Detection of Phishing URLs Using Machine Learning Techniques

Joby James
SCT College of Engineering,
Trivandrum.
jamesjoby@gmail.com

Sandhya L.
SCT College of Engineering,
Trivandrum.
lsandyaajith@gmail.com

Ciza Thomas
College of Engineering,
Trivandrum.
cizathomas@gmail.com

Abstract*— Phishing costs Internet users billions of dollars per year. It refers to luring techniques used by identity thieves to fish for personal information in a pond of unsuspecting internet users. Phishers use spoofed e-mail, phishing software to steal personal information and financial account details such as usernames and passwords. This paper deals with methods for detecting phishing web sites by analyzing various features of benign and phishing URLs by Machine learning techniques. We discuss the methods used for detection of phishing websites based on lexical features, host properties and page importance properties. We consider various data mining algorithms for evaluation of the features in order to get a better understanding of the structure of URLs that spread phishing. The fine-tuned parameters are useful in selecting the apt machine learning algorithm for separating the phishing sites from benign sites.

Keywords—Phishing; benign; URL; Page rank; WHOIS

I. INTRODUCTION

Phishing is a criminal mechanism employing both social engineering and technical tricks to steal consumers' personal identity data and financial account credentials. Social engineering schemes use spoofed e-mails, purporting to be from legitimate businesses and agencies, designed to lead consumers to counterfeit websites that trick recipients into divulging financial data such as usernames and passwords. Technical subterfuge schemes install malicious software onto computers, to steal credentials directly, often using systems to intercept consumers' online account user names and passwords [1].

Figure. 1 represents the webpage of the popular website www.facebook.com. Figure. 2 represents a webpage similar to that of facebook, but is the webpage of a site which spreads phishing activities. A user may misunderstand the second site as genuine facebook site and provide his personal identity details. The Phisher can thus steal that information and he may use it for vicious purposes.

A. The Technique of Phishing

The criminals, who want to obtain sensitive data, first create unauthorized replicas of a real website and e-mail, usually from a financial institution or another company that deals with financial information. The e-mail will be created using logos and slogans of a legitimate company. The nature and format of Hypertext Mark-up Language makes it very easy to copy images or even an entire website. While this ease

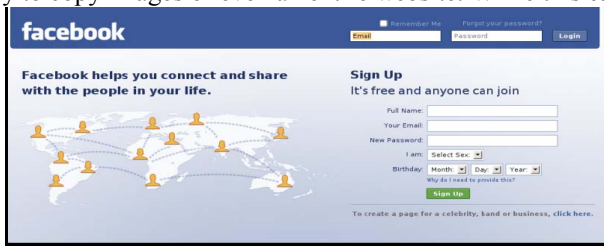


Figure 1.Original facebook webpage

Figure2. Phishing webpage [4]

of website creation is one of the reasons that the Internet has grown so rapidly as a communication medium, it also permits the abuse of trademarks, trade names, and other corporate identifiers upon which consumers have come to rely as mechanisms for authentication. Phisher then send the "spoofed" e-mails to as many people as possible in an attempt to lure them in to the scheme. When these e-mails are opened or when a link in the mail is clicked, the consumers are redirected to a spoofed website, appearing to be from the legitimate entity.

B. Statistics of Phihing attacks

Phishing continues to be one of the rapidly growing classes of identity theft scams on the internet that is causing both short term and long term economic damage. There have been nearly 33,000 phishing attacks globally each month in the year 2012, totalling a loss of $687 million [1].

An example of phishing occurred in June 2004. The Royal Bank of Canada notified customers that fraudulent e-mails purporting to originate from the Royal Bank were being sent out asking customers to verify account numbers and personal identification numbers (PINs) through a link included in the e-

mail. The fraudulent e-mail stated that if the receiver did not click on the link and key in his client card number and pass code, access to his account would be blocked. These e-mails were sent within a week of a computer malfunction that prevented customer accounts from being updated [2].

The United States continued to be the top country hosting phishing sites during the third quarter of 2012. This is mainly due to the fact that a large percentage of the world's Web sites and domain names are hosted in the United States. Financial Services remains to be the most targeted industry sector by Phishers [1].

II. RELATED WORK

Many researchers have analyzed the statistics of suspicious URLs in some way. Our approach borrows important ideas from previous studies. We review the previous work in the phishing site detection using URL features that motivated our own approach.

Ma et al. [3, 4] compared several batch-based learning algorithms for classifying phishing URLs and showed that the combination of host-based and lexical features results in the highest classification accuracy. Also they compared the performance of batch-based algorithms to online algorithms when using full features and found that online algorithms, especially Confidence-Weighted (CW), outperform batch-based algorithms.

The work by Garera et al. [5] uses logistic regression over hand-selected features to classify phishing URLs. The features include the presence of red flag keywords in the URL, features based on Google's Page Rank, and Google's Web page quality guidelines. It is difficult to make a direct comparison with our approach without access to the same URLs and features.

McGrath and Gupta [6] did not construct a classifier, but performs a comparative analysis of phishing and non phishing URLs with respect to datasets. They compared non phishing URLs drawn from the DMOZ Open Directory Project [7] to phishing URLs from PhishTank [8]. The features they analyze include IP addresses, WHOIS thin records containing date and registrar-provided information, geographic information, and lexical features of the URL such as length, character distribution, and presence of predefined brand names [6].

III. PROBLEM OVERVIEW

URLs sometimes known as "Web links" are the primary means by which users locate information in the Internet. Our aim is to derive classification models that detect phishing web sites by analysis of the lexical and host-based features of URLs. We analyze different classifying algorithms in Waikato Environment for Knowledge Analysis (WEKA) workbench and MATLAB.

IV. DESIGN FLOW

The work consists of host based, page based and lexical feature extraction of collected URLs and analysis. The first step is the collection of phishing and benign URLs. The host based, popularity based and lexical based feature extractions are applied to form a database of feature values. The database is knowledge mined using different machine learning methods. After evaluating the classifiers, a particular classifier is selected and is implemented in MATLAB. The design flow is shown in Figure 3.

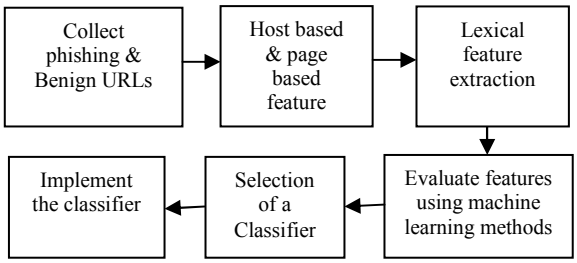


Figure3. Design flow graph

A. Collection of URLs

We collected URLs of benign websites from www.alexa.com [9] www.dmoz.org [7] and personal web browser history. The phishing URLs were collected from www.phishtak.com [8]. The data set consists of 17000 phishing URLs and 20000 benign URLs.

We obtained PageRank [10] of 240 benign websites and 240 phishing websites by checking PageRank individually at PR Checker [11].

We collected WHOIS [12] information of 240 benign websites and 240 phishing websites.

B. Host based analysis

Host-based features explain "where" phishing sites are hosted, "who" they are managed by, and "how" they are administered. We use these features because phishing Web sites may be hosted in less reputable hosting centers, on machines that are not usual Web hosts, or through not so reputable registrars.

The block schematic for the host based analysis is shown in Figure 4.

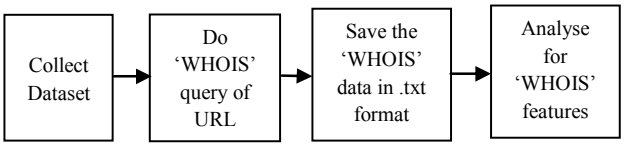


Figure4. Block diagram for host based analysis

The following are the properties of the hosts that are identified.

1) WHOIS properties: WHOIS [12] properties gives details about the date of registration, update and expiry, who is the registrar and the registrant. If phishing sites are taken down frequently, the registration dates will be newer than for legitimate sites.. A large number of phishing websites contain IP address in their hostname [5]. So getting the details of such hostnames will be helpful in efforts to point to phishing sites, which can be obtained from the Whois properties.

2) Geographic properties: Geographic properties give details about the continent/country/city to which the IP address belongs.

3) Blacklist membership: A large percentage of phishing URLs were present in blacklists. In the Web browsing context, blacklists are precompiled lists or databases that contain IP addresses, domain names or URLs of malicious sites the web users should avoid. On the other hand white lists contain sites that are known to be safe.

a) DNS-Based Blacklists: Users submit a query representing the IP address or the domain name in question to the blacklist provider's special DNS server, and the response is an IP address that represents whether the query was present in the blacklist. SORBS, [13] URIBL [14], SURBL [15] and Spamhaus [16] are examples of major DNS blacklist providers.

b) Browser Toolbars: Browser toolbars provide a client-side defense for users. Before a user visits a site, the toolbar intercepts the URL from the address bar and cross references a URL blacklist, which is often stored locally on the user's machine or on a server that the browser can query. If the URL is present on the blacklist, then the browser redirects the user to a special warning screen that provides information about the threat. McAfee SiteAdvisor [17], Google Toolbar [18] and WOT Web of Trust [19] are prominent examples of blacklist-backed browser toolbars.

c) Network Appliances*:* Dedicated network hardware is another popular option for deploying blacklists. These appliances serve as proxies between user machines within an enterprise network and the rest of the Internet. As users within an organization visit sites, the appliance intercepts outgoing connections and cross references URLs or IP addresses against a precompiled blacklist. IronPort acquired by Cisco in 2007 and WebSense are examples of companies that produce blacklist backed network appliances.

Limitations of blacklists: The primary advantage of blacklists is that querying is a low overhead operation: the lists of malicious sites are precompiled, so the only computational cost of deployed blacklists is the lookup overhead. However, the need to construct these lists in advance give rise to their disadvantage that blacklists become stale. Network administrators block existing malicious sites, and enforcement efforts take down criminal enterprises behind those sites. There is a constant pressure on criminals to construct new sites and to find new hosting infrastructure. As a result, new malicious URLs are introduced and blacklist providers must update their lists yet again. However, in this process, criminals are always ahead because Web site construction is inexpensive. Moreover, free services for blogs e.g., Blogger [20] and personal hosting e.g., Google Sites [21], Microsoft Live Spaces [22] provide another inexpensive source of disposable sites.

4) Page/Popularity Based Property: Popularity features indicate how popular a web page is among Internet users. Various popularity features are as follows:

a) PageRank [10]: It is one of the methods Google uses to determine a page's relevance or importance. The maximum PR of all pages on the web changes every month when Google does its re-indexing.

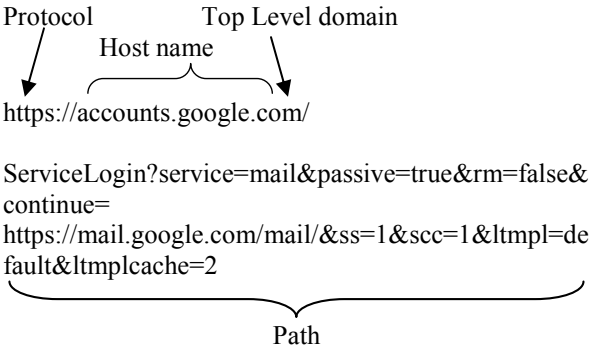
The PageRanks form a probability distribution over web pages, so the sum of all web pages' PageRanks will be equal to unity.

b) Traffic Rank details: Traffic Ranks of websites indicate a site's popularity. Alexa.com ranks various websites according to the Internet traffic based on previous 3 months. Traffic close to 1 is accurate. Ranks more than 100,000 are not so accurate since chance for error is high.

5) Lexical feature analysis: Lexical features are the textual properties of the URL itself, not the content of the page it points to. URLs are human-readable text strings that are parsed in a standard way by client programs. Through a multistep resolution process, browsers translate each URL into instructions that locate the server hosting the site and specify where the site or resource is placed on that host. To facilitate this machine translation process, URLs have the following standard syntax.

<protocol>://<hostname><path>

An example of URL resolution is shown below:



The <protocol> portion of the URL indicates which network protocol should be used to fetch the requested resource. The most common protocols in use are Hypertext Transport Protocol or HTTP (http), HTTP with Transport Layer Security (https), and File Transfer Protocol (ftp).

The <hostname> is the identifier for the Web server on the Internet. Sometimes it is a machine-readable Internet Protocol (IP) address, but more often especially from the user's perspective it is a human-readable domain name.

The <path> of a URL is analogous to the path name of a file on a local computer. The path tokens delimited by various punctuation marks such as slashes, dots, and dashes, show how the site is organized. Criminals sometimes obscure path tokens to avoid scrutiny, or they may deliberately construct tokens to mimic the appearance of a legitimate site.

The methodology used in our work to extract the lexical features from the URL list is as follows: The URLs of legitimate websites, collected from alexa.com and dmoz.org, are written into a notepad and the file is saved in the computer. Then the MATLAB program is executed. It will ask for input

file. Feed the benign URL list to the MATLAB program. The program processes the list and the feature list is obtained. The decision vector '0' is added. The list is saved in excel and csv format at location in the computer as specified in the program. The same procedure is done for phishing URL list. The decision vector '1' is added. The feature set comprises of host length, path length, number of slashes, number of path tokens etc. The Figure 5 shows the flowchart of feature extraction.

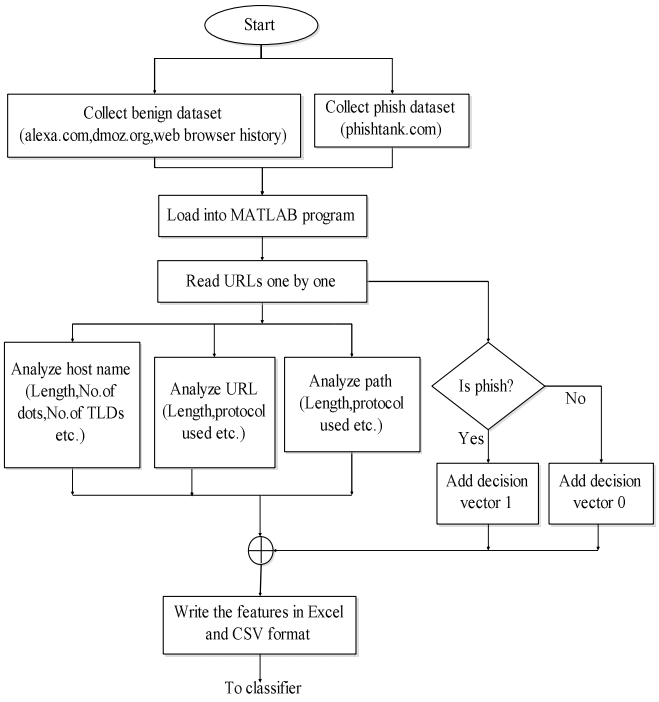


Figure5. Flow chart for feature extraction

C. Machine learning algorithms

The evaluation of the various classifying algorithm is done by using the workbench for data mining, Waikato Environment for Knowledge Analysis (WEKA) [22] and using MATLAB.

Four types of input data files i.e., Attribute Relation File Format (.arff), Comma Separated Values (.csv), C4.5, binary are allowed in WEKA. In our experiment .csv file format was used. The input file to the WEKA was obtained by a MATLAB program by appending 'YES' in place of decision vector '1' (phish) and 'NO' in place of decision vector '0' (benign) of the dataset generated by MATLAB from input URL list. The evaluation was done using percentage split 60%.

The input to the classifiers in MATLAB is four .txt files test.xls, testresult.xls, train.xls, trainresult.xls.

The four machine learning algorithms considered for processing the feature set are:

1) Naive Bayes: Naive Bayes is a simple probabilistic classifier based on applying Bayes' theorem (or Bayes's rule) with strong independence (naive) assumptions. Parameter estimation for Naïve Bayes models uses the maximum likelihood estimation. It takes only one pass over the training set and is computationally very fast.

2) J48 decision tree: A decision tree is a predictive machine-learning model that decides the target value (dependent variable) of a new sample based on various attribute values of the available data.

3) K-NN**:** It is based on closest training examples in the feature space. An object is classified by a majority vote of its neighbors.

4) 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 support vectors.

The program flow for the classifier performance is shown in Figure 6.

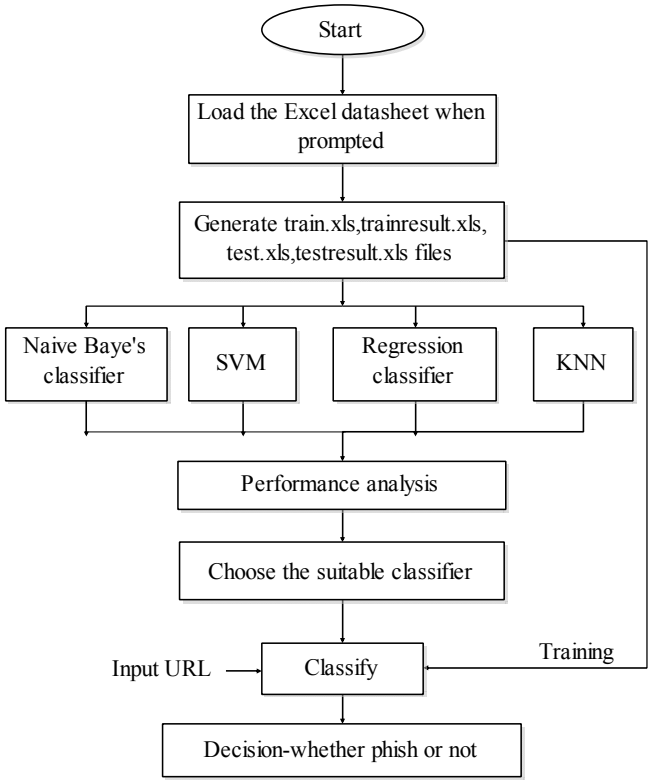


Figure 6. Program flow

V. RESULTS

The main findings of our preliminary work include:

- Phishing URLs and domains exhibit characteristics that are different from other URLs and domains.
- Phishing URLs and domain names have very different lengths compared to other URLs and domain names in the Internet.
- Many of the phishing URLs contained the name of the brand they targeted.

PageRank of benign and phishing websites were collected using Google PageRank Checker [11] and are presented in

Figure 7 and Figure 8. PageRank obtained for phishing sites are: Not Available, Non-Existing and 0.

The N/A pagerank (grey pagerank bar) might be due to one of the following reasons [11]:

- Web page is new, and it is not indexed by Google yet.
- Web page is indexed by Google, but it is not ranked yet.
- Web page was indexed by Google long ago, but it is recognized as a supplemental page.
- Web page or the whole website is banned by Google.

Supplemental Result is a URL residing in Google's secondary database containing pages of less importance, as measured primarily by Google's PageRank algorithm. Google used to place a "Supplemental Result" label at the bottom of a search result to indicate that it is in the supplemental index; however in July 2007 they discontinued this practice and it is no longer possible to tell whether a result is in the supplemental index or the main one[11].

PageRank for benign sites ranges from 0 to 9. We used 240 benign URLs and 240 malicious URL sites for the plot. It is inferred from the graph that the PageRank is pretty high for benign URLs compared to phishing websites. One exception is about newly registered websites. If we do the PageRank check we will get 'N/A' (Not Available) message from the PageRank Checker [11].

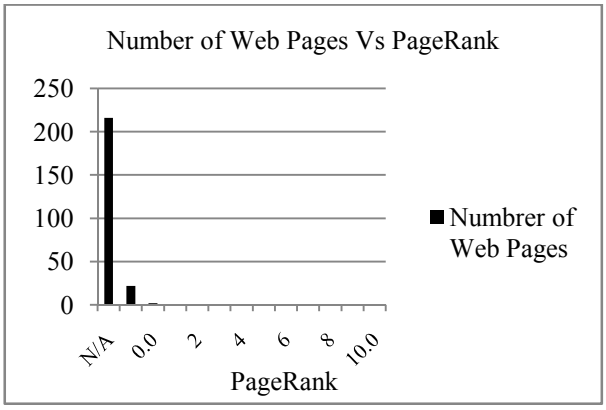


Figure 7.Number of phishing sites vs. PageRank

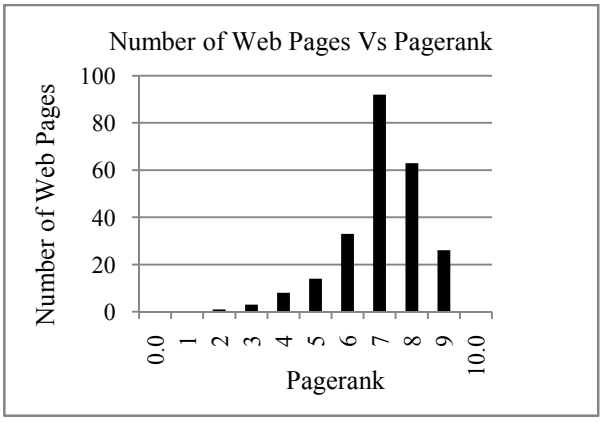


Figure8. Number of benign sites Vs. PageRank

We analyzed the prepared URL feature dataset using Naïve Bayes, J48 Decision Tree, k-NN, and SVM classifying algorithms in WEKA. The percentage split is set to 60% i.e., 40 percentage of the dataset is taken as training data and 60 percentage as test data. The performance is then evaluated based on Confusion matrix, Detection Accuracy, True Positive Rate and False Positive Rate. The result is tabulated in TABLE 1.

The analysis of the dataset is done using MATLAB also by setting the above said testing conditions and is tabulated in TABLE 2.

When we check the Success Rate in analysis by WEKA and MATLAB, it is seen that there are slight differences in values. The J48 Decision Tree has the highest Success Rate compared to other selected classifying algorithms in WEKA. By using only the lexical features, we were able to achieve a Detection Accuracy/Success rate of 93.2% for test split of 60%. When 90% of dataset is used, we got 93.78% Detection Accuracy. In MATLAB, using Regression Tree we got 91.08% detection accuracy when using 60% of dataset for testing and 85.63% detection accuracy when using 90% of data for testing.

TABLE 1. Classifier Performance - WEKA

Test options	Classifier	Confusion Matrix	Success Rate (%)	Error Rate (%)
Percentage split-60	**Naïve Bayes**	4438 3578 260 3945	68.60	31.40
	J48	7612 404 428 3777	93.20	6.80
	IBK	7042 974 455 3750	88.30	11.70
	SVM	7511 505 1459 2746	83.93	16.07
Percentage split-90	**Naïve Bayes**	1180 792 61 1022	72.08	27.92
	J48	1883 89 101 982	93.78	6.22
	IBK	1756 216 97 986	89.75	10.25
	SVM	1846 126 355 728	84.26	15.74

Figure 9 shows a comparison of TP Rate, FP Rate and Detection Accuracy of SVM, Naïve Bayes, Regression Tree and k-NN classifiers.

Figure 10 shows detection accuracy parameters of the classifiers with 60% and 90% test split.

TABLE2. Classifier performance – MATLAB

Test Options	Classifier	Confusion Matrix	Success Rate (%)	Error Rate (%)
Percentage split-60	Naïve Bayes	7281 303 3633 4042	74.20	25.80
	Regression Tree	10856 470 1166 5839	91.08	8.92
	KNN	11299 3025 723 3284	79.55	20.45
	SVM	9871 806 1082 3531	87.65	12.35
Percentage split-90	Naïve Bayes	13648 1018 2764 5500	83.50	16.50
	Regression Tree	15082 999 2951 8465	85.63	14.37
	KNN	16451 5080 1582 4384	75.77	24.23
	SVM	16416 5848 5 661	74.48	25.52

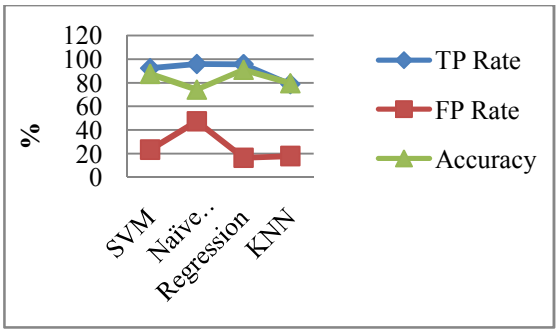


Figure9. Detection parameters

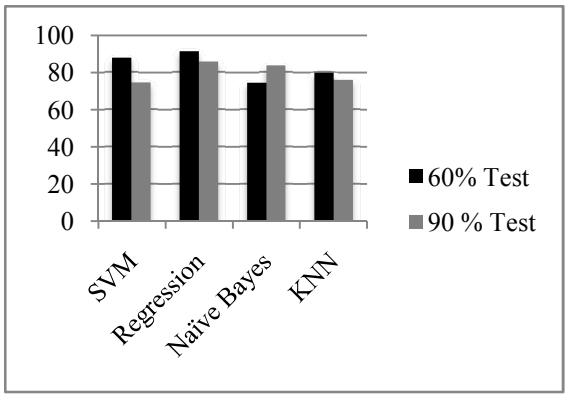


Figure10. Detection accuracy comparison

Apart from that another experiment done was to test whether an input URL is phish or not. The URL was loaded into the MATLAB program and extracted URL features. A feature set is created in .xls format. This is used as test data and the classifier makes the decision whether 'Benign' or 'Phish' with its specified accuracy.

VI. CONCLUSION

Several features are compared using various data mining algorithms. The results points to the efficiency that can be achieved using the lexical features. To protect end users from visiting these sites, we can try to identify phishing URLs by analyzing their lexical and host-based features. A particular challenge in this domain is that criminals are constantly making new strategies to counter our defense measures. To succeed in this contest, we need algorithms that continually adapt to new examples and features of phishing URLs.

Online learning algorithms provide better learning methods compared to batch-based learning mechanisms. Going forward we are interested in various aspects of online learning and collecting data to understand the new trends in phishing activities such as fast changing DNS servers.

REFERENCES

[1] Phishing Trends Report for Q3 2012, Anti Phishing Working Group. http://antiphishing.org.

[2] Report on Phishing, Binational Working Group on Cross-Border Mass Marketing Fraud, October 2006

[3] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker," Beyond Blacklists: Learning to Detect Phishing Web Sites from Suspicious URLs", Proc.of SIGKDD '09.

[4] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Learning to Detect Phishing URLs", ACM Transactions on Intelligent Systems and Technology, Vol. 2, No. 3, Article 30, Publication date: April 2011.

[5] Garera S., Provos N., Chew M., Rubin A. D., "A Framework for Detection and measurement of phishing attacks", In Proceedings of the ACM Workshop on Rapid Malcode (WORM), Alexandria, VA.

[6] D. K. McGrath, M. Gupta, "Behind Phishing: An Examination of Phisher Modi Operandi", In Proceedings of the USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET).

[7] DMOZ Open Directory Project. http://www.dmoz.org.

[8] PhishTank. http://www.phishtank.com.

[9] The Web Information Company, www.alexa.com.

[10] I. Rogers, "Google Page Rank – Whitepaper",

[11] http://www.sirgroane.net/google-page-rank/PR Checker, http://www.prchecker.info/check_page_rank.php

[12] WHOIS look up, www.whois.net, www.whois.com

[13] SORBS, Spam and Open-Relay Blocking System, www.sorbs.net

[14] URIBL, URI blacklist, www.uribl.com

[15] SURBL, www.surbl.org

[16] SPAMHAUS, www.spamhaus.org

[17] McAfee site advisor, www.siteadvisor.com

[18] Google toolbar, www.toolbar.google.com

[19] WOT Web of Trust. http://www.mywot.com.

[20] BLOGGER, www.blogger.com

[21] Google sites, www.sites.google.com

[22] Microsoft sites,www.microsoft.com/en/in/sitemap.aspx

[23] Data Mining with Open Source Machine Learning Software, www.cs.waikato.ac.nz/ml/weka/

[24] J. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, Elsevier Inc., 2006.