Email Classification using Behavior and Time Features

YEQIN SHAO[1], QUAN SHI[1]*, YANGHUA XIAO[2], NIK BESSIS[3]*, PETER NORRINGTON[4],
[1]Center of Modern Educational Technology, Nantong University, Jiangsu, CHINA
[2]School of Computer Science, Fudan University, Shanghai, CHINA
[3]Department of Computing and Mathematics, University of Derby, Derby, UK
[4]Institute for Research in Applicable Computing, University of Bedfordshire, Luton, UK
hnsyk@ntu.edu.cn, sq@ntu.edu.cn, shawyh@fudan.edu.cn, n.bessis@derby.ac.uk,
peter.norrington@beds.ac.uk

Abstract

The various forms, flexible sending tricks and tremendous number of spam emails have brought great challenges to accurate email classification. In this paper, we present a behavior- and time-feature-based email classification method. Based on email logs, email social networks are built through the extraction of entities and relations from the email records using the MapReduce model. By combining behavior features from social networks and time features from email sending intervals, we adopt a Support Vector Machine based classifier to identify spammers and non-spammers. Compared with the current email classification methods, the advantages of our method are: 1) in addition to the behavior-based features, our method integrates the time feature to facilitate email classification; 2) to efficiently handle the vast number of emails, we employ the MapReduce model to extract the behavior- and time-based features on the email social network. Experiments on real email data of three years show that the proposed method achieves better classification accuracy.

Keywords: Social Network, Email Spam, Classification, Support Vector Machine

1 Introduction

Email, as a cheap, fast and effective communication method, has become an integral part of our life. However, email spam, as a nuisance product, remains severe. According to Kaspersky's 2014 spam report [1], the percentage of e-mail (spam) in all emails is 66.76%. The widespread propagation of spam wastes user time, consumes email provider resources (CPU, storage, bandwidth), and spreads fraudulent messages, or even viruses or malware which attack the user's computer. Email classification aims to distinguish spam from legitimate emails for alleviating the harmful effects on users and email providers. Therefore, accurate email classification method is highly desired for users and email providers. To achieve accurate email classification, many methods have been proposed to detect and classify emails. These methods mainly fall into two categories, namely: content-based and behavior-based.

Content-based methods identify spam emails according to their contents. The first type of content-based method compares the email content with spam keywords to find the spam emails [2]. The second type is machine-learning-based methods [3-8]. They extract various features from email contents and employ a classifier such as Naïve Bayes, neural network, decision trees, or Support Vector Machine to detect spam. It is intuitive to identify spam emails according to their contents. However, to elude antispam tools, spammers usually disguise the spam message by embedding it into normal content, especially pictures and cartoons, from which it is difficult to identify the spam messages. In addition, content-based methods need to scan the email content, which breaches the privacy of users.

Behavior-based methods, which identify spammers by investigating the behaviors of senders, provide a special insight into email classification, without the need to scan the email contents, avoiding violation of user privacy and speeding up the classification procedure. The first type of behavior-based method extracts the features of the sender without an email social network [9-13]. They depend on the basic behavioral characteristics of senders to detect spams. Recently, social network analysis has been attracting increasing research interest [14-16]. The second type of behavior-based method adopts social network analysis to describe the behavior of senders [17-21]. They construct an email social network and extract the features, i.e., in-degree, out-degree and clustering coefficients etc. to identify spam. However, there are two disadvantages to the current behavior-based email classification methods: 1) they do not take into account email sending interval characteristics, which can actually distinguished between spammers and normal users; 2) due to the vast number of emails, accurate and quick email classification is almost infeasible on a single workstation or server.

In this paper, we present a behavior- and time-feature-based email classification method to distinguish this complicated and burdensome spam from legitimate emails under the MapReduce model. The behavior features from email social networks and time feature from email sending intervals are extracted to capture the sender's characteristics. Then a feature selection method is employed to project all features into a discriminative space. Finally, a Support Vector Machine classifier is employed to differentiate spam and legitimate emails. The remainder of this paper is organized as follows. Section 2 formulates the email classification and introduces the framework of our method. Sections 3 and 4 present the details of email social network and time features. Sections 5 and 6 briefly review the MapReduce-based Feature Extraction and Support Vector Machine classifier. Extensive experiments are performed in Section 7. The paper concludes in Section 8.

2 Problem Statement

Suppose M is the set of emails, M_s and M_n denote the spam emails and legitimate emails, respectively. $M_s \bigcup M_n = M$, $M_s \bigcap M_n = \Phi$.

Email classification seeks a map $f : M \rightarrow \{spam, non-spam\}$ to determine if an email is spam or non-spam. We take Support Vector Machine (SVM) as a classifier of our approach and exploit the discriminative hyper-plane of it to distinguish spam from legitimate emails. Figure 1 shows the system framework.

The first part is training (inside the dotted rectangle). Based on the email records, the proposed method extracts entities (senders and recipients) and relations (email sending-receiving relationship) to construct an email social network. According to the email social network, behavior-based and time-based features are extracted and selected to obtain discriminative features. Then, these features are put into a classifier to train a classification criterion, which is stored in the decision database.

The second part is testing (outside the dotted rectangle). When a new email appears, the proposed method extracts the new email's information (say, sender and recipient) and classifies the email according to the classification criterion. To make the method adaptive, the new email information will be used to update the existing email social network and the classification criterion.

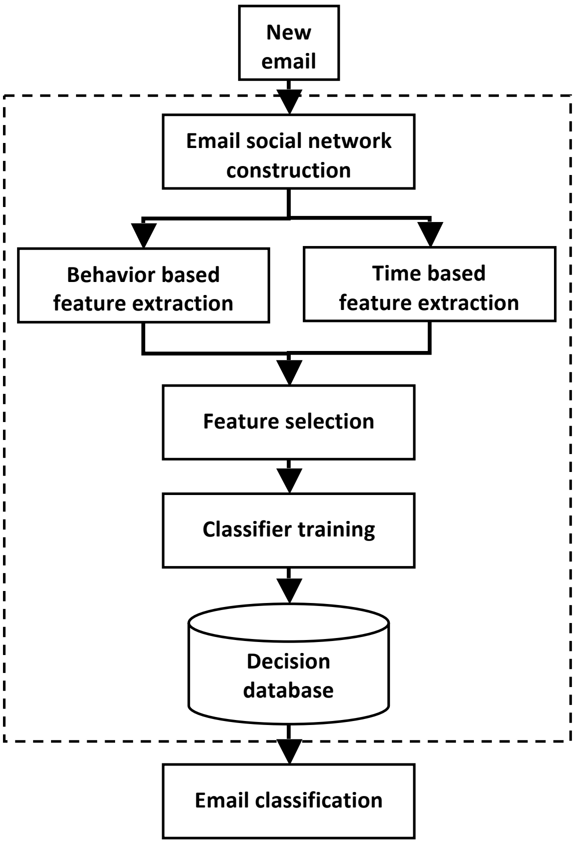

New email
Email social network construction
Behavior based feature extraction
Time based feature extraction
Feature selection
Classifier training
Decision database
Email classification

Figure 1: Framework of behavior and time feature email classification

3 Email Social Network Features

An email social network can be denoted as a weighted and directed graph $G(V, E)$, where V is the set of nodes each of which represents an email address. E is the set of edges. For nodes $v_i, v_j \in V$, if node v_i sends an email to node v_j, an edge $e_{ij} \in E$ which is from node v_i to node v_j will be added to the graph, and the weight of this edge is the number of emails from v_i to v_j. If v_i and v_j send emails to each other, the edge is bidirectional.

In general, a spammer tends to send out a large number of spam emails which will not be responded to. While, for normal users, they not only send emails to others, but also receive replies from others. From the perspective of a social network, the node corresponding to a normal user has a bidirectional edge. However, the spammer has only edges direct to others and the weight of these edges is almost 1, which is shown in Figure 2.

An egocentric network consists of a specific individual itself and their immediate contacts. In this paper, we mainly analyze the features of a one-hop egocentric network. Denote $G(V, E)$ as an egocentric network centered at node v; E is the set of nodes directly connected to v. We have extracted many features, such as out-degree, in-degree, the ratio of in-degree to out-degree, the average and variance of out-degree and in-degree, the number of bidirectional edges and so on; due to the space limitations here, six major features are introduced in detail.

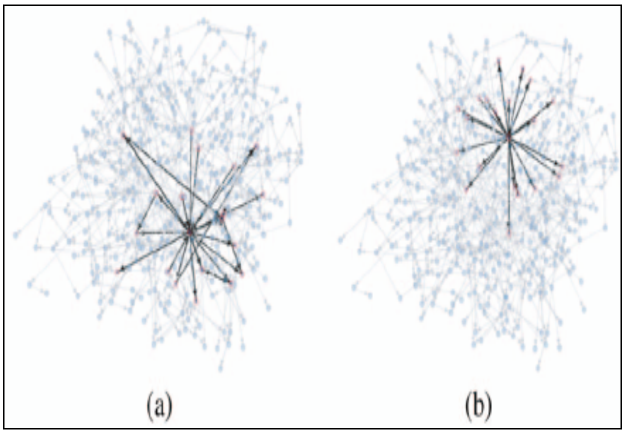


Figure 2: Egocentric network. (a) is a normal user and (b) represents a spammer (Source: [22])

Out-degree: Number of edges starting from node v. The larger the out-degree, the more users node v sends emails to. Generally speaking, in a certain time period, the number of mails sent by a normal user is not too many. As demonstrated in the top row of Figure 3, the out-degree of a normal user has an upper bound. While, spammers, for the purpose of broadcasting their messages, will send a tremendous number emails to as many people as possible, and hence their out-degree is large.

Average weight of outbound edges: Average number of emails sent by node v. A user usually sends emails to several recipients, so there are several outbound weights. The average weight of outbound edges reflects the number of emails sent out by node v. The larger this average, the more active a user's communication with others. In general, the average weight of a normal user is a random number, while that of spammer tends to 1.

In-degree: Number of edges direct to node v. The larger the in-degree is, indicates more users sending emails to node v. A normal user will receive emails from colleagues, friends and others, while a spammer only sends spam, which results in no-one sending emails to reply to it.

Reply ratio: Ratio between bidirectional edges and outbound edges of node v, which indicates the frequency of interaction with others. Typically, the communication between normal users is interactive, and hence the reply ratio will be a certain value. By contrast, the reply ratio of a spammer is almost 0.

Out-degree of IP address: Total number of users receiving emails from a IP address. To hide from anti-spam inspection, the spammer will change the email address constantly; in particular, each email uses a different sender address which is randomly generated. However, the IP address on which the emails are sent usually does not constantly change. There are still a great number of users receiving emails from a same IP address, which means the out-degree of the IP address is large. This feature can effectively separate spam and legitimate emails.

Outbound weight ratio of sender email address and IP address: Ratio between the emails sent by a specific email address on a IP address and all mails sent by all different senders on the same IP address.

$$F_{m\_ip} = out_m / out_{ip}$$

where $out_m = |V_{ik}|$ denotes the number of emails sent by a specific email address on a IP k address. $out_{ip} = \sum_{i=1,\dots,n} |V_{ik}|$ denotes the total number of emails sent by all different senders on the IP address k. V_{ik} represents the number of emails sent by i on IP address k. Since the spammer frequently changes the sender email address, the out_m of each sender address is almost 1, whereas the out_{ip} of the IP address is still large. As a result, the their ratio is small. For normal users, an IP address is used by one or a few sender email addresses, and hence the out_{ip} of this IP address is in a reasonable range. Therefore, for each sender email address, the outbound weight ratio of the sender email address and IP address is relatively bigger than that of a spammer, as shown in the bottom row of Figure 3.

These specific features as mentioned earlier on are summarized in Table 1. The cumulative distribution function of four typical features is illustrated in Figure 3.

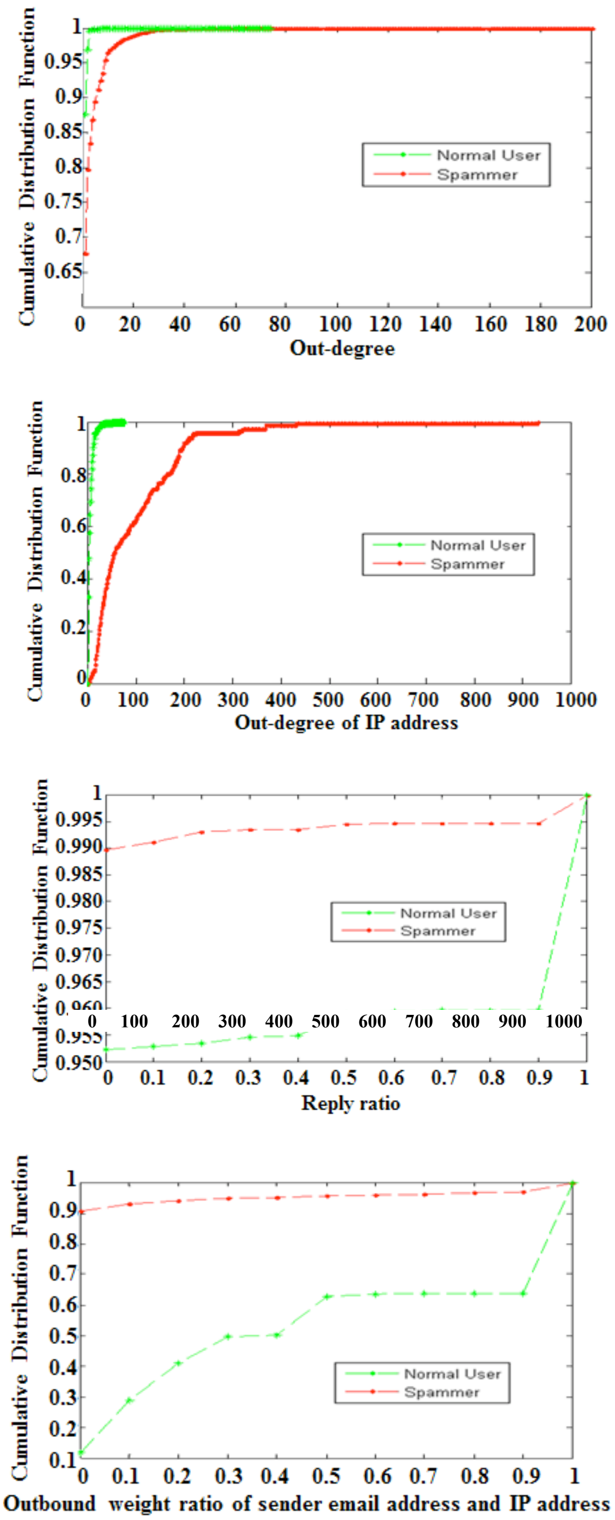


Figure 3: The major social network features of spammers and normal users based on our dataset in section 7: Experiments

Table 1: One-hop egocentric network features

Feature	***Formula***	***Description***
Out-degree	$\lvert V_i \rvert$	The number of outbound edges of node *i*
mean of weights of outbound edges	$\dfrac{\sum_{j=1,\cdots m} weight(e_{ij})}{m}$	The average weights of all outbound edge of node *i*
In-degree	$\lvert\{e_{ji}\}\rvert$	The number of inbound edges of node *i*
Reply ratio	$\dfrac{\lvert\{e_{ji}\}\rvert}{\lvert V_i\rvert}$	The ratio between bidirectional edges and all outbound edge of node *i*
Out-degree of IP address	$\sum_{i=1,\cdots n} \lvert V_{ik} \rvert$	The sum of out-degrees of nodes on a same IP address *k*
Outbound weight ratio of sender email address and IP address	$\dfrac{\lvert V_{ik}\rvert}{\sum_{i=1,\cdots n}\lvert V_{ik}\rvert}$	The ratio of outbound weights between a node *i* and its IP address *k*

4 Time Feature

To deliver the spam message to as many recipients as possible, a spammer needs to send spam to different email addresses continuously, therefore the email sending interval follows a certain regularity, while that of normal users is on demand and random. To present the time feature, we compute the sending interval of two adjacent emails and explore the distribution by the histogram of the intervals. The spammer sends a large number of spam emails in a short period of time, so the majority of its email sending intervals falls into bins near 0, while the distribution of normal users tends to be random. This is shown in Figure 4.

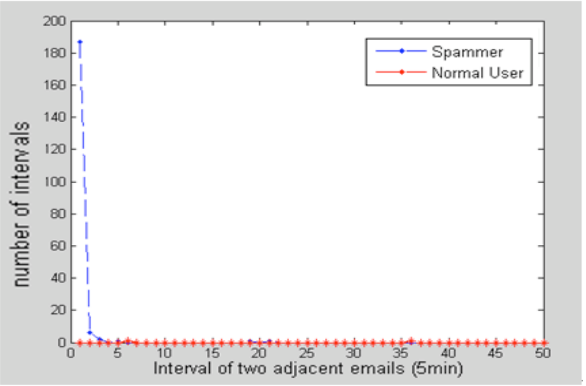


Figure 4: Histogram of sending interval for two adjacent emails of a user based on our dataset mentioned in section 7: Experiments

We employ the entropy of information theory to measure the imbalance and define the feature on time F_t thus: Given $m_{i,j}$ (j=1,2,…,n) as emails sent in a period of time by email address i, $t_{i,j}$ (j=1,2,…,n) as the sending time point, so the sending interval of two adjacent emails by email address i can be represented as $d_{i,j} = t_{i,j+1} - t_{i,j}$ (j=1,2,…,n). Based on the intervals, a histogram can be generated and the probability distribution can be presented as $p_i = p(x = i)$, where p_i indicates the proportion of emails of sending interval i in all outbound emails. Therefore the time feature can be defined as $F_t = -\sum p_i \ln(p_i)$. For spammers, the value of F_t is small, since the distribution is imbalanced; for normal users, the value of F_t is relatively big.

5 MapReduce-based Extraction

MapReduce [23], developed by Google, is a programming model for large data set processing in a parallel and distributed way on a cluster. The main components of MapReduce are Map and Reduce methods. The Map method filters and sorts key/value pairs to produce a set of intermediate key/value pairs. The Reduce method summarizes the intermediate values with the same intermediate key to generate the output key/value pairs. Both the Map and Reduce methods are highly parallel processable. To efficiently extract the aforementioned features from the massive email data, we apply the Java implementation of MapReduce, Hadoop, in our method. As Figure 5 shows, first, we extract the entities from the original email records on the email server to generate the sender-receiver list. The sender-receiver list contains items like <sender1, recv1, recv2, …>, which indicates that sender1 sent an email to recv1, recv2, etc. Then, we use the sender-receiver list as input of MapReduce to generate the desired features. To calculate various features, we need to determine the output key/value pairs. For in-degree feature of each entity, the output key is each email receiver, and the output value is the number of senders that delivers email to the specific receiver.

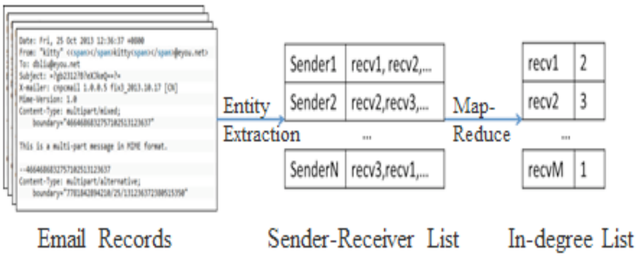


Figure 5: MapReduce-based in-degree feature extraction

6 Support Vector Machine (SVM)

Support Vector Machine (SVM) [24-25] is a generally applicable tool for classification and regression. Given m labeled training samples $\{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$, where x_i is the i-th sample of n dimensional features and y_i is the label of sample i (for the two-class case, y_i is usually -1 or +1), SVM uses the following formulas to seek an optimal discriminative hyper-plane $W^T x + b = 0$ in n-dimensional space to differentiate the samples into two different classes with maximal margin.

$$\min \tfrac{1}{2} w^T w + C \sum_{i=1}^{l} \varepsilon_i$$
$$s.t. y_i (w^T \phi(x_i) + b) \geq 1 - \varepsilon_i \ ,$$
$$\varepsilon_i \geq 0$$

where C is a cost coefficient, and $\phi(x_i)$ is used to map x_i to the high-dimensional space. In the feature space, the binary classification of emails is non-linear, and hence we employ RBF kernel $K(x_i, x_j) = \exp(-\gamma \| x_i - x_j \|^2), \gamma > 0$ in the SVM to train and classify the emails. The classification of a new sample is determined by its position ("above" or "below") with regarding to the optimal discriminative hyper-plane.

After the behavior-based and time-based features are obtained for each entity in the email social network, we apply a Fisher Separation Criterion to compute the discriminative ability of each feature and select six discriminative features (listed in Table 1) to classify the emails. The training samples with these selected features are put into the SVM to train a discriminative model which can classify the test emails into spam emails and legitimate emails. In the email feature space, the email classification is non-linear and hence, we adapt a Gaussian kernel based SVM to train the model and classify the emails.

7 Experiments

7.1 Experimental data

We conduct experiments on the real email data of three recent years (2010-2012) on an email server owned by an author's affiliation, Nantong University in Jiangsu of China. All entities and relations are extracted to construct an email social network and analyze the behavior features that we are interested in.

Table 2 takes the first week of each December of the three years, as an example, to demonstrate the email social networks. Since the number of emails is large, Hadoop is used to process and analyze all these data together.

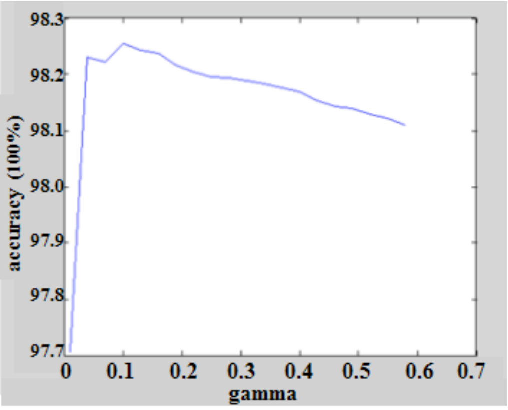
For this paper, we construct a Hadoop testbed consisting of a master node and nine data nodes. Each node has quadcore 2.90 GHz CPU, 16 GB memory, and 500 GB hard disk. HDFS is used as the cluster file system. All Hadoop nodes are connected with 1 Gigabit Ethernet cards.

Table 2: Demonstration of entities and relations of email social network

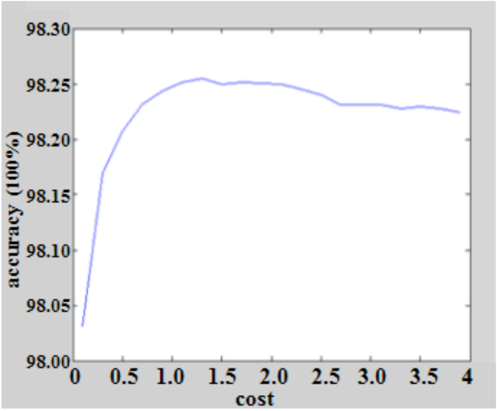
Year	*2010*	*2011*	*2012*
Entities #	30321	37837	54882
Spam #	23953	31026	46639
Legitimate (aka 'Ham') #	6368	6811	8233
Relations #	66964	70108	99349

7.2 Parameter tuning

To achieve good performance of SVM, appropriate hyper parameters and *C* (as mentioned in Section 5) need to be determined. We employ grid search to analyze the relation of hyper parameters and classification performance, as shown in Figure 5.



(a) Relation of *gamma* and accuracy



(b) Relation of *cost* and accuracy

Figure 6: SVM hyper parameters and classification accuracy

As Figure 6 shows, with parameter gamma increasing gradually, the classification accuracy increases rapidly at first and then decreases slowly. The accuracy maximum appears when gamma is 0.1. In the same way, with parameter cost increasing gradually, the classification accuracy increases at first and then decreases slowly. The accuracy maximum appears when cost is 1.3. Therefore, in our experiments, the gamma of SVM is 0.1, and the cost of SVM is 1.3.

7.3 Experimental results

For email classification, the classification accuracy is subject to two incorrect classification cases.

A) One case is mistaking the legitimate emails as spam emails (false positive), which may lead to removing the user's legitimate emails. For example, emails which are multicast by a user (teacher or supervisor) to inform members of a piece of notification or announcement, are usually mistaken as spam. Although multicasting means that bulk mails are sent in a short interval, the sending timing of a normal user is usually random as mentioned above, and the number of recipients is usually not too large. Therefore, the time feature is not outstanding and the outbound weight of the IP address is almost equivalent to that of a sender email address, which motivates us to employ the time feature and outbound weight ratio of sender email address and IP address to differentiate spammers and normal users.

B) The other case is mistaking the spam emails as legitimate emails (false negative), which may result in a waste of the user's time. Spammers, to evade anti-spam detection, usually frequently change the sender email address, which disguises them as normal users, because the out-degree of each sender email address is close to 1. However, the outbound weight and out-degree of the IP address, on which the spammer locates, is still large for there are numerous senders on this IP address, which inspires us to use out-degree and outbound weight of IP address to reduce this type of wrong classification.

Table 3: Comparison of SVM classification results on new features and basic features (Acc. denotes the accuracy)

	Acc. mean	**Acc. min**	**Acc. median**	**Acc. max**	**Chi-square value**
Basic+Time features	0.9470	0.8546	0.9474	0.9660	1757.4793
Basic+Weight ratio features	0.9386	0.8410	0.9422	0.9534	1101.6472
Basic+IP out-degree features	0.9289	0.8027	0.9286	0.9497	559.3627
Basic features	0.9024	0.8186	0.9100	0.9306	N/A

To analyze the effectiveness of the proposed features individually, we compare the classification performance before and after using each feature. We refer to the method using out-degree, in-degree and reply ratio features as method with basic features, and the method using each proposed features as method with new features (here, the new features are: time feature called Time, outbound weight ratio of sender email address and IP address called Weight ratio method, out-degree of IP address called IP out-degree). The accuracy of SVM on these three new features and the basic features is shown in Table 3.

It can be observed from Table 3 that, compared with the basic features, each additional new feature achieves a better classification accuracy, which shows that the new features are effective to identify spam emails from legitimate emails. At *p*=.05, the Chi-square values indicate that, using SVM classification, the new features are statistically significant (compared to the critical value 3.84).

To further evaluate the effectiveness of the new features, we adopt another classification approach, Logistic Regression, to classify the emails. The accuracy of Logistic Regression on the three new features and basic features is shown in Table 4. We also find that each additional new feature obtains a better classification performance and more information. At *p*=.05, the Chi-square values indicate that, using Logistic Regression classification, the new features are also statistically significant.

Table 4: Comparison of Logistic Regression classification results on new features and basic features (Acc. denotes the accuracy)

	Acc. mean	**Acc. min**	**Acc. median**	**Acc. max**	**Chi-square value**
Basic+Time features	0.9363	0.8506	0.9383	0.9557	1182.0849
Basic+Weight ratio features	0.9321	0.8488	0.9293	0.9506	915.3726
Basic+IP out-degree features	0.9163	0.8369	0.9154	0.9398	242.0517
Basic features	0.8981	0.8372	0.9085	0.9260	N/A

To validate the performance of the combination of the proposed features, we compare the classification performance of basic features and the combination of new features.

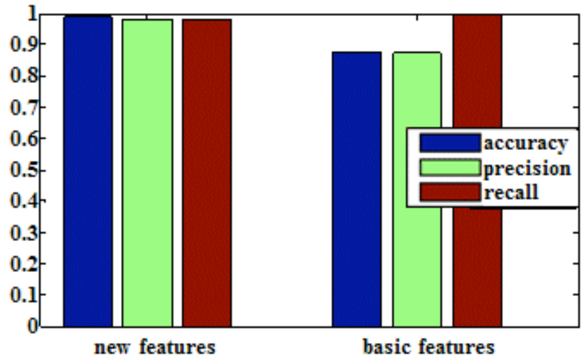


Figure 7: Comparison of SVM classification results on new features and basic features by accuracy, precision and recall

We refer to the method additionally using all our proposed features as method with new features. The performance of SVM on new features and basic features is shown in Figure 7. It can be observed from Figure 7 that the recall of basic feature is a little better than the new features, while the classification accuracy and precision of new features is better than those of basic features, which indicates that the basic features can well distinguish spam emails from legitimate emails, however, misclassify legitimate emails as spam. In contrast, due to the additional features: out-degree of IP address, outbound weight ratio of sender email address and IP address and feature on time, the new features can effectively detect the aforementioned misclassified legitimate users. Consequently, the accuracy and precision is significantly increased. These factors make the accuracy with the new features greatly improved.

At p=.05, the Chi-square value is greater than the critical value (3.84), which demonstrates that the new features work significantly better statistically than the basic features.

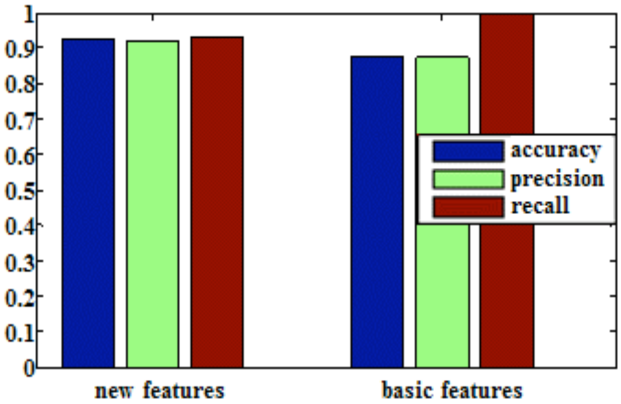


Figure 8: Comparison of Logistic Regression classification results on new features and basic features by accuracy, precision and recall

To further evaluate the effectiveness of new features, we also employ Logistic Regression to classify legitimate emails and spam emails by using new features and basic features. From Figure 8, we can see that the classification accuracy and precision of new features are still better than those of basic features, and the recall of basic features is better than new features as well, for the same reason as that of SVM.

On the other hand, it can be also observed, by comparing Figures 7 and 8, that using the same new features, the classification performance of SVM is better than that of Logistic Regression, which reflects the advantage of SVM.

To prove the role of MapReduce model in social-network-based feature extraction, we compute the extraction time of features from all email records based on different number of data nodes. In Figure 9, we take in-degree feature extraction as example.

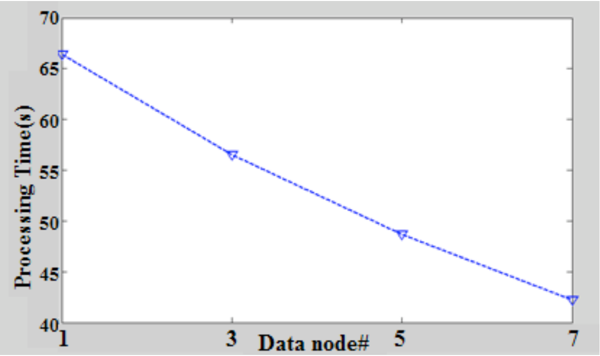


Figure 9: Extraction time of in-degree feature on different number of data nodes with Hadoop

It can be observed, from Figure 9, that the extraction time of in-degree feature decreases, with the number of data nodes increasing. Therefore, Hadoop is effective to speed up the feature extraction from large number of email records.

Due to the unavailability of either source codes or the datasets, or the small number of email records in the usual public datasets, used by other email classification methods, it is difficult to directly compare them with the proposed method.

Table 5: Quantitative comparison between our method and state-of-the-art method

Method	*Avg Accuracy*	*Avg Precision*	*Avg Recall*
Method of [10]	-	0.8330	-
Method of [12]	0.9740	-	0.9760
Method of [18]	0.9700	-	-
Method of [19]	0.9600	-	-
Our method	**0.9870**	**0.9801**	**0.9883**

To roughly understand the overall performance of our proposed method, we list the performance of our proposed method and those reported by other methods, using commonly used metrics (Accuracy, Precision, and Recall), as shown in Table 5. [10] exploited network-level characteristics (e.g., the IP sender's geographic coordinates) to classify email. [12] used the behavioral features (e.g., number of recipients) to customize the filtering rules for spam detection. [18] employed a locally stored friend-list to build the social network and inferred the relationship closeness and (dis)interests between individuals to detect spams. [19] constructed the social network according to sender and recipient fields in the email file and extracted social features to classify emails.

Compared to other behavior-based methods, our method achieves higher average accuracy, precision and recall than other state-of-the-art methods, due to the behavior- and time-based features. In particular, compared with [10], our method achieves a significant improvement. It is because our method adopts discriminative behavior and time features to accurately capture the characteristics of spammer and normal users, instead of exploiting the network layer features, i.e., the operation system of the sending device and IP address.

The experiments as a whole demonstrate that the proposed method can distinguish the spam emails from legitimate emails and outperform the state-of-the-art methods under comparison.

8 Conclusions

Email classification is a fundamental and important issue. According to the characteristics of the spammer, this paper proposes a time- and behavior-based email classification method. By extracting the sender and recipient of the email delivery record, we construct an email social network. Based on the email social network, we analyze the behavior features in a distributed way, as well as the time features. Then, a SVM-based classifier is adopted to detect the spam emails. The experiments on real email data show that the proposed method achieves accurate classification results.

In the future, we will investigate deep learning technology to find potential distinguishable features for effective email classification. We will try other Hadoop-based classifiers to detect spam on the large volume of training set.

Some features of our method are based on IP address. Under some circumstances, such as DHCP, the IP address will change after a period of time if the terminal device cannot renew the previous IP address. However, this period of time is not short, and in this period the IP address is still fixed, therefore, those features related to the IP address still work.

Acknowledgments

The paper is supported by National Science Foundation of China (61171132，61340037), Talent Project of Jiangsu Province of China (2014WLW 029), Technology Platform Projects of Nantong (CP2013001).

References

[1] Maria Vergelis, Tatyana Shcherbakova and Nadezhda Demidova, Kaspersky Security Bulletin: Spam in 2014, Kaspersky [Online], 24 March 2015, Available at http://securelist.com/analysis/monthly-spam- reports/ 59420/spam-report-march-2014/

[2] Tarek M Mahmoud, Ala Ismail El Nashar, Tarek Abd-El-Hafez and Marwa Khairy, An Efficient Three-phase Email Spam Filtering, British Journal of Mathematics & Computer Science, Vol.4, No.9, 2014, pp.1184-1201.

[3] Bing Zhou, Yiyu Yao and Jigang Luo, Cost-sensitive three-way email spam filtering, Journal of Intelligent Information Systems, Vol.42, No.1, 2014, pp.19-45.

[4] James Clark, Irena Koprinska and Josiah Poon, A neural network based approach to automated e-mail classification, Web Intelligence, International Conference on IEEE/WIC/ACM, Halifax, Canada, Oct., 2003, pp.702-705.

[5] Karl-Michael Schneider, A comparison of event models for Naive Bayes anti-spam e-mail filtering, Proc. 10th conference of European chapter of the Association for Computational Linguistics, Budapest, Hungary, April, 2003, Vol.1, pp.307-314.

[6] Islam, Md Rafiqul, Morshed U. Chowdhury and Wanlei Zhou, An Innovative spam filtering model based on support vector machine, International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, Vienna, Austria, Nov., 2005, pp.348-353.

[7] Carreras, Xavier and Lluis Marquez, Boosting trees for anti-spam email filtering. arXiv preprint cs/0109015, 2001.

[8] Drucker, Harris, S. Wu and Vladimir N. Vapnik, Support vector machines for spam categorization, IEEE Transactions on Neural Networks, Vol.10, No.5, 1999, pp.1048-1054.

[9] Yue, Xun, Ajith Abraham, Zhong-Xian Chi, Yan-You Hao and Hongwei Mo, Artificial immune system inspired behavior-based anti-spam filter, Soft Computing, Vol.11, No.8, 2007, pp.729-740.

[10] Cortez, Paulo, André Correia, Pedro Sousa, Miguel Rocha and Miguel Rio, Spam Email Filtering Using Network-Level Properties, 10th Industrial Conference on Data Mining, Berlin, Germany, July, 2010, pp.476-489.

[11] Husna, Husain, Santi Phithakkitnukoon, Srikanth Palla and Ram Dantu, Behavior analysis of spam botnets, 3rd International Conference on Communication Systems Software and Middleware and Workshops, Bangalore, India, Jan., 2008, pp.246-253 .

[12] Naksomboon, S., C. Charnsripinyo and N. Wattanapongsakorn, Considering behavior of sender in spam mail detection, 6th International Conference on Networked Computing (INC), Gyeongju, Korea, May, 2010, pp.1-5.

[13] Zamil, Mohammed Fadhil, Ahmed M. Manasrah, Omar Amir and Sureswaran Ramadass, A behavior based algorithm to detect spam bots, Collaborative Technologies and Systems (CTS). Illinois, USA, May, 2010, pp.453-462.

[14] John Scott, Social network analysis, Sage, 2012.

[15] Zhao, Yun Wei, Willem-Jan van den Heuvel and Xiaojun Ye, A Framework for Multi-Faceted Analytics of User Behaviors in Social Networks, Journal of Internet Technology, Vol.15, No.6, 2014, pp.985-994.

[16] Fu-Hong Lin, Chang-Jia Chen, Dan Guo and Hong-Ke Zhang, A Study of User Behaviors in an Online Social Network about the Games of the Kaixin-Net, Journal of Internet Technology, Vol.13, No.2, 2012, pp.173-179.

[17] Cailing Dong and Bin Zhou, Spam Detection, E-mail/Social Network, Encyclopedia of Social Network Analysis and Mining, Springer, 2014, pp.1954-1960.

[18] Haiying Shen and Ze Li, Leveraging social networks for effective spam filtering, IEEE Transactions on Computers, Vol.63, No.11, 2013, pp.2743-2759.

[19] Wang, Min-Feng, Meng-Feng Tsai, Sie-Long Jheng and Cheng-Hsien Tang, Social feature-based enterprise email classification without examining email contents, Journal of Network and Computer Applications, Vol.35, No.2, 2012, pp.770-777.

[20] DeBarr, Dave and Harry Wechsle, Using social network analysis for spam detection, Advances in Social Computing, Springer, 2010, pp.62-69.

[21] Lam, Ho-Yu and Dit-Yan Yeung, A learning approach to spam detection based on social networks, PhD diss., Hong Kong University of Science and Technology, 2007.

[22] Wang, Chen, Yibo Zhang, Xiaohan Chen, Zhiyu Liu, Lei Shi, Guang Chen, F. Qiu, Chun Ying and Wei Lu, A behavior-based SMS antispam system, IBM Journal of Research and Development, Vol.54, No.6, 2010, pp.3:1-3:16.

[23] Dean, Jeffrey and Sanjay Ghemawat, MapReduce: simplified data processing on large clusters, Communications of the ACM, Vol.51, No.1, 2008, pp.107-113.

[24] Vapnik, Vladimir Naumovich and Vlamimir Vapnik, Statistical Learning Theory, Vol. 1. New York: Wiley, 1998.

[25] Cortes, Corinna and Vladimir Vapnik, Support Vector Networks, Machine Learning, Vol.20, No.3, 1995, pp.273-297.