–328.

D. N. Tran, B. Min, J. Li, and L. Subramanian. 2009. Sybil-resilient online content voting. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, Vol. 9. 15–28.

Trustlet. 2012. Extended Epinions Trust Network Dataset: http://www.trustlet.org/extended_epinions.html. (2012).

Trustlet. 2014. Index of Advogato Dataset: http://www.trustlet.org/datasets/advogato/. (2014).

B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove. 2011. An analysis of social network-based sybil defenses. *ACM SIGCOMM Computer Communication Review* 41, 4 (2011), 363–374.

G. Wang, W. Jiang, J. Wu, and Z. Xiong. 2014. Fine-grained feature-based social influence evaluation in online social networks. *IEEE Transactions on Parallel and Distributed Systems* 25, 9 (2014), 2286–2296.

G. Wang and J. Wu. 2011a. FlowTrust: Trust inference with network flows. *Frontiers of Computer Science in China* 5(2) (2011), 181–194.

G. Wang and J. Wu. 2011b. Multi-dimensional evidence-based trust management with multi-trusted paths. *Future Generation Computer Systems (Elsevier)* 27(5) (2011), 529–538.

X. Wang, W. Cheng, P. Mohapatra, and T. Abdelzaher. 2013. ARTSense: Anonymous reputation and trust in participatory sensing. In *Proceedings of the 32nd IEEE International Conference on Computer Communications (INFOCOM)*. 2517–2525.

D. J. Watts. 1999. *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton University Press (1999).

J. Wu. 2009. Trust mechanisms and their applications in MANETs. *Keynote speech in IEEE TrustCom'09.*

H. Xia, Z. Jia, L. Ju, X. Li, and Y. Zhu. 2011. A subjective trust management model with multiple decision factors for MANET based on AHP and fuzzy logic rules. In *Proceedings of the 2011 IEEE/ACM International Conference on Green Computing and Communications (GreenCom)*. 124–130.

H. Yang, I. King, and M. Lyu. 2007. DiffusionRank: A possible penicillin for web spamming. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 431–438.

S. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. 2011a. Like like alike: Joint friendship and interest propagation in social networks. In *Proceedings of the 20th ACM International Conference on World Wide Web (WWW)*. 537–546.

S. Yang, B. Long, A. Smola, H. Zha, and Z. Zheng. 2011b. Collaborative competitive filtering: Learning recommender using context of user choice. In *Proceedings of the 34th International ACM Conference on Research and Development in Information Retrieval (SIGIR)*. 295–304.

X. Yang, H. Steck, and Y. Liu. 2012. Circle-based recommendation in online social networks. In *Proceedings of the 18th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. 1267–1275.

Y. Yao, S. Ruohomaa, and F. Xu. 2012a. Addressing common vulnerabilities of reputation systems for electronic commerce. *Journal of Theoretical and Applied Electronic Commerce Research* 7, 1 (April 2012), 1–20.

Y. Yao, H. Tong, F. Xu, and J. Lu. 2012b. Subgraph extraction for trust inference in social networks. In *ASONAM*. IEEE Computer Society, 163–170.

Y. Yao, H. Tong, X. Yan, F. Xu, and J. Lu. 2013. MATRI: A multi-aspect and transitive trust inference model. In *Proceedings of the 22nd ACM International Conference on World Wide Web (WWW)*. 1467–1476.

J. S. Yedidia, W. T. Freeman, and Y. Weiss. 2001. Generalized belief propagation. *Advances in Neural Information Processing Systems 13*, eds. T. K. Leen, T. G. Dietterich, and V. Tresp, MIT Press (2001).

J. S. Yedidia, W. T. Freeman, and Y. Weiss. 2002. *Understanding Belief Propagation and its Generalizations*. Technical Report. Mitsubishi Electric Research Laboratories. 239–269.

H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. 2008. Sybillimit: A near-optimal social network defense against sybil attacks. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*. 3–17.

H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. 2006. SybilGuard: Defending against sybil attacks via social networks. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*. 267–278.

H. Yu, Z. Shen, C. Miao, C. Leung, and D. Niyato. 2010. A survey of trust and reputation management systems in wireless communications. *Proceedings of the IEEE* 98(10) (2010), 1755–1772.

Y. Yu, K. Li, W. Zhou, and P. Li. 2012. Trust mechanisms in wireless sensor networks: Attack analysis and countermeasures. *Journal of Network and Computer Applications* 35, 3 (2012), 867–880. Special Issue on Trusted Computing and Communications.

W. Yuan, D. Guan, and Y.-K. Lee. April 2010. Improved trust-aware recommender system using small-worldness of trust networks. *Knowledge-Based Systems* 23 (April 2010), 232–238.

H. Zhang, T. N. Dinh, and M. T. Thai. 2013. Maximizing the spread of positive influence in online social networks. In *Proceedings of the 33rd IEEE International Conference on Distributed Computing Systems (ICDCS)*. 317–326.

H. Zhu, B. A. Huberman, and Y. Luon. 2012. To switch or not to switch: Understanding social influence in online choices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2257–2266.

C. Ziegler and G. Lausen. 2005. Propagation models for trust and distrust in social networks. *Information Systems Frontiers* 7, 4–5 (2005), 337–358.

Y. Zuo, W.-C. Hu, and T. O'Keefe. 2009. Trust computing for social networking. In *Proceedings of the 6th International Conference on Information Technology: New Generations (ITNG)*. 1534–1539.

Received December 2014; revised November 2015; accepted February 2016