Classifying Phishing Emails Using Confidence-Weighted Linear Classifiers

Ram B. Basnet
Computer Science & Engineering Department
New Mexico Tech
Socorro, NM 87801, USA
rbasnet@cs.nmt.edu

Andrew H. Sung
Computer Science & Engineering Department
New Mexico Tech
Socorro, NM 87801, USA
sung@cs.nmt.edu

Abstract— **Though Internet users are generally becoming more aware of phishing emails and phishing websites, cyber scammers are able to come up with novel schemes constantly that circumvent phishing filters and often succeed in fooling even savvy users. Using heuristic approaches and knowledge about the phishing techniques, researchers have developed several phishing specific unique features to detect phishing emails. In this paper, we propose a new and simple methodology to detect phishing emails utilizing Confidence-Weighted Linear Classifiers. We use the contents of the emails as features without applying any heuristic based phishing specific features and obtain highly accurate results compared to the best that have been published in the literature.**

Keywords- Phishing, confidence-weighted, linear classifier, filtering, email

I. Introduction

There are 3 major email categories: Ham, Spam and Phishing. Ham is solicited and legitimate email; Spam is unsolicited and legitimate email; and Phishing, on the other hand is unsolicited, deceitful, and potentially harmful email. According to Antiphishing.org [1] phishing is a criminal mechanism employing both social engineering and technical subterfuge to steal consumers' personal identity data and financial account credentials.

Phishing emails usually act on behalf of a trusted third-party to trick email receivers into performing some actions such as giving away personal information, e.g. bank accounts, social security numbers, usernames and passwords to online banking and popular social networking websites like Facebook, Orkut, Twitter, etc. Though much research on anti-phishing techniques has been done and new techniques and methodologies are being proposed regularly, online scammers manage to come up with new innovative schemes to circumvent the existing detection technology and lure potential victims.

Financial services are the most targeted sector of the phishing schemers. More brands are under attack than ever before, hitting record high in the 4th quarter of 2009. The United States continue its position as the top country hosting phishing sites [1].

According to the report published in [2], more than 420,000 scam e-mails are sent every hour in the UK and it is estimated that Britons were targeted by 3.7 billion phishing emails in the last 12 months alone. The report highlights that online banking fraud rose by 14% in the last 12 months. According to their survey a quarter of users admit to falling victim to e-fraudsters, with the average victim losing over £285. Fake banking emails are the most common method used by criminals, with 55% of those targeted receiving seemingly legitimate e-correspondence from high street banks [2]. Online scammers are good at taking advantage of real-world phenomena to trick Internet users into falling for their scam. For example, according to an article in [3], iPad's instant popularity inspired scammers to pull phishing tricks by promising to give an iPad to BETA testers after a couple of months of testing. The phishing site asks for responder's email account information including password to get enrolled in the scam beta testing program. Internet users were also being lured into getting free World Cup tickets. Streaming sites laden with adware are asking users to take surveys by entering their personal information such as email. Online scammers, phishers, identity thieves are looking into Google's Trend ranking of hot search topics to get the potential online victims looking into the topics [5][6].

Once the phishing email receivers are lured into a fraudulent website, even the experienced, security-minded users are often easily fooled to fulfill the website's primary goal. Dhamija et al. [4] have examined various aspects of bogus websites that make them credible. Successful phishers must not only present a high credibility web presence to their victim, they must also create a presence that is so impressive that it causes the victim to fail to recognize security measures installed in web browsers and/or corporate security systems. Data indicates that some phishing attacks have convinced up to 5% of their recipients to provide sensitive information to spoofed websites.

The design and implementation of effective phishing detection techniques to combat cyber crime and to ensure cyber security, therefore, is an important and timely issue that–as long as the cyber criminals are proceeding unabated in scamming Internet users–requires sustained efforts from the research community.

Phishing detection techniques based on the machine learning methodology has proved highly effective, due to the large phishing dataset available and the advances in feature mining and learning algorithms. Confidence-weighted linear classifiers (CWLC) have recently attracted much attention and demonstrated their great effectiveness in detecting malicious websites [22, 23]. In this paper, we experiment with using CWLC for detecting phishing emails and, based on a large phishing dataset and in comparison with other

978-1-4244-8870-4 /10/$26.00 © 2010 IEEE

learning machines, present impressive and highly accurate results as compared to those previously published.

This paper is organized as follows. Section II describes previous and related works. In section III we briefly introduce Confidence-Weighted Learning algorithm and text categorization technique. Section 1V is devoted to our phishing email classification approach: experimental setup, the datasets we used for our experiments and the experimental results and evaluations. We conclude with section V with some summary of results and conclusions in Section V.

II. Related Works

The idea of using contents of the text documents to automatically classify them into various pre-defined categories has a long history in text mining–although we do not know of any previous work that used only the text contents as features to classify phishing emails against their counterpart or hams.

The popular open source SpamAssassin[1] project [18] uses a variety of mechanism including header text analysis, Bayesian filtering, DNS blacklists, and collaborative filtering databases to combat Spam. Bergholz et al. [7] have proposed advanced email features generated by adaptively trained Dynamic Markov Chains and by novel latent Class-Topic Models. They show that classifiers trained using features extracted with these two techniques outperforms the previous benchmark. They used different classifiers, mainly the Support Vector Machine (SVM classifier implemented in the libSVM-library [19]. They have also run experiments using other classifiers, e.g., maximum entropy and decision trees. They noted that the difference in most cases were negligible but didn't report the results on the paper.

Fette et al. [8] proposed the method to detecting malicious phishing emails by incorporating features specifically designed to highlight the deceptive methods used to fool users. With their method they were able to accurately classify 92% of phishing emails, while maintaining a false positive rate on the order of 0.1%. Their experiments results were based on approximately 860 phishing emails and 6950 non-phishing emails. Though their research work was one of the earliest in this research area, their dataset is relatively far smaller than that used by Bergholz [19]. The accuracy of their methodology on their dataset was significantly higher than that of SpamAssassin, a widely-used spam filter.

In our previous work [9], we used the 10 most common features specific to phishing emails proposed by Fette et al. [8] and also added 6 groups of content based features that are most common in phishing emails pretending to come from legitimate financial institutions and ecommerce sites that ask for usernames, social security number, passwords etc. Thus, we applied five different popular machine learning algorithms such as (SVMs, SOM, NN) on a dataset with 16 features. We achieved more than 97% accuracy across the board with libSVM performing the best with 98.04% accuracy. We showed that the heuristic based keyword features had very high prominence in phishing emails.

III. Background

In this section we discuss motivation and the underlying techniques we employ to achieve our goal. In subsection III-A we discuss this fairly new algorithm Confidence-Weighted Linear Classifier (CWLC), in III-B we briefly explain text categorization method on which we based our phishing email classification problem, and in III-C the motivation behind our work.

A. Confidence-Weighted (CW) Learning

Dredze et al. [10] recently proposed confidence-weighted linear classifiers (CWLC), a new class of online learning method designed for Natural Language Processing (NLP) problems based on the notion of parameter confidence. Online learning algorithms operate on a single instance at a time, allowing for updates that are fast, simple and make few assumptions about the data, and perform well in wide range of practical settings. Online algorithm processes its input piece-by-piece in a serial fashion, i.e., in the order that the input is fed to the algorithm, without having the entire input available from the start.

Online algorithms operate in rounds. On round i the algorithm receives an instance $x_i \in \mathrm{R}^d$ to which it applies its current prediction rule to produce a prediction $y_i \in \{-1, +1\}$ (for binary classification.) It then receives the true label $\hat{y}_i \in \{-1, +1\}$ and suffers a loss $l(y_i, \hat{y}_i) = 1$ if $y_i \neq \hat{y}_i$ and $l(y_i, \hat{y}_i) = 0$ otherwise. The algorithm then updates its prediction rule and proceeds to the next round. Just like support vector machines, the prediction rules in CW are linear classifiers

$$f_w(x) : f_w(x) = sign(x . w). \tag{1}$$

The margin of an example (x, y) with respect to a specific classifier w is given by $y(w.x)$. The sign of the margin is positive iff the classifier w predicts correctly the true label y. The absolute value of the margin $|y(w.x)| = |w.x|$ is often thought of as the confidence in the prediction, with larger positive values corresponding to more confident correct predictions. The details on the algorithm can be found in [10].

Dredze et al. [10] have applied CWLC on a range of NLP tasks and showed that their methods improve over other state of the art online and batch methods, learns faster in the online setting, and lends itself to better classifier combination after parallel training.

Ma et al. [22, 23] have applied Confidence-Weighted (CW) algorithm on a large-scale URL datasets from real-time source, large Web mail provider. They showed that recently-developed online algorithms such as CW can be highly accurate classifiers, capable of achieving classification accuracies up to 99% on experiments over a live URL feed. CW clearly outperforms other online (Passive Aggressive and Logistic Regression with Stochastic Gradient Descent) and batch algorithms (LIBLINEAR [24]).

B. Text Categorization

The goal of text categorization is the classification of documents into a number of predefined categories. The first step in text categorization is to transform documents which typically are strings of characters, into a representation suitable for the learning algorithm and the classification task [14]. Information Retrieval research suggests that word stems work well as representation units and that their ordering in a document is of minor importance for many tasks. This leads to an attribute value representation of text. Each email document is an instance represented as a vector of stemmed words which is commonly called 'bag of words' representation. Each distinct term w_i corresponds to a feature with the number of times term w_i occurs in the document as its value. More on text categorization can be found in [16] and [21].

In texts classification tasks, millions of features derived from words and word combinations, most of which are binary and are infrequently on, can be weakly indicative of a particular class. These properties make the data very sparse which in turn demands large training sets, and very high dimensional parameter vectors. Therefore, the size and complexity of individual instances in the classification problem make it difficult to keep more than a small number of instances in main memory. These particularities make online algorithms, which process a single instance at a time, a good match for natural-language tasks [10].

C. Motivation

Fette et al. [8] hypothesized that phishing email classification appears to be simple text classification problem but, the classification is confounded by the fact that the class of "phishing" emails is nearly identical to the class of real emails. From a learning perspective, this is a challenging problem. However, they didn't experimentally verify it or prove otherwise. Nevertheless, they proposed a new method for detecting these malicious emails by incorporating features specifically designed to highlight the deceptive methods used to fool users. Though their method has been widely adopted by research communities and well studied over the period, very limited to no research work is found in the literature on classifying phishing emails as a text classification problem by solely using readily available actual contents as learning features. Thus, we are motivated to investigate the hypothesis to classify the phishing emails using recently-developed CW linear classifiers.

IV. Experiments and Results

In this section we present the results of running CW Linear Classifier on datasets of emails described in subsection IV-A. Subsection B explains our methodology. The results in classifying the datasets are shown in subsection C.

A. Datasets

We used publicly available datasets from two different sources. We used the phishing datasets available from [12]. To make ham corpora we used public dataset published by SpamAssassin Project [13]. The details on each dataset are summarized in Table I.

We generated 5 sets of datasets containing varying number of phishing and ham emails out of those public datasets. The datasets provided in [12] cover a variety of common phishing schemes. The phishing emails are collected at different times making them the most comprehensive public datasets. We used the first two of the datasets as they were and combined the last two into one so it would contain emails ranging from November 15, 2005 to August 7, 2007. This dataset cover many phishing schemes and contents that evolved over the years. Corpus4 contains all the phishing emails found in [12]. The ham corpus contains more than 20K emails. Corpus4 is the most comprehensive and largest dataset that contains all the phishing emails in [12] from dates ranging from November 27, 2004 to August 7, 2007.

B. Experimental Setup

We wrote a series of short Python scripts to generate files in specific input formats required by classifiers.

Stop words or functional words such as articles, prepositions, etc. that are not useful in the text categorization process were removed during preprocessing using a standard "stop" list in Information Retrieval. Natural Language Toolkit [15], an open source Python library, was used for preprocessing the email texts. Detail on texts preparation and feature extraction is given by Lewis et al. in [16] and in [21].

Fig. 1 shows the flowchart of the methodology we followed to carry out our experiments.

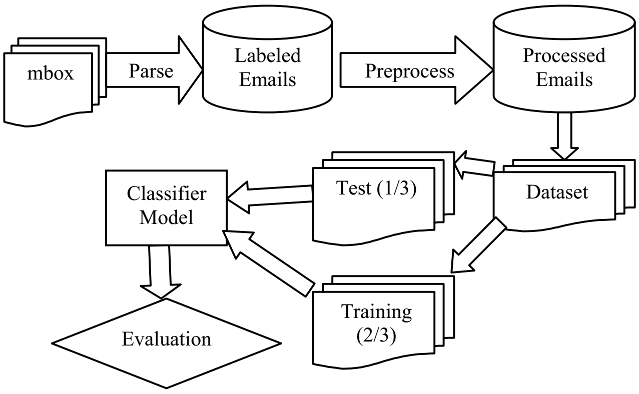

Figure 1. Experimental Setup.

We used the holdout method to train and evaluate the classifiers. Each dataset was divided into two groups, training and testing set, using $2/3^{rd}$-$1/3^{rd}$ split respectively. The training set was used to train the classifier and the test set to estimate the error rate of the trained classifier. We show the average classification results of those 10 random splits on each dataset.

We used the LIBLINEAR implementation of an SVM with a linear kernel as our batch algorithm. LIBLINEAR [24]

TABLE I. SUMMARY OF DATASETS

Dataset	**Examples**			**Training Size (2/3)**			**Test Size (1/3)**			**Feature Size**
	Total	***Phishing***	***Ham***	***Total***	***Phishing***	***Ham***	***Total***	***Phishing***	***Ham***	
Corpus0	1381	412	969	920	275	645	461	137	324	18953
Corpus1	1390	421	969	927	281	646	463	140	323	18647
Corpus2	5485	4516	969	3657	3011	646	1828	1505	323	31617
Corpus3	9396	4516	4880	6264	3011	3253	3132	1505	1627	66658
Corpus4	20026	5349	14677	13351	3567	9784	6675	1782	4893	139742

is a linear classifier for millions of instances and features. We tuned CW and LIBLINEAR classifiers parameters using 10 fold cross validation over training datasets.

C. Results

On these publicly available and highly used real-world datasets in research purpose, Confidence-Weighted Linear classifiers achieved the best accuracy of 99.77%, with false positive rate (FPR - ham emails marked as phishing) of less than one percent across all datasets. This very low false positive indicates that very few legitimate emails are miss classified. False negative rate (FNR - missing a phishing email) on the datasets is also less than one percent across the board. LIBLINEAR on the other hand gave the best accuracy of 99.58% with FPR less than 1% and the worst FNR of 2.3% on Corpus2. Though test accuracy, i.e., the fraction of correctly classified emails, is of limited interest in phishing classification, we report them for comparisons with related works.

Table II shows the classification accuracies of CWLC and LIBLINEAR on all the datasets. CWLC and LIBLINEAR both gave competitive results. CWLC gave better FPR while LIBLINEAR gave better FNR on all datasets. The results given by CWLC are much better than the results in [7, 8, 9, 20] though the experimental conditions and approaches are different.

We also applied Support Vector Machines (SVMs) on the datasets because SVMs have been one of the best classifiers for text categorization tasks [21]. However, the results were not that impressive as the best test accuracy result we found was 81.96% on Corpus2 dataset. The rest of the results were in the range of 70-75%.

Other performance criteria in evaluating binary classifier are specificity and sensitivity. Specificity or true negative rate (TNR) is the proportion of emails that are predicted as ham of all the emails that actually are ham (true negative plus false positive). It can be seen as the probability that the prediction is negative given that the email is not phishing. With higher specificity, fewer good emails are filtered out as phishing. Sensitivity or True Positive Rate (TPR) is the proportion of emails that were classified as phishing (True Positive) of all the emails that actually are phishing (True Positive plus False Negative). It can be seen as the probability that the classification is positive given that the email is phishing. The receiver operating characteristic (ROC) curve usually plots the relationship between sensitivity and specificity. Normally, the ROC curve is presented by plotting false positive (FP) rate vs. true positive (TP) rate.

Furthermore, we also calculated precision, recall and F-measure on the datasets. We achieved the best F-measure of 99.83% compared to 97.64% by Fette et al. [8] and 99.46% by Bergholz et al. [7]. The results are summarized below in Table III. These results are more impressive than the best result we've achieved in our previous work [9].

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have shown that it is possible to detect phishing emails with high accuracy by using Confidence-Weighted Linear Classifiers, using features that are readily available from the email contents without applying extra effort to retrieve heuristic-based phishing specific features.

As the text of phishing emails are often similar to the text of legitimate emails, learning rules like Naïve Bayes might not actually help the classifier [8]. We didn't closely examine our datasets to see if there were any highly similar ham and phishing emails. We would like to further investigate this matter to see how effectively CWLC can classify highly similar phishing email from its ham counterpart.

TABLE II. AVERAGE ACCURACY OF CWLC AND LIBLINEAR

Dataset	**Accuracy**		**FP Rate**		**FN Rate**	
	CWLC	***LIBLINEAR***	***CWLC***	***LIBLINEAR***	***CWLC***	***LIBLINEAR***
Corpus0	98.96%	99.28%	0.06%	0.74%	3.41%	0.68%
Corpus1	99.31%	99.33%	0.06%	0.49%	2.13%	1.08%
Corpus2	99.72%	99.45%	0.84%	2.30%	0.17%	0.17%
Corpus3	99.77%	99.58%	0.15%	0.63%	0.32%	0.20%
Corpus4	99.76%	99.46%	0.08%	0.48%	0.68%	0.68%

TABLE III. AVERAGE PERFORMANCE MEASURE OF CWLC AND LIBLINEAR

Dataset	Precision		Recall		F-Measure	
	CWLC	*LIBLINEAR*	*CWLC*	*LIBLINEAR*	*CWLC*	*LIBLINEAR*
Corpus0	99.84%	98.26%	96.59%	99.32%	98.18%	98.79%
Corpus1	99.86%	98.85%	97.87%	98.92%	98.85%	98.89%
Corpus2	99.82%	99.51%	99.83%	99.83%	99.83%	99.67%
Corpus3	99.84%	99.33%	99.68%	99.80%	99.76%	99.56%
Corpus4	99.79%	98.68%	99.32%	99.32%	99.55%	99.00%

The method we've proposed obviously will not work on image content. Phishers create images that contain the text of the message only in graphical form to bypass the content-based phishing filter. This is also an area we would like to further look into.

The results motivate future work to explore feature selection techniques and inclusion of those selected variables with the most common phishing email features as described in [8] to apply CWLC to see if the predictive accuracy of the classifier can be further improved. As future works we will apply CWLC on the feature sets proposed in [7,8,9] for accurate comparisons of various machine learning techniques on various feature sets.

ACKNOWLEDGMENT

The authors would like to acknowledge the support received from ICASA (the Institute for Complex Additive Systems Analysis, a division of New Mexico Tech). The authors are also grateful to Dr. Mark Dredze for providing us CWLC Library and necessary help in using it.

REFERENCES

[1] "Antiphishing.org. 2009 4th Quarter Report," 2010. [Online]. Available: http://www.antiphishing.org/reports/apwg_report_Q4_2009.pdf. [Accessed: June 13, 2010].

[2] "Help New Security. 420,000 scam emails sent every hour," 2010. [Online]. Available: http://www.net-security.org/secworld.php?id=9421. [Accessed: June 16, 2010].

[3] "Help Net Security. Ipad phishing scams still going strong," 2010. [Online]. Available: http://www.net-security.org/secworld.php?id=9483 [Acessed: June 14, 2010].

[4] R. Dhamija, J. D. Tygar and M. Hearst, "Why Phishing Works," CHI 2006, April 22-27, Montreal, Quebec, Canada.

[5] "Online scammers hope to score on online World Cup enthusiasts," 2010. [Online]. Available: http://infoworld.com/t/malware/online-scammers-hope-score-online-world-cup-enthusiasts-575?source=rss_security_central. [Accessed: June 18, 2010].

[6] "Google Trends," 2010. [Online]. Available: http://www.google.com/trends. [Accessed: June 16, 2010].

[7] A. Bergholz, J. D. Beer, S. Glahn, M-F Moens, G. Paab, S. Strobel, "New Filtering Approaches for Phishing Email," Fifth Conference on Email and Anti-Spam, CEAS 2008, Aug 21-22, 2008, Mountain View, CA.

[8] I. Fette, N. Sadeh and A. Tomasic, "Learning to Detect Phishing Emails", Technical Report CMU-ISRI-06-112, Institute for Software Research International, Carnegie Mellon University, June 2006.

[9] R. Basnet, S. Mukkamala, A. Sung, "Detection of phishing attacks: A machine learning approach," Studies in Fuzziness and Soft Computing 226:373-383, 2008.

[10] M. Dredze, K. Crammer, and F. Pereira, "Confidence-Weighted Linear Classification," Proceedings of the International Conference on Machine Learning (ICML), Omnipress, 2008, pp. 264-271.

[11] "Online Algorithm." [Online]. Available: http://en.wikipedia.org/wiki/Online_algorithm. [Accessed: June 9, 2010].

[12] "phishingcorpus homepage" Feburary 2010. [Online]. Available: http://monkey.org/~jose/wiki/doku.php?id=PhishingCorpus

[13] "Spamassassin public corpus," 2010. [Online]. Available: http://spamassassin.apache.org/publiccorpus/

[14] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," Machine Learning: ECML-98, Tenth European Conference on MachineLearning, 1998, pp. 137-142.

[15] "Natural Language Toolkit," 2010. [Online]. Available: http://www.nltk.org/

[16] D. D. Lewis, Y. Yand, T. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. 2004. JMLR, 5, 361–397.

[17] "SAwin32 – SpamAssassin for Win32," 2010. [Online]. Available: http://sawin32.sourceforge.net/

[18] "The Apach SpamAssassin Project" 2010 [Online]. Available: http://spamassassin.apache.org/

[19] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[20] S. Abu-Nimeh, D. Nappa, X. Wang, and S. Nair, "A Comparison of Machine Learning Techniques for Phishing Detection," APWG eCrime Researchers Summit, October 4-5, 2007, Pittsburgh, PA, USA.

[21] R. Basnet, G. Torres, A. Sung, B. Ribeiro. Translation Based Foreign Language Text Categorization. Unpublished.

[22] J. Ma, A. Kulesza, M. Dredze, K. Crammer, L. K. Saul, and F. Pereira, "Exploiting Feature Covariance in High-Dimensional Online Learning," Proceedings of the International Conference on Artificial Intellignece and Statistics (AISTATS), May 2010, pp. 493-500.

[23] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying Suspicious URLs: An Application of Large-Scale Online Learning," Proceedings of the International Conference on Machine Learning (ICML), June 2009, pp. 681-688.

[24] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," 2008. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/liblinear/