2013 Ninth International Conference on Computational Intelligence and Security

Random Walk with Pre-filtering for Social Link Prediction

Ting Jin[1], Tong Xu[1], Enhong Chen[1], Qi Liu[1], Haiping Ma[2], Jingsong Lv[2], Guoping Hu[2]
[1]*School of Computer Science and Technology ,University of Science and Technology of China*
[2]*Anhui USTC iFLYTEK Co., Ltd., China*
{*qingting,tongxu*}*@mail.ustc.edu.cn,* {*cheneh, qiliuql*}*@ustc.edu.cn,*{*hpma,jslv, gphu*}*@iflytek.com*

***Abstract*—The prosperity of content-oriented social media services has raised the new chances for understanding users' social behaviors. Different from traditional social networks, the links in social media are usually influenced by user preferences rather than the real world connections, thus the traditional methods based on social network evolvement may fail to reveal the adequate links. Meanwhile, the existing link prediction algorithms considering both social topology and nodes attributes might be too much computationally complex. To deal with these challenges, in this paper, we propose a two-steps link prediction framework, in which a filter is functioned to select the candidates firstly, and then the adapted Supervised Random Walk (SRW) is executed to rank the candidates for prediction. Experiments on the real world data set of social media indicate that our framework could effectively and efficiently predict the appropriate links, which outperforms the baselines including ordinary SRW with acceptable margin.**

***Keywords*-Link Prediction; Social Media; Supervised Random Walk**

I. Introduction

Rapid growth of social media platforms encourages the information explosion in Web 2.0 generation, while at the same time transforms the online social network services (SNS). The so-called "grassroot" users could now not only propose individual ideas or art works, but also interact with each other via shares, comments or connections. This phenomenon results in the prosperity of social media platforms and also raises new challenges for the administrators, who are required to understand the users' social behaviors, especially the social connections that directly affect the users' activity and loyalty of SNS.

Different from traditional social network which is usually based on the relationship in real world, the connections in the content-oriented social network may be due to the similar preference, topical authority or even the fashion trend. Particularly, in the asymmetric social network like Twitter and Flickr, links indeed mean "following" without permission of the followee, thus the ordinary users usually prefer to follow the stars or experts to achieve high-quality content. In this case, the traditional methods following some basic rules of network evolvement, such as power law [4], transitivity [14] or the small world phenomenon [10] may fail to reveal the correct links.

At the same time, the feature-based methods might either ignore the social network topology, or confront the high computational complexity. On one hand, those which based on pairwise similarity only might be inappropriate since they lose the global information from the entire network. On the other hand, those consider both node (user) attributes and network topology might be too much complicated. And since the networks are often huge in size and the majority will be negative samples for supervised learning, the training process could be severely time-consuming, and then lead to the failure of catching the evolving network in real-time.

To address this problem, in this paper, we propose a two-steps framework to effectively predict potential links with adapting the *"Supervised Random Walk"* (SRW) method [3] which partially ranks the candidate nodes. To be specific, in the first step, the pre-filtering will be functioned to select the potential links. Then in the second step, the SRW method will be executed to rank filtered candidates for prediction. Since the sizes of candidate sets are controlled, the computational complexity reduces sharply and the imbalanced classification problem is alleviated. The extensive experiments on the real world data set indicate that our framework could effectively and efficiently predict the adequate links, which outperforms the baselines with significant margin.

The reminder of this paper will be organized as follows. We firstly review literatures on link prediction in Section II. Then in Section III, we formulate the link prediction problem and propose our two-steps framework. In Section IV, the technical details will be explained for both feature engineering and proposed algorithm. Section V shows the experimental results on the real world data set, which validate the performance of our novel framework. Finally, we conclude the paper in Section VI.

II. Related Work

Indeed, the social link prediction has long been studied by social scientists and psychologists [13], while it does not, however, hinder the recent numerous efforts by computer scientists, especially when social media platforms become popular. Generally speaking, the existing methods could be roughly divided into two parts, i.e., the unsupervised models which predict links following some certain rules, or supervised models which attempt to train adequate classifiers.

Traditionally, the unsupervised methods focus on the structure of the network [12], e.g., the common neighbors [2] corresponding to the transitivity in social network, i.e.,

978-1-4799-2548-3/13 $31.00 © 2013 IEEE
DOI 10.1109/CIS.2013.36

Table I
LIST OF NOTATIONS

Symbol	Meaning	Symbol	Meaning
$G = \langle V, E \rangle$	(asymmetric) social network	$[t, t']$	sampling interval
$V = \{u_i\}$	set of nodes (users)	S	set of source users
$E = \{e_{ij}\}$	set of directed links	$\mathbf{w}_{ij}$	features' weight on link e_{ij}
C_i	candidate set for node u_i	p_{ij}	SRW score of u_i for u_j

friends of your friends are probably your friends. Some other methods focus on the path analysis with some special measures like Katz [8], Jaccard or Hitting Time [12] which is derived from the expectation of random walk. Besides, some complicated methods may consider more information, e.g., the matrix factorization algorithms like [16]. Usually, these models utilize a predictive score function or at least a threshold to measure the occurrence of edges.

Correspondingly, we have supervised models which treat the links as pairwise vertices, and then features are extracted to represent the pair nodes. After that, a classifier will be learned based on the training samples to predict the links may appear in the future. Thus, two factors should be determined: the features and the classification model. The features here are quite similar with those in unsupervised learning models, including node or edge attributes like rating [19] or location [15], or social factors like common neighbors and pairwise distance [5]. For the classifiers, basic classification models like Bayesian probabilistic models [7][17], or matrix methods like dimensionality reduction [11], could all be introduced to solve the link prediction task. Finally, as the *"learning to rank"* techniques introduced, the candidates could now be ranked instead of classified.

Besides, some other interesting applications could be developed based on the social link analysis, like the reversed link prediction [6], the disease spread [1] or multimedia tagging task [18].

III. OVERVIEW: FORMULATION AND FRAMEWORK

In this section, we will firstly formulate the asymmetric link prediction problem and then summarize the related mathematical notations. Secondly, the two-steps framework will be formally introduced, while the technical details will be introduced in Section IV.

A. Problem Formulation

Here we introduce some preliminaries related to the link prediction task. Usually, a social network is depicted by a graph $G = \langle V, E \rangle$, where V denotes the set of nodes (users), and E includes the links between users. What should be noted is that we discuss the directed edges here, i.e., the link e_{ij} only presents u_i follows u_j in the SNS, but not includes the reverse connection.

Then, a set of features could be extracted to describe both node (user) and edge (social link) according to not only users' profiles but also their behavior records. To be specific, we utilize $\mathbf{w}_{ij}$ to represent features' weight. The details will be illustrated in Section IV-A.

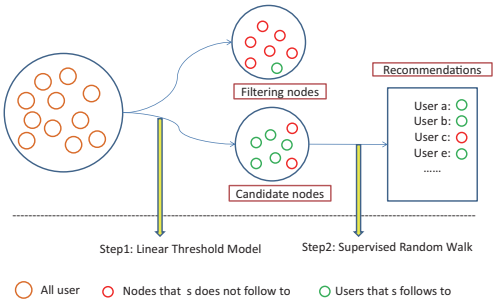


Figure 1. Flowchart of two-steps framework.

And finally, the link prediction task could be described as: if we treat the graph exists in the time interval $[t_0, t'_0]$ as training samples, and further predict the future links in the test interval $[t_1, t'_1]$, then we should traverse all the pairwise nodes $\langle u_i, u_j \rangle$ that $e_{ij} \notin E[t_0, t'_0]$ to reveal the most probable new links $\{e^*_{ij}\}$ will occur in $G[t_1, t'_1]$ based on the model learned by $G[t_0, t'_0]$. The related notations are summarized in Table I.

B. Two-steps Framework

As mentioned above, the supervised prediction models might be too much complicated mainly due to the huge size of social network. For a certain user, the potential links will be numerous, while only a few positive samples exist in the training set. Also, we realize that some basic rules could guide the link prediction process, e.g., users may prefer to follow those having similar preferences with them. Motivated by the two discoveries, we conduct a pre-filtering to control the size of candidates. The two-steps framework is defined as follow:

Pre-filtering. Firstly, we execute a filter to pick out the most probable candidates (to link), donated as C_i for node u_i. Filters will function based on some certain rules, including common interests and social connections. Those nodes which are impossible to be linked will be eliminated by this process.

Candidates ranking. Since the candidates are selected, in the second step, we attempt to rank the candidates through considering their attributes and social relationships. Here we introduced the Supervised Random Walk model to learn the partial relationship of filtered candidates, and the social structure will not be broken.

Indeed, these two steps reflect the progressively adjustment of prediction which improves the effectiveness and efficiency. The technical details will be explained in Section IV, and Figure 1 illustrates the complete flowchart of our framework.

IV. LINK PREDICTION WITHIN TWO STEPS

In this section, we will introduce the technical details of our two-steps framework. To be specific, firstly the features engineering is summarized including nodes attributes and link strength. Then, the pre-filtering based on a simple linear threshold model will be explained. And finally, we

will introduce how we adapt the Supervised Random Walk method to rank the filtered candidates.

A. Feature Engineering

Here we formally summarize the features we utilized. As mentioned above, in content-oriented social media platform users' behaviors usually indicate their preferences. At the same time, due to the social factors, users' decisions could be affected by friends or followees. Thus, both the behavior records and social characters should be considered. Depending on the data set, we extract two general sets of features to represent the nodes' attributes and links' strength separately, which are listed as follows.

Nodes attributes are extracted to represent the characters of individual users, which could be roughly categorized as ***user profile*** and ***user behavior***. The former ones contain personal information in which four features are selected, including *the gender, birthday, location and keywords in self-introduction*. The later ones are extracted from users' records in the social media platform, including *the number of followers, viewing history, rating history and tagging history*, which indicate their preference and topical authority in the SNS.

Link strength are extracted to represent the social interactions, which directly indicate the linking potential. Borrowing idea from traditional link prediction methods, here we select three classic features, including *the common neighbors, common viewing history and common tags*. In one word, these three features encompass both the social transitivity and the pairwise similarity metrics to comprehensively reflect the link strength.

B. Pre-filtering with Linear Threshold Model

Then we discuss about the pre-filtering process. In the online social media platforms, users prefer to interact with those who share similar preferences. At the same time, users will be affected by friends or popular trend, which reflects the social transitivity. Simple classifier may hardly conclude these rules. Besides, majority of training samples are negative due to the sparsity, since there are thousands of users in the network while only a few are connected, which results in severely imbalanced classification.

To deal with these problems, here we introduce a simple unsupervised method based on *Linear Threshold* (LT) model, which is one of the basic models in social influence simulation task. As discussed in [9], the LT model follows the intuitive assumption that the social influence could be counted as accumulating effects from activated neighbors, and if the influence surpasses a certain threshold, the node will be activated and further start to influence its inactivated neighbors. As the LT model introduced, we could now transfer the filtering task as social influence process. And definitely, for each node to be predicted, we will treat it as the initial node to filter candidate links.

There is another important issue about how to determine the influence strength in the LT model. Here we choose the tags, which is mentioned as individual feature in Section IV-A, to calculate the strength as follow:

$$S(u_i, u_j) = Cos(\mathbf{t}(u_i), \mathbf{t}(u_j)) = \frac{\mathbf{t}(u_i) \cdot \mathbf{t}(u_j)}{||\mathbf{t}(u_i)||_F * ||\mathbf{t}(u_j)||_F}. \quad (1)$$

Here $\mathbf{t}(u_i)$ represents the tag vector (term frequency vector) of node u_i. Based on the formulation, the strength will be easily estimated and the candidates will be filtered with a given threshold, while related discussion for threshold is mentioned in Section V.

C. Ranking candidates with SRW model

Finally, we discuss how to utilize the SRW model to rank the filtered candidates. We introduce the random walk with restart to model the features and network factors simultaneously. And also, as the labeled samples are nearly balanced after filtering, we could now solve it as a supervised learning task, i.e., utilizing the supervised random walk model.

The method of *Random Walk with Restart* (RWR) is common used in ranking graph nodes, since it makes full use of the attributes and network structure. Also, as the *"restart"* scheme adopted, which means during the random walk process, one can jump back to the start under a certain probability, the random walk score will not only reflect the authority in social network, but also the tight connection with start points. In other words, nodes with high score are more likely to be followed by the start node (the node to predict links) in the future.

Naturally, this method also needs to estimate the transition probability. Thus, we introduce the Supervised Random Walk [3] to rank the nodes, while at the same time refine the weights of features. The SRW model shares the similar random walk scheme with RWR, except the target of minimizing the loss function as follow:

$$min_w F(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2 + \lambda \sum_{u_i \in S} \sum_{u_p \in P_i, u_n \in N_i} loss(p_{u_i,u_p} - p_{u_i,u_n}). \quad (2)$$

In which S represents the source nodes set to predict links, P_i and N_i separately indicate the positive (existed link) and negative samples (not followed) of u_i, and $\mathbf{w}$ represents the weights of features, which will be learned during the training process. Definitely, the SRW targets at appropriately ranking the partial relationship between pairs of positive and negative samples. Here the loss function represents the error ranking loss which is formulated as:

$$loss(x) = \frac{1}{1 + e^{-x/b}}. \quad (3)$$

Since the SRW model is actually a learning-to-rank framework, after the model executed, we could derive the ranking of candidates. And then the top ones will be treated as the potential links in order.

Table II
STATISTICS OF DATA

User	Edge	Source user	Edge of source user
26,384	95,169	1,521	46,081

V. EXPERIMENTS

Data Set. We test our link prediction framework on the real world social data collected from the ihou.com, an online karaoke and social website, from July 2011 to October 2012. We find a long tail existing since only a few people follow a lot while the majority only follow a few. To ensure the training accuracy, only the users who follow more than 10 users are selected (i.e., we will predict links for them)[3] as source users and all of the users are covered to construct the network at the same time. Then, we split the linkage data into two parts according the the timestamp, i.e., the training set to learn the parameters w in SRW and test samples for validation. Statistics of dataset are show in Table II. By splitting, 21,852 edges exist in the training data and 24,229 edges in the test data for all the 1,521 users, which indicate extremely sparse situation.

Baseline Methods. We name our framework as Supervised Random Walk with pre-filtering (PF-SRW). For comparison, we choose the following baselines:

- Common neighbours (CN) [2]. It is a typical example of unsupervised method for link prediction. The basic idea of this method is simple and easy to implement. Since we consider only one-way links, we adapt the measure as common followees.
- Random walk with restart(RWR). The method has been briefly introduced in Section IV-C, which utilizes the network structure and "restart" scheme to highlight the significance of the start point.
- Supervised Random Walk (SRW) [3]. Also introduced in Section IV-C, and the difference with our framework is that no pre-filtering functioned here.

Evaluation Metrics. For each user u in the set of source users S we predict top-k nodes, and the prediction result is denoted as $L(u)$ while $T(u)$ indicates correct answers. Three metrics are chosen here, the precision indicates how many predictive links are correct and the recall measures how many correct links are predicted, which are calculated as follow:

$$precision = \frac{1}{n}\sum_{u\in S}\frac{|L(u)\cap T(u)|}{k} \quad recall = \frac{1}{n}\sum_{u\in S}\frac{|L(u)\cap T(u)|}{|T(u)|}$$

Then the F1-measure indeed reflects the balance between precision and recall as:

$$F1-measure = 2\cdot\frac{precision * recall}{precision + recall}$$

Definitely, higher scores of all the three measures indicate better results.

Generate Candidates. To evaluate the utility of the first step in our framework, we compare the candidates generated for only SRW and our PF-SRW framework. The statistics are shown in the Table III. We can see that our pre-filtering process generates fewer candidates for each user, while the candidates contain a higher proportion, i.e., the hit rate of ground truth in the test data.

Table III
COMPARE FOR DIFFERENT CANDIDATES GENERATION METHODS

Algorithm	Candidates size	Average candidates	Useful size	Proportion
SRW	1,010,297	664.2	7507	30.98%
PF-SWR	951,842	625.8	10886	44.93%

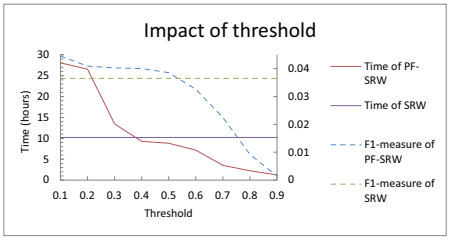


Figure 2. Efficiency and performance under different thresholds

Impact of Active Threshold. In linear threshold model, one node gets activated when the accumulative influence from activated neighbors exceeds the threshold. If the node is activated, it will be chosen as a member of candidates. Here the active threshold is set as unified to ease the modeling. The threshold will influence the size of candidates, since with larger threshold, the nodes are difficult to be activated, which leads to less candidates for a certain user. We compare the efficiency and performance between PF-SRW running under different thresholds with only SRW as standard. The result can be seen in Figure 2. We can find that smaller threshold leads to less time complexity but poorer performance. For instance, when threshold is 0.4, it takes less time to make prediction, while the F1-measure value is poorer than only SRW when threshold is greater than 0.6. Based on the results, we could make a trade-off for effectiveness and efficiency with adjusting the threshold.

Impact of Restart Parameter α. α determines the probability to jump back to the start during random walk. Small α allows the node jump far while large α constraints the jumping distance to stay only around the start points. We set α ranging from 0.1 to 0.9 and show the corresponding performance in Figure 3. When α is less than 0.6 the performance increases as α increases, and when α is larger than 0.6 the performance becomes poorer. Thus, we fix its value equal to 0.6 for the other experiments.

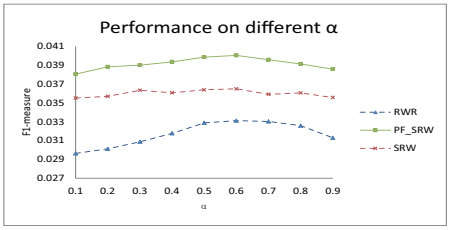
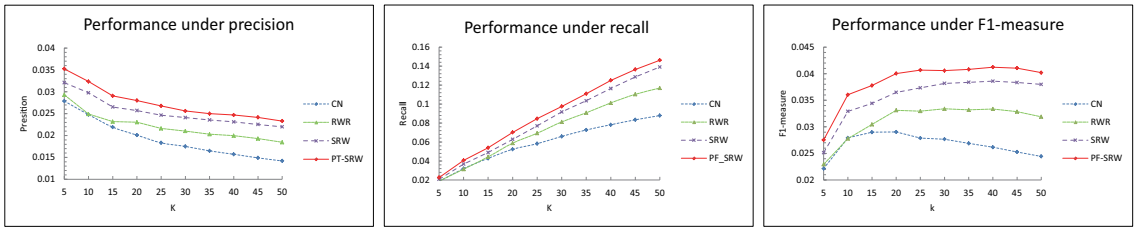


Figure 3. Impact of restart parameter α



(a) Top-k comparison under precision (b) Top-k comparison under recall (c) Top-k comparison under F1-measure

Figure 4. Comparison under different metrics

Comparison Under Different Metrics. The final performance comparisons based on different measures are shown in Figure 4(a),4(b) and 4(c), respectively. From these figures we can see that precision decreases when k ranges from 5 to 50 while the recall increases as well. The reason is that the more links we predict, the more correct links are revealed in the test data. However, the predictions ranking in the bottom do not capture as many correct links as the top ones. For the CN method, the F1-measure begins decreasing when k is more than 20, while for other methods (including ours), the F1-measure begins decreasing until k is more than 40. Thus, CN performs the worst. Among all these experiments, our PF-SRW performs the best with significant margin.

VI. Conclusion and Future work

In this paper, we proposed a two-steps framework to deal with the link prediction task in content-oriented (asymmetric) social network. Specifically, firstly a linear threshold based on tagging similarity network was executed to filter the adequate candidates, and then the adapted Supervised Random Walk (SRW) was executed to personally rank those candidates for the recommendation. Experiments on the real world data set indicated that our framework could effectively and efficiently predict the appropriate links, which outperforms the baselines including ordinary SRW with significant margin.

In the future, we will continue focusing on the social link analysis to distinguish the different motivations of connection in social media, e.g., the preference-oriented connection, or real world relationship based links. Some other related applications will also be studied, like annotating media contents based on social interaction analysis.

Acknowledgements. The work was supported by grants from Natural Science Foundation of China (Grant No. 61073110),the Key Program of National Natural Science Foundation of China (Grant No. 60933013) and Research Fund for the Doctoral Program of Higher Education of China (20113402110024).

References

[1] H. Kautz A. Sadilek and V. Silenzio. Modeling spread of disease from social interactions. In *ICWSM*, pages 322–329, 2012.

[2] L.A. Adamic and E. Adar. Friends and neighbors on the web. *Social networks*, 25(3):211–230, 2003.

[3] L. Backstrom and J. Leskovec. Supervised random walks: Predicting and recommending links in social networks. In *WSDM*, pages 635–644. ACM, 2011.

[4] A.L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[5] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM*, 2006.

[6] J. Hopcroft, T. Lou, and J. Tang. Who will follow you back? reciprocal relationship prediction. In *CIKM*, pages 1137–1146. ACM, 2011.

[7] H. Kashima and N. Abe. A parameterized probabilistic model of network evolution for supervised link prediction. In *ICDM*, pages 340–349. IEEE, 2006.

[8] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[9] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD*, pages 137–146. ACM, 2003.

[10] J.M. Kleinberg. Navigation in a small world. *Nature*, 406(6798):845–845, 2000.

[11] J. Kunegis and A. Lommatzsch. Learning spectral graph transformations for link prediction. In *ICML*, pages 561–568. ACM, 2009.

[12] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *JASIST*, 58(7):1019–1031, 2007.

[13] S. Milgram. The small world problem. *Psychology today*, 2(1):60–67, 1967.

[14] M. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):025102, 2001.

[15] A. Sadilek, H. Kautz, and J.P. Bigham. Finding your friends and following them to where you are. In *WSDM*. pages 723–732. ACM, 2012.

[16] H. Song, T. Cho, V. Dave, Y. Zhang, and L. Qiu. Scalable proximity estimation and link prediction in online social networks. In *SIGCOMM on IMC*, pages 322–335. ACM, 2009.

[17] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *ICDM*, pages 322–331. IEEE, 2007.

[18] T. Xu, D. Liu, E. Chen, H. Cao and J. Tian. Towards annotating media contents through social diffusion analysis. In *ICDM*, pages 1158–1163. IEEE, 2012.

[19] X. Yang, H. Steck, and Y. Liu. Circle-based recommendation in online social networks. In *KDD*, pages 1267–1275. ACM, 2012.