A Novel Approach for Phishing URLs Detection

Purva Agrawal[1], **Dharmendra Mangal**[2]

[1]M.E. Scholar,Dept. of Information Technology, Medicaps Institute of Technology and Management Indore (India)

[2]Assistant Professor, Department of Computer Science, Medicaps Institute of Technology and Management Indore (India)

Abstract: ***Seeking sensitive user data in the form of online banking user-id and passwords or credit card information, which may then be used by 'phishers' for their own personal gain is the primary objective of the phishing. With the increase in the online trading activities, there has been a phenomenal increase in the phishing scams which have now started achieving monstrous proportions. This paper gives strategies for distinguishing phishing sites by dissecting different components of phishing URLs by Machine learning systems. It talks about the systems utilized for identification of phishing sites in view of lexical features. We consider different data mining approaches for assessment of the features to show signs of improvement comprehension of the structure of URLs that spread phishing. We use KNN, Regression and SVM classifiers.***

Keywords: Lexical Analysis, Phishing URL, Machine Learning, KNN, Regression and SVM.

1. Introduction

A Phishing is an attempt by an individual or a group to steal personal confidential information such as passwords, credit card information from unsuspecting victims for identity theft, financial gain and other fraudulent activities. In the current scenario, when the end user wants to access his confidential information online (in the form of money transfer or payment gateway) by logging into his bank account or secure mail account, the person enters information like username, password, credit card no. etc. on the login page. But quite often, this information can be captured by attackers using phishing techniques(for instance, a phishing website can collect the login information the user enters and redirect him to the original site). There is no such information that cannot be directly obtained from the user at the time of his login input.

Whittaker et al. [5] define a phishing web page as "any web page that, without permission, alleges to act on behalf of a third party with the intention of confusing viewers into performing an action with which the viewers would only trust a true agent of a the third party." This definition, which is similar to the definition of "web forgery", covers a wide range of phishing pages from typical ones – displaying graphics relating to a financial company and requesting a viewer's personal credentials – to sites which claim to be able to perform actions through a third party once provided with the viewer's login credentials. Thus, a phishing URL is a URL that leads user to a phishing web page.

2. Literature Review

"The Phishing Guide" by Ollmann (2004) gives a detailed understanding of the different techniques often included in phishing attacks [1]. The phenomenon that started as simple URL persuading the receiver to reply with the information the attacker required has evolved into more advanced ways to deceive the victim. Links in URL and false advertisements sends the victim to more and more advanced fraudulent websites designed to persuade the victim to type in the information the attacker wants, for example to log into the fraudulent site mimicking the company's original. Ollmann also presents different ways to check whether websites are fraudulent or not. Apart from inspecting whether the visited site really is secure through SSL (Secure Sockets Layer), the user should also check that the certificate added to the website really is from the company it claims to be from and that it is signed by a trusted third party. Focusing more attention on the URL can also often reveal fraudulent sites. There are a number of ways for the attackers to manipulate the URL to look like the original, and if the users are aware of this they can more easily check the authentication of the visited site.

Watson et al. (2005) describe in their White Paper, "Know your enemy: Phishing", different real-world phishing attacks collected in German and United Kingdom honeynets [2]. Honeynets are open computer networks designed to collect information about different attacks out in the real world, for further forensic analysis. They noticed that phishing attacks using vulnerable web servers as hosts for predesigned phishing sites are by far the most common, compared to using self-compiled servers. A compromised server is often host for several different phishing sites. These sites are often only active for a few hours or days after being downloaded to the server.

Garera et al. (2007) focus on studying the structure of URLs employed in various phishing attacks. They find that it is often possible to tell whether or not a URL belongs to a phishing attack without requiring any knowledge of the corresponding page data. This paper describe several features that can be used to distinguish a phishing URL from a benign one. These features are used to model a logistic regression filter that is efficient and has a high accuracy. The paper use this filter to perform thorough measurements on several million URLs and quantify the prevalence of phishing on the Internet today [3].

Ma et al. (2009) propose a method to classify malicious URLs using variable number of lexical and host-based properties of the URLs. They describe an approach for problem based on automated URL classification, using statistical methods to discover the tell-tale lexical and host-based properties of malicious Web site URLs. These methods are able to learn highly predictive models by extracting and automatically analysing tens of thousands of features potentially indicative of suspicious URLs [4].

Whittaker et al. (2010) describe the design and performance characteristics of a scalable machine learning classifier that has been used in maintaining Google's phishing blacklist automatically. Their proprietary classifier analyses millions of pages a day, examining the URL and the contents of a page to determine whether or not a page is phishing. Their system classifies web pages submitted by end users and URLs collected from Gmail's spam filters. Though some URL based features are similar, we propose several new features and evaluate our approach with publicly available machine learning algorithms and public data sets. Unlike their approach, we do not use any proprietary and page content based features [5].

Zhang et al.(2007) present CANTINA, content-based approach to detect phishing websites, based on the TF-IDF information retrieval algorithm and the Robust Hyperlinks algorithm [6]. By using a weighted sum of 8 features (4 content related, 3 lexical, and 1 WHOIS-related) they show that CANTINA can correctly detect approximately 95% of phishing sites. The goal of our approach is to avoid downloading the actual web pages and thus reduce the potential risk of analysing the malicious content on user's system. In order to achieve this goal, we evaluate only the features related to URLs.

Besides machine learning (ML) based techniques, there exist many other approaches in phishing detection. Perhaps, the most widely used anti-phishing technology is the URL blacklist technique that most modern browsers are equipped with [7]. Other popular methods are browser based plug-in or add-in toolbars. SpoofGuard [8] uses domain name, URL, link, and images to evaluate the spoof probability on a webpage. The plug-in applies a series of tests, each resulting in a number in the range from 0 to 1. The total score is a weighted average of the individual test results. There has been an attempt to detect phishing attack using user generated rules [9]. Other anti-phishing tools include SpoofStick [10], SiteAdvisor [11], Netcraft anti-phishing toolbar [12], AVG Security Toolbar [13] etc.

3. Proposed Method

The work comprises of lexical feature extraction of collected URLs and investigation. The primary step is the gathering of phishing and benign URLs. The lexical based feature extractions is used to shape a database of feature values. The database is learning mined utilizing various machine learning strategies. Subsequent to assessing the classifiers, a specific classifier is chosen and is executed in MATLAB. Figure 1 shows the proposed flow diagram.

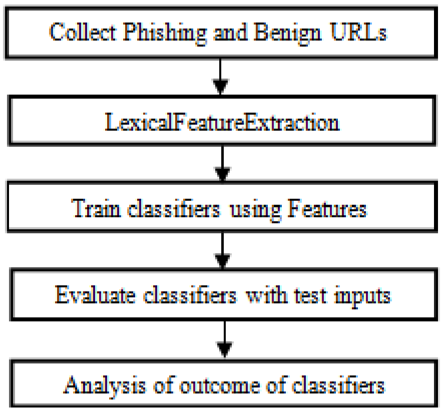
Collect Phishing and Benign URLs
LexicalFeatureExtraction
Train classifiers using Features
Evaluate classifiers with test inputs
Analysis of outcome of classifiers

Figure 1: Flow diagram for the proposed work

Collection of URLs

In this paper, we have taken URLs of benign websites from www.alexa.com [14] www.dmoz.org [15] and personal web browser history. The phishing URLs were collected from www.phishtak.com [16].

Lexical Feature Extraction

Lexical features are the textual properties of the URL itself, not the substance of the page it indicates. URLs are human-readable text strings that are parsed in a standard manner by customer projects. Through a multistep determination process, programs make an interpretation of each URL into guidelines that find the server facilitating the site and indicate where the site or asset is set on that host. Following Figure 2 shows Flow diagram of proposed researched.

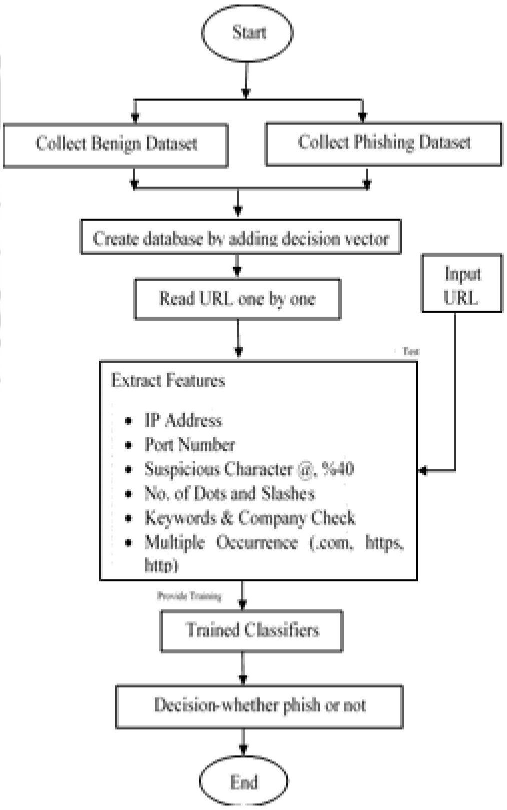

Figure 2: Flow diagram of proposed research

Following properties are recognized:

IP Address

Phishing URLs often contain IP addresses to hide the actual URL of the website. For example a website URL may be extremely long and look suspicious such as something like this "http://www.freewebhosting.com/markswebsite/todaysphishingpage.html" but the URL that contains the IP address is typically shorter and more standard such as this "http://66.135.200.145". Phishers use IP addresses to obscure the actual domain name of the website being visited. URL detection methods can look for an IP address in the URL and add to a phishing score if one is found. However legitimate websites sometimes use IP addresses especially for internal private devices that aren't accessible to the public. Network devices such as routers, servers, and network printers are every so often accessed using an IP address in a web browser.

Protocol

The <protocol> portion of the URL demonstrates which network protocol ought to be utilized to fetch the requested resource. The most widely used protocols are Hypertext Transport Protocol or (http), HTTP with Transport Layer Security (https), and File Transfer Protocol (ftp). Spoofguard [8] identified several standard port numbers as 21, 70, 80, 443, 1080. These correspond to common services used in web browsers such as FTP, Gopher, web, secure web, and SOCKS. If a suspicious unknown port number is used the phishing score is increased because attackers often use different port numbers to bypass security detection programs that may monitor a specific port number.

Number of Dots and Slashes

There are a numerous ways for attackers to create Legitimate-looking URLs. Of course, legitimate URLs also can contain a number of dots, and this does not make it a phishing URL, however there is still information conveyed by this feature, as its inclusion increases the accuracy in our empirical evaluations. It is likely that legitimate URLs contain slightly more dots in some cases, however, phishing URLs typically cannot have this number reduced considerably in that attackers typically have to attach the target domain/hostname in the phishing URL as a deception. This feature is simply the maximum number of dots ('.') contained in any of the links present in the URL, and is a continuous feature. Generally, the URL should not contains more number of slashes. If URL contains more than five slashes then that URL will be a phishing URL. [17][18]

Suspicious Character @ and %40

Some recent browser vulnerabilities have helped in misleading the users too. One such example was the Internet Explorer URL spoofing vulnerability. This vulnerability allows an attacker to alter the address displayed on the address bar of the browser, while a fake web site is opened. Checking URL against special symbols such as '@', is another feature because many of phishing URLs modified using these symbols which makes it possible to write URLs that appear legitimate but actually lead to different pages. Presence of @ symbol in the URL indicates that, all text before @ is comment. Whatever written before @ is ignored and the trailing URL is visited. For example http://www.usfca.edu@www.cse.scu.edu/~tschwarz/coen252_03/Lectures/URLObscuring.html. If this URL is visited, the user is actually visiting a page on www.cse.scu.edu/~tschwarz/coen252_03/Lectures/URLObscuring.html. This allows an attacker to modify the address displayed on the address bar of the browser, while a phished URL is opened. In some cases, Phishers use the ASCII encoding of the '@' character i.e. %40. Since '@' can seem phish so, phishers uses hexadecimal equivalent number for attack. For example: http://129.210.2.1%40www.usfca.edu. If this URL is visited, the user is actually visiting a page on www.usfca.edu .

Multiple Occurrence (.com, https, http)

The occurrence of multiple '.com', 'https', 'http' in an URL impose a threat of phishing by redirecting request to the followed http(s) URL. The nomenclature "=http://" or "=https://" allows the redirection attack. Occurrence of multiple '.com' in URL is also suspicious and may lead to the phishing attack by the means of URL redirection. Here the given example shows: http://www.google.com/url?q=http://www.badsite.comThis URL would refer a user from one site (in this case, google.com) to another site, badsite.com.

Keyword Check

There are a lot of variety of URL based properties which can be used in a phishing URL. In this research work, we find such properties to detect phishing URL. Coding part of this research contains following properties: "update", "click", "user", "termination", "confirm", "account", "banking", "secure", "ebayisapi", "webscr", "login", "free", "lucky", "bonus" and "signin".

Company Check

Phishing websites want to look as legitimate as possible so they every so often contain the name of the company they are aiming. Researchers from Google and Johns Hopkins University identified the most dominant phishing targets. The list includes eBay, Paypal, Volksbank, Wells Fargo, Bank of America, Private Banking, HSBC, Chase, Amazon, Banamex, and Barclays [3]. The matching domain names of these companies were determined and the company keyword list comprises: ebay, paypal, volksbank, wellsfargo, bankofamerica, privatebanking, hsbc, chase, amazon, banamex, and Barclays. The overall phishing score increases if one of the keywords listed above is found in the URL.

Machine Learning Algorithms

The input to the classifiers in MATLAB is two .txt files; newben.txt and newphis.txt. The three machine learning algorithms considered for processing the feature set are:

- ***K-NN:*** It is based on closest training examples in the feature space. An object is classified by a majority vote of its neighbors.
- ***SVM:*** The SVM performs classification by finding the hyper plane that maximizes the margin between two classes. The vectors that define the hyper plane are the supportvectors.
- ***Regression Classifier:*** Regression Trees are an axis parallel Decision Tree which can induce trees for predicting both categorical and real-valued targets, and

hence they can be used for regression as well as classification.

4. Simulation and Results

The performance of proposed algorithms which makes use of the proposed novel features to improve the effectiveness to detect phishing URLs has been studied by means of MATLAB simulation. The simulation results are depicted right the way through confusion matrix. The confusion matrix for each classifier SVM, k-NN, Regression shows the Accuracy, Specificity, True Positive Rate and False Positive Rate to show the signs of improvement.

Here we have used two split cases to verify the improvement that are:

- 60 – 40 Split Case
- 90 – 10 Split Case

60 – 40 split case

Here percentage split is set to 60-40 i.e. 40 percentage of the dataset is taken as training data with which classifier is then able to perceive a classification model and once the learning phase is complete with given 40 percentage of data, the classifier is given unclassified URLs as input here, 60 percentage as test data, and a predicted class is returned as output.

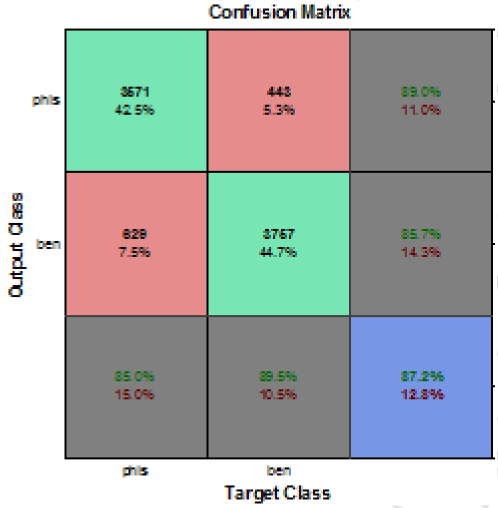

Figure 3: Confusion matrix for KNN

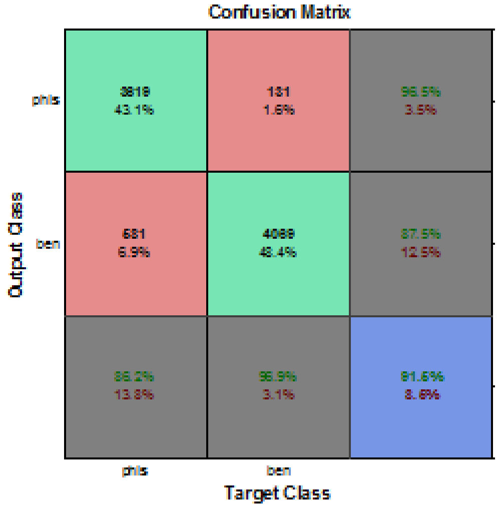

Figure 4: Confusion matrix for Regression classifier

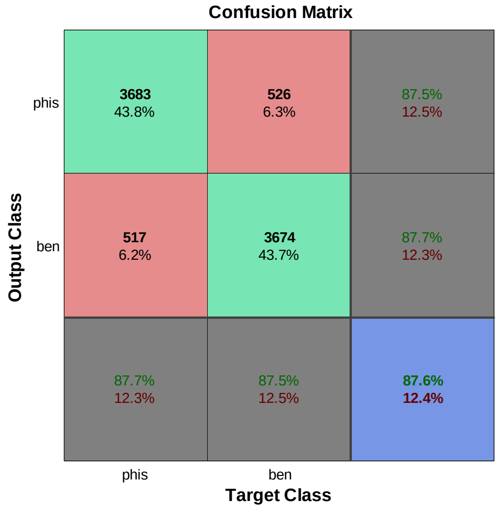

Figure 5: Confusion matrix for SVM

90-10 split case

Here percentage split is set to 90-10 i.e. 10 percentage of the dataset is taken as training data with which classifier is then able to perceive a classification model and once the learning phase is complete with given 10 percentage of data, the classifier is given unclassified URLs as input here, 90 percentage as test data, and a predicted class is returned as output.

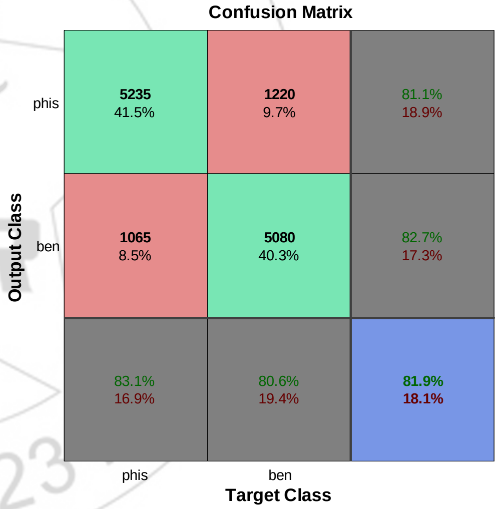

Figure 6: Confusion matrix for KNN

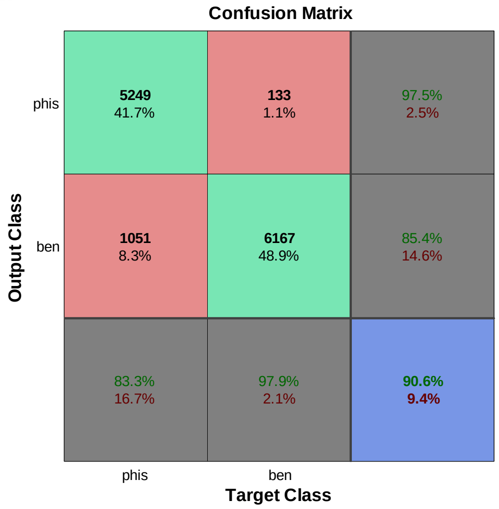

Figure 7: Confusion matrix for Regression classifier

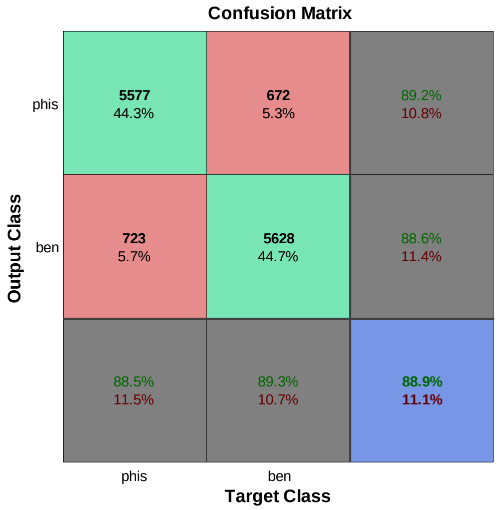

Figure 8: Confusion matrix for SVM

The efficiency of the proposed lexical URL analysis is calculated by measuring the performance of the proposed method. The objective is to measure the efficiency of the lexical analysis approach, discussed in proposed architecture as a distinguishing feature for phishing and legitimate URLs. We analysed phishing and legitimate URLs lexically to study their relation in predicting URL's class i.e. Benign or Phishing.

On the basis of real-life URLs obtained from different sources, we demonstrate that our proposed methodology is showing the signs of improvement by lexically analysing the URLs through classifiers. This study empirically confirms previous studies that URLs contain more information than simply identifying reachability to a resource, and with our enhancements resulted in very good classification accuracy. On observing Table 1 shown below, it was found that the proposed approach outperforms previous research work on the basis of accuracy for all three classifiers.

Table 1: Result compared to previous work

Test Options	Classifiers	Previous Work (Success Rate)	Proposed Approach
Percentage Split-60	SVM	85.3	87.6
	Regression	89.8	91.5
	KNN	86.1	87.2
Percentage Split-90	SVM	83.3	88.9
	Regression	81.4	90.6
	KNN	77.7	81.9

5. Conclusion

Phishing techniques have not only grown in number, but also in sophistication. Phishing recognition techniques are rapidly varying to keep up with the novel techniques used by phishers.

While generalizing about URLs, it is hard to conclude whether a website is legitimate or phishing just by the URL contents alone. One can on the other hand add to a phishing score if certain features are spotted that are more likely found in phishing URLs rather than legitimate URLs.

We used a simple classification techniques since our aim was to evaluate the feature, and not the classifiers. This work proved diagnostically that the proposed methodology is showing the signs of improvement utilizing different lexical features for detecting phishing URLs through classifiers.

6. Future Work

In subsequent studies, we will evaluate the effect of using other classifiers, along with other low-cost lexical features to further improve the classification accuracy. A particular challenge is a need of algorithm that continually adapt new examples and features of phishing URLs. As our future work, we plan to develop a framework using this approach and deploy it for a large-scale real-world test.

References

[1] Ollmann, G., “The Phishing Guide, Understanding and Preventing Phishing Attacks”, Online Available: https://www.nccgroup.trust/uk/our-research/the-phishing-guide-understanding-preventing-phishing-attacks/.

[2] Watson, D., Holz, T., and Mueller, S. “Know your enemy: Phishing, behind the scenes of Phishing attacks”, The Honeynet Project & Research Alliance, 2005, Online Available: https://www.honeynet.org/papers/phishing.

[3] S. Garera, N. Provos, M. Chew, A.D. Rubin, “A framework for detection and measurement of phishing attacks”, In: Proc. 5th ACM Workshop on Recurring Malcode, WORM‟07, ACM, New York, NY, USA, 2007, pp. 1-8.

[4] J. Ma, L.K. Saul, S. Savage, G.M. Voelker, “Beyond blacklists: Learning to detect malicious web sites from suspicious URLs”, In: Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, Paris, France, 2009, pp. 1245-1254.

[5] C. Whittaker, B. Ryner, M. Nazif, “Large-scale automatic classification of phishing pages”, In: Proc. 17th Annual Network and Distributed System Security Symposium, NDSS‟10, San Diego, CA, USA, 2010.

[6] Y. Zhang, J. Hong, L. Cranor, “CANTINA: a content based approach to detecting phishing web sites”, In Proc. 16th Int. Conf. World Wide Web, WWW‟07, Banff, Alberta, Canada, 2007, pp. 639-648.

[7] Google Safe Browsing API - Google Code, http://code.google.com/apis/safebrowsing/

[8] N. Chou, R. Ledesma, Y. Teraguchi, D. Boneh, J. Mitchell, “Client-side defense against web-based identity theft”, In: Proc. 11th Network and Distributed System Security Symposium, NDSS‟04, San Diego, CA, USA, 2004.

[9] R. B. Basnet, A.H. Sung, Q. Liu, Rule-based phishing attack detection, In: Proc. Int. Conf. Security and Management, SAM‟11, Las Vegas, NV, USA, 2011.

[10] SpoofStick Home, Online available at: http://www.spoofstick.com.

[11] McAfee Site Advisor Software – Website Safety Ratings and Secure Search, Online available at: http://www.siteadvisor.com

[12] Netcraft Anti-Phishing Toolbar, Online available at: http://toolbar.netcraft.com.

[13] AVG Security Toolbar, Online available at: http://www.avg.com/product-avg-toolbar-tlbrc#tba2

[14] The Web Information Company. Online available at: www.alexa.com

[15] DMOZ Open Directory Project. Online available at: http://www.dmoz.org.

[16] PhishTank. Online available at: https://www.phishtank.com/

[17] Xiang, G., Hong, J., Rose, C. P., and Cranor, L. 2011. CANTINA+: A feature-rich machine learning framework for detecting phishing Web sites. ACM Trans. Inf. Syst. Secur. 14, 2, Article 21 (September 2011), 28 pages.

[18] Xiaoqing GU, Hongyuan WANG, Tongguang NI 2013.An Efficient Approach to Detecting Phishing Web. Journal of Computational Information Systems 9: 14 (2013) 5553–5560.