TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.6, June 2014, pp. 4306 ~ 4313
DOI: 10.11591/telkomnika.v12i6.4562

Online Imbalanced Support Vector Machine for Phishing Emails Filtering

XiaoQing Gu, TongGuang Ni*, Wei Wang
School of Information Science and Engineering, Changzhou University,
Changzhou 213164, China
Telp 86-519-86330558, Fax 86-519-86330284
Corresponding author, e-mail: tiddyddd@163.com, hbxtntg-12@163.com*, super_asd@163.com

Abstract

Phishing emails are a real threat to internet communication and web economy. In real-world emails datasets, data are predominately composed of ham samples with only a small percentage of phishing ones. Standard Support Vector Machine (SVM) could produce suboptimal results in filtering phishing emails, and it often requires much time to perform the classification for large data sets. In this paper, an online version of imbalanced SVM (OISVM) is proposed. First an email is converted into 20 features which are well selected based on its content and link characters. Second, OISVM is developed to optimize the classification accuracy and reduce computation time, which is used a novel method to adjust the separation hyperplane of imbalanced date sets and an online algorithm to make the retaining process much fast. Compared to the existing methods, the experimental results show that OISVM can achieve significantly using a proposed expressive evaluation method.

Keywords: *phishing emails, filtering, support vector machine, imbalanced date, online*

1. Introduction

Phishing email has increased enormously over the last years and is a serious threat to global security and economy. Phishing email is the act of attempting to fraudulently acquire through deception sensitive personal information such as passwords and credit card details by assuming another's identity in an official-looking email. The user is provided with a convenient link in the same email that takes the email recipient to a fake webpage appearing to be that of a trustworthy company. When the user enters his personal information on the fake page, it is then captured by the fraudster. According to a report from RSA [1], the number of phishing attacks in the year of 2011 increased 37% compared to that in the year of 2010, and approximately one in every 300 emails delivered on the Internet in the year of 2011 was a phishing email. Phishers can obtain $4500 in stolen funds in each phishing attack. PhishTank, an organization tracks 31,850 unique phishing attacks during July 2012. In addition there are phishing attacks against non-traditional sites, such as automotive associations. Highly targeted attacks on the employees or members within a certain company, government agency, or organization are called "spear phishing". Here the phisher wants to gain access to a company's computer system.

Among the countermeasures used against phishing, three main alternatives have been used: Black list/white list, network and encryption based countermeasures and content based filtering [2]. The first alternative consists in using public lists of malicious phishing websites (black list) and lists of ham non-malicious websites (white list), where each link in an email must be checked in both lists. The blacklist-based anti-phishing toolbars are developed by many companies such as Netcraft. The main problem of this countermeasure is that phishing websites are short-lived, it makes difficult to keep an up-to-date list of malicious websites.

The second alternative is based on email authentication methods. Email authentication mechanisms allow receiving mail agents to accept mail from known good senders, reject mail from known spammers, or use reputation mechanisms such as blacklists to decide how to handle mail from other senders. Herzberg et al. [3] have invented an authentication mechanisms based on DNS-based email sender, which use the DNS system to identify the sender. Dhanalakshmi et al. [4] identified spoofed emails using various techniques such as Sender Policy Framework, Sender ID and Domain Keys Identified Mail.

Received September 29, 2013; Revised December 23, 2013; Accepted January 22, 2014

The third alternative is based on content-based phishing filtering. Filtering attempts to distinguish phishing emails from legitimate emails using machine learning techniques. Fette et al. [5] developed a scheme in to filter phishing emails based on the features collected information from internal and external information of emails. Garera et al. [6] identified a set of fine-grained heuristics from URLs, and applied a logistic regression model to these URL signatures. Zhuang et al. [7] have proposed an anti-phishing framework using multiple classifiers combination. Chen et al. [8] have adopted a hybrid text and data mining model that used key phrase extraction technique to discover important semantic categories from the textual content of the phishing alerts. Shih et al. [9] designed and implemented an email virus filter with an embedded system.

As the phishing emails are often nearly identical to legitimate websites, current detection approaches have limited success in detecting these attacks. In the other hand, email data sets in real-world usually have class imbalance problems, due to the fact that ham emails is represented by a much larger number of instances than phishing emails. In this paper, we proposed a new online imbalanced SVM (OISVM) to provide phishing filtering. Based on standard SVM, an imbalanced algorithm and an online learning strategy are combined, which overcomes the imbalanced problem in SVM and use the incremental training samples in re-training. As a result, the training time can be reduced greatly without much loss of the classification precision.

The contributions of our work are: (1) A number of new features of emails are incorporated, in particular content features and link features. (2) A new online imbalanced SVM to the phishing filtering problem is developed. It is easy modeling and fast implemented which gives stable classification results when testing different datasets.

2. Content and Link Features of Phishing Emails

In this section we discuss the content and link features used across all the phishing emails with the intension of identifying a set of generic features to be used for filtering.

Link identity: The objective of this module is to extract the link identity of an email, owing to link identity defined by analyzing the hyperlinks structure of an email. The hyperlinks of a regular email often link to its own domain, while phishing emails are usually the opposite. A phishing email often contains hyperlinks that point to a foreign domain. Here anchor links are analyzed, specifically the href attribute of <a> and <area> tags. For each anchor links, the base domain is extracted part from the URL, and then the occurrence is counted for each base domain. The base domain which has the highest occurrence will be the link identity.

Next, the feature generation step would determine the feature values of an email based on its content and link characters. The features that server as input to our filtering are presented according to [5, 6]. But these features also differ from the list proposed above. First, some features have been changed along with the phishing techniques; second, the features that require special information are not included in our approach, such as the age of linked-to domains, spam-filter output; third, all features in our approach are binary features.

Feature 1: HTML format. Phishing emails tend to use some formatting of the content to display the logo or design of the corresponding message. For this reason, it is common for phishing emails to be in the HTML format.

Feature 2: Using IP addressed instead of URL. Frequently, phishing attempt to conceal the destination website by obscuring the URL. Due to the low cost of phishing, many phishing emails can only be addressed by an IP address URL instead of a domain or host name. On the other hand, legitimate companies rarely link to pages by an IP address, and so such a link in email is a potential indicate of a phishing attack.

Feature 3 and 4: Dots in URL and slash in URL. To construct legitimate-looking URL, there may be a lager number of dots in a phishing URL. Legitimate URL also can contain a number of dots, but a URL could be less credible if there are too many dots in it. The average number of dots of all URLs in an email is computed by Equation (1):

$$AVG_{dots} = \sum_u d_u / |U| \quad (1)$$

Where u is an URL in an email d, d_u is the number of dots in the URL u, and |U| is the number of URLs. Feature 3 is a binary feature to compare with AVG_{dots} and five dots. Similar to feature 3,

$AVG_{flashes}$ is the average number of slashes in all URLs in an email. Feature 4 is also a binary feature to compare with $AVG_{flashes}$ and five lashes.

Feature 5: Usage of well-defined underlying contents or “Here” links. Most of the phishing emails use the underlying contents or “Here” links such as invoking a sense of false urgency, threat, wheedle, and concern to deceit the users in clicking on the visited hyperlink.

Feature 6: Domain in href is different from the display string. In phishing emails, the link text seen in the emails is usually different from the actual link destination. For example, www.eBay.com, the URL referring to the display string is eBay, but it redirects the user discretely to a website which its domain is eBay123.

Feature 7: Domain in header fields is different from link identity. Companies normally tend to host their own mail server and web servers within their own network domains. On the other hand, phishers often use a free email count from public email service providers. To mislead recipients of such messages, phishers often use the name of the target as part of the email account name and the full user name of the email account. Therefore, link identity is compared to the following three header fields: “From:”, “Return-Path:” and “Reply-to:”.

Feature 8: Country in header fields. This feature obtains the geographic location of the network domain of the claimed email addresses found on the headers: “Return-Path:”, “From:” and “Reply-To:”. The location for all three should be consistent, that is, in the same country. It is noted that a subset of phishing emails rely on email communications between the phishers and recipients to carry out the phishing attack, instead of relying on redirecting recipients to a fraudulent web site. In this case, country code in domain in headers is compared to each other.

Feature 9-19: Keywords. Given the nature of phishing email, they often contain some distinctive words. We use a positive word list, i.e., a list of words hinting at the possibility of phishing. For each word in the list we record is a binary feature of whether or not the word occurs in the email. The list contains a total of ten word stems: account, update, password, bank, log, inconvenience, security, access, verify, credit.

Feature 20: Spam Filter. A trained, off-line version of SpamAssassin is used to generate a feature: the class assigned to the email either “ham” or “spam”. This is a binary feature using the trained version of SpamAssassin with the default rule weights and threshold. This feature’s importance is discussed in more detail in [5].

3. Online imbalanced SVM

3.1. Imbalanced SVM

Support vector machine (SVM) learning is a promising pattern classification technique proposed by Cortes and Vapnik [10]. SVM learning aims at minimizing an upper bound of the generalization error through maximizing the margin between the separating hyperplane and the data. Although SVMs often work effectively with balanced datasets, they could produce suboptimal results with imbalanced datasets. More specifically, an SVM classifier trained on an imbalanced dataset often produces models which are biased towards the majority class and have low performance on the minority class.

A novel method is proposed in [11] for the separation hyperplane of binary classification imbalanced data. Firstly, the original samples are preliminarily trained by the standard support vector machine, and a normal vector of the separation hyperplane is obtained. Secondly, one-dimensional data are generated by projecting the high dimensional data onto the normal vector. Then, the ratio of the two-class penalty factors is determined based on the information derived from the standard deviation of the projective data and the two-class sample sizes. Finally, a new separation hyperplane is presented by the second training.

Given a training set of N samples $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$, where $\boldsymbol{x}_i \in R^d$ represents an n-dimensional data point and $y_i \in \{+1, -1\}$ represents the label of the class of that data point, for $i = 1,\ldots,n$. Let $\phi(\mathbf{X})$ denote the date matrices in feature space **H**, $\phi(\mathbf{X}) = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \ldots, \phi(\mathbf{x}_n)]$, then the kernel function **K** can be found such that $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$. Thus, the nonlinear OISVM can be achieved by solving the following quadratic problem:

$$\min \frac{1}{2}\|w\|^2 + C^+ \sum_{i=1}^{n^+} \xi_i + C^- \sum_{i=n^++1}^{n} \xi_i$$

$$\text{s.t.} \quad y_i((\mathbf{w}\cdot\varphi(\mathbf{x}_i))+b) \geq 1-\xi_i, \quad \xi_i \geq 0 \tag{2}$$

$$\sum_{i=1}^{n^+} \xi_i > 0, \quad \sum_{i=n^++1}^{n} \xi_i > 0, \ i=1, 2,\ldots, n$$

In imbalanced SVM, the SVM soft margin objective function is modified to assign two misclassification costs, such that C^+ is the misclassification cost for positive class examples, while C^- is the misclassification cost for negative class examples. Here we also assume positive class to be the minority class and negative class to be the majority class. Here $C^+ = Cs^+ / n^+$, $C^- = Cs^- / n^-$, C is a constant; s^+ is the projective standard deviation of positive class; s^- is the projective standard deviation of negative class, and $n^- = n - n^+$.

To solving Equation (2), the original samples are preliminarily trained by standard SVM, and finding the optimal value of α_i, $\boldsymbol{w}_1$ can be recovered as:

$$\boldsymbol{w}_1 = \sum_{i=1}^{n} a_i y_i \varphi(\boldsymbol{x}_i) \tag{3}$$

So the rejection value:

$$\boldsymbol{w}_1 \cdot \varphi(\boldsymbol{x}_j) = \sum_{i=1}^{n} a_i y_i \varphi(\boldsymbol{x}_i)\cdot\varphi(\boldsymbol{x}_j) = \sum_{i=1}^{n} a_i y_i k(\boldsymbol{x}_i, \boldsymbol{x}_j) \ \ (j=1, 2,\ldots, n) \tag{4}$$

As a result, the parameters s^+ and s^- computations are described as the following equations:

$$\begin{aligned} s^+ &= \sqrt{\frac{1}{n^+-1}\sum_{j=1}^{n^+}[\sum_{i=1}^{n} a_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)] - \frac{1}{n^+-1}\sum_{j=1}^{n^+}[\sum_{i=1}^{n} a_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)]^2} \\ s^- &= \sqrt{\frac{1}{n^--1}\sum_{j=1}^{n^-}[\sum_{i=1}^{n} a_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)] - \frac{1}{n^--1}\sum_{j=1}^{n^-}[\sum_{i=1}^{n} a_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)]^2} \end{aligned} \tag{5}$$

To solve this optimization problem Lagrangian is constructed:

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C^+\sum_{i=1}^{n^+}\xi_i + C^-\sum_{i=n^++1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i(y_i(\boldsymbol{w}\cdot\varphi(\boldsymbol{x}_i)+b)-1+\xi_i)$$
$$-\sum_{i=1}^{n^+}\beta_i\xi_i - \sum_{i=n^++1}^{n}\gamma_i\xi_i \tag{6}$$

With Lagrangian multipliers $\alpha_i \geq 0$, $\beta_i \geq 0$ and $\gamma_i \geq 0$. The derivatives of $L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ with respect to the primal variables using the Karush-Kuhn-Tucker (KKT) conditions should vanish,

$$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} + \sum_{i=1}^{n}\alpha_i y_i \varphi(\boldsymbol{x}_i) = 0 \tag{7}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N}\alpha_i y_i = 0 \tag{8}$$

$$\frac{\partial L}{\partial \xi_i} = C^+ - \alpha_i - \beta_i = 0 \tag{9}$$

$$\frac{\partial L}{\partial \xi_j} = C^- - \alpha_i - \gamma_j = 0 \tag{10}$$

Substituting (7)-(10) into (6), we obtain the dual form of the optimization problem:

$$\max \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

$$\text{s.t. } \sum_{i=1}^{n} a_i y_i = 0 \tag{11}$$

$$0 \le a_i \le Cs^+ / n^+ \text{, } y_i\text{=+1, } i\text{=1,2,…,}n^+$$

$$0 \le a_j \le Cs^- / n^- \text{, } y_i\text{=-1, } j\text{= } n^+\text{+1, …, } n$$

Equation (11) is a typical convex quadratic programming problem which is easy to be numerically solved. Suppose a training sample $\boldsymbol{x}_i (1 \le i \le n)$ called a Support Vector (SV) if the corresponding Lagrange multiplier $\alpha_i > 0$. Denote the SV sets as $SV_1 = \{\boldsymbol{x}_i \mid 0 \le \alpha_i \le Cs^+ / n^+, 1 \le i \le n^+\}$ and $SV_2 = \{\boldsymbol{x}_j \mid 0 \le \alpha_j \le Cs^- / n^-, 1+n^+ \le j \le n\}$. Suppose $\alpha^* = [\alpha_1^*, \ldots, \alpha_N^*]$ can be used to solve the above optimization problem, and the optimal threshold b^* is computed by the following formula:

$$b^* = \frac{|SV_1| \sum_{\boldsymbol{x}_i \in sv_1}\sum_{j=1}^{N} \alpha_j^* y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) + |SV_2| \sum_{\boldsymbol{x}_i \in sv_2}\sum_{j=1}^{N} \alpha_j^* y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)}{-|SV_1| \cdot |SV_2|} \tag{12}$$

Finally, the SVM decision function can be given by:

$$f(\boldsymbol{x}) = sign(\sum_{i=1}^{n} a_i^* y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b^*) \tag{13}$$

The dual optimization problem can be solved in the same way as solving the standard SVM optimization problem. The modified SVM algorithm would not tend to skew the separating hyperplane towards the minority class examples to reduce the total misclassifications as the minority class examples are now assigned with a higher misclassification cost.

3.2. Online Imbalanced SVM (OISVM)

In standard SVM applications, an SVM is trained on an entire set of training data, and is then tested on a separate set of testing data. Phishing emails filtering is typically tested and deployed in an online setting, which proceeds incrementally. Online learning is performed in a sequence of trials. At trial *t* the algorithm first receives an instance $\boldsymbol{x}_t$ and is required to predict the label associated with that instance. After the online learning algorithm has predicted the label, the true label is revealed and the algorithm pays a unit cost if its prediction is wrong. The ultimate goal of the algorithm is to minimize the total number of prediction mistakes it makes along its run. To achieve this goal, the algorithm may update its prediction mechanism after each trial so as to be more accurate in later trials.

Based on Relaxed Online Support Vector Machine (ROSVM) algorithms described by Sculley in [12], the proposed OISVM classifier is stated in Table 1.

Initially training in imbalanced SVM is only a small fraction of training emails end up as support vectors. Given an incoming message $\boldsymbol{x}_i$ and a label y_i, if the Classifier's optimal strategy is satisfied well, it will not change the hypothesis; thus it is not necessary to re-train. If the Classifier's optimal strategy is not satisfied, the hyperplane parameters are updated using the imbalanced SVM algorithm over the seen messages (seenData set). The training would use

only the historical support vector samples and the incremental training samples in re-training. All non-SV samples are discarded after previous training. Consequently, the training time can be reduced greatly without much loss of the classification precision. A parameter *m* is used to set the definition of well classified, which is used to reduce the number of updates. A parameter *p* is used to set the number of messages in seenData.

Table 1. Pseudo Code for Proposed OISVM Classifier

```
(1) Initialize:
    Data set X = (x1, y1),. . . ,(xn, yn)
    Seed imbalanced SVM classifier with a few examples of each class;
    Train an initial imbalanced SVM filters;
(2) Online Learning
    For Each xi ∈ X do:
      Classify xi
      IF yif(xi) < m
          Find w', b' with imbalanced SVM with parameters C+, C- on seenData,
          using w, b as seed hypothesis.
          set (w, b) := (w', b')
      IF size(seenData) > p
          Remove oldest example from seenData
      Add xi to seenData
(3) Finishing:
    Repeat until xn is finished
```

Our algorithm seems similar to ROSVM; however, they are used in a different context. First, we use an imbalanced SVM. Second, ROSVM uses the linear kernel as it assumes that phishing and ham are linearly separable. However, in most real-life emails dataset, the datasets are not completely linearly separable even though they are mapped into a higher dimensional feature space. For OISVM we use the Gaussian Radial Basis Function (RBF) kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \left\| \boldsymbol{x} - \boldsymbol{x}' \right\|^2)$.

4. Experimental Settings and Results

In this section, we present the experiments conducted and discuss the results. All classification modeling is carried out on a computer with an Intel Xeon at 1.86 GHz and 8 GB of memory. The features describing the properties of emails are extracted as described in section 2 and the size of each feature vector is 20.

4.1. Dataset Description and Evaluation Criteria

We rely on four different datasets to carry out the evaluation studies of our work. The first one is a phishing dataset containing phishing emails collected between 2005 and 2008 by Jose Nazario [13]. The second one is also a ham dataset collected from the Apache SpamAssassin Project [14]. Using these two collections of phishing emails and ham emails we constructed a dataset NAZA. To perform experiments on data from a real-world mailbox, we use the dataset REAL which were collected over a period of 10 months in 2012 and gathered from several users' personal mailboxes. To simplify our evaluation studies, all emails in four datasets are in English. The summary of the key figures of each used dataset is given in Table 2.

Table 2. Summary of the used Dataset

Dataset	Size	Training(Ham,Phinshing)	Testing(Ham,Phinshing)
NAZA	10520	7890(5523,2367)	2630(1841,789)
REAL	4208	2524(2272,252)	1684(1515,169)

In this paper, a group of performance metrics in classification problems are used for the evaluation of the results, consisting of FPR, FNR, accuracy, precision, recall and ROC. True Positives (TP) means correctly classified phishing emails, True Negative (TN) means correctly classified ham emails, False Positive (FP) means wrong classified ham emails as phishing, and False Negative (TN) means wrong classified phishing messages as ham. Therefore, The False

Positive Rate (FPR) and the False Negative Rate (FNR) as the proportion of wrongly classified ham and phishing email messages respectively (FPR = FP/ (FP+TN), FNR = FN/ (TP+FN)). Accuracy states the overall percentage of correct classified email messages (Accuracy = (TP+TN)/ (TP+FP+TN+FN)). Precision as the classifier's safety, states the degree in which messages identified as phishing are indeed malicious (Precision = TP/ (TP+FP)). Recall as the classifier's effectiveness, states the percentage of phishing messages that the classifier manages to classify correctly (Recall = TP/ (TP+FN)). Receiver Operating Characteristic (ROC) as a classifier's balance ability between its FPR and its FNR is a function of varying a classification threshold.

The classification algorithms, online SVM, imbalanced SVM, ROSVM and OISVM are implemented. We use Platt's SMO algorithm as a core SVM solver, and imbalanced SVM classifier implemented in the libSVM-library. Since an email is only considered as a legitimate or a phishing, it is naturally a binary classification problem. The SVM would produce output in two classes: +1 means phishing, and -1 means legitimate. The robustness of the classifiers is evaluated using 10-fold cross validation. For the online setting, the ROSVM [12] was used. For training the SVM classifier, we need to specify two parameters, the γ value in the kernel function, and C the penalty values. In online SVM and ROSVM, the RBF kernel with parameters C = 100 and γ =0.1 turned out to be most accurate and stable. ROSVM and OISVM parameters tuning were estimated over a 20% subsets from the training dataset. According to the size of datasets, it is setting m=0.8 and p=1000 for the threshold. The value of penalty values C^+ and C^- used in imbalanced SVM and OISVM is given in Table 3, where C^+ is for phishing examples, and C^- is for ham examples.

Table 3. The Optimal Value of Penalty Values C^+ and C^-

Dataset	imbalanced SVM		OISVM	
	C^+	C^-	C^+	C^-
NAZA	22.86	4.19	26.35	3.24
REAL	31.50	2.39	36.21	2.02

4.2. Results

We compared online SVM, imbalanced SVM, ROSVM and OISVM for two datasets using 10-fold cross validation. The results are shown in Table 4 and 5. The training time for classifiers is the training of the classification, not including the preprocessing of the emails. The training time is expressed in seconds. As we can see from the Table 4, imbalanced SVM is much more effective then online SVM and ROSVM, but imbalanced SVM is expensive in terms of time. Our proposed OISVM although can achieve a similarly accurate classification performance in far less time. The results demonstrate that OISVM outperformed all of the other SVM approaches in the detection of phishing email viruses.

Table 4. Performance of the Methods for the NAZA Dataset

Method	Accuracy	FPR	FNR	Precision	Recall	ROC	Training time (s)
OnSVM	92.52%	6.68%	6.60%	93.06%	94.01%	96.46%	120.2
ImSVM	97.35%	2.74%	2.16%	98.00%	98.91%	98.51%	200.9
ROSVM	92.17%	7.16%	6.91%	92.58%	93.23%	96.09%	10.6
OISVM	97.16%	2.99%	2.43%	97.69%	97.75%	98.22%	13.8

Table 5: Performance of the methods for the REAL dataset

Method	Accuracy	FPR	FNR	Precision	Recall	ROC	Training time (s)
OnSVM	90.47%	8.25%	7.84%	91.52%	91.74%	90.12%	72.3
ImSVM	95.13%	5.62%	5.26%	96.04%	95.89%	95.67%	100.5
ROSVM	90.28%	8.67%	8.03%	90.36%	90.25%	90.00%	8.5
OISVM	95.01%	6.01%	5.49%	96.22%	95.33%	95.24%	9.6

We can observe the results on the REAL datasets are somewhat inferior in Table 5. The FPR and FNR are increased compared to NAZA, and Accuracy, Precision, Recall and ROC are decreased a little. The cause in the fact is that the public dataset are somewhat artificial in

that they are collected from diverse sources and even cover different time periods. Since we are using SVM for classification, the detection results also depend on the quality and quantity of the training dataset. If the training dataset of REAL could represent all characteristics of ham and phishing emails then the detection performance would become better.

5. Conclusion

This paper addresses the problem of filtering phishing emails from ham ones with imbalanced and online learning SVM (OISVM). In OISVM, the SVM soft margin objective function is modified to assign two misclassification costs. By assigning a higher misclassification cost for the minority class examples than the majority class examples, the effect of class imbalanced could be reduced. Furthermore, the online learning algorithm is used, whereby only subsets of the data are to be considered at any one time and results subsequently combined, can make the retraining process much faster and avoid the much storage cost. Thus OISVM can be scaled up to handle extremely large data sets. The experiments show that OISVM is able to obtain very good results with the different validation datasets employed. Furthermore, a number of features have described that are particularly well-suited to filtering phishing mails which are binary features and selected by its content and link characters.

In our future works, we plan to adjust existing feature extraction methods, and seek for more relevant features to get a better result. Furthermore the method used to collect a dataset must be improved.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under contact (61070121).

References

[1] Sanchez F, Duan Z. *A sender-centric approach to detecting phishing emails.* Proceedings of ASE International Conference on Cyber Security. USA. 2012: 248-257.
[2] Bergholz A, Beer J, Glahn S. New filtering approaches for phishing email. *Journal of Computer Security*. 2010; 18(1): 7-35.
[3] Herzberg A, Jbara A. Security DNS-based email sender authentication mechanisms: A critical review. *Computers & security*. 2009; 28(8): 731-742.
[4] Dhanalakshmi R, Kavisankar L, Chellappan C. Enhanced Enhanced Email Authentication Against Spoofing Attacks To Mitigate Phishing. *European Journal of Scientific research*. 2011; 54(1): 165-170.
[5] Fette L, SADEH N TOMASIC A. *Learning to Detect Phishing Emails.* Proceedings of the International World Wide Web Conference Committee (IW3C2). Canada. 2007; 649-656.
[6] Garera S, Provos N Chew M. *A framework for detection and measurement of phishing attacks*. Proceedings of the 2007 ACM Workshop on Recurring Malcode. Germany. 2007; 1–8.
[7] Zhuang W, Jiang Q. Intelligent Anti-phishing Framework Using Multiple Classifiers Combination. *Journal of Computational Information Systems*. 2012; 8 (17): 7267- 7281.
[8] Chen. X, Bose. I. Assessing the severity of phishing attacks: A hybrid data mining approach. *Decision Support Systems*. 2011; 50(4): 662 – 672.
[9] Shih D, Chiang H, Yen D. An intelligent embedded system for malicious email filtering. *Computer Standards & Interfaces*. 2013; 35(5): 1289-1302.
[10] Cortes C, Vapnik V, Support vector networks. *Machine Learning*. 1995; 20(3): 273–297.
[11] Liu W, Liu S, Xue Z. Balance Method for imbalanced Support Vector Machines. *Pattern Recognition & Artificial intelligence*. 2008; 21(2): 136-141.
[12] Sculley D, Wachman G. *Relaxed online SVMs for spam filtering*. Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. USA. 2007: 415–422.
[13] Gomez J, Moens M. PCA document reconstruction for email classification. *Computational Statistics and Data Analysis.* 2012; 56(3): 741–751.
[14] Ramanathan V. Phishing detection and impersonated entity discovery using Conditional Random Field and Latent Dirichlet Allocation. *Computers & Security*. 2013; 34(5): 123-139.