IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

Genetically Optimized Neural Network with Lexical Features for Phishing Detection

Varsha Malik
M. Tech. Scholar
Dept. of Computer Science and Engineering
BSA College of Engineering and Technology,
Mathura (India)
varsha.mlk@gmail.com

Sanjay Kumar
Asst. Professor
Dept. of Computer Science and Engineering
BSA College of Engineering and Technology,
Mathura (India)
sanjaysingh.chauhan@bsacet.org

Abstract –**Over the past few years, we applied several different methods to detect phishing web by means of known and new features. This paper gives strategies for distinguishing phishing sites by dissecting different components of phishing URLs. It talks about the systems utilized for identification of phishing sites in view of lexical features. We propose several novel highly effective lexical features to study the anatomy of URLs. We considered different data mining approaches for assessment of the features to show signs of improvement comprehension of the structure of URLs that spread phishing. This paper utilized Genetically Optimized Neural Network classifier algorithm. The simulation results provide the accuracy, specificity, true positive rate and false positive rate which is evaluated on the basis of Confusion matrix.**

In order to detect and filter such kind of emails, Neural Network and Genetically Optimized Neural Network classifier are proposed to use for email classification and detection of spam mails. The performance of both the classifiers is evaluated in terms of Accuracy, Error, Time, Precision and Recall.

Keywords –**E-Mail, GA-NN, SPAM, URL.**

I. INTRODUCTION

Phishing is an attempt to steal personal confidential information such as passwords, credit card information from innocent victims for financial gain, identity theft and other fraudulent activities by an individual or a group. The current scenario, when the user desires to access his confidential information online (like payment gateway or money transfer) by logging into his secure mail account or bank account, the individual enters information like credit card no., username, password etc. on the login page. But quite often, this information can be taken by intruders using phishing techniques (for example, when a user provides login information on a phishing website his data is stolen and then he is redirected to the genuine site). There is no such information that cannot be directly obtained from the user at the time of his login input.

Phishing web pages are fake web pages that are made by malicious individuals to mimic Web pages of genuine web sites. Most of these types of web pages have great visual similarities to trick their victims. Some of these types of web pages look exactly like the genuine ones. Victims of phishing web pages may expose their credit card number, password, bank account or other vital information to the phishing web page owners. It includes techniques such as deceiving customers through URL, screen captures, spam messages, emails and installation of key loggers.

Several ideas were borrowed from Spoofguard and additional checks were added to figure out the trends within the phishing websites. However, In spite of different scenarios it is difficult to provide maximum accuracy. The core problem is to decrease the detection of false positives and increase the true positives thereby increase the overall accuracy of the system.

The major concern of this research is to design a framework intended for assessment of the lexical features to show signs of improvement through comprehensively studying the components of the URLs which promote phishing, by the means of Genetically Optimized Neural Network classifier algorithms.

II. PROPOSED METHODOLOGY

The classifier takes unclassified URLs as input, and returns a predicted binary class as output (either Phish or Benign). Our aim is to evaluate the effectiveness of URL features as discriminating features.

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

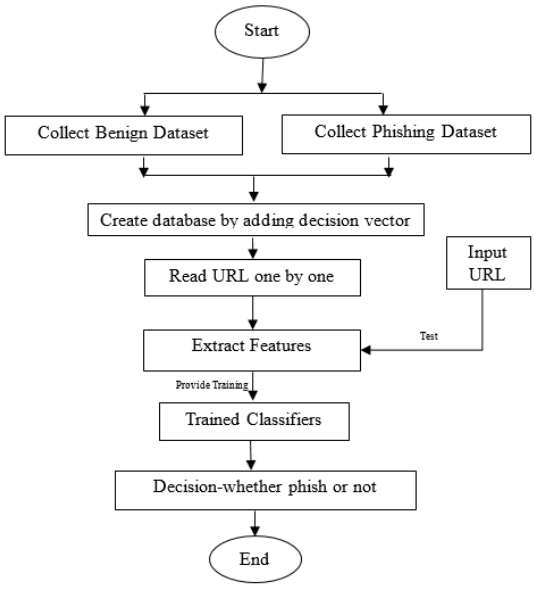

Figure 1: Flow diagram of proposed architecture

We started with collection of URLs and then after loading the URLs we started by reading URLs one by one for feature extraction. To facilitate feature extraction, each URL was split into three sections: protocol, domain, and path. All subsequent feature extraction was performed on these sub-regions. After collecting of URL features, the classifier's life initiates by a supervised learning phase. During this phase, the classifier is fed with pre-classified URL along with their pre-defined class. The classifier is then able to perceive a classification model. Once the learning phase is complete, the classifier is given unclassified URLs as input, and a predicted class is returned as output.

Architectures also hold room for checking a particular URL for Phishing. A random URL is provided to the trained classifier for recognizing the class (Phishing or Benign) of the given URL.

Collection of URLs

Here in this research work, we have taken URLs of benign websites from www.alexa.com [1] www.dmoz.org [2] and personal web browser history. The phishing URLs were collected from www.phishtak.com [3].

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.

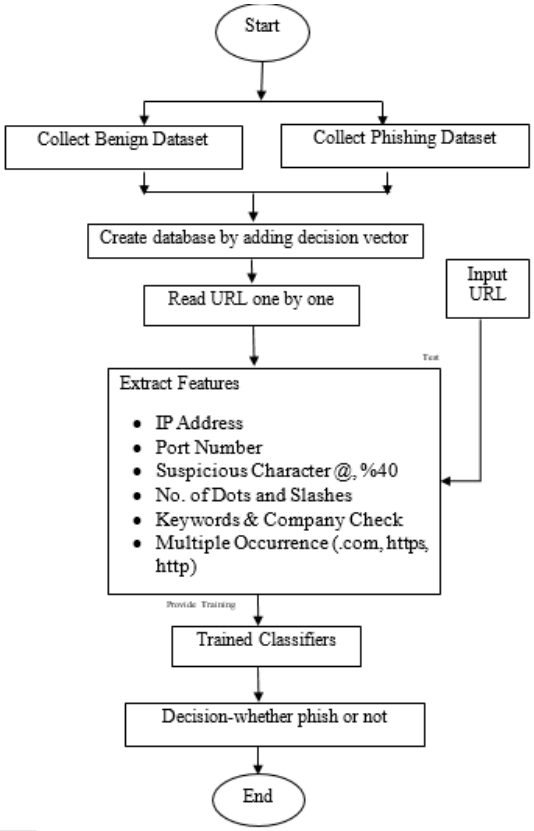

Figure 2: Flow diagram for lexical feature extraction

IP Address

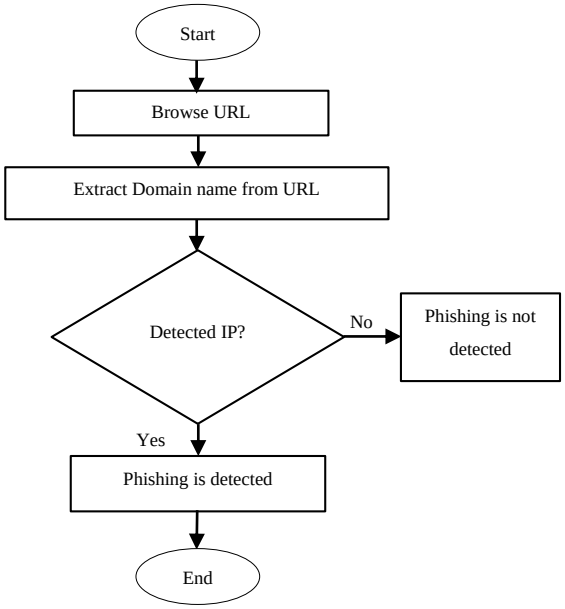

Figure 3: Phishing URL detection using IP address

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

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

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 at times 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 [4] 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.

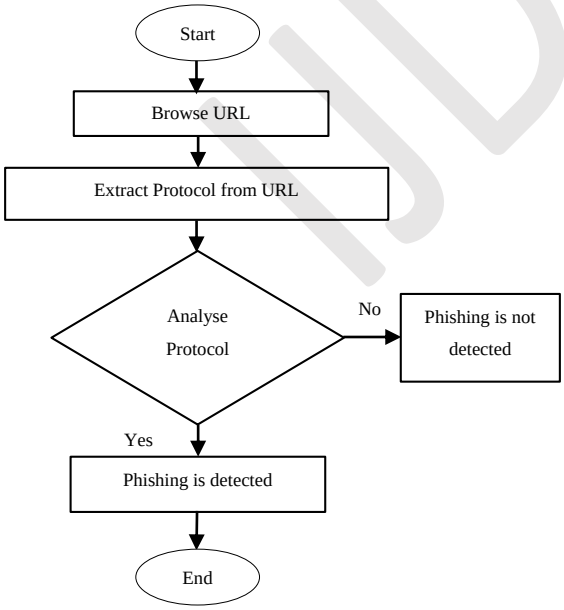

Figure 4: Phishing URL detection using protocol

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 various 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 [5]. 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 [6].

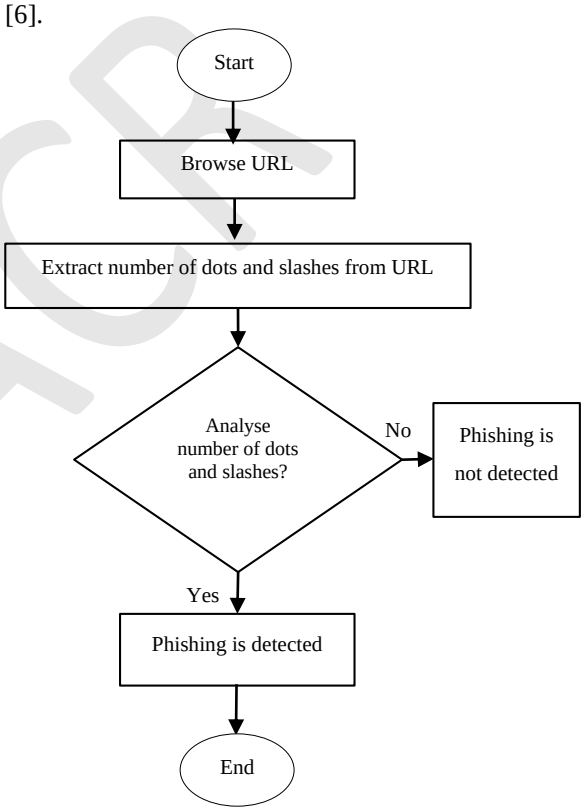

Figure 5: Phishing URL detection using number of dots and slashes

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

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

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. A few cases illustrate, 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.

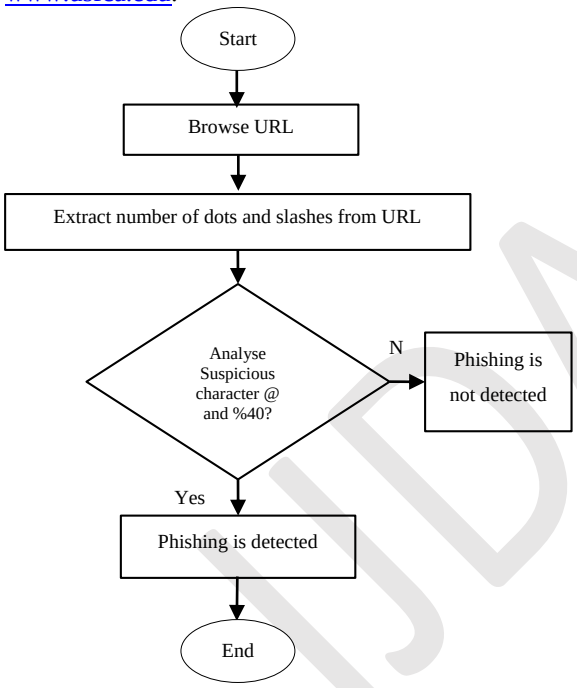

Figure 6: Phishing URL detection using suspicious character @ and %40

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.com This URL would refer a user from one site (in this case, google.com) to another site, badsite.com.

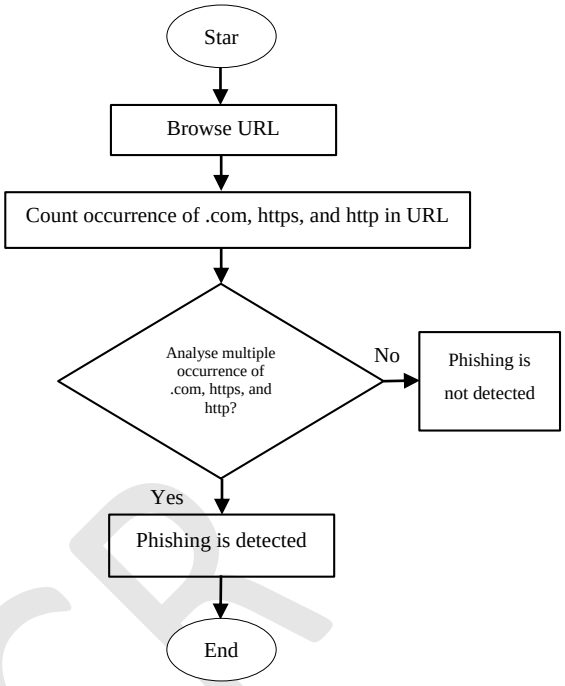

Figure 7: Phishing URL detection using multiple occurrence of .com, https, and http

Keyword Check

There are a lot of variety of URL based properties which can be used in a phishing URL. Here 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".

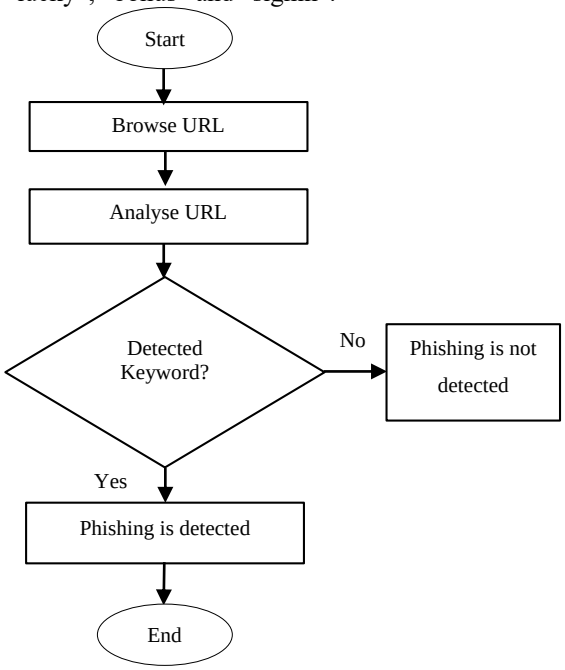

Figure 8: Phishing URL detection using keyword check

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

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 [7]. 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.

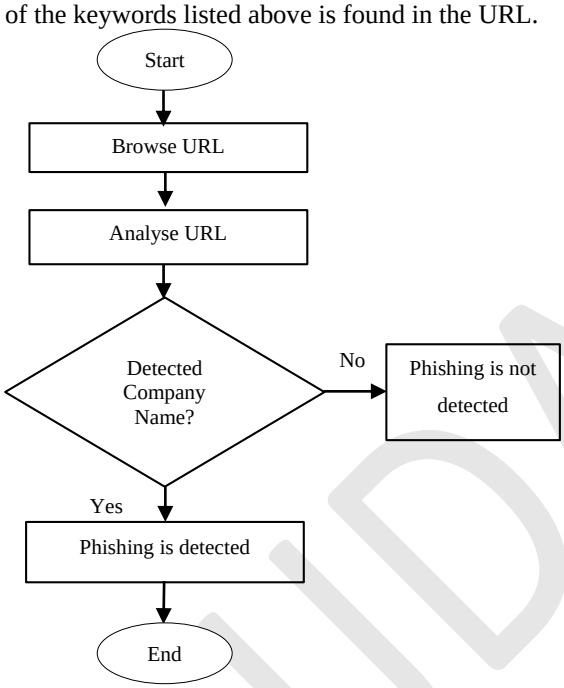

Figure 9: Phishing URL detection using company check

Classification

The input to the classifiers in MATLAB is two .txt files; newben.txt and newphis.txt. The Genetically optimized Neural Network classifier is considered for processing the feature set are:

Genetic Algorithm

Genetic Algorithm of GA is an optimization tool that lies on the platform of Heuristic Approaches. Based on the proposal of Darwin principle of fittest survival, this method was introduced to commence optimization problems in soft computing [8]. The first category of results is termed as initial population and all the individuals are candidate solution. Simultaneous study of the population including all candidates and next phase of solutions are generated following the steps of GA [9].

An iterative application of operators on the selected initial population is the initiative process of GA. Further steps are devised based on valuation of this population. The typical routing of GA is described in following pseudo code:

1. Randomly generate initial population.
2. Employ fitness function for evaluation.
3. Chromosomes with superior fitness are valued as parents.
4. New population generation by parent's crossover with probability function.
5. Chromosome mutation with probability to defend system from early trap.
6. Repeat step 2.
7. Terminate algorithm based on satisfaction criteria.

Training with Optimized Neural Network

In previous phase, neural network is optimized using particle swarm optimization then the optimized NN is used to train extracted class data using back propagation algorithm.

Objective Function

Back propagation neural network is a type of multi-layer feed forward network in which each layer is connected by transfer functions and can fulfil arbitrary nonlinear mapping. It is widely applied in stock price, petroleum price, economic time sequence, network flow and other nonlinear areas and attained satisfactory performance. The basic learning process of the back propagation neural network algorithm is as follows [8]:

1. Initialize the connection weights w_{ij}, v_{jt} and threshold θ_j in the back propagation neural network.
2. Input the first learning sample couples to the back propagation neural network.
3. Compute the input u_j of each neural unit and the output h_j in the hidden layer. The equation is:

$$u_j = \sum_{i=1}^{n} w_{ij} x_i - \theta_j \quad (1)$$

$$h_j = f(u_j) = \frac{1}{1+\exp(-u_j)} \quad (2)$$

4. Compute the input l_t of each neural unit and the output y_t in the output layer. The equation is:

$$l_t = \sum v_{jt} h_j - \gamma_t \quad (3)$$

$$y_t = \frac{1}{1+\exp(-l_t)} \quad (4)$$

5. Compute the weights error δ_t which is connected to the neural unit t in the output layer.

$$\delta_t = (c_t - y_t) y_t (1 - y_t) \quad (5)$$

In the equation (5), c_t represents the expectation of the sample.

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

6. Compute the weights error δ_j which is connected to the neural unit j in the hidden layer.
$$\delta_j = \sum_{t=1}^{q} \delta_t v_{jt} h_j (1 - h_j) \quad (6)$$
7. Update the connection weights v_{jt} and threshold γ_t in the back propagation neural network.
$$v_{jt}(N+1) = v_{jt}(N) + \alpha \delta_t h_j \quad (7)$$
$$\gamma_t(N+1) = \gamma_t(N) + \beta\delta \quad (8)$$
8. Update the connection weights w_{jt} and threshold $\theta_j$in the back propagation neural network.
$$w_{jt}(N+1) = w_{jt}(N) + \alpha \delta_j x_i \quad (9)$$
$$\theta_j(N+1) = \theta_j(N) + \beta\delta_j \quad (10)$$
9. Input the next learning sample and go to the step 3 until all of the samples are trained.
10. Back propagation neural network go to a new round of learning. If it meets the equation (11), the training of the back propagation network can be ended.
$$\left|\sum_{k=1}^{z} E_k\right| \le \varepsilon \quad (11)$$

In the equation (11), ε represents the accuracy requirement of back propagation neural network, E_k represents the mean square error and the definition are as follows:

$$E_k = \frac{1}{2}\sum_{t=1}^{q}(c_t - y_t)^2 \quad (12)$$

Before training the back propagation neural network, proper connection weights w_{ij} and v_{jt} of the back propagation neural network should be chosen. Normally the initialization is randomly which can cause the convergence is slow and the defect of local optimal solutions.

III. Simulation Results

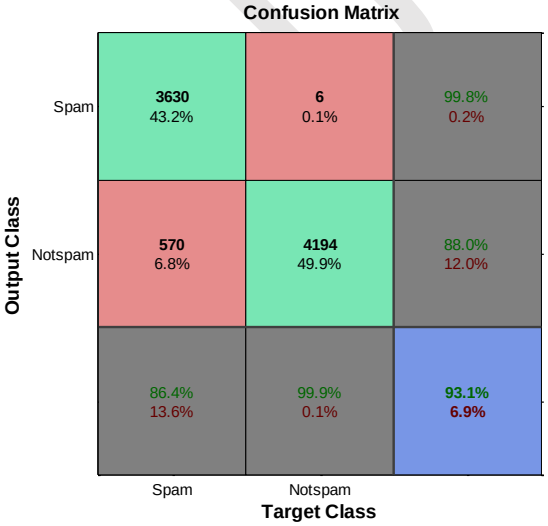

Figure 10: Confusion matrix for NN classifier

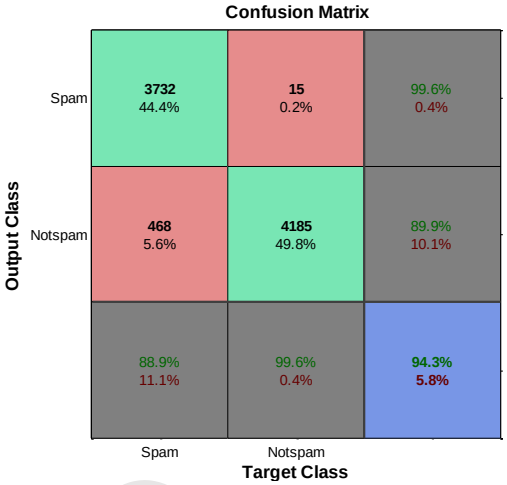

Figure 11: Confusion matrix for GA-NN classifier

On observing Table 1 shown below, it was found that the proposed GA-NN approach outperforms the standalone NN approach.

Table 1: Result comparison

Test Options	Classifiers	Proposed Approach (Success Rate)
Percentage Split-60	GA	93.1
	GA-NN	94.3

IV. Conclusion

Phishing recognition techniques are rapidly varying to keep up with the novel techniques used by phishers. Combating phishing is an ongoing battle that will perhaps never end much like the ongoing battle with spam emails. Phishers have various methodologies and procedures to conduct a well-designed phishing attack.

While generalizing about URLs, it is hard to conclude if a website is genuine or phishing just by the contents of the URL 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 have made use of Genetically Optimized Neural Network classifier as our intention was the evaluation of the feature. This work proved diagnostically that the proposed GA-NN methodology is showing the signs of improvement utilizing different lexical features for detecting phishing URLs.

Reference

[1] "DMOZ Open Directory Project," [Online]. Available: http://www.dmoz.org.
[2] "PhishTank," [Online]. Available: https://www.phishtank.com/.
[3] Xiang G., Hong J., Rose C. P. and Cranor L., "CANTINA+: A feature-rich machine learning framework for detecting phishing Web sites," ACM

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 3, October 2016)

Trans. Inf. Syst. Secur. 14, 2, Article 21, p. 28, September 2011.

[4] N. Chou, R. Ledesma, Y. Teraguchi, D. Boneh and J. Mi, "Client-side defense against web-based identity theft," in Proc. 11th Network and Distributed System Security Symposium, NDSS'04, San Diego, CA, USA, 2004.

[5] Xiaoqing GU, Hongyuan WANG and Tongguang NI, "An Efficient Approach to Detecting Phishing Web," Journal of Computational Information Systems, 2013.

[6] R. B. Basnet, A.H. Sung and Q. Liu, "Rule-based phishing attack detection," in Proc. Int. Conf. Security and Management, SAM'11, Las Vegas, NV, USA, 2011.

[7] S. Garera, N. Provos, M. Chew and 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.

[8] J. H. Holland, "Adaptation in Natural and Artificial Systems", University of Michigan Press, 1975.

[9] D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, 1989.

IJDACR