Friendship Prediction on Social Network Users

Kuan-Hsi Chen and Tyne Liang
Department of Computer Science
National Chiao Tung University
Hsinchu, Taiwan
{kschen,tliang}@cs.nctu.edu.tw

Abstract—**Undoubtedly friendship is one of key factors which keep social networking service users active and the whole community expanding. Hence, predicting friendships becomes an indispensable service provided by the platforms like Plurk, Twitter and Facebook. In this study, an empirical prediction resolution is presented by taking into account the interactions among Plurk users in Taiwan. Both response links and content information extracted from the interaction corpus are used as features in the implementation of the vector space machine based prediction. Experimental results show that the presented approach outperforms those bag-of-word based methods presented in previous studies.**

Keywords—social network; link prediction; friendship; interaction;

I. Introduction

Nowadays many people have been accustomed to communicate and make friends through social networking services like Facebook, Plurk, Twitter, Weibo and other platforms. However, it is challenging for the services to obtain all possible social connections of users due to relationship dynamics. Hence effective relationship prediction is demanded for popular services so that they can infer whether two users are connected or not, which kind of connection it is and when the connection will be established. Undoubtedly good prediction mechanisms can facilitate the applications like recommendation, advertisements and friend suggestion.

Essentially social networks may change and grow over time through user's participation and interaction. In previous studies, relationship or link prediction has been investigated in two aspects. One concerns modeling the evolution of social networks [1][2][3][4][5][6]. Nowell and Kleinberg [7] developed link prediction inferring co-author relationships by measuring the proximity between two users according to friend overlap, social paths, etc.. Hasan et al. [8] also tried to predict the addition of links on co-authorship networks with a support vector machine (SVM) model and reported 91% F-score. Choudhury et al. [9] proposed a linear regression model to predict future communication on threshold networks using network features and node features extracted from emails. Bramsen et al. [10] extracted n-grams and n-POS-grams from Enron corpus and trained a SVM model for social power prediction.

On the other hand, finding the missing links from social networks is also concerned [11]. Popescul and Ungar [12] employed logistic regression model to select features from relational data and obtained 90.9% accuracy at predicting citation links made in scientific literature. Leskovec et al. [13] applied theories of balance and status on a logistic regression classifier to forecast whether a user trusts another one or not by extracting the in-degrees, the out-degrees, and other features from these two users and their friends. Yang et al. [14] predicted friendships by a factor-based random walk model with the extraction of the features like common interests and friends. In [15], Sun et al. extracted topological features from DBLP bibliographic network to learn a generalized linear model for time prediction. On the other hand, Gilbert and Karahalios [16] applied an effective statistical method and employed age difference, groups in common and mutual friends for friendship strength prediction. Xiang et al. [17] trained a latent variable model for relationship strength measurement by employing user profiles and interactions.

However, most of the researches mentioned above dealt with small datasets or predicted relationships on specific networks like co-authorship networks, which are drastically different from the networks preserved by social networking services. It becomes challenging to predict the links on a large-scale and sparse-connection network for the diversities of user interaction and relationships. In this paper, we presented an empirical friendship prediction by analyzing the user interactions on a realistic social network preserved by Plurk[1], a popular microblogging service in Taiwan. Except link features which are concerned in previous studies, sentiment indication and response content are taken into account in order to overcome the sparse connection exhibited in such large-scale network. The prediction is built on the basis of support vector machine for its success in many classification applications. The proposed approach is verified on a real corpus in terms of different measurements. Experimental results on Plurk dataset report that our method yielded 87% prediction accuracy on friendship prediction and outperformed two previous methods and two other popular learning models.

II. Plurk and Its Texts

As a popular platform for Chinese-speaking users, Plurk allows its users to post Plurks (messages), or responses in no more than 140 characters to share with their friends and fans. They are connected by "Friends" and "Fans". "Friends" on Plurk means a reciprocal connection, as "FRIENDS" to Facebook; "Fans" represents a nonreciprocal relationship like "follow" on Twitter. Figure 1 is an example of a Plurk thread (a Plurk message together with its responses) in which emotion

---

[1]http://www.plurk.com/

icons (like 😊 , 🐱) and qualifiers (like says) are used to express users' feelings and motivations respectively.

Fig. 1.   An example of a plurk thread

In order to develop the presented friendship prediction, we gathered the data from 5,628 users who responded respectively to the Plurks of Ying-Jeou Ma and Ing-wen Tsai (2012 presidential candidates in Taiwan) from July 31st, 2011 to Feburary 15th, 2012. For each user, we collected the profiles, Plurk messages and the responses, and friend connection by breadth-first search. Under the restriction of Plurk API, the data we collected are from 203,426 unique Plurk users and 13,185,045 unique friend pairs. Figure 2 displays the cumulative distribution function of the amount of "friends"in the dataset and its curve follows a power law distribution. These 203,426 users posted 13,271,338 Plurk messages and received 55,525,227 responses in total. It is also observed that the distribution of the average amount of Plurk message posted per day per person and this distribution follows power law too. Our collected data show that a Plurk user posted 0.74 Plurk message every day and each message receives 4 responses in average. A message and a response contain 19.1 and 12.1 tokens (namely, Chinese word, English word, icon, url, punctuation mark) respectively in average. In addition, the friend's responses contain less number of tokens (10.8 tokens) than the ones in those non-friend's responses (12.8 tokens). It is also found that Plurk users like to share URLs while they communicate to each other. There are 10.4 million URLs in the collected Plurk threads and out of them 82.7% (8.6 million) URLs are expressed in the responses. Furthermore, 6.5 million URLs out of 8.6 million URLs are found in friend's responses.

III.   OUR APPROACH

Our goal is to predict whether two Plurk users, v_i and v_j, are friends or not according to their social activities on Plurk. Hence such prediction can be framed as classification task. It is known that statistical learning models like SVM have been widely employed for their classification robustness on a large dataset. Nevertheless, their performance relies on appropriate selection of features extracted from the dataset at model developing. Different from previous researches focusing single type of features only, the presented approach considers multiple features which turn out to be useful even when few training data (2000 instances) are used. Following subsections describe the presented features which will be extracted from the collected Plurk threads and used in the presented SVM-based prediction.
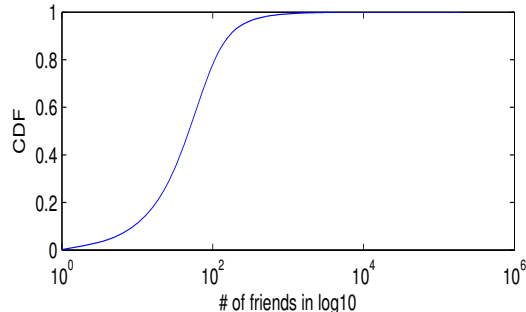

Fig. 2.   Distribution of number of "friends"

TABLE I.   FRIENDSHIP FEATURES

Feature	Description
f_1	same user-ID
f_2	same Plurk-ID
f_3	sentiment agreement
f_4	sentiment disagreement
f_5	average positive emoticon number
f_6	average negative emoticon number
f_7	average response length
f_8	average count of URL sharing in responses

A. Feature Extraction

In general, there are two types of features commonly addressed in various relationships prediction on social networks. One is personal features like personal friends, hobbies and background. So, friendship prediction can be approached on the basis of homophily theory [18] to evaluate how much commonality between two persons. The other is text features extracted from texts such as emails, forums, microblogs, etc. Table I lists the features which are employed for the proposed friendship prediction. They can be categorized into three types, namely, linkage, semantic, and behavior features, indicating the measurement of the commonality between two persons from three aspects. Linkage features (f_1 and f_2) are from the aspect of reaction activity; semantic features (f_3 to f_6) from users' attitudes to message content; behavior features (f_7 and f_8) from personal closeness.

Same User-ID.

A unique user-ID is assigned to each Plurk user when he or she is registered and a unique Plurk-ID is assigned to each Plurk message when it is posted. So we may calculate the amount of the same user-ID's to which two different users v_i and v_j (different user-ID) respond to the same message (Plurk-ID) that user v_k posts. So this feature f_1 can be treated as the linkage between two users and is defined as equation (1).

$$f_1 = \frac{|v_i \cap v_j|}{|v_i|} \times \frac{|v_i \cap v_j|}{|v_j|} \tag{1}$$

In equation (1), $|v_i|$ and $|v_j|$ are the number of users whose Plurk messages are responded by user v_i and v_j respectively in a given social graph. $|v_i \cap v_j|$ is the number of users who are responded by both v_i and v_j.

Same Plurk-ID.

TABLE II. AN EXAMPLE OF MESSAGE-LEVEL TAGGING FOR PLURK THREAD P_i

Plurk thread	Author	Polarity	Message-agree tag: $Tm(m_i, m_j)$
m_0 (Plurk)	U_1	Neutral	
m_1	U_2	Positive	$T_m(m_1, m_0)$:Agree
m_2	U_3	Negative	$T_m(m_2, m_0)$:Disagree
m_3	U_4	Positive	$T_m(m_3, m_0)$:Agree
m_4	U_1	Negative	
m_5	U_3	Positive	$T_m(m_5, m_4)$:Disgree
m_6	U_2	Negative	$T_m(m_6, m_4)$:Agree
m_7	U_4	Positive	$T_m(m_7, m_4)$:Disagree

TABLE III. DECISION TABLE OF MESSAGE-LEVEL AGREE TAG

Condition	Polarity		Message-agree tag: $T_m(m_i, m_j)$
	m_i	m_j	
$i < j$	Positive	Positive	Agree
$i < j$	Positive	Negative	Disagree
$i < j$	Negative	Positive	Disagree
$i < j$	Negative	Negative	Agree
$i = 0, i < j$	Neutral	Positive	Agree
$i > 0, i < j$	Neutral	Positive	Neutral
$i = 0, i < j$	Neutral	Negative	Disagree
$i > 0, i < j$	Neutral	Negative	Neutral

Same Plurk-ID feature f_2 can be seen as another indicator of linkage between two users v_i and v_j by counting the number of the same Plurk messages (Plurk-IDs) to which v_i and v_j respond. It is defined as equation (2).

$$f_1 = \frac{r(v_i, v_j)}{r(v_i)} \times \frac{r(v_i, v_j)}{r(v_j)} \qquad (2)$$

Here $r(v_i, v_j)$ is the count of messages replied by both v_i and v_j; r(v_i) and r(v_j) indicate the count of v_i's and v_j's responses respectively.

Sentiment Agreement and Disagreement.

In this paper, we check whether the sentiment polarity is agreed or not between each Plurk message and its responses. We start from investigating message-level sentiment agreement in each Plurk thread so as to understand user-level sentiment agreement. Table II is an example of a tagged Plurk thread in which we record message sequence, message author, message polarity and message-level agreement. The message polarity is detected by implementing Nave Bayes probability model presented in [19]; the message-level agreement is determined by comparing the polarity of each response with that of the most recent message given by U_1, the author of the Plurk. For example, both m_3 and m_5 given by U_3 are compared with m_0 while m_7 is compared with m_6. Similarly m_1 and m_4 given by U_2 are compared with m_0 while m_8 is compared with m_6. As shown in Table IV, user-level agreement is made if the the count of message-level agreement is greater than that of message-level disagreement; otherwise a disagreement tag is made. If the count of message-level agreement and disagreement are the same, then a neutral tag is made at user-level agreement (like U_1 and U_4 in Table IV).

We examined 13,271,338 Plurk threads (containing 55 million responses) and tag sentiment polarity (positive, negative, or neutral tag) for all messages and responses with the employment of two outer resources, namely NTUSD[2] ("National Taiwan University Sentiment Dictionary") and Plurk icon set.

TABLE IV. THE USER-LEVEL AGREE TAGGING RESULT FROM TABLE II

Author		Message level		User-level agree tag: T_u
Plurk	response	# of agree	# of disagree	
U_1	U_2	3	0	Agree
U_1	U_3	0	3	Disagree
U_1	U_4	1	1	Neutral

TABLE V. STATISTICS OF USER-LEVEL AGREE TAGS

	Agree %	Disagree %	Neutral %	Total
Friend	18.33	13.27	68.38	23.6M
Non-friend	17.44	13.25	69.31	3.8M
Plurk author	28.44	31.44	40.12	3.2M

Our statistical result shows that 60% of the message-level agreements are detected in friend's responses. Table V shows that there are more number of agreement and disagreement tags identified among friends' Plurk threads than those of non-friends' ones. Such observation can be also found from the statistical results of agreement and disagreement for all the user pairs who have communications. Out of 1,128,637 friend pairs, 69% pairs have at least two agree/disagree tags, but only 39% non-friend pairs have at least two agree/disagree tags. So the sentiment agreement at user level is taken into account the presented friendship prediction.

In this paper, agreement feature, f_3, and disagreement feature, f_4, are measured by equation (3) and (4).

$$f_3 = \sum_{PT_k \in Agree_{ij}} \left(\frac{1}{D(PT_k)} \right) \qquad (3)$$

$$f_4 = \sum_{PT_k \in Disagree_{ij}} \left(\frac{1}{D(PT_k)} \right) \qquad (4)$$

Here $Agree_{ij}$ is the Plurk thread set in which every thread PT_k has T_u = agree and $Disagree_{ij}$ is the set with $T_u = disagree$. In addition, the average response distance $D(PT_k)$ between user pair v_i and v_j in PT_k is taken into account since users are assumed to be more related as $D(PT_k)$ is smaller. $D(PT_k)$ is calculated by equation (5)

$$D(PT_k) = \begin{cases} avg(\sum_{pt_y \in v_j} d(pt_y)), & if \ PT_k \in v_i \\ avg(\sum_{pt_y \in v_i} d(pt_y)), & if \ PT_k \in v_j \\ avg(\sum_{pt_y \in v_i \ or \ v_j} d(pt_y)), & otherwise \end{cases} \qquad (5)$$

where pt_y is the response posted by one of the users of pair (v_i, v_j) in PT_k and $d(pt_y)$ calculates the distance between pt_y and the latest message posted by the other user .

Emoticons.

We assume that people may use more positive emoticons when responding to friends' Plurks. According to [20] [21] [22], we manually classify Plurk emoticon into positive and negative classes. Table VI is the statistics of emoticons from 55,000,000 responses which are classified according to the relationships among Plurk author and responders. For each response, we count the numbers of positive and negative emoticons. Then we calculate the average emoticon numbers over tokens for each set. We observe that more positive icons are used in friend's responses. Meanwhile, less negative icons are used in friends' interaction than those expressed

TABLE VI.   AVERAGE EMOTICON COUNT

	Friend	Non friend
# of positive icons per 100 tokens	1.55	1.43
# of negative icons per 100 tokens	0.45	0.58

in non-friend's communication. The result is identical with our assumption. Therefore, we calculated average amount of positive and negative emotion icons as features by equation (6) and (7).

$$f_5 = \frac{positive\_icon(R_{ij})}{|R_{ij}|} \qquad (6)$$

$$f_6 = \frac{negative\_icon(R_{ij})}{|R_{ij}|} \qquad (7)$$

Here R_{ij} is the response set between user v_i and v_j and $positive\_icon(R_{ij})$ counts the number of positive icons in R_{ij}. $negative\_icon(R_{ij})$ counts the number of negative icons in R_{ij}.

Average Response Length.

As mentioned in Section 2, friend's response length is shorter than non-friend's response length. In fact, it is indeed easier for friends to communicate with each other than the one to strangers. So the average response length f_7 between user v_i and v_j becomes an significant indicator for friendship prediction and it is calculated by equation (8).

$$f_7 = \begin{cases} \frac{len(R_{ij})}{|R_{ij}|} + \frac{len(R_{ji})}{|R_{ji}|} & if\ len(R_{ij}) > 0 \\ & and\ len(R_{ji}) > 0 \\ -(\frac{len(R_{ij})}{|R_{ij}|} + \frac{len(R_{ji})}{|R_{ji}|}) & otherwise \end{cases} \qquad (8)$$

R_{ij} is the response set which user v_i responds to v_j, and R_{ji} is the set from user v_j to v_i. $total\_len(R_{ij})$ and $total\_len(R_{ji})$ are the number of total length of set R_{ij} and R_{ji}.

Average Count of URL Sharing in Responses.

It is known from the collected data that Plurk users love to share multimedia resources to their friends. So the average count of URL sharing f_8 is considered and calculated by equation (9).

$$f_8 = \begin{cases} \frac{url(R_{ij})}{|R_{ij}|} + \frac{url(R_{ji})}{|R_{ji}|} & if\ url(R_{ij}) > 0 \\ & and\ url(R_{ji}) > 0 \\ (\frac{url(R_{ij})}{|R_{ij}|} + \frac{url(R_{ji})}{|R_{ji}|}) & otherwise \end{cases} \qquad (9)$$

R_{ij} is the response set which user v_i responded to v_j, and R_{ji} is the response set which user v_j responded to v_i. $url(R_{ij})$ is the amount of URLs in set R_{ij}.

B. Friendship Prediction Model

In this paper, we assume that a non-linear boundary exists among the friend pairs and non-friend pairs in the feature space defined above. Hence support vector machine (SVM) based model is applied for friendship prediction since it is capable of classifying data whose class boundaries are non-linear by projecting data from a lower dimensional feature space to a higher one via a kernel function to find a linear boundary.

Moreover, using SVM can help us classify those friend pairs which are near non-friends in the feature space since SVM determines the boundary according the nearest nodes. In this paper, we implement the SVM classifier with LIBSVM[3] for friendship prediction.

IV.   EXPERIMENTAL RESULTS

As described in Section 2, we collected the profiles, Plurk threads, and friend connections by breadth-first search from 5,628 users and their friends. In the experiments, we randomly selected 35,000 friend pairs and another 35,000 non-friend pairs as training data. The same amount of testing source is generated in the same way and from which seven sets of 5000 positive and 5000 negative instances are randomly extracted. VIII shows the experimental results yielded by the mentioned models in terms of accuracy, precision and recall which are defined respectively as follows.

$$accuracy = \frac{a + d}{a + b + c + d} \qquad (10)$$

$$precision = \frac{a}{a + b} \qquad (11)$$

$$recall = \frac{a}{a + c} \qquad (12)$$

Here a, b, c and d are the number of classification results as indicated in Table VII.

TABLE VII.   CONTINGENCY TABLE

Truth \ Prediction	# of friend pairs	# of non-friend pairs
# of friend pairs	a	b
# of non-friend pairs	c	d

A. Model Comparison

First, our SVM-based prediction is compared with two previous approaches based on bag-of-word (BOW) model. One is Classification Based on Associations approach (CBA for short) proposed by Liu et al. [23]. In the prediction experiments, CBA is applied by treating and selecting those common words in Plurk threads as association rules. To implement CBA, we generate a transaction for each response. The transaction contains the nouns, verbs, adjectives and adverbs occurring in one specific corresponding response, and a friendship tag indicating the relationship between Plurk author and its responder. In this paper, the CBA classifier is implemented using "LUCS-KDD implementation of CBA"[4] .

For method comparison, we implemented the LIBSVM and extracted the binned text features in the same way as done in [10] who calculated the feature scores of all one to six token n-grams and one to three POS tag n-grams from the data. This approach is called as SPM (social power model). In the implementation, we preserved those six token n-grams whose absolute frequencies and relative frequencies are greater than or equal to corresponding thresholds, namely, 5 and 1.5. Also we retained three POS tag n-grams whose absolute frequencies and relative frequencies are greater than or equal to thresholds,

---

[3]http://www.csie.ntu.edu.tw/ cjlin/libsvm
[4]http://cgi.csc.liv.ac.uk/ frans/KDD/Software/CBA/cba.html

TABLE VIII.   FRIENDSHIP PREDICTION WITH DIFFERENT METHODS

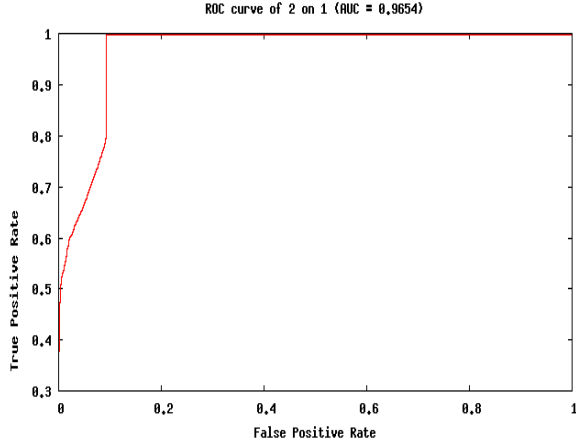| Model	Accuracy %	Friend prediction (w/ inactive users)		Friend prediction (w/o inactive users)	
		precision %	recall %	precision %	recall %
CBA	N/A	100.0	20.8	100.0	20.8
LIBSVM - binned text feature	60.6	95.6	13.5	95.6	16.8
LIBSVM - our feature	79.5	95.4	61.9	95.4	77.6

Fig. 3.   ROC curve of our prediction result.

namely, 20 and 5. The reserved n-grams are then binned according to their relative frequency ratios.

Table VIII shows that our method (indicated as LIBSVM-our features) can recommend a friend list for each user with 80% accuracy. It outperforms both CBA and SPM in the experiment containing 70,000 testing instances, showing the proposed features are more informative than words and POS tags for friendship prediction. On the other hand, the binned text features employed by SPM are not common so that most of the friend pairs were misclassified. The accuracy of CBA is not available since association rules used for prediction cannot help us identify negative instances. There are quite few rules discovered by CBA, making it unable to yield high recall.

We evaluate the discrimination ability of our method through receiver operating characteristic curve (ROC). Figure 3 demonstrates the ROC curve of our prediction result. The area under curve (AUC) value is 0.9654, implying that our approach can effectively discriminate friends from non-friends.

It also shows that lower recall is obtained since our method is limited by the frequencies of interactions between user pairs. Most of failure classification instances are attributed to zero number of interactions in many friend pairs. When we remove 6,956 friend pairs whose interactions are zero in our testing data, the recall becomes 77.6

B. Feature Evaluation

The importance of the addressed features is evaluated by leave-one-out test. Table IX indicates that f_7 (average response length), which was not used for friendship prediction in previous researches, turns out to be the most important feature. Such experimental result is consistent with the previous investigation described in Section 2 in which the lengths

TABLE IX.   FRIENDSHIP PREDICTION RESULT OF LEAVE-ONE-OUT TEST

Feature	5-fold cross validation Accuracy %
All-f_1	78.0
All-f_2	79.1
All-f_3	79.4
All-f_4	79.4
All-f_5	79.4
All-f_6	79.4
All-f_7	**76.5**
All-f_8	79.4

of friend's responses are different from those of non-friend's responses. The second most important feature is "the same user ID" which captures mutual friend information among user pairs. The leave-one-out results also prove that the links between users are important for prediction. As to the sentiment features, more advanced processing of response polarities is demanded for agreement evaluation. In addition, the sentiment targets of responses should be considered since responders may reveal sentiments toward not only Plurk authors but also topics or other responders.

Table X lists the prediction results by using single feature only. It also show that f_7 (average response length) turns out to be the most effective feature compared to other features; while f_1 (same user ID) feature yield the second best recall due to homophily phenomena, inferring that friends tend to have some mutual friends. Nevertheless, the networking service users may not be friends to each other even they have many mutual friends because the lowest precision was generated by using f_1.

By running forward selection, we obtain the best feature set which contains all the mentioned features except f_4 (sentiment disagreement) and f_6 (average negative Plurk icon). Such results indicate that the effect of negative sentiment is complicated when it is considered together with other features.

TABLE X.   FRIENDSHIP PREDICTION RESULTS OF SINGLE FEATURE

Feature	Accuracy %	Prediction on Friends		
		precision %	recall %	F-measure %
f_1	69.00	89.78	42.87	58.03
f_2	60.07	93.97	21.51	35.01
f_3	55.20	99.35	10.48	18.96
f_4	69.91	99.87	39.87	56.99
f_5	66.12	99.74	32.33	48.84
f_6	56.67	99.96	13.34	23.54
f_7	76.65	99.79	53.40	69.57
f_8	61.10	99.83	22.23	36.36

C. User Relationship Prediction

User relationship prediction is motivated to recommend a friend list by social networking services. So we randomly sampled 200 Plurk users as test set. For user v_i in the set, we generated a user pair set which consists of their friends' connections and 200 randomly sampled users who are not friends of v_i. In total, all the user pair sets contain 14,896

friend pairs and 40,000 non-friend pairs. We then applied our friendship prediction processing to predict the relationships of each user pair set. The results show that there are 83.6% average accuracy, 82.3% average precision and 31.7% average recall. Such results indicate that the proposed method can filter out those users who are not relevant and preserve the most relevant users as friend suggestions. Although the prediction result is accurate, the recall is lower because many friend pairs do not have any interactions. Our investigation shows that 8,058 user pairs, out of 9,912 falsely-predicted friend pairs, have no interactions. It is believed that most people using network services like Plurk would like to make friends with those they know each other, but they will frequently communicate with those who are their acquaintances.

V. Conclusions

As one of key functions provided by social networking services, friendship prediction indeed provides an effective way to identify their friends. In this paper, we proposed a SVM-based friendship prediction with the employment of multiple features extracted from Plurk user interactions. The experimental results report that such approach is capable to achieve 80% identification accuracy and 95% precision for Plurk users. Nevertheless, lower recall (61.9%) is obtained due to the fact that many user pairs do not interact frequently. As service platforms become popular, human networks will become more complex than before. So how to efficiently process and cluster various kinds of nodes and relationships should be taken into account for advanced applications like group detection and information acquisition.

References

[1] E. Jin, M. Girvan, and M. Newman, "Structure of growing social networks," *Phys. Rev. E*, vol. 64, no. 4, p. 046132, Sep. 2001.

[2] A. L. Barabasi, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," 2002.

[3] J. Davidsen, H. Ebel, and S. Bornholdt, "Emergence of a small world from local interactions: Modeling acquaintance networks," *Phys. Rev. Lett.*, vol. 88, p. 128701, Mar 2002. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.88.128701

[4] B. Taskar, M. fai Wong, P. Abbeel, and D. Koller, "Link prediction in relational data," in *in Neural Information Processing Systems*, 2003.

[5] H. Kashima and N. Abe, "A parameterized probabilistic model of network evolution for supervised link prediction," in *Proceedings of the Sixth International Conference on Data Mining*, ser. ICDM '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 340–349. [Online]. Available: http://dx.doi.org/10.1109/ICDM.2006.8

[6] W. Tang, H. Zhuang, and J. Tang, "Learning to infer social ties in large networks," in *Proceedings of the 2011 European conference on Machine learning and knowledge discovery in databases - Volume Part III*, ser. ECML PKDD'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 381–397. [Online]. Available: http://dl.acm.org/citation.cfm?id=2034161.2034187

[7] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, ser. CIKM '03. New York, NY, USA: ACM, 2003, pp. 556–559. [Online]. Available: http://doi.acm.org/10.1145/956863.956972

[8] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[9] M. De Choudhury, W. A. Mason, J. M. Hofman, and D. J. Watts, "Inferring relevant social networks from interpersonal communication," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 301–310. [Online]. Available: http://doi.acm.org/10.1145/1772690.1772722

[10] P. Bramsen, M. Escobar-Molano, A. Patel, and R. Alonso, "Extracting social power relationships from natural language," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 773–782. [Online]. Available: http://dl.acm.org/citation.cfm?id=2002472.2002570

[11] D. S. Goldberg and F. P. Roth, "Assessing experimentally derived interactions in a small world," *Proceedings of the National Academy of Sciences*, vol. 100, no. 8, pp. 4372–4376, 2003. [Online]. Available: http://www.pnas.org/content/100/8/4372.abstract

[12] A. Popescul, R. Popescul, and L. H. Ungar, "Statistical relational learning for link prediction," 2003.

[13] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 641–650. [Online]. Available: http://doi.acm.org/10.1145/1772690.1772756

[14] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha, "Like like alike: joint friendship and interest propagation in social networks," in *Proceedings of the 20th international conference on World wide web*, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 537–546. [Online]. Available: http://doi.acm.org/10.1145/1963405.1963481

[15] Y. Sun, J. Han, C. C. Aggarwal, and N. V. Chawla, "When will it happen?: relationship prediction in heterogeneous information networks," in *Proceedings of the fifth ACM international conference on Web search and data mining*, ser. WSDM '12. New York, NY, USA: ACM, 2012, pp. 663–672. [Online]. Available: http://doi.acm.org/10.1145/2124295.2124373

[16] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, pp. 211–220. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518736

[17] R. Xiang, J. Neville, and M. Rogati, "Modeling relationship strength in online social networks," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 981–990. [Online]. Available: http://doi.acm.org/10.1145/1772690.1772790

[18] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001. [Online]. Available: http://www.annualreviews.org/doi/abs/10.1146/annurev.soc.27.1.415

[19] Y.-T. Sun, C.-L. Chen, C.-C. Liu, C.-L. Liu, and V.-W. Soo, "Sentiment classification of short chinese sentences," in *Proceedings of the 22nd Conference on Computational Linguistics and Speech Processing*, ser. ROCLING '10, 2010, pp. 184–198. [Online]. Available: http://aclweb.org/anthology-new/O/O10/O10-1013.pdf

[20] J. Russell, "A circumplex model of affect," *Journal of personality and social psychology*, vol. 39, no. 6, pp. 1161–1178, 1980.

[21] R. Picard, E. Vyzas, and J. Healey, "Toward machine emotional intelligence: analysis of affective physiological state," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 10, pp. 1175 –1191, oct 2001.

[22] C. Yang, K. H.-Y. Lin, and H.-H. Chen, "Emotion classification using web blog corpora," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, ser. WI '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 275–278. [Online]. Available: http://dx.doi.org/10.1109/WI.2007.50

[23] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," 1998, pp. 80–86.