Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 3, March 2016, pg.621 – 647

Supremacy of Rotation Forest with LMT Base Classifier in Prediction of Phishing Websites

Manish Kumar*

Department of Computer Science, Banaras Hindu University, Varanasi-221005, India

E-mail: *manish.bhu14@gmail.com*

Abstract: Phishing is the skill of publishing a website of a credible organisation with the aim to acquire user's secretive data such as bank account detail, usernames, passwords and many more personal information. Now-a-days the growth of the phishing websites seems to be astonishing. Although the users of web are observant of these kinds of phishing attacks, however lot of users become victim of the attack. Therefore, predicting and stopping this attack is vital and compulsory to protect online trading. Most users will feel safe against phishing attacks if they use tool that can predict the phishing websites. In this study, the experiments were conducted for prediction of phishing websites on the dataset obtained from UCI Machine Learning Repository separately using twelve machine learning algorithms with ten-fold cross validation. We obtained classifier Rotation Forest with LMT as base classifier is outperforming and obtained the AUC, accuracy and MCC value up to **0.996, 0.974** and **0.946** respectively.

Keywords: AUC, Machine Learning, Phishing websites, Rotation Forest.

© 2016, IJCSMC All Rights Reserved

1. Introduction

Now a days, there are numerous people uses internet amenities because it saves money and time in addition to efforts. Unfortunately, the convenience of online services has been threatened by large-scale phishing attacks raised alongside internet users. Phishing is the identity theft in which attackers trying to access personal information and financial credentials of online consumers. The growth of the phishing websites appears to be surprising. Even though the web users are attentive of these kinds of phishing attacks, lot of users become victim. Web users believe that they are interacting with a trusted entity but that is phishing and looks similar to genuine. Communications from widespread web sites, auction sites, online payment processors and many more are the commonly used source to trap the unsuspicious public. Only specialists can recognise these kinds of phishing websites instantaneously but all of the web users are not specialist and hence become victim by providing their personal details. Phishing is constantly progressing since it is easy to duplicate the whole website using the HTML source code and by making minor modifications in the source code, it is possible to direct the victim to believe in the existing phishing website. Phishers practice lot of performances to trap the unsuspected web user by sending common greetings to the customers to check their account instantaneously and threat messages indicating to update their account immediately otherwise their account will be closed. Thus an effective mechanism is requisite to recognise the phishing websites from the genuine websites in order to save credential information.

1.1 Related Work

Even though numerous solutions were presented to solve phishing problem, but most of these solutions are not proficient to make a decision faultlessly. In this section, we are going to review the approaches and techniques applied in finding solutions

Aburrous M et. al. [1] used contrasting associative classification algorithms in their experiment. They collected 27 different features from various websites and ranged them among three fuzzy set values ‘‘legitimate, genuine and doubtful’’. To calculate the selected features, the authors conducted experiments using MCAR, CBA, C4.5, PRISM, PART and JRip. The results showed an important relationship between ‘‘domain identity’’ and ‘‘URL’’ features. **Aburrous M** et.al afterward [2], used the 27 features to construct a model to predict phishing websites using fuzzy techniques. **Pan Y and Ding X** [3] suggested a way to identify phishing websites by bagging abnormal behaviours demonstrated by these websites. They used two components as phishing detectors. First one is the **identity extractor** which is the organization’s full name abbreviation along with a unique string presented in the domain name and second is **page classifier** that is some Web properties, i.e. structural features that are significant to the site uniqueness and cannot be fictional. So, six structural features: (Abnormal URL, abnormal DNS record, abnormal anchors, Server form handler, abnormal cookies and abnormal certificate in SSL) are chosen and support Vector Machine classifier [4] was used to conclude whether the website is phisher or not. Experiments were conducted on a dataset comprise of 279 phishing websites, and 100 genuine websites conclude that the ‘‘identity extractor’’ performs better in dealing with phishing pages because the genuine websites are independent from each other, whereas some of the phishing sites are correlated. Furthermore, “page classifier” performance mostly depends on the result dig up from ‘‘identity extractor’’ and the accuracy in this method obtained was 84 %. **Zhang Y, Hong J and Cranor L** [5] utilized ‘‘CANTINA’’ stands for ‘‘Carnegie Mellon Anti-phishing and Network Analysis Tool’’, which is a content-based procedure to identify phishing websites using the term frequency–inverse document frequency (TF-IDF) measures [6].

CANTINA work can be described as follows:

1. Calculate the TF-IDF value for a given web page.

2. Consider the five highest TF-IDF terms and add them to the URL to catch the lexical signature.

3. The lexical signature is provide to the search engine.

If the N tops search results having the current web page, it is considered a genuine web page. If not, it is a phishing web page. In experiments N was set to 30 **Sanglerdsinlapachai N, and Rungsawang A** [7] also utilizes CANTINA with an additional attributes They have used 100 phisher websites and 100 genuine ones, and eight features for identifying phishing websites (domain age, known image, suspicious URL, suspicious link, IP address, dots in URL, forms and TF-IDF). They performed three types of experiments for their dataset where the first one evaluated a reduced CANTINA feature set ‘‘dots in URL, IP address, suspicious URL and suspicious link’’. The second experiment elaborate testing whether the new features ‘‘domain top page similarity’’ are important enough to play a significant role in detecting website type. The third and last experiment evaluated the results after adding the new proposed feature to the reduced CANTINA features applied in the first experiment. All compared classification algorithms revealed that the new feature played a key role in detecting the type of the website. The best accurate algorithm was neural network with an error rate equals to 7.5 %, after that SVM and random forest with an error rate equals to 8.5 %, Adaboost with 9.0 % and J48 with 10.5 %, whereas Naıve Bayes gave the worst result with a 22.5 % error rate. **Sadeh N, Tomasic A and Fette I** [8], compared numerous commonly used machine-learning methods including SVM, rule-based techniques, decision trees and Bayesian techniques and the Random Forest algorithm was implemented in ‘‘PILFER’’ stands for Phishing Identification by Learning on Features of email Received, which essentially aim to detect phishing emails. In the experiments a dataset consisting of

860 phishing emails and 6,950 genuine emails was used. The proposed method appropriately detected 96 % of the phishing emails with a false positive rate of 0.1 %. They used 10 features for distinguishing phishing email's which are: IP-based URL's, age of domain, non-matching URL's, having a link within the e-mail, HTML emails, number of links within the e-mail, number of domains appears within the e-mail, number of dot's within the links, containing JavaScript and spam filter output''. The results discovered that PILFER has a false positive rate of 0.0022 % if it is being installed without a spam filter. If PILFER is joined with Spam Assassin, the false positive rate decreased to 0.0013 %, and the detection accuracy rises to 99.5 %. **Wenyin L, Huang G, Xiaoyue L, Min Z, Deng X** [9] discovered type of websites based on visual similarity by comparing phishing websites with the genuine ones. This method initially decomposed the web page into prominent block regions depending on ''visual cues.'' The visual similarity between phishing web page and genuine one is then evaluated using following three metrics: block level similarity, layout similarity and overall style similarity based on the matching of the prominent block regions. A web page is considered phisher if any metric has a value higher than a predefined threshold. They collected 8 phishing web pages and 320 official bank pages, and conducted experiment which shows a 100 % true positive and 1.25 % false positive. Even though the results were extraordinary, but this work suffers from weaknesses of low size dataset and potential instability attributed to the high flexibility of the layout within the HTML documents. **Dhamija R, Tygar JD** [10], proposed a new method, called ''dynamic security skins''. This methodology used a shared secret image that permits a remote server to verify its identity to the user in a way that supports easy authentication by users based on comparing the user expected image with an image generated by the server. They implemented their technique by developing an extension to ''Mozilla Firefox browser''. The main shortcoming of this technique is that the users bear the problem of deciding whether the website is phishing or

not. This approach also recommends an essential change in the Web infrastructure for both servers and clients, so it can succeed only if the whole industry's support it. In addition, this technique does not provide security if the users logged-in from a public computers. **Miyamoto D, Hazeyama H, and Kadobayashi Y**[11] presented a survey aimed to evaluate the performance of machine-learning-based detection methods including ''AdaBoost, Bagging, SVM, Classification and Regression Trees, Logistic Regression, Random Forests, NN, Naive Bayes and Bayesian Additive Regression Trees'' showed that 7 out of 9 of machine-learning-based detection methods outperformed CANTINA in predicting phishing. In the experiments a dataset consisting of 1,500 phishing websites and 1,500 legitimate websites used.

So, the growth of the phishing websites appears to be surprising. Even though the web users are attentive of these kinds of phishing attacks, lot of users become victim. Therefore, predicting and stopping this attack is vital to protect online trading. Most users will feel safe against phishing attacks if they use tool that can predict the phishing websites. In this study, the experiments were conducted for prediction of phishing websites using machine learning algorithms and tried to search the best one for the considered domain.

2. Material and Methods

2.1 Dataset

For study, I downloaded the dataset from the UCI Machine Learning Repository named "phishing website" dataset uploaded in 2015, having number of instances 11055 and number of features 31 [12, 13, 14]. These features can be categorized into four category: Address Bar based Features, Abnormal Based Features, HTML and JavaScript based Features, Domain based Features**.** Moreover, description and subcategory of each feature is explained in the next section.

2.2 Features of Phishing Websites

2.2.1 Address Bar based Features

2.2.1.1 Using the IP Address

If an IP address is used as an alternative of the domain name in the URL, such as "http://125.98.3.123/phisher.html", users can be sure that somebody is trying to snip their personal information. Sometimes, the IP address is even converted into hexadecimal code as shown in the following link "http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html".

$$\underline{Rule}\text{: IF}\begin{cases}\text{If The Domain Part has an IP Address } \rightarrow \text{ Phishing}\\ \text{Otherwise} \rightarrow \text{ Legitimate}\end{cases}$$

2.2.1.2 Long URL to Hide the Suspicious Part

Phishers can use long URL to hide the suspicious part in the address bar. For example:

http://federmacedoadv.com.br/3f/aze/ab51e2e319e51502f416dbe46b773a5e/?cmd=_home&amp;dispatch=11004d58f5b74f8dc1e7c2e8dd4105e811004d58f5b74f8dc1e7c2e8dd4105e8@phishing.website.html. If the length of the URL is greater than or equal 54 characters then the URL classified as phishing.

$$\textit{Rule: IF}\begin{cases} URL\ length < 54 \ \rightarrow \ feature = \text{Legitimate}\\ else\ if\ URL\ length \geq 54\ and\ \leq 75 \ \rightarrow \ feature = Suspicious\\ otherwise \rightarrow \ feature = \text{Phishing}\end{cases}$$

2.2.1.3 Using URL Shortening Services "TinyURL"

URL shortening is a method on the "World Wide Web" in which a URL may be made considerably smaller in length and still lead to the requisite webpage. This is accomplished by means of an "HTTP Redirect" on a domain name that is short, which links to the webpage that has a long URL. For example, the URL "http://portal.hud.ac.uk/" can be shortened to "bit.ly/19DXSk4".

Rule: IF$\begin{cases} \text{TinyURL} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.1.4 URL's having "@" Symbol

Using "@" symbol in the URL leads the browser to ignore everything preceding the "@" symbol and the real address often follows the "@" symbol.

Rule: IF $\begin{cases} \text{Url Having @ Symbol} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.1.5 Redirecting using "//"

The existence of "//" within the URL path means that the user will be redirected to another website. An example of such URL's is: "http://www.legitimate.com//http://www.phishing.com". We examine the location where the "//" appears. We find that if the URL starts with "HTTP", that means the "//" should appear in the sixth position. However, if the URL employs "HTTPS" then the "//" should appear in seventh position.

Rule: IF $\begin{cases} \text{ThePosition of the Last Occurrence of "//" in the URL} > 7 \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.1.6 Adding Prefix or Suffix Separated by (-) to the Domain

The dash symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name so that users feel that they are dealing with a legitimate webpage. For example http://www.Confirme-paypal.com/.

Rule: IF $\begin{cases} \text{Domain Name Part Includes } (-)\text{Symbol} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.1.7 Sub Domain and Multi Sub Domains

Let us consider we have the following link: http://www.hud.ac.uk/students/. A domain name might include the country-code top-level domains (ccTLD), which in our example is "uk". The "ac" part is shorthand for "academic", the combined "ac.uk" is called a second-level domain (SLD) and "hud" is the real name of the domain. To produce a rule for extracting this feature, we firstly have to omit the (www.) from the URL which is in fact a sub domain in itself. Then, we have to remove the (ccTLD) if it exists. Finally, we count the remaining dots. If the number of dots is greater than one, then the URL is classified as "Suspicious" since it has one sub domain. However, if the dots are greater than two, it is classified as "Phishing" since it will have multiple sub domains. Otherwise, if the URL has no sub domains, we will assign "Legitimate" to the feature.

$$\text{Rule: IF}\begin{cases}\text{Dots In Domain Part} = 1 \rightarrow \text{Legitimate}\\ \text{Dots In Domain Part} = 2 \rightarrow \text{Suspicious}\\ \text{Otherwise} \rightarrow \text{Phishing}\end{cases}$$

2.2.1.8 .HTTPS

(Hyper Text Transfer Protocol with Secure Sockets Layer)

The existence of HTTPS is very important in giving the impression of website legitimacy, but this is clearly not enough. The authors in (Mohammad, Thabtah and McCluskey 2012) (Mohammad, Thabtah and McCluskey 2013) suggest checking the certificate assigned with HTTPS including the extent of the trust certificate issuer, and the certificate age. Certificate Authorities that are consistently listed among the top trustworthy names include: "GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster and VeriSign". Furthermore, by testing out our datasets, we find that the minimum age of a reputable certificate is two years.

$$\text{Rule: IF}\begin{cases}\text{Use https and Issuer Is Trusted and Age of Certificate} \geq 1\text{ Years} \rightarrow \text{Legitimate}\\ \text{Using https and Issuer Is Not Trusted} \rightarrow \text{Suspicious}\\ \text{Otherwise} \rightarrow \text{Phishing}\end{cases}$$

2.2.1.9 Domain Registration Length

Based on the fact that a phishing website lives for a short period of time, we believe that trustworthy domains are regularly paid for several years in advance. In the dataset, we find that the longest fraudulent domains have been used for one year only.

Rule: IF$\begin{cases} \text{Domains Expires on} \leq \text{ 1 years } \rightarrow \text{ Phishing} \\ \qquad\qquad \text{Otherwise} \rightarrow \text{ Legitimate} \end{cases}$

2.2.1.10 Favicon

A favicon is a graphic image (icon) associated with a specific webpage. Many existing user agents such as graphical browsers and newsreaders show favicon as a visual reminder of the website identity in the address bar. If the favicon is loaded from a domain other than that shown in the address bar, then the webpage is likely to be considered a Phishing attempt.

Rule: IF$\begin{cases} \text{Favicon Loaded From External Domain} \rightarrow \text{ Phishing} \\ \qquad\qquad\qquad \text{Otherwise} \rightarrow \text{ Legitimate} \end{cases}$

2.2.1.11 Using Non-Standard Port

This feature is useful in authorizing if a particular service (e.g. HTTP) is up or down on a particular server. In the aim of controlling intrusions, it is much better to simply open ports that you need. Several firewalls, Proxy and Network Address Translation (NAT) servers will, by default, block all or most of the ports and only open the ones selected. If all ports are open, phishers can run almost any service they want and as a result, user information is threatened.

Rule: IF$\begin{cases} \text{Port \# is of the Preffered Status} \rightarrow \text{ Phishing} \\ \qquad\qquad \text{Otherwise} \rightarrow \text{ Legitimate} \end{cases}$

2.2.1.12 The Existence of "HTTPS" Token in the Domain Part of the URL

The phishers may add the "HTTPS" token to the domain part of a URL in order to trick users. For example: http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/.

Rule: IF$\begin{cases} \text{Using HTTP Token in Domain Part of The URL} \rightarrow \text{ Phishing} \\ \text{Otherwise} \rightarrow \text{ Legitimate} \end{cases}$

2.2.2 Abnormal Based Features

2.2.2.1 Request URL

Request URL examines whether the external objects contained within a webpage such as images, videos and sounds are loaded from another domain. In legitimate webpages, the webpage address and most of objects embedded within the webpage are sharing the same domain.

Rule: IF $\begin{cases} \text{\% of Request URL } < 22\% \rightarrow \text{ Legitimate} \\ \text{\%of Request URL} \geq 22\% \text{ and } 61\% \rightarrow \text{ Suspicious} \\ \text{Otherwise} \rightarrow \text{ feature} = \text{Phishing} \end{cases}$

2.2.2.2 URL of Anchor

An anchor is an element defined by the <a> tag. This feature is treated exactly as "Request URL". However, for this feature we examine:

1. If the <a> tags and the website have different domain names. This is similar to request URL feature.
2. If the anchor does not link to any webpage, e.g.:

   A. <a href="#">

   B. <a href="#content">

   C. <a href="#skip">

   D. <a href="JavaScript ::void(0)">

© 2016, IJCSMC All Rights Reserved

Rule: $\text{IF}\begin{cases} \text{\% of URL Of Anchor } < 31\% \rightarrow \mathit{Legitimate} \\ \text{\% of URL Of Anchor } \geq 31\% \text{ And } \leq 67\% \rightarrow \text{Suspicious} \\ \text{Otherwise} \rightarrow \text{Phishing} \end{cases}$

2.2.2.3 Links in <Meta>, <Script> and <Link> tags

Given that our examination covers all angles likely to be used in the webpage source code, we find that it is common for legitimate websites to use <Meta> tags to offer metadata about the HTML document; <Script> tags to create a client side script; and <Link> tags to retrieve other web resources. It is expected that these tags are linked to the same domain of the webpage.

Rule:

IF

$$\begin{cases} \text{\% of Links in “} < \text{Meta} > \text{”,”} < \text{Script} > \text{” and “} < \text{Link>”} < 17\% \rightarrow \text{Legitimate} \\ \text{\% of Links in } < \text{Meta} > \text{”,”} < \text{Script} > \text{” and “} < \text{Link>”} \geq 17\% \text{ And } \leq 81\% \rightarrow \text{Suspicious} \\ \text{Otherwise} \rightarrow \text{Phishing} \end{cases}$$

2.2.2.4 Server Form Handler (SFH)

SFHs that comprise an empty string or “about: blank” are considered doubtful because an action should be taken upon the submitted information. In addition, if the domain name in SFHs is different from the domain name of the webpage, this reveals that the webpage is suspicious because the submitted information is rarely handled by external domains.

Rule: $\text{IF}\begin{cases} \text{SFH is “about: blank” Or Is Empty} \rightarrow \text{Phishing} \\ \text{SFH Refers To A Different Domain} \rightarrow \text{Suspicious} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.2.5 Submitting Information to Email

Web form allows a user to submit his personal information that is directed to a server for processing. A phisher might redirect the user’s information to his personal email. To that end, a server-side script language might be used such as “mail()” function in PHP. One more client-side function that might be used for this purpose is the “mailto:” function.

Rule: IF $\begin{cases} \text{Using “mail()” or mailto: Function to Submit User Information} \rightarrow \text{ Phishing} \\ \qquad\qquad\qquad\qquad\qquad \text{Otherwise } \rightarrow \text{ Legitimate} \end{cases}$

2.2.2.6 Abnormal URL

This feature can be extracted from WHOIS database. For a legitimate website, identity is typically part of its URL.

Rule: IF $\begin{cases} \text{The Host Name Is Not Included In URL } \rightarrow \text{ Phishing} \\ \qquad\qquad \text{Otherwise} \rightarrow \text{ Legitimate} \end{cases}$

2.2.3 HTML and JavaScript based Features

2.2.3.1 Website Forwarding

The fine line that distinguishes phishing websites from legitimate ones is how many times a website has been redirected. In our dataset, we find that legitimate websites have been redirected one time max. On the other hand, phishing websites containing this feature have been redirected at least 4 times.

Rule: IF $\begin{cases} \#\text{ofRedirect Page} \leq 1 \rightarrow \text{ Legitimate} \\ \#\text{of Redirect Page} \geq 2 \text{ And} < 4 \rightarrow \text{ Suspicious} \\ \qquad \text{Otherwise } \rightarrow \text{ Phishing} \end{cases}$

2.2.3.2 Status Bar Customization

Phishers may use JavaScript to show a fake URL in the status bar to users. To extract this feature, we must dig-out the webpage source code, particularly the “onMouseOver” event, and check if it makes any changes on the status bar.

Rule: IF $\begin{cases} \text{onMouseOver Changes Status Bar} \rightarrow \text{ Phishing} \\ \text{ It Does}'\text{t Change Status Bar} \rightarrow \text{Legitimate} \end{cases}$

2.2.3.3 Disabling Right Click

Phishers use JavaScript to disable the right-click function, so that users cannot view and save the webpage source code. This feature is treated exactly as “Using onMouseOver to hide the Link”. Nonetheless, for this feature, we will search for event “event.button==2” in the webpage source code and check if the right click is disabled.

Rule: IF $\begin{cases}\text{Right Click Disabled} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$

2.2.3.4 Using Pop-up Window

It is unusual to find a legitimate website asking users to submit their personal information through a pop-up window. On the other hand, this feature has been used in some legitimate websites and its main goal is to warn users about fraudulent activities or broadcast a welcome announcement, though no personal information was asked to be filled in through these pop-up windows.

Rule: IF $\begin{cases}\text{Popoup Window Contains Text Fields} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$

2.2.3.5 IFrame Redirection

IFrame is an HTML tag used to display an additional webpage into one that is currently shown. Phishers can make use of the “iframe” tag and make it invisible i.e. without frame borders. In this regard, phishers make use of the “frameBorder” attribute which causes the browser to render a visual delineation.

Rule: IF $\begin{cases}\text{Using iframe} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$

2.2.4 Domain based Features

2.2.4 1 Age of Domain

This feature can be extracted from WHOIS database (Whois 2005). Most phishing websites live for a short period of time. By reviewing our dataset, we find that the minimum age of the legitimate domain is 6 months.

Rule: IF $\begin{cases}\text{Age Of Domain} \geq 6 \text{ months } \rightarrow \text{ Legitimate} \\ \qquad\qquad \text{Otherwise } \rightarrow \text{ Phishing}\end{cases}$

2.2.4 2 DNS Record

For phishing websites, either the claimed identity is not recognized by the WHOIS database (Whois 2005) or no records founded for the hostname. If the DNS record is empty or not found then the website is classified as “Phishing”, otherwise it is classified as “Legitimate”.

Rule: IF$\begin{cases}\text{no DNS Record For The Domain } \rightarrow \text{ Phishing} \\ \qquad\qquad \text{Otherwise} \rightarrow \text{ Legitimate}\end{cases}$

2.2.4 3 Website Traffic

This feature measures the popularity of the website by determining the number of visitors and the number of pages they visit. However, since phishing websites live for a short period of time, they may not be recognized by the Alexa database (Alexa the Web Information Company., 1996). By reviewing the dataset, we find that in worst scenarios, legitimate websites ranked among the top 100,000. Furthermore, if the domain has no traffic or is not recognized by the Alexa database, it is classified as “Phishing”. Otherwise, it is classified as “Suspicious”.

Rule: IF$\begin{cases}\text{Website Rank} < 100{,}000 \rightarrow \text{ Legitimate} \\ \text{Website Rank} > 100{,}000 \rightarrow \text{Suspicious} \\ \qquad\qquad \text{Otherwise } \rightarrow \text{ Phish}\end{cases}$

2.2.4. 4 PageRank

PageRank is a value ranging from “0” to “1”. PageRank aims to measure how important a webpage is on the Internet. The greater the PageRank value the more important the webpage. In our datasets, we find that about 95% of phishing webpages have no PageRank. Moreover, we find that the remaining 5% of phishing webpages may reach a PageRank value up to “0.2”.

Rule: IF$\begin{cases} \text{PageRank} < 0.2 \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.4 5 Google Index

This feature examines whether a website is in Google’s index or not. When a site is indexed by Google, it is displayed on search results (Webmaster resources, 2014). Usually, phishing webpages are merely accessible for a short period and as a result, many phishing webpages may not be found on the Google index.

Rule: IF$\begin{cases} \text{Webpage Indexed by Google} \rightarrow \text{Legitimate} \\ \text{Otherwise} \rightarrow \text{Phishing} \end{cases}$

2.2.4.6 Number of Links Pointing to Page

The number of links pointing to the webpage indicates its legitimacy level, even if some links are of the same domain (Dean, 2014). In our datasets and due to its short life span, we find that 98% of phishing dataset items have no links pointing to them. On the other hand, legitimate websites have at least 2 external links pointing to them.

Rule: IF$\begin{cases} \text{\#Of Link Pointing to The Webpage} = 0 \rightarrow \text{Phishing} \\ \text{\#Of Link Pointing to The Webpage} > 0 \text{ and } \leq 2 \rightarrow \text{Suspicious} \\ \text{Otherwise} \rightarrow \text{Legitimate} \end{cases}$

2.2.4.7 Statistical-Reports Based Feature

Several parties such as PhishTank (PhishTank Stats, 2010-2012), and StopBadware (StopBadware, 2010-2012) formulate numerous statistical reports on phishing websites at every given period of time; some are monthly and others are quarterly. In our research, we used 2 forms of the top ten statistics from PhishTank: “Top 10 Domains” and “Top 10 IPs” according to statistical-reports published in the last three years, starting in January2010 to November 2012. Whereas for “StopBadware”, we used “Top 50” IP addresses.

$$\text{Rule: IF}\begin{cases}\text{Host Belongs to Top Phishing IPs or Top Phishing Domains } \rightarrow \text{ Phishing}\\ \text{Otherwise } \rightarrow \text{ Legitimate}\end{cases}$$

2.3 Rotation Forest

The Rotation Forest classifier chooses decision tree as base classifier and Principle Component Analysis (PCA) as the feature extraction method. The main factor for the outperformance of the rotation forest is that the transformation matrix used to evaluate the extracted features.

***Rotation Forest Algorithm*:** Let X be the training set kernel vectors and Y *be* the class labels of the corresponding input kernel vector and F be the feature set. Assume that there are N numbers of training instances with n features, and then X will become an N by n matrix. Let Y take values from set of class labels $\{\omega_1, . . . \omega_c\}$ denoted by ω. The feature set of dataset is assumed to be partitioned into K subsets and the decision trees numbers of Rotation Forest algorithm is to be L with notation of $\{D_1,..., D_L\}$. The data used in training of base classifier is created with a randomly split K feature set [15, 16].

The training set for classifier D_i is handled in three steps:

(i) As a first step, F is divided into K feature sets randomly with each subset of $M = n/K$ number of features.

(ii) In this second step, let F_{ij} denote the j^{th} subset of features to train classifier D_i and X_{ij} be the set of data for F_{ij}, being subset of features. A nonempty random subset is drawn from X_{ij} and then with bootstrap to form a new training set the 75% of this training data is selected as *X*'. A linear transformation is operated on *X*' to generate the coefficients of matrix C_{ij}. Each matrix *X*' has size of M ×1 and the coefficients of this matrix are $a_{ij}^{(1)} \ldots a_{ij}^{(M_j)}$

(iii) In this last step, having obtained the coefficients of matrix C_{ij}, a sparse rotation matrix R_i then formed

$$R_i = \begin{bmatrix} a_{i1}^{(1)} \ldots a_{i1}^{(M_1)} & [0] & \ldots & [0] \\ [0] & a_{i2}^{(1)} \ldots a_{i2}^{(M_2)} & \ldots & [0] \\ \ldots & \ldots & \ldots & \ldots \\ [0] & [0] & \ldots & a_{iK}^{(1)} \ldots a_{iK}^{(M_K)} \end{bmatrix}$$

Here at this point, the columns of R_i are rearranged with respect to the original feature set and the new rotation matrix is represented as $\mathrm{R_i^a}$. The transformed training set for classifier D_i will become $\mathrm{XR_i^a}$. By means of this approach, the classifiers are provided with parallel training.

While the classification phase is evaluated for a given instance *x*, let the probability of this instance being classified by classifier D_i to one of classes is denoted with $\mathrm{d_{ij}}(x\mathrm{R_i^a})$. From this point, the confidence of a class is calculated by Eq. (1), and *x* is assigned to a class with the largest confidence calculated.

$$\mu_j(x) = \frac{1}{L} \sum_{i=1}^{L} d_{ij}\left(xR_i^a\right), j = 1 \ldots c \tag{1}$$

The Rotation Forest algorithm applies Principal Component Analysis (PCA) transformation to each *K* subset to determine principal components that are expected to preserve the variability of information in the data. By means of *K* axis rotations, the new features for base classifier are formed. The Rotation approach in this method serves the ensemble with accuracy and diversity. In traditional the Rotation Forest algorithm, decision trees are chosen for rotation task, because of their sensitivity to rotation of the feature axes. And hence the name ‘forest’ is inspired from this scheme.

3. Results and Discussion

For testing our proposed method the experiments were conducted for prediction task of phishing websites by separately applying twelve machine learning algorithms namely: Rotation Forest with LMT as base classifier (ROF+LMT), Rotation Forest with J48 as base classifier (ROF+J48), Logistic Model Tree (LMT), J48, Random Forest (RF), Aggregating One-Dependence Estimators (AODE), Logistic, Multilayer Perceptron Classifier (MLPC), Multilayer Perceptron Classifier (MLPC), Radial Basis Function Classifier(RBFC), Naive Bayes, Simple Logistic (SLG) and Sequential Minimum Optimisation (SMO) using Weka 3.7.12 [17]. The classification performances of the classifiers were analysed with respect to the standard performance parameters, namely: Accuracy, Specificity, Sensitivity, Precision, Receiver Operating Characteristic (ROC) Area [18], Matthew’s Correlation Coefficient (MCC) besides time taken for training (learning). The formula for calculating these parameters are given below:

$$Sensitivity = \frac{tp}{tp + fn} * 100 \qquad (2)$$

$$Specificit\,y = \frac{tn}{tn + fp} * 100 \qquad (3)$$

$$Accuracy = \frac{tp + tn}{tp + fp + tn + fn} \qquad (4)$$

$$\Pr ecision = \frac{tp}{tp + fp} \qquad (5)$$

$$MCC = \frac{(tp * tn) - (fp * fn)}{\sqrt{(tp + fn) * (tn + fp) * (tp + fp) * (tn + fn)}} \qquad (5)$$

where

tp is the number of true positives,

tn is the number of true negatives,

fp is the number of false positives and

fn is the number of false negatives.

The table 1 shows the values of Sensitivity, Specificity, Accuracy, Precision, MCC, AUC performance metrics besides their training time for all the twelve classifiers separately for our chosen dataset.

(Insert Table 1)

The sensitivity indicates the ability of the classifier to identify positive instances correctly, the specificity indicates the ability of the classifier to identify negative instances correctly and accuracy indicates the percentage of correct classification of both positive class as well as negative class instances. The ROF+LMT performs better than other classifiers with sensitivity, specificity and accuracy values 0.962, 0.982 and 0.974 respectively.

The Mathews Correlation Coefficient (MCC) is another important parameter to evaluate the performance of the binary class classifiers. A coefficient of +1 represents a perfect classification, 0 an average random classification and −1 an inverse classification. It can be observed from the table 1 that, that classifier having high value of accuracy performance parameter for a particular family also have high MCC. In our experiment the MCC value we achieved is 0.946 for ROF+LMT.

The area under ROC curve (AUC) is an important statistical property to compare the overall relative performance of the classifiers. AUC can take values from 0 to 1. The value 0 for the worst case, 0.5 for random ranking and 1 indicates the best classification as the classifier has ranked all positive examples above all negative example. The figure 1 shows

© 2016, IJCSMC All Rights Reserved

that AUC value of ROF+LMT classifier is greater than other classifier for our considered dataset equals to 0.996.

(Insert Fig. 1)

4. Conclusion

We have compared the performance of twelve classifiers (including SVM, which was reported as the better performing classifier by the previous studies) in the prediction of phishing websites. The experimental results of our proposed method have demonstrated that ROF+LMT has produced superior prediction performance in terms of classification accuracy, AUC and MCC respectively for selected dataset. It was also observed that few classifiers have yielded poor classification accuracy like SMO and RBF. This problem will be investigated in our future study by (i) Exploring all possible combination of various different types of input features and different machine learning algorithms, (ii) By deal with various factors that affects prediction performance (such as class imbalance, incomplete learning etc.) for improving the prediction accuracy and finally identifying the exact cause (through checking very high similarity by generating human interpretable rules through PART algorithm. In future I am also planning to develop a web tool based on our discovered algorithm which will be helpful in prediction of phishing website.

References:

1. Aburrous M, Hossain MA, Dahal K, Fadi T (2010) Predicting phishing websites using classification mining techniques. In: Seventh international conference on information technology, Las Vegas, Nevada, USA
2. Aburrous M, Hossain MA, Dahal K, Thabtah F (2010) Intelligent phishing detection system for e-banking using fuzzy data mining. Expert Syst Appl Int J 37(12):7913–7921.

© 2016, IJCSMC All Rights Reserved

3. Pan Y, Ding X (2006) Anomaly based web phishing page detection. In: ACSAC ‘06: Proceedings of the 22nd annual computer security applications conference, Washington, DC.
4. Cortes C, Vapnik V (1995) Support-vector networks. Machine Learning 20(3):273–297.
5. Zhang Y, Hong J, Cranor L (2007) CANTINA: a content-based approach to detect phishing web sites. In: Proceedings of the 16th World Wide Web conference, Banff, Alberta, Canada.
6. Manning CD, Raghavan P, Schutze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge.
7. Sanglerdsinlapachai N, Rungsawang A. (2010) Using domain top page similarity feature in machine learning-based web. In: Third international conference on knowledge discovery and data mining, Washington, DC.
8. Sadeh N, Tomasic A, Fette I (2007) Learning to detect phishing emails. In: Proceedings of the 16th international conference on World Wide Web, pp 649–656.
9. Wenyin L, Huang G, Xiaoyue L, Min Z, Deng X (2005) Detection of phishing webpages based on visual similarity. In: Proceeding WWW ‘05 Special interest tracks and posters of the 14th international conference on World Wide Web, New York, NY.
10. Dhamija R, Tygar JD (2005). The battle against phishing: dynamic security skins. In: Proceedings of the 1st symposium on usable privacy and security, New York, NY.
11. Miyamoto D, Hazeyama H, Kadobayashi Y (2008). An evaluation of machine learning-based methods for detection of phishing sites. Aust J Intell Inf Process Syst 10(2):54–63.

12. Mohammad, Rami, McCluskey, T.L. and Thabtah, Fadi (2012) An Assessment of Features Related to Phishing Websites using an Automated Technique. In: International Conferece For Internet Technology And Secured.
13. Mohammad, Rami, Thabtah, Fadi Abdeljaber and McCluskey, T.L. (2014) Predicting phishing websites based on self-structuring neural network. Neural Computing and Applications, 25 (2). pp. 443458. ISSN 09410643.
14. Mohammad, Rami, McCluskey, T.L. and Thabtah, Fadi Abdeljaber (2014) Intelligent Rule based Phishing Websites Classification. IET Information Security, 8 (3). pp. 153160. ISSN 17518709.
15. Juan J., Rodriguez, L., I., Kuncheva, (2006). Rotation forest: a new classifier ensemble method. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(10), 1619–30.
16. Ludmila I., Kuncheva, Jaun J., Rodriguez, (2007). An experimental study on rotation forest ensembles. Proceedings of the 7th International Conference on Multiple Classifier Systems, Springer-Verlag, Prague, Czech Republic, 459–68
17. I., H., Witten and H., Ian, (2011). Data mining: practical machine learning tools and techniques. Morgan Kaufmann Series in Data Management Systems.
18. Tom Fawcett, (2003). ROC graphs: Notes and practical considerations for data mining researchers. Technical report, HP Laboratories

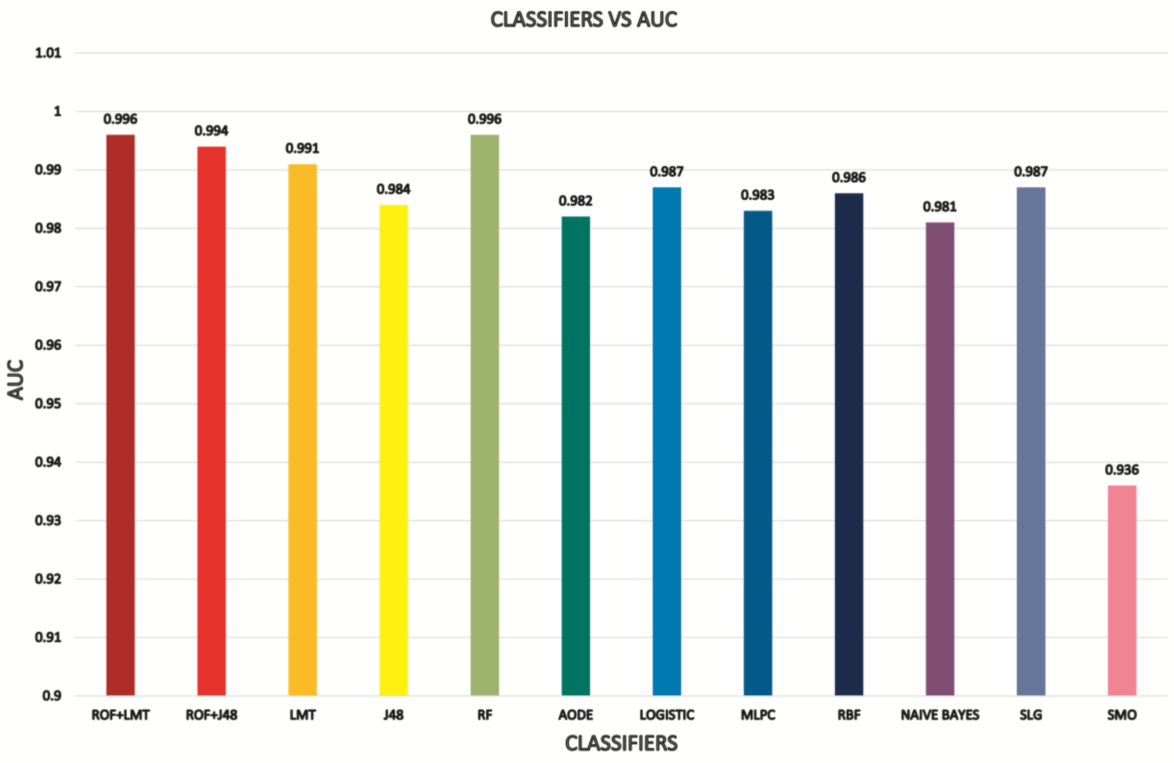
FIGURE CAPTIONS

Fig 1: AUC of selected classifiers for phishing website dataset

TABLE CAPTIONS

Table 1: Performance of twelve classifiers for phishing website dataset

Fig 1: AUC value of selected classifiers for phishing website dataset



© 2016, IJCSMC All Rights Reserved

Table 1: Performance of twelve classifiers for phishing website dataset

Classifiers	**Sensitivity**	**Specificity**	**Accuracy**	**Precision**	**MCC**	**AUC**	**Training Time (in sec)**
ROF+LMT	**0.962**	**0.982**	**0.974**	**0.974**	**0.946**	**0.996**	**442.64**
ROF+J48	**0.956**	**0.977**	**0.968**	**0.968**	**0.935**	**0.994**	**021.11**
LMT	**0.956**	**0.979**	**0.969**	**0.969**	**0.937**	**0.991**	**058.44**
J48	**0.942**	**0.972**	**0.959**	**0.959**	**0.916**	**.984**	**000.44**
RF	**0.961**	**0.982**	**0.973**	**0.973**	**0.944**	**0.996**	**002.90**
AODE	**0.906**	**0.952**	**0.931**	**0.932**	**0.861**	**0.982**	**000.16**
LOGISTIC	**0.923**	**0.953**	**0.940**	**0.940**	**0.878**	**0.987**	**002.30**
MLPC	**0.943**	**0.952**	**0.948**	**0.948**	**0.894**	**0.983**	**018.14**
RBF	**0.923**	**0.953**	**0.940**	**0.940**	**0.877**	**0.986**	**010.22**
Naive Bayes	**0.904**	**0.950**	**0.930**	**0.930**	**0.858**	**0.981**	**000.05**
SLG	**0.921**	**0.953**	**0.939**	**0.939**	**0.876**	**0.987**	**000.987**
SMO	**0.920**	**0.953**	**0.938**	**0.938**	**0.874**	**0.936**	**037.24**