2015 IEEE Trustcom/BigDataSE/ISPA

Trust-Aware Review Spam Detection

Hao Xue*, Fengjun Li*, Hyunjin Seo†, and Roseann Pluretti†
*EECS Department, †School of Journalism and Mass Communications
The University of Kansas
Lawrence, KS, USA

***Abstract*—Online review systems play an important role in affecting consumers' behaviors and decision making, attracting many spammers to insert fake reviews to manipulate review content and ratings. To increase utility and improve user experience, some online review systems allow users to form social relationships between each other and encourage their interactions. In this paper, we aim at providing an efficient and effective method to identify review spammers by incorporating social relations based on two assumptions that people are more likely to consider reviews from those connected with them as trustworthy, and review spammers are less likely to maintain a large relationship network with normal users. The contributions of this paper are two-fold: (1) We elaborate how social relationships can be incorporated into review rating prediction and propose a trust-based rating prediction model using proximity as trust weight; and (2) We design a trust-aware detection model based on rating variance which iteratively calculates user-specific overall trustworthiness scores as the indicator for spamicity. Experiments on the dataset collected from Yelp.com show that the proposed trust-based prediction achieves a higher accuracy than standard CF method, and there exists a strong correlation between social relationships and the overall trustworthiness scores.**

I. Introduction

Online review systems are getting increasingly popular recently. For example, Yelp.com, known mainly for restaurant reviews, had a total of 71 million reviews of businesses and a monthly average of 135 million unique visitors to the site as of 2014, and TripAdvisor.com, specializing in travel-related services, reached 315 million unique monthly visitors and over 200 million reviews until 2014. With a large volume of information and opinions regarding products or services available on these platforms, people tend to read the reviews before making purchasing decisions and their behaviors and decisions may be significantly affected by others' opinions. Typically driven by financial motivations, individuals or a group of individuals utilize deceptive reviews to maliciously promote or demote a product or a service.

Many approaches have been proposed to identify or alleviate review spamming. For example, Amazon marks reviews written by those who actually bought the products with an "Amazon Verified Purchase" tag and Yelp.com filters out suspicious reviews with heuristic rules. However, due to the subjective nature of the reviews, user idiosyncrasies, and high-dimensional user features in the context of the review, spam review detection still remains a challenging problem. The severity of this problem has attracted attention of researchers in recent years. Since [1], many learning-based approaches have been proposed to identify fake reviews and/or spam reviewers from textual features [1], [2], temporal features [3], [4], individual or group behavior patterns of spammers [5]–[7], and sentiment disparities [8]. These approaches perform well in detecting poorly-composed spam reviews such as empty, duplicate, irrelevant reviews or advertisements.

However, changes have been observed in spammers' behaviors, which indicates that spammers are evolving to avoid being detected. This makes rule-based and behavior-based methods less effective. For example, spammers may imitate the writing pattern of regular reviewers to generate less suspicious content. Previous methods that are based on textual similarity would fail in detecting such well-written fake reviews. What makes things worse is the emergence of professional spammers who perform spamming activities in a well-organized way, such as recruiting human workers from crowdsourcing platforms to generate fake reviews. Moreover, for fast and effective manipulation, spammers may control a large number of accounts or work in groups to insert bogus reviews in a short period of time. To avoid being detected, the accounts may be used only for limited times, which makes it difficult to distinguish them from socially "lazy" users who submitted very a few reviews. While sophisticated approaches that take a large set of features into account may be able to address these challenges, they are susceptible to user subjective bias and individual idiosyncrasies.

Inspired by the structural analysis [9], [10], we propose to utilize the social relationships among users in the review systems. Many online review systems encourage interactions among their users. For example, *Yelp.com* and *Last.fm* allow registered users to form friendships; *Amazon* supports sending "helpfulness" tags to reviews that a user finds useful; *Epinions* and *Ciao* allow a user to add others in a trust list. We argue that spam reviewers, while trying their best to pretend as genuine users, still behave abnormally in large-scale social interactions in general. Recruited with monetary incentives, they are reluctant to devote a large amount of effort that is typically required in social interactions. As a result, spammers tend to be more isolated in the social graphs than regular active users.

Moreover, users tend to trust those who are socially connected with them more than strangers, indicating a correlation between trust and social relationship strength among users. Since the primary objective of review spamming is rating manipulation, rating variance metrics seem to be effective in distinguishing spam reviews from regular reviews. However, due to the data sparsity problem that is typical in review systems, the straightforward adoption of rating deviation based detection mechanisms performs poorly. From this consideration, we propose to first utilize social relations to predict "trustworthy" ratings for items that a user has not yet reviewed. Then, the predicted ratings will be incorporated into a model that evaluates the quality of reviews according to rating vari-

978-1-4673-7952-6/15 $31.00 © 2015 IEEE
DOI 10.1109/Trustcom.2015.440

ances.

Rating prediction is a typical problem in recommender systems. For instance, collaborative filtering methods are commonly used to predict ratings or preferences that a user would give to target items. Moreover, trust propagation and trust networks are used in making trustworthy predictions [11], while trust, reputation, and similarity are combined in [12] to improve the quality of recommendations. Recent trust-aware recommendation approaches take users' trust relations into account and incorporate social relationships among users to improve traditional recommendation systems [13]–[16]. While a few systems let users to explicitly express the perceived trust about other users, e.g., *Epinions* allows a user to add another user to her trust list if she likes or agrees with the review issued by this user, most of them provide indirect mechanisms for inferring the trust, e.g., the "helpfulness" votes that a user gives to reviews when the content is considered as useful. Our goal is different from these approaches since we define the concept of trust as the belief of a user in a review that it is not a spam. Therefore, we rely on social ties that indicate strong trust relationships than review quality or prediction accuracy.

In summary, in this paper, we propose a trust-based rating prediction method by applying random walk with restarts in the social graph to compute the proximity between users. The predicted ratings combined with original ratings form a pseudo user-item matrix, which is further used to compute a trustworthiness score for every user for determining if the user is a suspicious spammer. Our contributions are summarized as: (1) We propose a method based on random walk with restart to utilize two social relationships in rating prediction. (2) We propose an iterative model to calculate the trustworthiness of each user based on rating behaviors and trustworthy predictions. And (3) we analyze the relationship between user's trustworthiness and social relationships in the rating system and show a strong correlation exists between the two.

II. Related Work

A. Review spam detection

The problem of spam review was first studied in [1]. A supervised learning method was applied to identify duplicate fake reviews. Later on, various methods have been proposed based on patterns or behaviors to identify suspicious users, such as correlated temporal anomalies [4], rating behavior [5], and unexpectedness [17]. These methods only work if the behavior patterns of spammers remain unchanged. However, once the spammer is aware of these detection mechanisms, he would react quickly to change his behavior to make behavior-based models less effective.

Machine learning techniques have been applied in detection of spamming, both classification [18] and clustering [7] are used. The features chosen are based on subjective observations, which would introduce bias in the detection. So, aspects other than behavior should be taken into account. In [6], an iterative structure-based model was proposed by calculating three intertwined scores for reviewer, review, and store respectively. However, they treated all links equally and didn't consider features on the link. Our approach, on the other hand, not only takes users' behaviors like rating variance into consideration, but also uses relationship strengths as the weights on the links for spamming detection. Besides, we adopt the rating prediction technique in recommendation systems to incorporate trust into our detection framework.

B. Recommender systems

Recommender systems provide suggestions about suitable items for users based on their previous preferences and rating behaviors. Among the various models developed, collaborative filtering (CF) [19] has been the most successful one. Typically, CF methods can be divided into two categories, model-based and memory-based. In model-based methods, machine learning techniques [20], [21] are applied on data related to users' behaviors to learn models for predictions. Memory-based models work under the assumption that users who agree in the past will also agree in the future. Similarities are usually calculated using cosine similarity or Pearson Correlation. For a particular user, a missing rating is calculated by aggregating ratings from k most similar users.

However, the standard CF method suffers from the sparsity problem and performs poorly for cold-start users, who newly joined the system and have few review history. Also, standard CF method is not attack-resistant. Because of the existence of malicious users and spamming activities, various approaches seek to incorporate trust into the recommender system to improve the quality and trustworthiness of predictions. Trust and reputation were first integrated in [12], but its trust and reputation purely rely on ratings, which limits the effectiveness. Trust propagation was taken into account in [11], [22]. However, these methods still rely on user-user similarity and thus cannot yield accurate results for cold-start users. Our method overcomes the sparsity problem by taking social relationships into account to measure users' closenesses. For users without much review history, our approach is still able to make trustworthy predictions as long as social relationships exist. In this work, user-user relationships can be represented using graphs.

C. Random walk

The formalization of a sequence of some random steps on a graph or a web is a random walk. The relations among items and users can be represented using graphs where objects and relationships are represented as nodes and weighted edges (directed or undirected) respectively. Thus, similarities and closenesses of two nodes can be measured using transition probabilities by applying random walks on graphs [23].

Several researches applied this idea on recommender systems. [20] proposed a random walk method to capture the transitive similarities along the item-item matrix to alleviate the sparsity problem in CF. However, since the type of items may vary, it is arbitrary to say that the captured transitive similarity would be accurate. A trust-based and item-based model for recommender system was proposed in [24]. It used the ratings of connected nodes directly as recommendations, which introduced bias into the recommendations. A random walk with restarts (RWR) method was proposed in [15] to measure the closeness between among users, music tracks, and tags for collaborative recommendation. This work showed the effectiveness of modeling closenesses among nodes, but compared with our work, we strengthened the connections

among nodes by incorporating multiple relationships. Also, our model runs more efficiently by focusing on only a partition that contains the starting point and achieve good local approximation.

III. Overview of the Problem and Solution

Nowadays, many review systems provide more rich functionalities than merely rating and review content. Results from an analysis of Yelp.com's reviews showed that when evaluating the usefulness of online reviews, profile information and reputations of users would influence the perceived usefulness [25]. Attributes of users are significantly associated with how reviews are evaluated. Our preliminary study showed that, a review should be detailed, in suitable length, balanced and consistent in order to be perceived as helpful. In terms of credibility, more factors of the user should be taken into account. A credible user should be someone who uses the real name and has a real profile picture. On the profile, the number of friends, the number of compliments received, and whether the user is an "Elite" would affect the judgment. Our study also showed that when assessing reviews, the ones written by a friend would be considered as more trustworthy than the ones from a stranger. In this work, we argue that users with less or none social interactions will be more suspicious to be review spammers than users who are socially active.

In this paper, we focus our work on Yelp.com, but our algorithm can be extended to any social review system with complex user interactions. Yelp aims at building a community rather than merely being a rating platform. It provides rich functionalities to its users. Besides writing text reviews and assigning ratings, users can also upload photos with their reviews or use mobile App to "check-in". Popular and active users can be promoted as "Yelp Elite" as a recognition of role models on and off site. Besides, Yelp encourages users to form social relationships and interact with each other in various means, including following other users, sending compliments to other users, sending private message to other users, and tagging other users' reviews (e.g., cool, funny, and useful), etc. Typically, users of Yelp.com would take efforts to maintain a positive image online. Besides writing faithful reviews, they would also devote time to form social relationships with other users. On the other hand, driven by financial motivation, review spammers are hired to promote or demote their target items. It is natural to think that they devote most of their time into posting fake reviews and would not take much effort interacting with other users online. Thus, they behave very differently from normal users in the aspect of social interactions. However, existing work on spamming detection didn't take this into account.

In this work, we propose an approach that integrates trust-based rating prediction using random walk with restart and rating deviation based iterative model for spam detection. In the problem of spam detection, rating is an important factor. A review's rating about an item reflects its opinion [6]. Nowadays, spam reviews can hardly dominate the system, so the opinions of the majority should reflect the actual quality of an item at certain extent. Conceptually, if a large portion of reviews of a user deviate much from the majority's views, it is reasonable for us to consider this user as a suspicious spammer. Rating deviation has been used as a major feature for spam detection [5]–[7]. However, review data on most review systems is sparse: for many users, the number of reviews is not large enough to derive a stable credibility. In the Yelp.com dataset that we used in our experiment, about 80.59% users wrote less or equal to 2 reviews and about 93.37% users wrote less than 5 reviews. Under this circumstances, it's necessary to find reliable methods to make trustworthy predictions to fill the missing entries in the sparse user-item rating matrix. An effective solution would be applying random walks on graphs, [20], [24] proved the effectiveness, but their models have limitations, which are discussed in the previous section.

IV. Trust-Aware Spam Detection

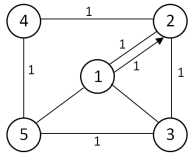
In this section, we describe an iterative model to calculate the overall trustworthiness of all the reviewers in the system and use it as an indicator to determine the likelihood of being a review spammer. More specifically, we first introduce a random walk with restart approach to infer the perceived trustworthiness of one user for another user based on the social relations between them, and then present our trust-based rating prediction model to derive proximity-based predictions to overcome the data sparsity problem. Finally, we elaborate the design of the iterative model to compute an overall trustworthiness score for each user as the spamicity indicator.

A. Inferring trust from social relations

The goal of this work is to detect suspicious content and actions in online review systems with third-party user-generated content (UGC). It is conceivable that social relations among users can be utilized to measure the trustworthiness of a user perceived by others and extend it to the UGC he submits.

Trust-aware recommendation systems or social collaborative recommendation systems are developed based on the assumption that users have similar tastes with other users they trust or connect to. However, this hypothesis may not always be true in the real world. For example, one's friends may have variant opinions about a same item, and thus social regularization is introduced to treat friends differently based on how similar a friend is with the target user [16]. From this aspect, our goal is different from these approaches since we define the concept of trust as the belief of a user in the UGC that it is submitted by a legal reviewer but not a suspicious spammer. Such belief can be generated from the interactions among the users. In particular, we consider two relationships available in our dataset collected from *Yelp.com*: the social *friendship* that often reflects a strong tie between users with mutual and cooperative interactions, and the unilateral *compliment* relationship (similar to up-votes, helpfulness votes, etc. in other online review systems) that does not require a confirmation in the reverse direction. Under our definition of trust, for a target user, the one-way compliment relationship represents an equally trustful relationship as the two-way friendship to other users since it indicates a subjective perception of trust.

Based on these considerations, we propose to represent the inherent relational structure among the users in a graph G and model the trustworthiness that a user i gives to other users as the *proximity* from node i to any other nodes in G. Among various proximity measures, we adopt the random walk with restart (RWR) model to measure the distance between two



(a) An example graph with 5 nodes

	Node 1	Node 2	Node 3	Node4	Node 5
Node 1	0.5671	0.1678	0.1184	0.0486	0.0979
Node 2	0.1248	0.5845	0.1197	0.1030	0.0678
Node 3	0.1379	0.1357	0.5621	0.0453	0.1189
Node 4	0.0624	0.1672	0.0598	0.5515	0.1589
Node 5	0.1248	0.0845	0.1197	0.1030	0.5678

(b) Converged proximity matrix

Fig. 1: Example graph with friendship and compliment relationships and its proximity matrix

nodes since RWR explores the geometry of the graph and takes all the possible paths into account. Moreover, RWR can model the multi-faceted relationship between two nodes, which makes it an ideal proximity measurement for the problem we study in this work. RWR model starts a random walk at node i and computes the proximity of every other node to it. The RWR proximity from node i to node j is the probability that a random walk starting from i reaches j after infinite time, considering that at any transition the random walk will restart at i with a probability α ($0 < \alpha < 1$) or randomly move to another node along the link with a probability (1 - α). From an initial state (denoted as a column vector $\mathbf{q}$), the state of node i at step $k+1$ can be calculated as [15]

$$\mathbf{p}_i^{(k+1)} = (1-\alpha)\mathbf{S}\mathbf{p}_i^{(k)} + \alpha\mathbf{q} \tag{1}$$

where $\mathbf{p}_i^k$ is the *proximity vector* of node i at step k with $p_i(j)^k$ denoting the probability at step k that the random walk is at node j, and $\mathbf{S}$ represents the column normalized transition probability matrix for all nodes in the graph with $S_{i,j}$ denoting the transition probability of moving from a current state at node i to the next state at node j. From an initial state, we recursively apply equation (1) until it converges after l steps. Then, the steady-state transition probability $p_i^l(j)$ represents the proximity from node j to the target user i. Figure 1 illustrates a toy graph of 5 nodes and the proximity matrix computed from the graph. As shown in the figure, a link between two nodes denotes the bidirectional friendship relationship and a directed arrow points to the receiver of a compliment. Here, we treat the strengths of friendship and compliment equally and set them both to be 1. The distribution of $\mathbf{p}_i$ is highly skewed such that the calculated proximity drops exponentially with the increase in the distance between two nodes. To speed up the computation of the proximity, it is suggested to perform RWR only on the partition that contains the starting point and iteratively approximate a local estimation. To achieve a desirable (near) realtime response, we define the *neighborhood* of a target node as the partition with nodes of a maximum distance of m and restrict the iteration number by l steps. Moreover, as different types of relations may indicate different levels of trust, we further define *link strength* for each type of relationship links and take them into account when column normalizing the transition matrix $\mathbf{S}$.

B. Trust-based rating prediction

The proximity measured from the friendship and compliment relations in the RWR model demonstrates the perceived trustworthiness of a user to others who are socially connected with him, and thus can be used as a trustworthiness weight in the user-based collaborative filtering approach to weight each user's contribution to the rating prediction. In particular, in an online review system with $|\mathcal{U}|$ users and $|\mathcal{I}|$ items, we model the trustworthiness of the target user a to another user u in the system as a function of $p_{a,u} = \mathbf{p}_a(u)$ and adopt Resnick's standard prediction formula [19] to calculate the predicted rating of user a to any item i that a has not yet reviewed:

$$\hat{r}_{a,i} = \bar{r_a} + \frac{\sum_{u \in \mathcal{U}_N(a), u \neq a} (r_{u,i} - \bar{r_u})\omega_{a,u}}{\sum_{u \in \mathcal{U}, u \neq a} \omega_{a,u}} \tag{2}$$

where $\bar{r_a}$ and $\bar{r_u}$ are the average ratings of user a and u, respectively, $r_{u,i}$ is the rating of user u to item i, and $\omega_{a,u} = f(p_{a,u})$. $f(\cdot)$ is a linear trust function to relate the perceived trustworthiness with the relationship-based closeness. For simplicity, we consider $\omega_{a,u} = p_{a,u}$ in this work. The prediction is based on the ratings to item i from all the users in the neighborhood of the target a (i.e., $\mathcal{U}_N(a)$) who has reviewed i. This predicted rating aggregates the contributions of users who are considered trustworthy by the target user a but neglects the contributions from users with common preference judgements in the past, which makes it different from traditional user-based CF approaches. This is because our goal is to find a trusted prediction of the rating whose value falls into a reasonable range (with non-suspicious rating variance) but not the most accurate prediction of the rating. From this consideration, our model is more tolerant to small inaccuracies in rating predictions than the CF model and its variants, and thus can support several relaxations for a better efficiency. Social relations are employed in our model to overcome the data sparsity problem, however, it should be noted that for users with no social interactions, it is still impossible to predict the ratings for items that they have not reviewed. Finally, with the trust-based predictions, we form a "pseudo user-item rating matrix" of $|\mathcal{U}|$ users and $|\mathcal{I}|$ items with three types of elements, the original ratings $r_{u,i}$, the predicted ratings $\hat{r}_{u,i}$ and empty ratings "-".

C. Trust-aware detection based on rating variance

In recommendation systems, rating variance that is inversely related to the recommendation accuracy is often considered as a confidence measurement [26]. Hybrid recommendation approaches have been proposed to first adopt any existing CF algorithm as a "black box" to predict ratings of unrated items, and recommend the top-N items by filtering out the ones with rating variances larger than a deviation threshold, which can be user-specified or item-specific standard deviation. Based on the observation that the accuracy of predictions monotonically decreases with the increase of rating variance [27], we propose to calculate a *quality score*, q_i, for every item i based on all the ratings (with both original and predicted ones) received on i, denoted as $R_{u,i}$, as well as the item-specific rating variance. In particular, $R_{u,i}$ is from the pseudo user-item rating matrix, which is either $r_{u,i}$ or $\hat{r}_{u,i}$. A straightforward approach is illustrated in an iterative model as below:

$$q_i = \frac{\sum_{u=1}^{|\mathcal{U}|} R_{u,i} v_{u,i}}{\sum_{u=1}^{|\mathcal{U}|} v_{u,i}} \tag{3}$$

where $v_{u,i}$ is a rating-variance-based vote defined as:

$$v_{u,i} = \begin{cases} 1, & |R_{u,i} - q_i| \leq \Delta \\ 0, & |R_{u,i} - q_i| > \Delta \end{cases} \tag{4}$$

Here, Δ is the deviation threshold defining the maximum acceptable rating variance for any trust-based rating predication that is considered accurate. The value of Δ can be selected arbitrarily from a suggested range or defined as the standard deviation to model the worst case scenario. With a large Δ, ratings of any arbitrary reviews are more likely to be included in the quality estimation of an item. On the contrary, when the Δ is small, the quality estimation is more likely to be biased. In hybrid recommendation systems, it is commonly suggested to select Δ in a range from 0.8 to 1.2 [27]. As discussed in section IV-B, our model is less sensitive to rating inaccuracy than CF based approaches, therefore, we suggest to tolerate a reasonably larger inaccuracy with a large Δ in order to incorporate more trusted ratings. In the experiment, we learned the value of Δ (=2.011) from a small set of labeled data.

The total number of votes received by a user reflects the trustworthiness of the user by taking into account both social relational structure with all the neighboring nodes and the rating deviations of all the items rated by the user and his neighbors. Therefore, it is natural to derive an *overall trustworthiness score*, $\omega_u \in [0, 1]$, for any user u in the system:

$$\omega_u \leftarrow \frac{\frac{\sum_{i=1}^{n} v_{u,i}}{\#ofreviews(u)}}{max_{a \in \mathcal{U}}(\frac{\sum_{i=1}^{n} v_{a,i}}{\#ofreviews(a)})} \tag{5}$$

where ω_u is defined as the per-review vote count normalized by the maximal per-review vote count among all the users. This is to limit the impact (and the potential bias) of less active users who only reviewed a small number of items on the estimated item quality. The overall trustworthiness score ω_u is updated iteratively each round according to the deviation-based votes. To incorporate the trustworthiness into item quality estimation defined in Equation (3), we re-define it as the weighted quality score:

$$q_i = \frac{\sum_{u=1}^{|\mathcal{U}|} R_{u,i} \omega_u}{\sum_{u=1}^{|\mathcal{U}|} \omega_u} \tag{6}$$

We describe the process of the iterative model in Algorithm 1. Finally, the overall trustworthiness scores for all users calculated from the iterative model are used as indicators to distinguish regular reviewers and the potential spam reviewers. In particular, users with $\omega_u \leq \tau_L$ is considered as spammers and users with $\omega_u \geq \tau_U$ is considered non-spammers, where τ_L and τ_U are pre-selected lower and upper thresholds. In the next section, we will discuss the detection results based on experiments over a dataset of 50,304 reviews collected from a representative online review system, *Yelp.com*, and evaluate the performance of our trust-aware iterative detection model.

Algorithm 1 Iterative model to calculate the overall trustworthiness

Input:
Sets of items $\mathcal{I}$ and users $\mathcal{U}$;
Initial ω_u for all users in $\mathcal{U}$;
Rating deviation threshold Δ;
Output:
Item quality scores q_i for all items in $\mathcal{I}$, overall trustworthiness scores ω_u for all users in $\mathcal{U}$;
repeat
Compute the quality scores for all items using (6)
Count trust votes and compute per-review vote counts for all users using (4)
Update the overall trustworthiness score using (5)
until converged

V. Experiments & Evaluations

In this section, we first introduce the dataset we use in the experiments, and then present the experiment results for evaluations.

A. Data collection and dataset

In the proposed iterative model, we consider social relationships among the users and calculate the proximity to a target user as an indicator of the perceived trust. Since there is no publicly available dataset that includes both reviews and social relationships (e.g., friendships and other uni- or bi-directional relationships), we have collected data from Yelp.com, a widely used online review system known mainly for restaurant reviews, for experimentation. We have crawled approximately 9 million (9,314,945) reviews submitted by 1,246,453 reviewers for 125,815 stores in 12 cities in the United States between 2004 and 2013, and completed the entire data collection process by March 2013. In this work, we extracted a smaller dataset of the city of Palo Alto, CA, with 300 stores, 22,877 users, and 50,304 reviews, and conducted experiments over this dataset.

B. Experiment results and evaluations

1) Trust-based rating predictions: We applied our trust-based rating prediction with RWR algorithm on three social graphs that include compliment relationship only, friendship only, and the two-faceted relationship, respectively, and compare the resulted prediction accuracy with a baseline approach that adopts the user-based collaborative filtering model. For performance evaluation, we consider two metrics, the Mean Absolute Error (MAE) and the Mean Absolute User Error (MAUE), defined as below:

$$MAE = \frac{\sum_{u=1}^{|\mathcal{U}|} \sum_{i=1}^{|\mathcal{I}_u|} |\hat{r}_{u,i} - r_{u,i}|}{\sum_{u=1}^{|\mathcal{U}|} |\mathcal{I}_u|} \tag{7}$$

and

$$MAUE = \frac{\sum_{u=1}^{|\mathcal{U}|} (\sum_{i=1}^{|\mathcal{I}_u|} (\hat{r}_{u,i} - r_{u,i}) / |\mathcal{I}_u|)}{|\mathcal{U}|} \tag{8}$$

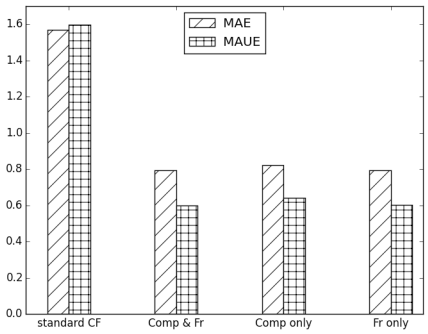


Fig. 2: Comparing prediction accuracy of the proposed model using three social graphs and the baseline CF approach

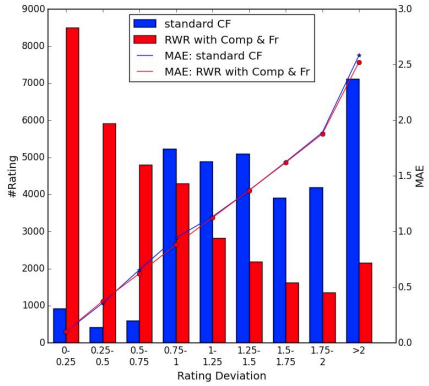


Fig. 3: Average MAE and rating deviation distribution

where $\mathcal{I}_u$ is the set of items on which user u has both actual and predicted ratings, denoted as $r_{u,i}$ and $\hat{r}_{u,i}$, respectively. From definitions, MAE measures the average absolute deviation between users' predicted ratings and actual ratings on the items in the evaluation set [28]. Different from MAE, MAUE denotes the average of the mean errors of all users [11]. We compare the performance of four approaches in terms of their MAEs and MAUEs and show the results in Table I and Figure 2. Obviously, our proposed method outperformed the baseline CF method under both metrics. Let us further compare the performance of the proposed trust-based rating prediction using RWR in three social graphs. While all three groups of relationships yield very close MAEs and MAUEs denoting similar performances in prediction accuracy, the two-faceted relationship that combines compliments and friendships achieves the best performance, while the compliments-only approach performed worst among the three. This is consistent with our expectations since friendship is a stronger relationship due to bilateral agreements from both sides. The results also showed that the predicted ratings are accurate estimations of user's opinions that are derived in a collaborative means.

Methods \ Metrics	MAE	MAUE
Standard CF	1.5672	1.5956
Compliments-only	0.8232	0.6423
Friendships-only	0.7934	0.6033
Two-faceted	0.7921	0.5985

TABLE I: Comparing prediction accuracy of four approaches using MAE and MAUE

Fig 3 shows further details about the distribution of rating variances and the average MAE. As shown in the figure, we divided the rating deviation into 9 ranges and compare the predication accuracy of standard CF and RWR on compliments-and-friendships graph. It is obvious that our method outperformed standard CF since a large number of predicated ratings fall into the ranges with smaller deviations. The potential causes of why the standard CF did not yield a satisfying performance will be discussed in Section VI.

2) Overall trustworthiness scores: In the experiment, the initial trustworthiness ω score for all users were set to 0.5. The

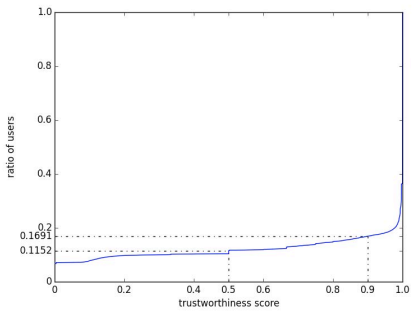
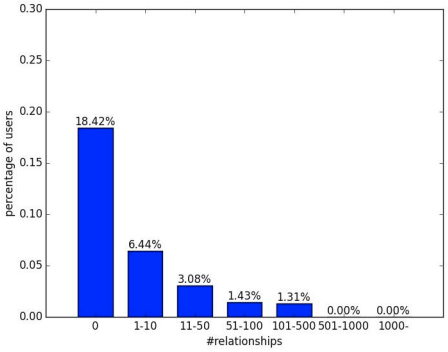


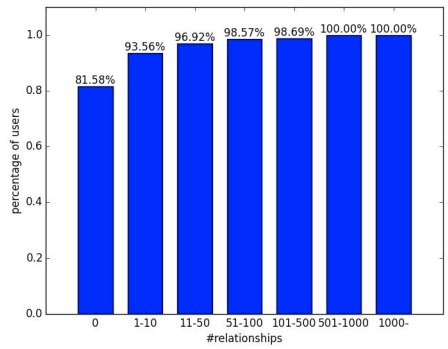
Fig. 4: CDF plot of the overall trustworthiness score ω_u

model iteratively follows the process shown in Algorithm 1. The algorithm is considered converged when the summation of the differences of all ω is less than or equal to $\varepsilon = 0.05$. The deviation threshold Δ is learned from a small set of labeled data using a discretization method - Recursive Minimal Entropy Partitioning (RMEP) [29]. The small dataset was manually labeled by three graduate students independently. The threshold learned was 2.011. The cumulative distribution function (CDF) of the overall trustworthiness scores is shown in Figure 4. From the figure, we see that about 11.52% of the users have trustworthiness scores less or equal to 0.5; 16.91% of the users have trustworthiness scores less or equal to 0.9. So about 80% of the users have high trustworthiness scores larger than 0.9. The results seem reasonable since most of the users in the system should be normal users rather than suspicious spammers, no matter how subjective their ratings are.

In order to show the correlation between social relationships and trustworthiness scores, we divided the number of relationships (both friendship and compliment) into 7 consecutive ranges in the ascending order. The relationship between the number of relationships and the percentage of users, with trustworthiness scores below or above 0.5 and 0.9, respectively, is shown in Figure 5 and Figure 6. It is obvious that the ratios of users with high trustworthiness scores increase along with the increase of the number of relationship links. It supports a conclusion that users who are more socially-active have higher overall trustworthiness scores.

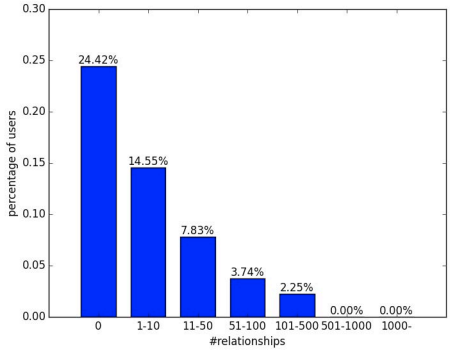


(a) % of users with $\omega < 0.5$

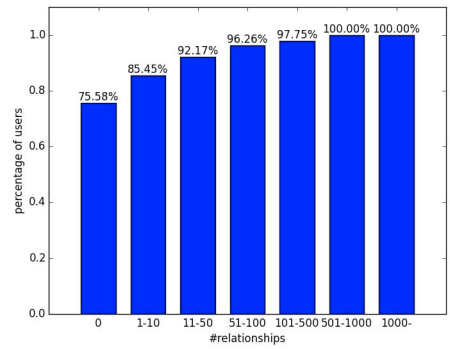


(b) % of users with $\omega > 0.5$

Fig. 5: The relationship between the overall trustworthiness score and the number of friends when ω is larger or smaller than 0.5



(a) % of users with $\omega < 0.9$



(b) % of users with $\omega > 0.9$

Fig. 6: The relationship between the overall trustworthiness score and the number of friends when ω is larger or smaller than 0.9

3) Evaluations: For evaluating the experiment results, we adopt the idea presented in [5]. We first ranked all users based on their trustworthiness scores in a descending order and selected the top-40 users and bottom-40 users. Then, we mixed all the selected users together so that the results to be evaluated demonstrate no relationship between the order and the trustworthiness scores. All related reviews of the selected users were retrieved from the dataset for evaluation. Two human evaluators were recruited for evaluation. They were all instructed with the background of review spam detection and the evaluation criteria. The content of this task was to read the reviews and manually assign a binary label of whether a user is suspicious or not based on their best judgments. The process was conducted independently between the two evaluators. Results showed that both evaluators detected less suspicious users but more normal users than the model did. The agreement between evaluators was higher than the agreement between the evaluators and the model.

VI. Discussions

Our experiment results showed that the standard CF method is not as effective as expected, with various possible reasons. First, one of the weaknesses of CF is that it is not attack-resistant [11], [30]. When review spammers post fake reviews on the system, the ratings of fake reviews significantly affect the overall rating of the target items and mislead other users. As a result, it is difficult for the CF model to achieve the expected accuracy. Moreover, these ratings deviate largely from the majority. For their aspect, most of users have a negative similarity with them. Therefore, the rating predicted for them is not accurate, which further affects the overall performance of the CF model. Our model, on the other hand, works under the assumption that review spammers tend to be socially inactive. Many of them would be isolated or barely connected with other users in the system. Our prediction model only aggregates the ratings from trusted users, which potentially filters out the influence of spammers.

The second possible reason is that CF cannot address cold-start users, which are users just joining the system with little review history. Relying on similarities to make predictions, CF is not effective for cold-start users because the similarities for them are not reliable. However, on Yelp.com, users are notified when their friends on Facebook or other social networks are registered. As a result, cold-start users without much review histories may have many social relationships to support our model. In other approaches of spam detection, data of these users are typically removed from datasets since it would be difficult to judge whether they are spammers or not. However,

our model can still effectively take them into account as long as social relationships exist.

Interestingly in our experiment results, some users with no social relationship still achieved a high trustworthiness score. This is because we assume spammers have less social relationships, but not vice versa. It does not necessarily mean that users with limited social relationships are suspicious. If a socially-lazy user whose opinions on different items always agree with the majority, he should be considered as not suspicious. While other models either remove these cases from their detection or perform poorly, our model detects the socially-lazy user with a high accuracy.

VII. Conclusion

In this paper, we study the problem of detecting review spammers using contextual social relationships that are available in several online review systems. We first present a trust-based rating predication algorithm using local proximity derived from social relationships, such as friendships and complements relationships, using the random walk with restart. We then incorporate the predicted ratings into a pseudo user-item matrix to overcome the sparsity problem and compute the overall trustworthiness score for every user in the system, which is used as the spamicity indicator. Experiments on the collected Yelp dataset show that the proposed trust-based prediction achieves a higher accuracy than standard CF method. Results also show a strong correlation between social relationships and the computed trustworthiness scores.

References

[1] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, ser. WSDM '08. New York, NY, USA: ACM, 2008, pp. 219–230.

[2] A. Mukherjee, B. Liu, J. Wang, N. Glance, and N. Jindal, "Detecting group review spam," in *Proceedings of the 20th International Conference Companion on World Wide Web*, ser. WWW '11, 2011, pp. 93–94.

[3] G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh, "Exploiting burstiness in reviews for review spammer detection." in *ICWSM*. Citeseer, 2013.

[4] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via temporal pattern discovery," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '12. New York, NY, USA: ACM, 2012, pp. 823–831.

[5] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. New York, NY, USA: ACM, 2010, pp. 939–948.

[6] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Identify online store review spammers via social review graph," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 4, pp. 61:1–61:21, Sep. 2012.

[7] A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos, and R. Ghosh, "Spotting opinion spammers using behavioral footprints," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 632–640.

[8] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*.

[9] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, Sep. 1999.

[10] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Technical Report 1999-66, November 1999, previous number = SIDL-WP-1999-0120.

[11] P. Massa and P. Avesani, "Trust-aware collaborative filtering for recommender systems," in *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*. Springer, 2004, pp. 492–508.

[12] C. Than and S. Han, "Improving recommender systems by incorporating similarity, trust and reputation," *Journal of Internet Services and Information Security (JISIS)*, vol. 4, no. 1, pp. 64–76, 2014.

[13] P. Bedi, H. Kaur, and S. Marwaha, "Trust based recommender system for the semantic web," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07, 2007, pp. 2677–2682.

[14] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: Social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, ser. CIKM '08, 2008, pp. 931–940.

[15] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation," in *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '09, 2009, pp. 195–202.

[16] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, ser. WSDM '11, 2011, pp. 287–296.

[17] N. Jindal, B. Liu, and E.-P. Lim, "Finding unusual review patterns using unexpected rules," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. New York, NY, USA: ACM, 2010, pp. 1549–1552.

[18] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "Fake review detection: Classification and analysis of real and pseudo reviews," Technical Report UIC-CS-2013-03, University of Illinois at Chicago, Tech. Rep., 2013.

[19] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '94, 1994, pp. 175–186.

[20] H. Yildirim and M. S. Krishnamoorthy, "A random walk method for alleviating the sparsity problem in collaborative filtering," in *Proceedings of the 2008 ACM Conference on Recommender Systems*, ser. RecSys '08. New York, NY, USA: ACM, 2008, pp. 131–138.

[21] L. H. Ungar and D. P. Foster, "Clustering methods for collaborative filtering," in *AAAI workshop on recommendation systems*, vol. 1, 1998.

[22] P. Massa and B. Bhattacharjee, "Using trust in recommender systems: An experimental analysis," in *Trust Management*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2004, vol. 2995, pp. 221–235.

[23] L. Lovász, "Random walks on graphs: A survey," *Combinatorics, Paul erdos is eighty*, vol. 2, no. 1, pp. 1–46, 1993.

[24] M. Jamali and M. Ester, "Trustwalker: A random walk model for combining trust-based and item-based recommendation," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '09. New York, NY, USA: ACM, 2009, pp. 397–406.

[25] L. F. P. R. X. H. . V. S. Seo, H., "Perceived usefulness of online reviews: Effects of review characteristics and reviewer attributes," in *Proceedings of the 65th International Communication Association Annual Conference*, May 2015.

[26] Y. Kwon, "Improving top-n recommendation techniques using rating variance," in *Proceedings of the 2008 ACM Conference on Recommender Systems*, ser. RecSys '08, 2008, pp. 307–310.

[27] G. Adomavicius, S. Kamireddy, and Y. Kwon, "Towards more confident recommendations: Improving recommender systems using filtering approach based on rating variance," in *17th Workshop on Information Technology and Systems (WITS07)*, 2007.

[28] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, Jan. 2004.

[29] K. B. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning."

[30] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Proceedings of the 2007 ACM Conference on Recommender Systems*, ser. RecSys '07. New York, NY, USA: ACM, 2007, pp. 17–24.