Feature Selection for Improved Phishing Detection

Ram B. Basnet[1,2], Andrew H. Sung[1,2], Quingzhong Liu[3]
[1]Computer Science and Engineering, New Mexico Tech, Socorro, NM, USA
[2]ICASA, New Mexico Tech, Socorro, NM, USA
{rbasnet|sung}@cs.nmt.edu
[3]Computer Science, Sam Houston State University, Huntsville, TX, USA
qxl005@shsu.edu

Abstract. Phishing – a hotbed of multibillion dollar underground economy – has become an important cybersecurity problem. The centralized blacklist approach used by most web browsers usually fails to detect zero-day attacks, leaving the ordinary users vulnerable to new phishing schemes; therefore, learning machine based approaches have been implemented for phishing detection. Many existing techniques in phishing website detection seem to include as many features as can be conceived, while identifying a relevant and representative subset of features to construct an accurate classifier remains an interesting issue in this particular application of machine learning. This paper evaluates correlation-based and wrapper-type feature selection techniques using real-world phishing data sets with 177 initial features. Experiments results show that applying an effective feature selection procedure generally results in statistically significant improvements in the classification accuracies of—among others—Naïve Bayes, Logistic Regression and Random Forests, in addition to improved efficiency in training time.

Keywords: feature selection, phishing detection, phishing webpage, evolutionary algorithms, anti-phishing

1 Introduction

Phishing has become something of a plague on the Internet. A typical phishing webpage may mimic a trusted third party such as a bank, a financial or e-commerce entity, etc. and induces Internet users to divulge their private information, e.g., username, password, bank account, credit card number, etc. Phishing attacks can cost not only the individual consumers but also well-known organizations and corporations whose brands are compromised in the attacks. Despite the efforts by the research community, the industry, and law enforcement to develop solutions to tackle the problem, phishing has shown no sign of abating. A recent report by the Anti-phishing Working Group (APWG) [1] indicated more sophisticated schemes seem to have been used in phishing attacks that also exploited an increased number of brands.

Since a black-list of phishing sites is unable to detect "zero-day" or new attacks [27], a machine-learning approach has been proposed to train a classifier with large amount of data. The classifiers reported in the literature [6, 9, 21], however, seem to include very large numbers of features. Since each feature included can increase the cost (storage, preprocessing, training time, etc.) of a system without possibly contributing to the classifier's performance, there is a strong motivation to design and implement systems with small feature sets as, according to M. Hall, "a good feature subset is one that contains features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other" [2].

In this paper, we evaluate two common feature selection techniques – correlation-based and wrapper-based techniques – for phishing detection. We compare these feature selection techniques using two feature space searching techniques (genetic and greedy forward selection) and conduct the experiments and evaluate results on a real-world dataset with more than 16,000 phishing webpages and more than 32,000 non-phishing webpages.

2 Related Work

A number of recent papers have evaluated various machine learning techniques in detecting phishing emails, URLs, and webpages [5, 6, 8, 9, 12, 21, 25, etc.]. Most research works, however, use all the features that can be conceived at the time and as a result feature selection study in phishing detection can be found sparingly.

In [9] Whittaker et al. describe the design and performance characteristics of Google's phishing blacklist. Their proprietary classifier, implementation of the online gradient descent logistic regression learning algorithm, performs the automatic feature selection—finding potential useful features to include in classification model and discarding the ones that do not contribute to the model.

Toolan et al. apply feature selection techniques to phishing and spam email classification using 40 features [17].

3 Feature Selection Methods

We evaluate two commonly used feature selection techniques in this paper.

3.1 Correlation-based Feature Selection (CFS)

CFS exploits the inter-dependency or predictability of one variable with another to generate the optimal subset of features with the goals of improving classification performance and reducing the feature dimension. As a simple filter algorithm that evaluates an importance of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them, CSF essentially ranks feature subsets in the search space of all possible feature subsets according to a correlation based heuristic evaluation function:

$$M_s = \frac{k\overline{\mathrm{r}_{\mathrm{cf}}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}, \tag{1}$$

where M_s is the heuristic "merit" of feature subset S containing k features, $\overline{r_{cf}}$ is the mean feature-class correlation ($f \in S$), and $\overline{r_{ff}}$ is the average feature-feature inter-correlation [2]. The numerator of (1) provides an indication of how predictive of the class a set of features are and the denominator provides how much redundancy there is among the features.

3.2 Wrapper Feature Selection (WFS)

Wrapper feature selector evaluates feature subsets by using a machine learning algorithm with the rationale that the induction method that will ultimately use the feature subset should provide a better estimate of accuracy. Though using the induction algorithm itself as the measure stands the best chance of identifying the “optimal” feature subset, wrapper feature selectors give highly variable cross-validation accuracy when the number of instances is small [3], and are prohibitively slow on large data sets using cross-validation [7].

3.3 Search Techniques

Searching the space of feature subsets within reasonable time constraints is vital in any feature selection technique. There are several search heuristics such as forward selection, backward elimination, best first, search using genetic algorithms, etc. Forward search and backward elimination are common and simple techniques where the algorithms consider only additions or deletions respectively to the feature subset [22], [23]. We evaluate greedy forward search and genetic algorithm in this study.

Genetic algorithms (GA) are adaptive search techniques based on the principles of natural selection and mutation in biology [26]. GA typically maintains a constant sized population of individuals which represent samples of the space to be searched. Each individual is evaluated on the basis of its overall fitness—how good a feature subset is with respect to an evaluation strategy. The solution space is searched in parallel which helps in avoiding local optima. The algorithm is an iterative process where new individuals (offspring) for the next generation are formed by using two main genetic operators such as crossover and mutation to the members of the current generation. Mutation randomly changes (thus adding or deleting features) one or more components of selected individuals. Crossover combines different from a pair of subsets into a new subset. Better feature subsets have a greater chance of being selected to form a new subset through crossover or mutation, effectively evolving good subsets over time [2], [4], [15].

4 Experiments and Results

We used 10 times 10-fold cross-validation (unless otherwise stated) to estimate the test accuracy. The experiments were run on a machine with 2 dual-core 2 GHz Intel processors with 4 GB memory. To conduct all the experiments, we used WEKA (Waikato Environment for Knowledge Analysis) data mining framework [14] with default parameter values where appropriate.

We compare feature selection and search techniques using 3 commonly used machine learning algorithms in problems similar to ours: Naïve Bayes (NB) [17], Logistic Regression (LR) [28] and Random Forests (RF) [16]. We also tried evaluating C4.5 [24] and Multilayer Perceptron, but the Wrapper feature selection technique was prohibitively slower taking months for these slower classifiers.

4.1 Data Sets

For phishing webpages, we used confirmed phishing URLs from PhishTank [11]. PhishTank, operated by OpenDNS, is a collaborative clearing house for data and information about phishing on the Internet. A phish once submitted is verified by a number of registered users to confirm it as phishing. We collected first set of phishing URLs from June 1 to October 31, 2010. Phishing tactics used by scammers evolve over time. In order to investigate these evolving tactics and to closely mimic our experiments as in the real-world scenario, we collected second batch of confirmed phishing URLs that were submitted for verification from January 1 to May 3, 2011. We used scripts [13] to automatically detect and expand the shortened URLs provided by online service longurl.org.

We collected our legitimate webpages from two public data sources. One is the Yahoo! directory[1], the web links in which are randomly provided by Yahoo's server redirection service [10]. We used this service to randomly select a URL and download its page contents along with server header information. In order to cover wider URL structures and varieties in page contents, we also made a list of URLs of most commonly phished targets. We then downloaded those URLs, parsed the retrieved HTML pages, and harvested and crawled the hyperlinks therein to also use as benign webpages. We made the assumption, which we think is reasonable, to treat those webpages as benign, since their URLs were extracted from a legitimate sources. These webpages were crawled between September 15 and October 31 of 2010. The other source of legitimate webpages is the DMOZ Open Directory Project[2]. DMOZ is a directory whose entries are vetted manually by editors.

Based on the date on which phishing URLs were submitted to PhishTank for verification, we generated two data sets. The first data set, we refer to it as DS1, contains 11,240 phishing webpages submitted before October 31, 2010 and 21,946 legitimate webpages from Yahoo! and seed URLs. The second data set, we refer to it as DS2, contains 5,454 phishing webpages submitted for verification between January 1 and May 3 of 2011 and 9,635 randomly selected legitimate webpages from DMOZ. We discarded the URLs that were no longer valid as the page couldn't be accessed to extract features from their contents.

4.2 Features

We start with a set of 177 features of which 38 are content-based and the rest are URL-based. Content-based features are mostly derived from the technical (HTML) contents of webpages e.g., counting external and internal links, counting IFRAME tags, and checking whether IFRAME tag's source URLs are present in blacklists and search engines, checking for password field and testing how the form data is transmitted to the servers (whether Transport Layer Security is used and whether "GET" or "POST" method is used to transmit form data with password field), etc.

URL-based features include lexical properties of URLs such as counting number of ".", "-", "_", etc. in various parts of URLs, checking whether IP address is used and what type of notation is used to represent the IP address in place of a domain name.

[1] http://dir.yahoo.com
[2] http:www.dmoz.org

URLs and domain part of the URLs are checked against top 3 search engines (Google, Yahoo, and Bing) indexes to see if the URLs are indexed. Features also include checking IPs and domain name of the URLs against the top list of IPs and domains historically popular for hosting phishing and other malicious websites. Features also include a list of eye-catching keywords (e.g., log, click, pay, free, bonus, bank, user, etc.) that are more commonly used in phishing URLs to deceive the end users.

4.3 Feature Selection

Table 1 displays the classification accuracies of Naïve Bayes, Logistic Regression and Random Forests classifiers with and without feature selection using CFS on DS1 data set. Genetic search technique resulted in a subset of 42 features out of 177 features; whereas greedy forward search (Greedy FS) selected all the features (results are not shown as they are same as without feature selection, grayed row). Genetic search technique improved Naïve Bayes classifier's results the most with its error improving from 2.2% to 1.7% with the significant reduction in both FPR and FNR.

Table 1. Classification results of Naïve Bayes, Logistic Regression and Random Forests classifiers using correlation-based feature selection method with genetic search and greedy forward selection search techniques on DS1 data set

Search Technique	# Features	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection	177	2.2	2.2	2.2	0.5	0.3	0.8	0.4	0.3	0.7
Genetic Search	42	1.7-	1.6-	1.9-	0.9+	0.9+	1.0+	0.5+	0.4+	0.6-

+,- statistically significant degradation or improvement

Table 2 shows the classifiers' results using Wrapper feature selection technique. Classification accuracies on Naïve Bayes, Logistic Regression and Random Forests are compared using two search techniques, genetic search and greedy forward selection. Unlike CFS, Wrapper based technique selected smaller subsets of features for all three classifiers using both genetic search and greedy forward search techniques. Though Wrapper based technique was notably slower compared to CFS technique, it aided in significant improvement in the classification accuracies of all classifiers. For example, with the subset of 14 selected features using greedy forward search technique, RF yielded the best error rate of 0.3% along with the best FPR and FNR of 0.2% and 0.5%, respectively on DS1 data set. Besides yielding higher accuracy, the reduced feature subset noticeably improved the training time.

Table 2. Classification results of Naïve Bayes, Logistic Regression and Random Forests classifiers using wrapper feature selection method with genetic search and greedy forward Cselection search techniques on DS1 data set

Search Technique	Classifier Performance											
	Naïve Bayes				Logistic Regression				Random Forests			
	# Features	Error (%)	FPR (%)	FNR (%)	# Features	Error (%)	FPR (%)	FNR (%)	# Features	Error (%)	FPR (%)	FNR (%)
Without feature selection	177	2.2	2.2	2.2	177	0.5	0.3	0.8	177	0.4	0.3	0.7
Genetic Search	62	1.3-	0.9-	2.1-	70	0.3-	0.2-	0.6-	91	0.4	0.2-	0.6-
Greedy FS	12	1.5-	1.0-	1.6-	13	0.4-	0.2-	0.8	14	0.3-	0.2-	0.5-

+,- statistically significant degradation or improvement

4.4 Concept Drift

Phishers come up with new tactics over time to invade the existing filters. Features developed and selected from observing a particular data set can yield highly accurate classification results when trained and tested on disjoint subsets of the same data set. But do these results hold on testing new data (possibly from different sources) using the features extracted and selected from older data set? We try to investigate this question in the following experiments.

First, using the selected features from DS1 data set, we ran our experiments on DS2 data set and show the results in Table 3. With CFS and genetic search combination, we see slightly better results for NB, but no improvement in overall error rates for LR and RF classifiers compared to the results using all the features.

Table 3. Results of using selected features from DS1 data set on DS2 data set.

Feature Selection	Search Technique	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection		0.8	0.2	1.9	0.4	0.3	0.7	0.3	0.0	0.7
CFS	Genetic Search	0.7-	0.2	1.5-	0.4	0.1-	0.9+	0.3	0.1+	0.6-
Wrapper	Genetic Search	1.6+	1.0+	2.7+	0.5+	0.4+	0.6-	0.3	0.1+	0.6-
	Greedy FS	2.7+	2.3+	3.4+	0.2-	0.0-	0.6-	0.4+	0.2-	0.8+

+,- statistically significant degradation or improvement

Wrapper feature selection method with both genetic and greedy forward search techniques, on the other hand, degraded classification accuracy of most of the classifiers. A subset of 42 features selected from DS1 data set using CFS and genetic search combination yielded a small improvement in error rate for Naïve Bayes on DS2 data set. The same feature subset, however, didn't improve the error rates of Logistic Regression and Random Forests. The combination of wrapper feature selection and greedy forward search technique improved classification accuracy of LR but decreased accuracies for NB and RF classifiers.

Table 4 shows the experimental results on testing newer data set DS2 using the models generated from training older data set DS1. As expected, the classification accuracy degraded significantly for all the classifiers. Interestingly, Naïve Bayes' performance results degraded the least while Random Forests' performance degraded the worst with or without performing feature selection in this context. Results show that the complete features are better than selected smaller subsets when it comes to classifiers' robustness towards concept drift in this context. Results suggest that as phishing tactics change over time, so must the data models in order to keep the models fresh and achieve optimal performance results.

Table 4. Results of training on older data set DS1 and testing the models on newer data set DS2 for the combinations of CFS and wrapper based feature selection techniques with genetic and greedy search techniques

Feature Selection	Search Technique	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection		3.2	0.5	8.0	3.8	0.2	10.2	4.0	0.0	11.1
CFS	Genetic Search	3.6+	0.2-	9.6+	4.8+	0.1-	13.2+	5.2+	0.0	14.2+
Wrapper	Genetic Search	8.5+	0.5	22.7	8.7+	0.1-	24.0+	8.3+	0.0	23.0+
	Greedy FS	5.8+	0.0-	16.0+	7.4+	0.0-	20.6+	16.1+	0.0	44.5+

+,- statistically significant degradation or improvement

5 Conclusions and Future Work

In this paper, we evaluated two common feature selection techniques: correlation based and wrapper based feature selection techniques for phishing website detection. We also evaluated two search methods: genetic search and greedy forward selection. Applying the techniques on real-world data sets, we experimentally demonstrated that feature selection technique can improve classification results when training and testing on the disjoint subsets of a data set.

Though wrapper based feature selection technique was extremely slow (taking several weeks) for slower classifier like C4.5 and Multilayer Perceptron (results not shown) as compared to correlation based feature selection (CFS) technique (taking hours or days), wrapper based technique improved classifiers accuracies significantly compared to CFS technique for the evaluated classifiers. Using all the features, however, yielded better results when training with older data set and testing the generated models with newer data set.

As future work, it would be interesting to evaluate other feature ranking and selection techniques such as principle component analysis, latent semantic analysis, chi-squared attribute evaluation, etc. and other feature space search methods such as greedy backward elimination, best first, etc.

Acknowledgment

This research work was supported by Institute for Complex Additive Systems Analysis (ICASA), a research division of New Mexico Tech and by National Institute of Justice award no. 2010-DN-BX-K223.

References

1. APWG Phishing Activity Trends Report- 2nd Half 2010, http://apwg.org/reports/apwg_report_h2_2010.pdf. Accessed on October 21 (2011)
2. Hall, M.A.: Correlation-based Feature Selection for Machine Learning. Hamilton, NewZealand (1999)
3. Kohavi, F., G. H. John, G.H.: Wrappers for Feature Subset Selection. Artificial Intelligence. 97, 273-324 (1997)
4. D. E Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
5. Basnet, R.B., Mukkamala, S., Sung, A.H.: Detection of phishing attacks: A machine learning approach. In: Prasad, B. (ed.) Studies in Fuzziness and Soft Computing, vol. 226, pp. 373-383. Springer, Heidelberg (2008)
6. Ma, J., Saul, L.K., Safage, S., Voelker, G.M.: Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs. In: ACM SIGKDD, pp. 1245-1253. Paris, France, (2009)
7. Caruna, R., Freitag, D.: Greedy Attribute Selection. In: 11th International Conference in Machine Learning. Morgan Kaufmann, San Francisco (1994)
8. Zhang, Y., Hong, J., Cranor, L.: CANTINA: A Content-Based Approach to Detecting Phishing Web Sites. In: WWW 2007, Banff, Alberta, Canada, ACM Press (2007)
9. Whittaker, C., Ryner, B., Nazif, M.: Large-Scale Automatic Classification of Phishing Pages. In: 17th Annual Network and Distributed System Security Symposium, California, USA (2010)
10. Yahoo! Inc.: Random Link – random, http://random.yahoo.com/fast/ryl
11. PhishTank - Out of the Net, into the Tank, http://www.phishtank.com/developer_info.php
12. Garera. S., Provos, N., Chew, M., Rubin, A.D.: A Framework for Detection and Measurement of Phishing Attacks. In: 5th ACM Workshop on Recurring Malcode (WORM ‘07), pp. 1-8. ACM Press, New York (2007)

13. PyLongURL - Python Library for LongURL.org, http://code.google.com/p/pylongurl/
14. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. ACM SIGKDD Explorations. 11, 1-8 (2009)
15. Vafaie, H., Jong, K.D.: Robust Feature Selection Algorithms. In: International Conference on Tools with Artificial Intelligence (ICTAI), pp. 356-363 (1993)
16. Breiman, L.: Random Forests. Machine Learning. 45, 5-32 (2001)
17. John, G., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: 11th International Conference on Uncertainty in Artificial Intelligence, pp. 338-345. San Mateo, USA (1995)
18. Toolan, F., Carthy, J.: Feature Selection for Spam and Phishing Detection: In: eCrime Researchers Summit (eCrime), pp. 1-9. Dallas, TX (2010)
19. Miyamoto, D., Hazeyama, H., Kadobayashi, Y.: A Proposal of the AdaBoost-Based Detection of Phishing Sites. In: 2nd Joint Workshop on Information Security (2007)
20. Fette, I., Sadeh, N., Tomasic, A.: Learning to Detect Phishing Emails. In: 16th International Conference on World Wide Web, pp. 649-656 (2007)
21. Basnet, R.B., Sung, A.H.: Classifying Phishing Emails Using Confidence-Weighted Linear Classifiers. In: International Conference on Information Security and Artificial Intelligence, pp. 108-112, Chengdu, China (2010)
22. Kittler, J.: Feature Set Search Algorithms. In: Chen, C.H. (ed.) Pattern Recognition and Signal Processing. The Netherlands (1978)
23. Miller, J.: Subset Selection in Regression. Chapman and Hall, New York (1990)
24. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA, (1993)
25. Basnet, R.B., Sung, A.H., Liu, Q.: Rule-Based Phishing Attack Detection. In: International Conference on Security and Management (SAM'11), Las Vegas, NV (2011)
26. Holland, J. H.: Adaption in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, MI (1975)
27. Ludl, C., McAllister, S., Kirda, E., Kruegel, C.: On the Effectiveness of Techniques to Detect Phishing Sites. *DIMVA '07 - the 4th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment* (pp. 20-39). Springer-Verlag (2007).
28. le Cessie, S., van Houwelingen, J.C.: Ridge Estimators in Logistic Regression. Applied Statistics. Applied Statistics. 41, 191-201(1992)