Leveraging Online Social Relationships for Predicting User Trustworthiness

Jun Zou and Faramarz Fekri
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
Email: {junzou, fekri}@ece.gatech.edu

***Abstract*—Online social networks are becoming important platforms where users make social connections and share information. However, they are vulnerable to malevolent activities by malicious users. Hence, it necessitates effective automatic methods to predict user trustworthiness. The existing works mostly predict the trustworthiness of individual users separately from other users, ignoring the fact that users are related to each other through online social relationships. In this paper, we propose a probabilistic model based on Pairwise Markov Random Field (PMRF) that takes into account both user features and social relationships. In addition, we apply the Belief Propagation (BP) algorithm to perform inference efficiently in PMRF. The complexity of the algorithm grows only linear in the number of users. The experiment results on the Twitter datasets show that the proposed PMRF model can effectively exploit the social relationships to significantly improve the prediction performance.**

I. Introduction

The online social networks such as Facebook and Twitter are becoming important platforms for making social connections and sharing information. Meanwhile, the powerful personal mobile computing devices, e.g., smart phones and tablets, and the ubiquitous wireless communications enable users to participate in on-line social activities, generate and disseminate data in multiple forms, e.g., texts, images, and videos, anywhere any time. Hence, social computing applications have attracted significant interest from both industry and government [1]. However, due to the open nature of such social networking platforms, they are inherently vulnerable to malicious users or spammers who misuse social networks to perform malevolent activities such as spamming [2], phishing [3], and spreading computer virus. Those misbehaviors, if unattended, can greatly undermine the utility and functionality of social networks. Therefore, there is an urgent need to develop algorithms to effectively predict the trustworthiness of users in social networks, so as to identify suspicious users and limit their activities or suspend their accounts.

Several existing works have proposed automatic methods for analyzing user trustworthiness [4]–[6]. They discovered effective patterns and features of users that are useful for the prediction task, and applied machine learning techniques, e.g., support vector machine, to classify users based on their features. However, most existing approaches predict the trustworthiness of each user separately from other users, ignoring the fact that users in online social networks are related to each other through social connections and interactions.

In most online social networks, users can connect to other people they know or share common interests. The establishment of bidirectional connections between users often indicates some degree of similarity in their trustworthiness. In some social networks like Facebook, the connecting request initialized by one user requires explicit approval from the other user before they become connected, and in other social networks like Twitter, although users can unilaterally establish directed connections, e.g., following other people, bidirectional connections are mostly established between friends who know each other in real life, or between people who share mutual interests. Hence, users with close social relationships are more likely to have similar roles in social networks. For example, the study on the Twitter social network has revealed that criminal user accounts tend to be socially connected to form a small-world network [7].

In this work, we propose a probabilistic model based on Pairwise Markov Random Field (PMRF) that takes into account both user features and social relationships. We use PMRF to express a proper factorization of the joint distribution of users based on their social relationships. As PMRF can be conveniently represented by a probabilistic graph consisting of edges and nodes, we can apply the Belief Propagation (BP) [8] algorithm to exploit the graph structure to perform inference efficiently, and hence, the complexity of the algorithm grows only linear in the number of users. In the experiment, we apply the proposed algorithm to predict spammers in the Twitter datasets, and show that the proposed PMRF model can effectively exploit the social relationships to significantly improve the prediction performance.

II. Related Works

With the emergence of online social networks, e.g., Twitter and Facebook, user trustworthiness in social networks attracted wide interest from both government and industry [9]. Many works proposed to predict user trustworthiness based on user behavior patterns in social networks. The works in [4]–[6] introduced various features extracted from user profiles and user generated social content, and used them to detect spammers using classification tools in machine learning. Although such feature-based classification approaches can capture behavioural characteristics of individual users, they ignore the relationships between users.

A number of other works utilized the connection patterns in social network structure. In [10] the authors evaluated user trustworthiness based on the density of interconnections between users, assigning higher trustworthiness to users with higher degree of connections. In [11], the authors proposed to use social network characteristics of community formation to

978-1-4799-5952-5/15/$31.00 ©2015 IEEE

learn classification models for identifying spammers. However, in today's social networks, there can be a significantly larger number of malicious users and they can connect to each other to increase their social connections. Indeed, the authors in [7] examined the Twitter network as a particular case and found that criminal accounts tend to be socially connected to form a small-world network.

In [12], a socially regularized matrix factorization model was used to learn latent user features from social activities for spammer detection in social networks. By exploiting users' social relationships, it imposed a social regularization term on matrix factorization, as the latent factors of a user should be similar to their connected users since they share similar interests and may perform similar social activities. In [13], the authors employed social relationships to regularize the least squares model for spammer classification, where an extra penalty is added to the loss function during training if two users with close social relationships have different predicted labels. However, the user's social relationships are not utilized to predict the label of an unknown target user. In [14], the authors proposed a social network aided Bayesian spam email filter, which adjusts the keyword weights based on the social closeness between the email receiver and sender, as users with high closeness are less likely to send spam emails to each other.

The BP algorithm is well-known for its efficiency in computing marginal functions from global functions of many variables. There are many successful applications of BP to solving inference problems in probabilistic graphical models. In the graph-based iterative probabilistic decoding of turbo codes and low-density parity-check (LDPC) codes [15], BP algorithms have shown to achieve performance close to the optimal maximum likelihood decoding scheme, yet at much less computational cost. Previously, the work in [16]–[18] proposed to use BP for trust and reputation management based on the explicit ratings exchanged between users for peer-to-peer (P2P) networks and delay tolerant networks, and the work in [19] applied BP to detect spam users in recommendation systems who give spam ratings to targeted items. In this work, we consider user trustworthiness prediction in general social networks, where users are inter-related via the social relationships.

III. User Feature-Based Methods

In this section, we briefly introduce how user features are derived and used to predict user trustworthiness in social networks. As the research results in [5], [6] show, the different patterns of trusted users and untrusted users can be captured by carefully designed user features. Generally, the user features are mainly extracted from the following two major categories of social data:

1) User Profiles: The profiles of trusted users tend to be more complete and verifiable than those of untrusted users. For example, trusted users usually provide more details of personal information, such as his professional title and employer, and they may also provide links to personal webpages, so that other people can actually verify the user's true identity in real-life. In addition, the existing time of a trusted user' account since it was first registered is usually longer, whereas many untrusted users are newly registered users.

2) User Content: Users generate data and disseminate information in online social networks, e.g., post tweets in Twitter. We can directly look into user generated content to determine if a user is behaving improperly, e.g., posting spam messages that include phishing URLs. Also, we can extract certain patterns of users' posting behaviors, such as the number of duplicated messages users post. The specific features that are effective depend on the form of the social network, and need to be designed accordingly.

The basic approach to analyzing user trustworthiness is using the classification machine learning tools [4]–[6], e.g., Support Vector Machine and Decision Tree. The classifiers are first trained with labelled user data, and given a unknown user they generate classification results as the predicted user trustworthiness. However, such classifiers treat each individual user separately from others in social networks, and ignore the social relationships between users.

IV. Modelling Trustworthiness on PMRF

In order to exploit the social relationships to improve user trustworthiness prediction, we formulate the problem as a probabilistic inference problem using a PMRF graphical model, where the social relationships can be conveniently modeled as edges between hidden nodes. In addition, the inference in PMRF can be carried out efficiently by using the BP algorithm.

A. Probabilistic Problem Formulation

We assume a set $\mathbb{U}$ of M users in the social network, $\mathbb{U} = \{1, \ldots, M\}$. Our goal is to infer the trustworthiness r_u of each user $u \in \mathbb{U}$. We model r_u as a discrete random variable, which takes values from a discrete set $\Gamma = \{s_1, s_2, \ldots, s_L\}$, $|\Gamma| = L$. For example, with $\Gamma = \{0, 1\}$, we can use 0 and 1 to represent "malicious/untrusted" and "normal/trusted", respectively. For user u, we also represent the user features by vector $\boldsymbol{\theta}_u = [\theta_1, \theta_2, \ldots, \theta_p]$ with length p, which can be extracted from the social data as discussed in Sec. III. Let $\mathbb{R} = \{r_1, \ldots, r_M\}$ denote the set of trustworthiness variables, and $\Theta = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$ denote the set of user features for all users.

To take into account the dependency between users due to social relationships, we need to jointly model all variables in $\mathbb{R}$. Let $P(\mathbb{R}|\Theta)$ be the joint posterior distribution of $\mathbb{R}$, given the observed user features in Θ. Then to compute r_u, we infer the marginal posterior distribution $P(r_u|\Theta)$ from $P(\mathbb{R}|\Theta)$. The direct computation can be expressed as follows

$$P(r_u|\Theta) = \sum_{r_1 \in \Gamma} \cdots \sum_{r_{u-1} \in \Gamma} \sum_{r_{u+1} \in \Gamma} \cdots \sum_{r_M \in \Gamma} P(\mathbb{R}|\Theta), \quad (1)$$

which is summation over all variables except r_u. Obviously, the computational complexity grows exponentially as $\mathcal{O}(L^M)$. Considering the large number of users in social networks, (1) is computationally infeasible. In the following, we propose a PMRF model that properly factorizes the joint distribution $P(\mathbb{R}|\Theta)$ into local functions according to social relationships between users, and hence we can apply the BP algorithm to infer $P(r_u|\Theta)$ efficiently.

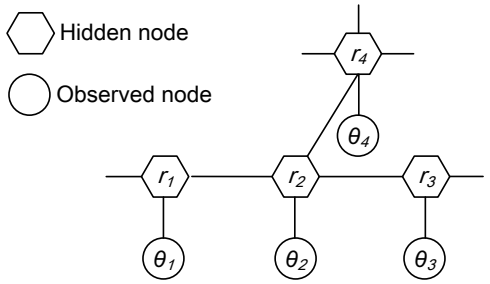

Fig. 1: Illustration of the PMRF model.

B. Modelling via PMRF

A PMRF consists of hidden and observed nodes [20], in which the statistical dependencies between two connected hidden nodes are represented by compatibility functions, and the dependency relations between the observed nodes and hidden nodes are represented by local evidence functions. We model the joint posterior probability distribution $P(\mathbb{R}|\Theta)$ in a PMRF $\mathcal{G}$ as illustrated in Fig. 1. For each user u, whose trustworthiness r_u needs to be predicted, we assign a hidden node (or variable node), shown as a hexagon, and we further connect it to an observed node (or feature node), shown as a circle, which represents the observed features $\boldsymbol{\theta}_u$ of user u. To capture the dependence among variables induced by social relationships, variable nodes r_u and r_v are connected via an edge, if there exists a social relationship between users u and v. Let $\mathcal{E}$ denote the set of all edges between variable nodes. We can express a proper factorization of $P(\mathbb{R}|\Theta)$ represented by PMRF $\mathcal{G}$ as follows

$$P(\mathbb{R}|\Theta) = \frac{1}{Z} \prod_{(u,v)\in\mathcal{E}} \psi_{uv}(r_u, r_v) \prod_{u=1}^{M} \phi_u(r_u|\boldsymbol{\theta}_u) \tag{2}$$

where Z is a normalization factor, $\psi_{uv}(r_u, r_v)$ is a compatibility function between r_u and r_v, and $\phi_u(r_u|\boldsymbol{\theta}_u)$ is the local evidence function for r_u given $\boldsymbol{\theta}_u$.

Firstly, we specify the local evidence function $\phi_u(r_u|\boldsymbol{\theta}_u)$, which describes the probability distribution of r_u given the observed user feature $\boldsymbol{\theta}_u$. We can use the output probability distribution from the probabilistic classifiers. As one example, we use the Logistic Regression [21] to classify users as spammers ($r_u = 0$) or normal users ($r_u = 1$), in which the probability of $r_u = 1$ given the observed feature vector $\boldsymbol{\theta}_u$ can be computed as follows

$$P_{\text{reg}}(r_u = 1|\boldsymbol{\theta}_u) = \frac{1}{1 + \exp\{-(c_0 + c_1\theta_1 + \ldots + c_p\theta_p)\}}, \tag{3}$$

where $\mathbf{c} = [c_0, c_1, \ldots, c_p]$ is a logistic regression coefficient vector to be learnt from training datasets, and

$$P_{\text{reg}}(r_u = 0|\boldsymbol{\theta}_u) = 1 - P_{\text{reg}}(r_u = 1|\boldsymbol{\theta}_u). \tag{4}$$

Hence, we can let $\phi_u(r_u|\boldsymbol{\theta}_u) = P_{\text{reg}}(r_u|\boldsymbol{\theta}_u)$.

The compatibility function $\psi_{uv}(r_u, r_v)$ needs to be defined properly to reflect the influence users u and v have on each other in the social network. Basically, r_u and r_v should be compatible and have similar values if there exists a strong social relationship between users u and v. In this work, we define $\psi_{uv}(r_u, r_v)$ as

$$\psi_{uv}(r_u, r_v) \propto \exp\left\{-\alpha(r_u - r_v)^2\right\}, \tag{5}$$

where $\alpha > 0$ is some parameter that adjusts the influence of one user on his connected users. Since r_u only takes discrete values from Γ, $\psi_{uv}(r_u, r_v)$ can also be explicitly expressed as

$$\psi_{uv}(r_u = s_i, r_v = s_j) = q_{ij}, \tag{6}$$

where $0 \le q_{ij} \le 1$ and $\sum_{1\le j\le L} q_{ij} = 1$. We can interpret q_{ij} as the probability of $r_v = s_j$ given $r_u = s_i$. A larger q_{ii} means users u and v are more likely to have similar trustworthiness, i.e., $r_u = r_v = s_i$.

V. Inference Using BP

The PMRF model expresses the factorization of $P(\mathbb{R}|\Theta)$ into many local functions as shown in (2), and hence, we apply the BP algorithm to exploit such structure to efficiently infer the marginal probability distribution $P(r_u|\Theta)$, $\forall u \in \mathbb{U}$. This is one important computational advantage of PMRF models.

A. BP Algorithm

BP is a message-passing algorithm that operates on probabilistic graphs, where messages are exchanged between nodes along edges. By exploiting the graph structure, BP efficiently computes marginal functions from complex global functions of large number of variables. When the graph has a tree structure, BP computes the exact results. Even in graphs with loops, we can obtain very good approximate results by applying the loopy BP algorithm [22], in which the probabilistic messages are iteratively exchanged between variable nodes until convergence. In this work, since users in social networks can easily connect to each other to form loops, the proposed PMRF model will have loops in many cases, and hence, we need to apply the iterative loopy BP algorithm.

During each iteration n, at any variable node b, we update the messages sent to its neighbors. To compute $m_{b,a}^{(n)}(r_a)$, the message sent from variable node b to neighbor node a, we compute the product of all incoming messages in last iteration from its neighbors except the one from node a, and multiply it with the local evidence function and the compatibility function between nodes a and b. This message is given as follows

$$m_{b,a}^{(n)}(r_a) \propto \sum_{r_b\in\Gamma} \psi_{ab}(r_a, r_b)\phi_b(r_b|\boldsymbol{\theta}_b) \times \prod_{c\in\mathcal{N}(b)\setminus a} m_{c,b}^{(n-1)}(r_b), \tag{7}$$

where $\mathcal{N}(b)$ denotes the set of users who are connected to user b, whose corresponding variable nodes are connected to r_b in PMRF.

After BP converges, the marginal distribution $P(r_u|\Theta)$ of r_u can be computed according to

$$P(r_u|\Theta) \propto \phi_u(r_u|\boldsymbol{\theta}_u) \prod_{v\in\mathcal{N}(u)} m_{v,u}^{(n)}(r_u). \tag{8}$$

The predicted trustworthiness $\hat{r}_u$ for user u is given by

$$\hat{r}_u = \text{argmax}_{r_u\in\Gamma} P(r_u|\Theta). \tag{9}$$

Algorithm 1 BP Algorithm on PMRF

- Initialization. Initialize $m_{a,b}(r_b)$ as $m_{a,b}^{(0)}(r_b) = \frac{1}{|\Gamma|}$, and set iteration counter $n = 1$.
- Iterative message-passing until convergence.
  (1) In iteration n, at each node b, update $m_{a,b}^{(n)}(r_b)$ for its neighors $\mathcal{N}(b)$ using Eqn. (7);
  (2) $n = n + 1$, and repeat step (1) until convergence.
- Compute the marginal probabilities $P(r_u|\Theta)$ using Eqn. (8);
- Compute the trustworthiness of all users using Eqn. (9).

We summarize the BP algorithm in Algorithm 1.

B. Complexity Analysis

The complexity of updating a message using (7) is $\mathcal{O}(|\Gamma|K)$, where K is the average degree of each variable node on PMRF. In each iteration, each variable node updates and sends out K messages to its neighbor nodes, and the total number of updated messages is $\mathcal{O}(MK)$. Hence, the overall complexity in terms of multiplications is $\mathcal{O}\left(|\Gamma|MK^2\right)$. Note that the proposed algorithm converges quickly, on the average in 10 iterations. Hence, the complexity of the algorithm grows only linear in the number of users.

VI. Experimental Evaluation

A. Datasets

We use the Twitter dataset prepared by [6], in which the authors crawled the real Twitter user profiles and identified spammers that post URL links of malicious websites, e.g., phishing websites. The dataset includes 1,000 spammers and 10,000 non-spammers (normal users). The crawled data for each user contain the basic user profile information, e.g., account creation time, the list of followers and followees, and the 40 most recent Tweets. From the user data we can extract various user features such as those introduced in [5], [6]. In our experiment, we use the following three user features in Table I that are reported to be very effective in discriminating spammers from normal users. We also extract user relationships using the bidirectional following connections, i.e., user pairs that follow each other in the Twitter dataset.

TABLE I: User features in Twitter dataset

Feature	Definition
Account age	How long ago the user account was created.
URL rate	The average number of posted URLs per tweet.
Reply ratio	The ratio of the number of tweets that are replies to other users to the total number of all tweets.

B. Performance Evaluation

In the experiment, we predict the trustworthiness of users by classifying the users into spammers and non-spammers. To evaluate classification performance, we first create balanced training data from the dataset. We include all 1,000 spammers of the original dataset, and randomly sample 1,000 non-spammers. We measure the performance in terms of overall accuracy as well as the precision, recall, and F_1-score metrics for both spammer and non-spammer classes. For all introduced metrics, the higher the value the better the performance. However, the individual metric of precision or recall does not reflect the performance of the algorithms very well by itself, since high precision may be achieved at low recall, and high recall may be achieved at low precision, while the F_1-score that calculates the harmonic mean of precision and recall is a more balanced metric.

In the PMRF model, we let $r_u = 0$ to indicate that user u is a spammer and $r_u = 1$ for non-spammer. The compatibility function in (6) is specified by the transition probability $q_{ii} = 0.95$, $\forall i = 0, 1$, and $q_{ij} = 0.05$ for $i \neq j$. In addition, we use the probabilities computed by the Logistic Regression classifier (3) as $\phi_u(r_u|\boldsymbol{\theta}_u)$. We compare the performance of the proposed PMRF-based algorithm with the following classification algorithms based only on users features, including:

- **Support Vector Machine (SVM)**: We use the `svmtrain()` function from the Statistics and Machine Learning Toolbox in Matlab to train a linear support vector machine classifier;
- **Logistic Regression**: To learn the logistic regression coefficients, we use the multinoimal logistic regression function `mnfit()`;
- **Naive Bayes**: We train the Naive Bayes classifier using the `NaiveBayes.fit()` function.

We split the data such that 80% of users in each class are used for training and the rest 20% for testing. We summarize the results on the testing data in Table II. For all algorithms, we use the same features shown in Table I. The results show that our PMRF-based algorithm outperforms other classification algorithms significantly, and it consistently achieves better performance in all metrics, improving classification performance for both spammer and non-spammer classes. This confirms that online social relationships between users can be utilized for user classification, and our proposed algorithm effectively exploits such relationships.

For all other classification algorithms, their accuracy results are quite close. Among them, the SVM classifier has the best accuracy, yet 18% worse than the proposed PMRF algorithm. Noticeably, all of them classify users individually. They take each user's features as input and predict his/her label, disregarding the labels of other users. This suggests that it is difficult to gain further performance improvement without taking user relationships into account.

C. Impact of Parameters

We investigate the impact of the compatibility function (6) on classification performance. In the experiment, we always let $q_{00} = q_{11}$, so that the effects of the compatibility function on both spammer and non-spammer classes are identical. In Fig. 2, we show the overall accuracy of the proposed algorithm with q_{ii} increasing from 0.5 to 1. We can see that when $q_{ii} = 0.5$, the proposed algorithm has the same performance as that of Logistic Regression, since setting $q_{ii} = 0.5$ means

TABLE II: Classification performance comparison for the spammer and non-spammer classes.

Algorithm	All	Spammer Class			Non-Spammer Class		
	Accuracy	Precision	Recall	F_1 score	Precision	Recall	F_1 score
Proposed PMRF	0.895	0.891	0.900	0.896	0.899	0.890	0.895
SVM	0.735	0.730	0.745	0.738	0.740	0.725	0.732
Logistic Regression	0.733	0.734	0.730	0.732	0.731	0.735	0.733
Naive Bayes	0.725	0.686	0.830	0.751	0.785	0.620	0.693

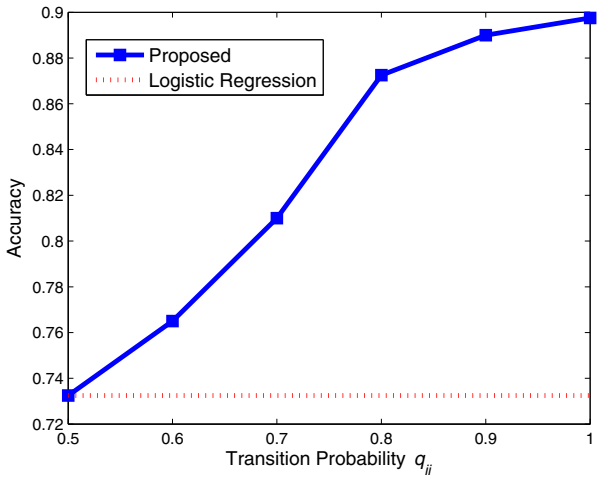

Fig. 2: Impact of the compatibility function on accuracy.

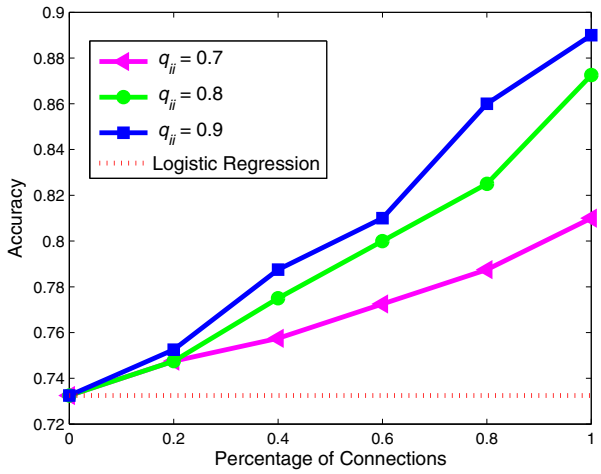

Fig. 3: Impact of social relationships on accuracy.

no compatibility is imposed on the labels of two connected users, i.e., given the label of a user, the other connected user can have either label, spammer or non-spammer, with equal probabilities. As q_{ii} increases, the accuracy first improves significantly, but then the performance begins to saturate when q_{ii} is large enough. This is due to several factors, e.g., some users are not connected to other users and thus changing compatibility function does not effect their predicted labels, or some users' probabilities of labels are strongly dominated by their individual features.

We also like to note that even though in this experiment setting $q_{ii} = 1$ seems to provide the best performance, in general the appropriate value of q_{ii} should be chosen based on the cross-validation results on the datasets. Choosing a moderate value can reduce the influence of a single user, and hence only when there are large enough number of users with identical labels socially connected to a user, his/her label will be significantly influenced. This can be useful for some scenarios, e.g., where spammers have tricked some new non-spammers to connect with them. Also, when the local evidence $\phi_u(r_u|\boldsymbol{\theta}_u)$ computed based on the user feature has low prediction accuracy, a smaller q_{ii} (greater than 0.5) can reduce false label propagation compared to $q_{ii} = 1$.

D. Impact of Social Relationships

We next examine how the richness of social connections impacts the proposed algorithm. From the original dataset, we randomly sample a proportion of the social connections between users to generate the required datasets for our experiment. In Fig. 3, we show the accuracy results for varying percentage of social connections (relative to the original dataset), under $q_{ii} = \{0.7, 0.8, 0.9\}$. As the percentage of social connections decreases, the accuracy of the proposed algorithm decreases. When no social relationships can be utilized, it degenerates to Logistic Regression that classifies users individually. Comparing $q_{ii} = 0.9$ to $q_{ii} = 0.7$, we also find that the performance for larger q_{ii} is more sensitive to the richness of social connections.

VII. Conclusion

In this paper, we proposed a probabilistic graphical model based on PMRF for predicting user trustworthiness in online social networks, where social relationships between users are modelled as edges in PMRF. We applied the BP algorithm to exploit the graph structure to perform inference efficiently. The computational complexity is linear in the number of users. In the experiment on the Twitter dataset, we applied the proposed algorithm to predict spammers, and the results showed that it outperforms other classification algorithms in terms of both accuracy and F_1 score. Hence, the proposed algorithm effectively leveraged online social relationships to improve prediction performance.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1115199.

REFERENCES

[1] F.-Y. Wang, K. M. Carley, D. Zeng, and W. Mao, "Social computing: From social informatics to social intelligence," *IEEE Intelligent Systems*, vol. 22, no. 2, pp. 79–83, 2007.

[2] G. Brown, T. Howe, M. Ihbe, A. Prakash, and K. Borders, "Social networks and context-aware spam," in *Proc. of ACM Conference on Computer Supported Cooperative Work*, 2008, pp. 403–412.

[3] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Commun. of the ACM*, vol. 50, no. 10, pp. 94–100, 2007.

[4] G. Stringhini, S. Barbara, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in *Proceedings of Computer Security Applications Conference (ACSAC'10)*, 2010.

[5] K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: Social honeypots + machine learning," in *Proc. of the 33rd International ACM SIGIR conference on Research and development in information retrieval*, 2010, pp. 435–442.

[6] C. Yang, R. Harkreader, and G. Gu, "Die free or live hard? Empirical evaluation and new design for fighting evolving twitter spammers," in *Proceedings of the 14th International Symposium on Recent Advances in Intrusion Detection (RAID'11)*, Menlo Park, CA, USA, 2011.

[7] C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu, "Analyzing spammers social networks for fun and profit: A case study of cyber criminal ecosystem on twitter," in *Proc. of the 21st International World Wide Web Conference (WWW'12)*, pp. 71–80.

[8] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., 1988.

[9] W. Sherchan, S. Nepal, and C. Paris, "A survey of trust in social networks," *ACM Computing Surveys*, vol. 45, no. 4, pp. 47:1–47:33, Aug. 2013.

[10] V. Buskens, "The social structure of trust," *Social Networks*, vol. 20, no. 3, pp. 265–289, 1998.

[11] S. Y. Bhat and M. Abulaish, "Community-based features for identifying spammers in online social networks," in *Proc. of IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'13)*, 2013, pp. 100–107.

[12] Y. Zhu, X. Wang, E. Zhong, and N. N. Liu, "Social spammer detection in microblogging," in *Proc. of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI'12)*, 2012.

[13] X. Hu, J. Tang, Y. Zhang, and H. Liu, "Social spammer detection in microblogging," in *Proc. of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, 2013.

[14] H. Shen and Z. Li, "Leveraging social networks for effective spam filtering," *IEEE Transactions on Computers*, vol. 63, no. 11, pp. 2743–2759, Nov. 2014.

[15] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 219–230, Feb. 1998.

[16] E. Ayday and F. Fekri, "Iterative trust and reputation management using belief propagation," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 3, pp. 375–386, 2012.

[17] ——, "BP-P2P: Belief propagation-based trust and reputation management for P2P networks," in *Proc. of the 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2012, pp. 578–586.

[18] ——, "An iterative algorithm for trust management and adversary detection for delay tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1514–1531, Sept. 2012.

[19] J. Zou and F. Fekri, "A belief propagation approach for detecting shilling attacks in collaborative filtering," in *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM'13)*, 2013.

[20] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *In Exploring artificial intelligence in the new millennium*, pp. 239–269, 2003.

[21] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[22] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Constructing free energy approximations and generalized belief propagation algorithms," *IEEE Transactions on Information Theory*, vol. 51, pp. 2282–2312, 2005.