Users Connection across Social Media Sites Based On Users' Relationship Vector

Zhou YAN[1,*], Shu-dong LI[2], Wei-hong HAN[1], Bin ZHOU[1] and Wen-xiang HAN[1]

[1]candle531@126.com

[1] College of Computer, National University of Defense Technology, Changsha, Hunan 410073, China

[2] College of Mathematics and Information Science, Shandong Institute of Business and Technology, Yantai, Shandong 264005, China

Keywords: Online social networks, Multiple identities, Machine learning, Feature vector

Abstract. The identification and association of multiple identities in different online social networks (*osns*) is an important problem, and also is the basis for many applications. At present, most of technologies try to solve this problem by matching the username of social networks or calculating the similarity of a pair of users' personal information from different platforms. However, due to the anonymity of social networks, these methods often fail to identify and associate multiple virtual identities. In this paper, we propose a classification method based on machine learning. Our method jointly consider the time, the text and the topic of the similarity to construct the feature vector to characterize the user's relationship. And we use the feature vectors to train the classifier. The model is evaluated on real world dataset, the *twitter* and *sina weibo*. The experimental results show that our method is effective.

Introduction

Today,the online social networks have penetrated into all aspects of daily life, and also have played an important role in the impact of social politics, economy, culture and other fields of the industry. Different social networking platforms offer a variety of web applications [1], such as the exchange of information, games, chat and other services. It is a common phenomenon that a user has more than one account on different platforms. This phenomenon, which brings convenience to users, at the same time, also causes the information fragmentation and becomes the bottleneck of users' behavior analysis on social networks. It has important implications to associate with accounts on different social networks, such as open intelligence collection, find specific groups, specific information analysis and information dissemination or product recommendati-on.

Maybe it is an important evidence that the user's unique *ip* address or *mac* address to solve this problem. Unfortunately, the *ip* address and *mac* address in these platforms belong to the user's privacy information, it is difficult for us to obtain such information. Some methods are proposed by matching username or user's information. However, usernames in social networks are often anonymous and users' information is often error or unreal, as a result, which causes these methods cannot solve the problem effectively. So it is needed to propose another more effective method to connect multiple virtual identities from a real person.

In this paper, we calculate the number of the posts from two users at each time interval and their similarity. Then, we get the similarity of two users' posts by the*tf-idf* model. And we calculate the topic similarity between two users through the topic model. Finally we construct the relation vector between the users with the results of these three kinds of similarity, which as the input of the classifier. At the same time, we compare the three different classifiers in order to get a good classification effect.

Related Work

In this paper, we make a summary of research on the identification and correlation of multiple virtual identity.

In recent years, many methods to solve the problem of multiple virtual identity recognition and association are by comparing the user attributes in a number of platforms. Zafarani et al.[2]propose a method by calculating the string similarity of users' identity attributes such as usernames and names. Goga et al.[3]use the language model to compare the attributes of the blog post, such as the length of the posts. Bartunov et al.[4] solve the problem by the user's friend network attributes such as the number of friends and of ties with friends nature built into the graph model for comparison.

Other methods to solve this kind of problem are by clustering method. An unsupervised clustering algorithm based on [5] Morrison et al. is proposed, which is based on the role of users in the discussion board, and the method will be related to the different virtual identities of the same user. Chan et al.[6] use persistence, popularity, egocentric, networks, users of reciprocity and other features in the community to discuss the identity of the user's identity for clustering.

Yang et al.[7] use the writing features, but the stylistic features of no dependece on the topic. Therefore, this method more suitable for in a variety of themes and contents. Nie et al.[11] regards that the user's core interests will not be changed wiht the platform to change, they put forward a method of identity recognition by comparing the user's preferences. Dahlin et al.[8] present a method of combing the output results of the field matching, graph matching and the matching technology based on text. The accuracy of this method has been improved, but not experimental results have been given. Johansson et al.[9] draw on the framework of the paper[8], it has been improved and realized, and get better results, but the [9] does not take into account the same real people in different platforms released contents topic factors. We have made further expansion work on the basis of the work[9].

Problem Statement

Now we use the following definitions and concepts to define the issues that are being studied. User identity u_s from $osn_a$is denoted as source identity set, user identity u_c from osn_b is denoted as candidate identity set.

The progressively formed user image is a three tuple set *pu*, each of which is composed of a time vector *time*, a text representation vector *contents*, and a text topic vector *topics*, such as :

$$pu_i = \{time_i, contents_i, topics_i\} \tag{1}$$

,where the $time_i$ said user u_i each time interval within a certain time published contents number vector, $contents_i$ represents a quantitative representation of posts and $topics_i$ represents the topic distribution probability vector of ther user's blog post.

Provide a user u_s from osn_a and a user u_c from osn_b, we can get their user images pu_s and pu_c through statistics and calculations.We calculate the similarity vector

$$similar_{S,C} =< ti\text{-} similar, co\text{-} similar, to\text{-} similar > \quad (2)$$

,according to the users' image to determine whether they belong to the same real person i.

There are many ways to solve this problem. Such as, a heuristic method based on rules, collaborative approaches like crowd sourcing and manual tagging, and matching learning algorithm to associate the virtual identity.

As shown in Fig.1, we model the multiple virtual identity association problem, and the data is converted to a classification problem with the following three steps:

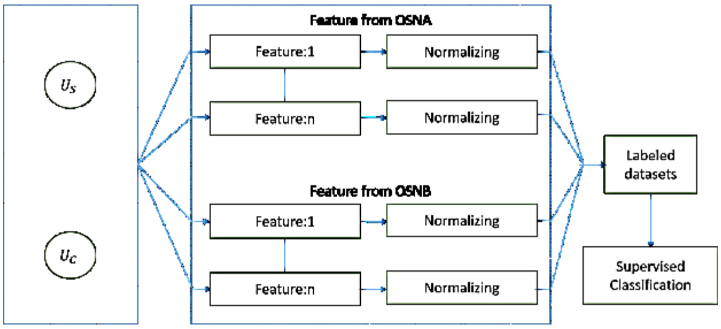


Figure1:user connection model

(1)feature extraction: extracting user's time, text, topic and other features;
(2)label the dataset: label "1" means that two users for the same real man, label "0" means that users are not the same real person;
(3)Training a classifier to establish the association of multiple virtual identity.

Connection Technology

Feature Extraction

When users are active on different platforms, they can always show the unique characteristics of their identity. Chen et al.[10] analysis users from multiple social networks showed that 85% of users in different social networking platforms can match 50% of the attributes.

MEQ Model.

Horne et al.[14] put forward a kind of "morningness/eveningness questionnaire"(meq) model, they identify the user's identity according to the habit of users, who are defined as "morning type" and "evening type". "Morning type" refers to those who just get up early in the morning, the more active users, "evening type" refers to those who become active in the evening. According to the model proposed by [14], the time of the posts which users publish on different platform can be an important clue to identify whether multiple identities belong to the same reality. In our experiments, we construct the time-posts curve graph by the number of posts in a period of time. We put this period of time by the time window of the same size. For example, assuming $u_{s,i}$ posts 100 blogs in tree months, we divide the period into 24 zones, respectively, 00:00, 01:00, 02:00, 03:00, ... , 23:00.

First, we construct a feature vector corresponding the number of posts in each time period *tc*:

$$tc{=}{<}tc_0,...,tc_i,tc_{i+1},...,tc_n{>} \quad (3)$$

We are more concerned about he distribution of the number of posts in a period of time, rather than the number of published. So, the next step is to use the min-max standard formula to normalize the *tc* vector:

$$\text{min-max:}tc_i{}^* = \frac{tc_i - tc_{min}}{tc_{max} - tc_{min}} \quad (4)$$

tc_i represents a normalized element in the *tc* vector,tc_{min} represents the minimum value in the *tc* vector,tc_{max} represents the maximum value in the *tc* vector,$tc_i{}^*$ represents the normalized result of tc_i. Normalized results tc' of *tc* vector using min-max standard formula:

$$tc' ={<}tc'_0,....,tc'_i,tc'_{i+1},...,tc'_n{>} \quad (5)$$

Fig.2 shows time-posts curve graph:

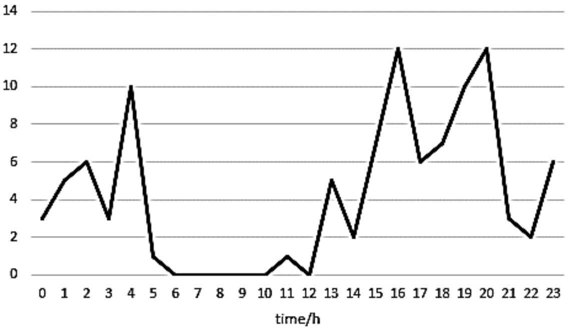


Figure2:time-posts graph

By using the above model, a normalized *tc* vector is created for each user. As shown in Fig.3 is a sample of u_s and u_c distribution diagram.

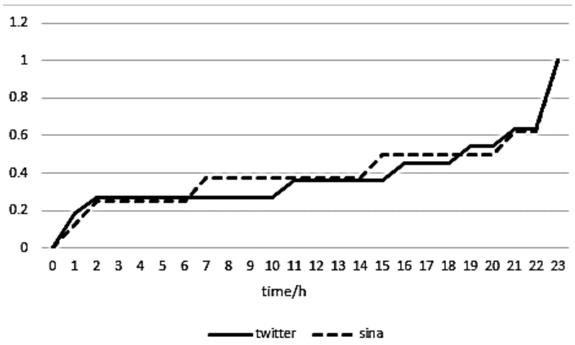


Figure3:U_S and U_C distribution diagram

Once the *tc* vector is constructed, we compute the similarity between the source user's and the candidate user's time-posts via the Euclidean distance. The shorter the distance between the two users the more likely to be the same user. The Euclidean formula between vector tc_s and tc_c is as follows:

$$\mathrm{d}(tc_s, tc_c) = \sqrt{\sum_{i=1}^{n}(tc_{s,i} - tc_{c,i})^2} \quad (6)$$

In fact,only by comparing the two user's time-posts of vector similarity, we cannot fully explain whether the two users belong to the same user. But we can regard the average value according to the distance of each time interval as a feature of the virtual identity.

Text Similarity Model.

A user's writing habit is formed in the long term, which is difficult to avoid completely [6]. Atig et al.[12] point out that the word is the most effective feature set, our experiment calculates the similarity of posts by the word in the statistics.

We extract the key words in the blog through the *tf-idf* model to calculate the weight of each word.And then we calculate the frequency of the key words to build a blog in an expression vector *tfv*=<$v_0, \ldots, v_i, v_{i+1}$,...,$v_n$>. Calculating the cosine similarity between tfv_s and tfv_c:

$$\cos\theta = \frac{\sum_{i=1}^{n}(v_i^s \times v_i^c)}{\sqrt{\sum_{i=1}^{n}(v_i^s)^2} \times \sqrt{\sum_{i=1}^{n}(v_i^c)^2}} \tag{7}$$

The v_i^s presents the frequency of the i-th word from source user,v_i^c presents the frequency of the i-th word from candidate user. The more close to the cosine value of 1 the more similar the two posts.

By calculating the similarity of the text, we can get the similarity of the writing habits of the blog which published by two users. We identify the similarity of posts as one of the important features of the users.

Topic Similarity Model.

Novak et al.[13], through the analysis of the text word, sentence, emotion to dig the user's writing habits, as a characterization of the identity of a main feature to solve the problem of multiple virtual identity. However, when users are active on multiple platforms, sometime will not publish the same content. The [11] supports that a user's interest is usually not changed according to the platform, in other words, the user's multiple platforms are related to topic is always similar. We use the topic model algorithm to calculate the probability of each topic released by the user, which constitutes the topic of the probability vector *topics*, then calculate the *jensen-shannon* distance of these *topics* vectors to judge the topic similarity of user's posts.

Lda[15-18] is the limited topic distribution of each text in the dataset, each topic is a three layers Bayes mode that is organized in the form of probability. Lda ignores the order of words in the text and the grammatical relations, the text with a *dirichelet* distribution of the distribution of *kdimensional implicit random variables* to represent the document, which to simulate the formation process of the document.

In [20], a more accurate *lda* generation model is obtained by adding the *dirichelet* a priori to the β parameter. We use the *lda* model to convert the user's posts into topic vector *topics*=[$t_0, \ldots, t_i, \ldots, t_k$], t_i indicates the probability value of the blog distribution on various topics, which *k*represents the number of topics. There are many methods to calculate the similarity between two probability densities, and we use the *js distance* formula to calculate the similarity:

$$js(topics_s||topics_c) = \frac{1}{2}[kl(topics_s||\overline{m}) + kl(topics_c||\overline{m})] \tag{8}$$

,where $\overline{m} = \frac{1}{2}(topics_s + topics_c)$, and *kl* divergence is

$$kl(topics_s||topics_c) = \sum_{i=1}^{|topics\ s|} topics_s \cdot \log\frac{topics\ _s}{topics\ _c} \tag{9}$$

,here,$topics_s$ indicates the topic vector of blogs from source user,$topics_c$ represents the topic vector of blogs from candidate user.

We get the *topics* vector of each blog by computing the user u_s and the user u_c, calcuating the *js distance* between them, and calcuating the average value of the distance as their topic similarity. The average value of their object distance is smaller, the more they are likely to be the same as the real man$i$. We regard the *js distance* average as a major feature to join our experiments.

Classifier

We combine the above features to form the similarity vector between users, using *gaussian bayes classifier*, *decisiontree classifier* and *randomforest classifier* for experimental comparison.

Gaussian Bayes Classifier.

Because the *naive bayes classifier* cannot effectively use the dependency information between attributes. Therefore, we use a *derivative bayes classifier* called *gaussian bayes classifier*proposed by the [21]. *Gaussian bayes classifier* uses the classical *gaussian function* to estimate the edge density, which makes the classifier can better fit the data and effectively improve the information between attributes.

We put the test set into the trained classifier, and get the probability that the testset belongs to each category. The classification of the maximum probability of selection is the classification of the test data.

In our experiments, we use the labeled vectors $similar_{s,c}$ to calculate the conditional probability estimates of each feature attribute in each category.Using the *gaussian function* to calculate the conditional probability of each attribute, the formula of *gaussian function* is as follows:

$$\mathrm{p}(x_i|\mathrm{y}) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right) \tag{10}$$

As each characteristic attribute is independent of the condition, according to the Bayes theorem, given a testset $similar'_{s,c}$, it can be calculated that it belongs to the probability of each category:

$$\mathrm{p}\left(\mathrm{y}\middle|similar'_{s,c}\right) = \frac{p(y)p(ti\text{-}simila\,r_i|y_i)p(co\text{-}similar\,_i|y_i)p(to\text{-}similar\,_i|y_i)}{p(ti\text{-}similar\,,co\text{-}similar\,,to\text{-}similar\,)} \tag{11}$$

DecisionTree Classifier.

Decision tree classifier is a tree structure of the classifier, each internal node represents a test on the attribute, each branch represents a test output, each leaf node represents a class.

In our experiment, we build the *decisiontree model* by using *id3* algorithm and *c4.5* algorithm. The labeled $similar_{s,c}$ vector is used as the training set. According to the mapping relationship between the object attribute and the object value, the test set is predicted.

RandomForest Classifier

Decisiontree algorithm has many good characteristics, such as the training time complexity is low, the prediction process is relatively fast, and the model is easy to show. At the same time, a single decision tree has some defects, such as easy to over-fitting, although by pruning can reduce this kind of situation, but it still cannot avoided completely. We investigate the *randomforest algorithm* in the experiment.

Randomforests are the combination of decision trees, each decision tree is trained to produce a new set of data from the original data set, the result of random forest decision is the result of the most decision trees. It uses the bootstrap re-sampling method to extract multiple samples from the original sample, and then, a decision tree model is established for each bootstrp sample. Finally, a number of decision trees are combined to predict, and the final prediction results are obtained by voting. Its principle is shown in Fig.4:

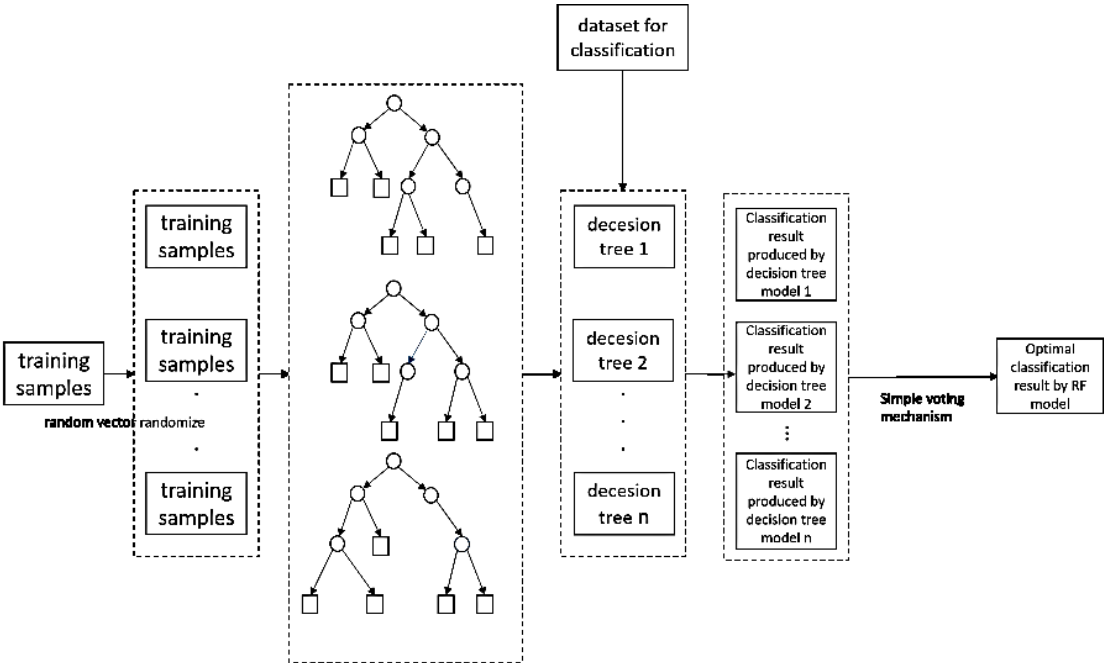

dataset for classification
training samples
random vector randomize
training samples
training samples
training samples
decesion tree 1
decesion tree 2
decesion tree n
Classification result produced by decision tree model 1
Classification result produced by decision tree model 2
Classification result produced by decision tree model n
Simple voting mechanism
Optimal classification result by RF model

Figure4:the principle of RandomForest

As we can see from Fig.4, we use the training data set$similar_{s,c}$ for training, generate a set of all decision trees {h(x,$similar_{s,c,i}$), k=1,2,...,n}, where x represents the number of training data set. Each decision tree model h(x,$similar_{s,c,i}$) has one vote to select the results of the test samples:

$$\bar{h}(\mathrm{x}) = \max_y \sum_{i=1}^{n} g(h_i(x) = y) \tag{12}$$

, where $\bar{h}(\mathrm{x})$ indicates the classification result of the random forest,$h_i(x)$ is the result of a single decision tree, y represents a classification target, *g(·)* as the indicator function. RF classification model using simple voting strategy to complete the final classification.

Experiments

We evaluate our model using three classifiers on *twitter* and *sina* data from two social network platforms. We crawl 2000 pairs of famous user data from these two platforms, which construction of the experimental data contains 2000 pairs of positive samples (belong to the same user's virtual identity) and 3998000 pairs of counterexamples (does not belong to the same user's virtual identity).

First, extract the release time of the posts of all users in the two platforms, the release time of the user's posts is processed by statistics and normalization, and the corresponding *tc* vector is formed, then, calculate the similarity between the *tc* vector of the users. The similarity demo as shown in Table 1.

Table1:TC similarity

Sina Users	Twitter Users	TC Similarity
5key	5key	0.589876672043761
5key	Jiu jiu de qi miao mao xian	1.6945653437983357
Bi ying	Bian zhi ren sheng wang	0.7659310429956397
Jiu jiu de qi miao mao xian	5key	0.8191151809672059
......	......	......

Next, to filter the stop words of posts, and text segmentation. Calculate the average similarity of posts of two users. The similarity demo as shown in Table 2.

Table2:Contents similarity

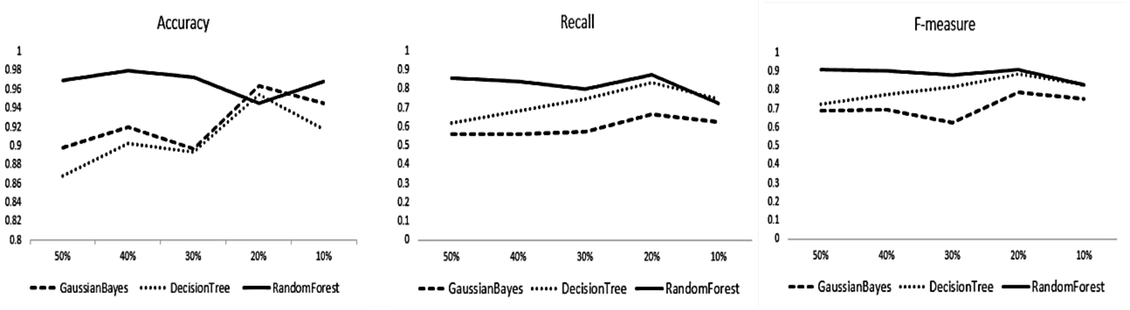
Sina Users	Twitter Users	Contents Similarity
5key	5key	0.9993788898642476
5key	Jiu jiu de qi miao mao xian	0.0076803650317414
Bi ying	Bian zhi ren sheng wang	0.9972188343613786
Jiu jiu de qi miao mao xian	5key	0.551630122809687
......	......	......

Use the *lda* model with a number of topics to get the topic probability vector *topics* of the user's blog, and calculate the topic similarity of the content published by the users. The result demo as shown in Table 3:

Table3:Topic similarity

Sina Users	Twitter Users	Topic Similarity
5key	5key	0.11505877945927491
5key	Jiu jiu de qi miao mao xian	0.08718837257862945
Bi ying	Bian zhi ren sheng wang	0.05261342114518521
Jiu jiu de qi miao mao xian	5key	0.053549736494677154
......	......	......

4000000 pairs of $similar_{s,c}$ vectors are generated by the three similarity features mentioned in the appeal.We use three different classifiers for 5-cross validation, and the results are shown in Fig.5.



a b c

Figure5: Multiple Classifier Results

In Fig.5, the range between 10% and 50% indicates the cross- validation rate. For example, 10% represents that use 90% of the data as a training set, and 10% of the data is used as a test set.

Fig.5(a) shows the accuracy of all algorithms with types of cross-validation. According to the result in it, for the accuracy, the ratio of *randomforest* outperform other algorithms in average accuracy. When the threshold value is in [20%, 50%], the ratio of *randomforest* has the best performance. But when the value in 20%, the accuracy of the *gaussianbayes* is the best. Fig.5(b) shows the recall of all algorithms. It can be seen that the recall of the *randomforest* and *decisiontree* have the highest recall. And Fig.5(c) shows the f-measure achieved by all algorithms. The *randomforest* has the highest f-measure. But when the threshold value is in [10%, 20%], the ratio of *randomforest* and *decisiontree* tend to be the same.

Conclusion and Future Work

We use three different dimensions to construct the user's feature vector: the time dimension, the content of the posts and the topic of the posts. Through the experiment of two different social network platform users, using the attributes of these three dimensions to unique describe the user's identity, and using supervised classifiers quickly identify two virtual identity is a realistic person. Our experimental results indicate that our method is effective in solving this problem.

In the future wok, we will continue to apply this method by adding more features, such as, user tag attributes, friends relationship graph attributes and so on. At the same time, we hope to develop an effective unsupervised learning model in next work, so that we can identify and associate multiple virtual identities faster and more accurately.

Acknowledgement

This research is supported by the State Key Development Program of Basic Research of China (no. 2013CB329601), NSFC (Grant No.61202362, 61472433), and the Project funded by the China Postdoctoral Science Foundation (no. 2013M542560, 2015T81129).

Reference

[1] Zafarani R, Liu H. Connecting users across social media sites: a behavioral-modeling approach[C]. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 41-49.

[2] Liu S, Wang S, Zhu F, et al. Hydra: Large-scale social identity linkage via heterogeneous behavior modeling[C]. Proceedings of the 2014 ACM SIGMOD international conference on Management of data. ACM, 2014: 51-62.

[3] Goga O, Lei H, Parthasarathi S H K, et al. Exploiting innocuous activity for correlating users across sites[C]. Proceedings of the 22nd international conference on World Wide Web. ACM, 2013: 447-458.

[4] Bartunov S, Korshunov A, Park S T, et al. Joint link-attribute user identity resolution in online social networks[C]. Proceedings of the 6th International

Conference on Knowledge Discovery and Data Mining, Workshop on Social Network Mining and Analysis. ACM. 2012.

[5] Morrison D, McLoughlin I, Hogan A, et al. Evolutionary Clustering and Analysis of User Behaviour in Online Forums[C]. ICWSM. 2012.

[6] Chan J, Hayes C, Daly E M[J]. ICWSM, 2010, 10: 215-218.

[7] Yang C C, Ng T D. Terrorism and crime related weblog social network: Link, content analysis and information visualization[C]. Intelligence and Security Informatics, 2007 IEEE. IEEE, 2007: 55-58.

[8] Dahlin J, Johansson F, Kaati L, et al. Combining entity matching techniques for detecting extremist behavior on discussion boards[C]. Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012). IEEE Computer Society, 2012: 850-857.

[9] Johansson F, Kaati L, Shrestha A. Detecting multiple aliases in social media[C]. Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ACM, 2013: 1004-1011.

[10] Chen T, Kaafar M A, Friedman A, et al. Is more always merrier?: a deep dive into online social footprints[C]. Proceedings of the 2012 ACM workshop on Workshop on online social networks. ACM, 2012: 67-72.

[11] Nie Y, Huang J, Li A, et al. Identifying users based on behavioral-modeling across social media sites[C]. Asia-Pacific Web Conference. Springer International Publishing, 2014: 48-55.

[12] Atig M F, Cassel S, Kaati L, et al. Activity profiles in online social media[C]. Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on. IEEE, 2014: 850-855.

[13] Novak J, Raghavan P, Tomkins A. Anti-aliasing on the web[C]. Proceedings of the 13th international conference on World Wide Web. ACM, 2004: 30-39.

[14] Horne J A, Ostberg O[J]. International journal of chronobiology, 1975, 4(2): 97-110.

[15] Blei D M. Probabilistic models of text and images[D]. University of California, Berkeley, 2004.

[16] Andrieu C, De Freitas N, Doucet A, et al[J]. Machine learning, 2003, 50(1-2): 5-43.

[17] Shudong Li, Lixiang Li, Yan Jia, Xinran Liu, and Yixian Yang, Identifying Vulnerable Nodes of Complex Networks in Cascading Failures induced by Node-based Attacks[J]. Mathematical Problems in Engineering, 2013, Volume 2013, Article ID 938398.

[18] D. Zhao, L. Li, H. Peng, Q. Luo, and Y. Yang[J]. Physics Letters A, vol. 378, no. 10, pp. 770–776, 2014.

[19] Yuanping Nie, Yan Jia, Shudong Li, Xiang Zhu, Aiping Li, Bin Zhou[J]. Neurocomputing, 2015. Article in press. doi:10.1016/j.neucom.2015.10.147.

[20] Griffiths T L[J]. Proceedings of the National academy of Sciences, 2004, 101(suppl 1): 5228-5235.

[21] Perez A, Larranaga P, Inza I[J]. International Journal of Approximate Reasoning, 2006, 43(1): 1-25.