Phishing Identification: An Efficient Neuro-Fuzzy Model Without Using Rule Sets

Luong Anh Tuan Nguyen, Huu Khuong Nguyen
Ho Chi Minh City University of Transport
Email: {nlatuan, nhkhuong}@hcmutrans.edu.vn

***Abstract*—The Internet has brought enormous benefits to mankind, but it could be many potential risks. Internet crimes are growing rapidly, phishing is one of the new type of online crime. Phishing site is a fake-site aimed to steal personal information such as password, banking account and credit card information, etc. Most of these phishing pages look similar to the real pages in terms of interface and uniform resource locator (URL) address. Many techniques have been proposed to identify phishing sites. However, the numbers of victims have been increasing due to inefficient protection technique. In this paper, we develop a neuro-fuzzy model for phishing identification efficiently. The model eliminates the subjective factors to improve efficiency such as if-then rule sets, the parameters of membership functions, etc. Moreover, the efficiency features to identify phishing were used for the neuro-fuzzy model. The effectiveness of the proposed technique is examined with large-scale datasets collected from phishing sites and legitimate sites. The results show that the proposed technique can identify over 99% phishing sites.**

***Index Terms*—Phishing, URL-Based, Neuro-Fuzzy**

I. INTRODUCTION

Phisher use a number of techniques to fool their victims aimed to steal personal information including email messages, instant messages, forum post, phone calls and social networking. These activities of phishing cause severe economy loss all over the world. According to Fortune Magazine, in 2011, 83% of Americans and 85% of Europeans regularly shopped online. Meanwhile, phishing sites are also growing rapidly in quality and quantity. Therefore, the risk of stealing user information is extremely high. For of these reasons, identifying phishing problem is very urgent, complex and extremely important problem.

In this paper, an efficient method is proposed to identify the phishing sites that focuses on the features of URL (PrimaryDomain, SubDomain, PathDomain) and Google's parameters (PageRank, BackLink, GoogleIndex). Then, a proposed neuro-fuzzy model is a system which reduces the error and increases the performance. The neuro-fuzzy model uses computational models to perform without using if-then rule sets. The proposed technique achieved identification accuracy above 99% with low false signals.

The remainder of this paper is organized as follows: Section II presents the related works. System design is detailed in section III. Section IV evaluates the accuracy of the method. Finally, Section V concludes the paper and figures out the future works.

II. RELATED WORKS

Methods for identifying phishing can be divided into three groups: blacklist, heuristic and machine learning. The blacklist-based technique [1][2][3][4] maintains a list of phishing websites called blacklist. The technique is inefficient due to the rapid growth in the number of phishing sites. Therefore, the heuristic and machine learning approaches have received more attraction of researchers.

Cantina [5] presented the TF-IDF algorithm based on 27 features of webpage. This technique can identify 97% phishing sites with 6% false positives. Although this technique is efficient, the time extracting 27 features of webpage is too long to meet real time demand and some features are not necessary for improving the phishing identification accuracy. Similarly, Cantina+ [6] used machine learning techniques based on 15 features of webpage and only six of 15 features are efficient for phishing identification such as bad form, Bad action fields, Non-matching URLs, Page in top search results, Search copyright brand plus domain and Search copyright brand plus hostname. In [7], the author used the URL to identify phishing sites automatically by extracting and verifying different terms of a URL through search engine. Even though this paper proposed a new interesting technique, the identification rate is quite low (54.3%). The technique [8] developed a content-based approach to identify phishing called CANTINA, which considers the Google PageRank value of a page, the evaluation dataset is quite small. The characteristic of the source code is used to identify phishing sites in [9].

The authors in [10] have proposed fuzzy technique based on 27 features of webpage, classified into 3 layer. Each feature has three linguistic values: low, moderate, high. The technique has built a rule set, triangular and trapezoidal membership functions. The achieved rate of the technique is 86.2%. But, there exist many drawbacks in [10]. First, the rule sets are not objective and greatly depend on the builder. Second, the weight of each main criteria is used without any clarification. Finally, the used heuristics are not optimal and really effective.

The authors in [11] have proposed neural network technique. Three layers were used in the neural network including input layer, hidden layer and output layer. The best achieved rate of the technique is 95%. However, there exist some drawbacks in [11]. First, a number of hidden nodes and activation function must be determined through experimentation. Second, the authors do not explain why using one hidden layer. Third,

978-1-4799-7862-5/15/$31.00 © 2015 IEEE

the value of features do not know how is it calculated. Finally, the datasets are not enough to verify the accuracy.

With respect to previous techniques, URL plays a minor role for identifying phishing websites. In this paper, we design a new neuro-fuzzy model based on URL's features and Google's parameters to identify phishing sites. The model without defining the if-then rule sets is developed from [12] with new aspects: i) The new heuristics have been proposed to identify phishing website more effectively and rapidly. ii) The parameters of the membership functions are eliminated, so the fuzzy values are calculated more objective. iii) The input values are normalized to enhance the accuracy and the convergence of training phase. Besides, The aspects in [12] also supports the new model as follows: i) The weights are trained by neural network, so the model is more efficient. ii) The if-then rules are not utilized. Hence, the result will be more precise and objective.

III. SYSTEM DESIGN

A. URL

A URL (uniform resource locator) is used to locate the resources[13].

The structure of URL is as follows:

$< protocol >: // < subdomain > . < primarydomain > . < TLD > / < pathdomain >$

For example, with the URL: http://www.paypal.abc.net/login/web/index.html, there are six components as follows: Protocol is http, Subdomain is paypal, Primarydomain is abc, TLD is net, Domain is abc.net, Pathdomain is login/web/index.html

B. Features of URL

Phishers usually try to make the Internet address (URL) of phishing sites look similar to legitimate sites to fool online users. They cannot use the exact URL of the legitimate site, they make more spelling mistake the features of URL such as PrimaryDomain, SubDomain, PathDomain. For example, the URL www.Paypall.com looks similar to well known website www.Paypal.com, or http://www.Paypal.attack.com. If users are not careful, they will think that they are on the Paypal site.

C. Features of Google's Parameters

Obviously, Phishing page is a new page and exist for a short time, so the ranking of phishing page is very low and the number of links from the other pages is very few. Therefore, the Google's parameters such as PageRank, BackLink and GoogleIndex can support to identify phishing.

D. System Model Design

The model can be depicted in Fig 1.

1) Phase I - Selecting four features of URL: Four features are extracted from URL such as *Domain*, *PrimaryDomain*, *SubDomain* and *PathDomain*.

2) Phase II - Calculating six values of the heuristics: Six values of the heuristics are calculated and six heuristics are six input nodes of the neuro-fuzzy network.

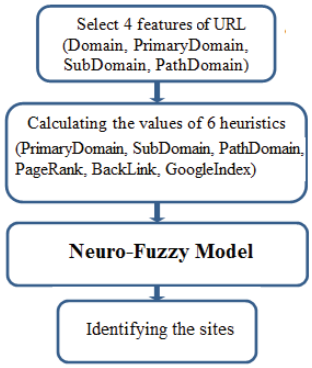

Fig. 1. The System Model

3) Phase III - Neuro-Fuzzy Network: The neuro-fuzzy network performs to calculate the value of the output node.

4) Phase IV - Identifying the sites: We based on the value of the output node to decide whether a site is a phishing site.

E. Neuro-Fuzzy Network Model

1) The model: The neuro-fuzzy network model was designed as in Fig 2. The model was designed with five layers as follows:

- The first layer, called the input layer, contains six nodes that are six heuristics such as PrimaryDomain, SubDomain, PathDomain, PageRank, BackLink, GoogleIndex.
- The second layer contains 12 nodes. The value of each node is calculated from the left sigmoid membership functions and the right sigmoid membership function.
- The third layer contains two nodes which are π_L and π_P. π_L and π_P are calculated by (1) and (2).

$$\pi_L = \prod_{i=1}^{6} L_i \tag{1}$$

$$\pi_P = \prod_{i=1}^{6} P_i \tag{2}$$

- The fourth layer contains two nodes which are NL (Normalization Legitimate) and NP (Normalization Phishing). NL and NP are calculated by (3) and (4).

$$NL = \frac{\pi_L}{\pi_L + \pi_P} \tag{3}$$

$$NP = \frac{\pi_P}{\pi_L + \pi_P} \tag{4}$$

- The fifth layer, called the output layer, has one output node.

The neural network performs from the fourth layer to the output layer. The weights are trained by the training algorithm and the sigmoid activation function is used in the proposed model, so the output value of the output node ranges from 0 to 1. The proposed model is classified into two classes so the site is phishing if the value of the output node is less than 0.5 and the site is legitimate, if the value is greater than or equal to 0.5.

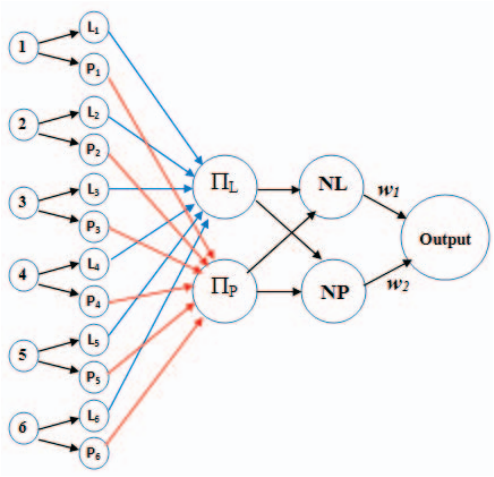

Fig. 2. The neuro-fuzzy network model

2) The value of six input nodes: Each input value varies from 0 to 1. If the input value close to zero, the site is doubted phishing site. If the input value closes to 1, the site will be a legitimate site. Six values of the input node are calculated as follows:

- Calculating the value of "PrimaryDomain": The value of " PrimaryDomain" is based on the Levenshtein distance [14] between " PrimaryDomain" and the result of GOOGLE search engine spelling suggestion. The algorithm is shown in Algorithm 1.
- Calculating the value of "SubDomain" and "PathDomain": Like "PrimaryDomain", the algorithm for calculating the value of "SubDomain" and "PathDomain" is shown in Algorithm 2.
- Calculating the value of "PageRank", "BackLink", "GoogleIndex": The values can be obtained from [15]. The algorithms for calculating the values are shown in Algorithm 3, Algorithm 4 and Algorithm 5.

3) Convert the value of six input nodes: In the second layer, the sigmoid membership functions are used to calculate the fuzzy values. the variable x of the membership functions ranges from -10 to 10. So, the value of the input nodes must be converted by equation (5).

$$Value_{new} = Value_{old} * (Max - Min) + Min \quad (5)$$

Where, Max is 10 and Min is -10. $Value_{old}$ ranges from 0 to 1. $Value_{new}$ ranges from -10 to 10.

4) The value of 12 nodes in the second layer: Each of these heuristics is classified into linguistic variables as "Phishing" and "Legitimate". Equation (6) and (7) are two membership functions that are built to calculate fuzzy values and the graph of the membership functions is shown in Fig 3 .

$$L(x) = \frac{1}{1+e^{-x}} \quad (6)$$

$$P(x) = \frac{e^{-x}}{1+e^{-x}} \quad (7)$$

Data: PrimaryDomain
Result: The value of heuristic "PrimaryDomain"
if *PrimaryDomain is IP* **then**
  $value = 0$; //doubt phishing
else
  $Result = Suggestion\_Google(PrimaryDomain)$;
  if *Result is NULL* **then**
    $value = 1$; //No doubt phishing
  else
    if *Levenshtein(Result, PrimaryDomain)=0* **then**
      $value = 1$; //No doubt phishing
    else
      $value = 1 - (1/Levenshtein(Result, PrimaryDomain))$;
    end
  end
end

Algorithm 1: Calculating the value of "PrimaryDomain"

Data: m //m is SubDomain or PathDomain
Result: The value of heuristic m
if *m is NULL* **then**
  $value = 1$; //No doubt phishing
else
  $Result = Suggestion\_Google(m)$;
  if *Result is NULL* **then**
    $value = 1$; //No doubt phishing
  else
    if *Levenshtein(Result, PrimaryDomain)=0* **then**
      $value = 1$; //No doubt phishing
    else
      $value = 1 - (1/Levenshtein(Result, m))$;
    end
  end
end

Algorithm 2: Calculating the value of "SubDomain/PathDomain"

Data: URL
Result: The value of heuristic PageRank
$value = Google\_PageRank(URL)$;
if $value <= 0$ **then**
  $value = 0$; //doubt phishing
else
  $value = 1 - (1/value)$;
end

Algorithm 3: Calculating the value of "PageRank"

5) Neural Network Training Algorithm: The proposed algorithm is shown in Fig 4. The algorithm performs two phases as follows:

- The *"propagation"* phase calculates the input value of the output node and the output value of the output node. The input value of the output node is calculated by (8)

Data: URL
Result: The value of heuristic BackLink
$value = Google\_BackLink(URL)$;
if $value <= 0$ **then**
| $value = 0$; //doubt phishing
else
| $value = 1 - (1/value)$;
end
Algorithm 4: Calculating the value of "BackLink"

Data: URL
Result: The value of heuristic GoogleIndex
$value = Google\_Index(URL)$;
if $value <= 0$ **then**
| $value = 0$; //doubt phishing
else
| $value = 1 - (1/value)$;
end
Algorithm 5: Calculating the value of "GoogleIndex"

$$O_I = \sum_i W_i * I_i \tag{8}$$

Where O_I, I_i and W_i are the input value of the output node, the value of the ith input node and the weight of the ith input node respectively.
The output value of the output node is calculated by (9)

$$O_O = \frac{1}{1 + e^{-O_I}} \tag{9}$$

Where O_O and O_I are the output value of output node and the input value of output node respectively.

- The *"weight update"* phase calculates the error of the output node and updates the weights. The error of the output node is calculated by (10)

$$Err = O_O * (1 - O_O) * (T - O_O) \tag{10}$$

Where T is the real value of sample in training dataset. The weights are updated by (11)

$$W_i = W_i + R * Err * O_O \tag{11}$$

Where R and W_i are learning rate and the weight of the ith input node respectively.

IV. EVALUATION

We have collected 11,660 phishing sites from PhishTank [1] and 10,000 legitimate sites from DMOZ [16]. The training dataset contains 6,660 phishing sites from PhishTank and 5,000 legitimate sites from DMOZ. We build 2 testing datasets, each of which contains 5,000 phishing sites or 5,000 legitimate sites. Experimental procedure is divided into 2 phases (Training and Testing) through PHP and MYSQL.

A. Training Phase

1) Import Training Dataset: Training dataset is imported into MYSQL. The result is shown in the Fig 5.

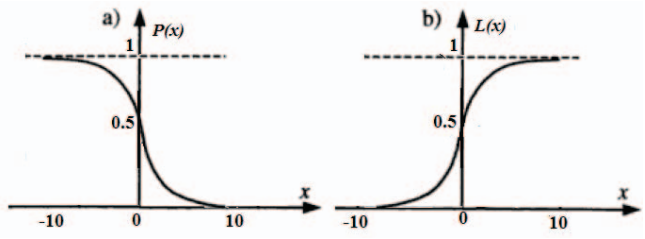

Fig. 3. Graph of membership functions

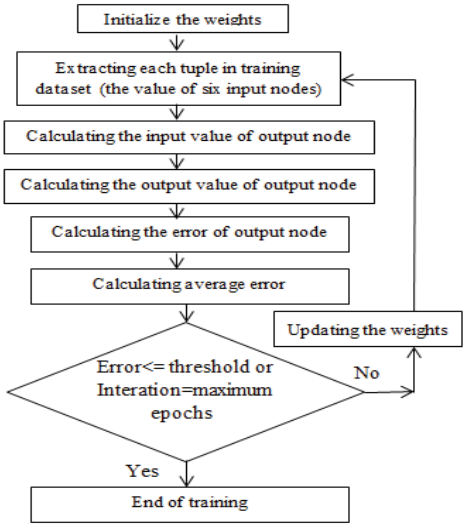

Fig. 4. Neural Network Training Algorithm

2) Extracting four features of URL: Four features (Primary-Domain, SubDomain, PathDomain and Domain) are extracted. Fig 6 shows the obtained result.

3) Calculating the value of six input nodes: Google search engine spelling suggestions and Google API are used to calculate the value of the input nodes. Then, the values are converted into from -10 to 10. The result is shown in the Fig 7.

4) Calculating the fuzzy value of 12 nodes in the second layer : Two membership functions left sigmoid and right sigmoid are used to calculate the value of the nodes in the second layer. The result is shown in the Fig 8

5) Network Training phase: We performed the network training with 9 values of learning rate. In the training phase, the parameters are set as follows:

- Learning rate: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 and 0.9
- Mean error threshold value: 1%
- Number of Epochs: 10,000
- The weights: initialize weights random values from 0 to 1

B. Testing Phase

In this phase, the proposed technique is tested with 2 testing datasets based on the weights of the network training with learning rate of 0.1, 0.2, 0.3, 0.4, 0.5, 0.5, 0.6, 0.7, 0.8,

phish_id	url	phish_detail_url	submission_time	verified	verification_time
2110838	http://www.paypal.com.uk.webapp.filippotteau.be/.d...	http://www.phishtank.com/phish_detail.php?phish_id...	2013-11-17 00:41:35	yes	2013-11-17 13:04:59
2110837	http://klapkasuli.hu/wp-admin/includes/remax	http://www.phishtank.com/phish_detail.php?phish_id...	2013-11-17 00:41:25	yes	2013-11-17 07:14:39
2110836	http://www.umbrellacreative.com.au/wp-content/plug...	http://www.phishtank.com/phish_detail.php?phish_id...	2013-11-17 00:41:16	yes	2013-11-17 13:50:14
2110835	http://www.livingabroadmagazine.co.uk/googledocss/...	http://www.phishtank.com/phish_detail.php?phish_id...	2013-11-17 00:41:10	yes	2013-11-17 13:04:59
2110831	http://www.paypal.com.login.account.eecp.org/5cc46...	http://www.phishtank.com/phish_detail.php?phish_id...	2013-11-17 00:40:26	yes	2013-11-17 13:50:14

Fig. 5. Training dataset of 11,660 sites in MYSQL

phish_id	domain	primarydomain	subdomain	pathname
2110838	filippotteau.be	filippotteau	paypal,com,uk,webapp	.dd
2110837	klapkasuli.hu	klapkasuli		wp,admin,includes,remax
2110836	umbrellacreative.com.au	umbrellacreative		wp,content,plugins,2013gdocs,index.htm
2110835	livingabroadmagazine.co.uk	livingabroadmagazine		googledocss,googledocss,sss
2110831	eecp.org	eecp	paypal,com,login,account	webscr,cmd%3D,account.php

Fig. 6. Four features are extracted

phish_id	PrimaryDomain	SubDomain	PathDomain	PageRank	BackLink	GoogleIndex
2110838	5	10	5	-10	-10	-10
2110837	5	5	10	-10	-10	-10
2110836	7.1	5	5	-10	-10	-10
2110835	5	5	10	-10	-10	-10
2110831	10	10	10	-10	-10	-10

Fig. 7. The value of the input nodes ranges from -10 to 10

phish_id	P1	P2	P3	P4	P5	P6	L1	L2	L3	L4	L5	L6
2110838	0.00699	0.00005	0.00699	0.99995	0.99995	0.99995	0.99331	0.99995	0.99331	0.00005	0.00005	0.00005
2110837	0.00669	0.00669	0.00005	0.99995	0.99995	0.99995	0.99331	0.99331	0.99995	0.00005	0.00005	0.00005
2110836	0.00079	0.00669	0.00669	0.99995	0.99995	0.99995	0.99921	0.99331	0.99331	0.00005	0.00005	0.00005
2110835	0.00669	0.00669	0.00005	0.99995	0.99995	0.99995	0.99331	0.99331	0.99995	0.00005	0.00005	0.00005
2110831	0.00005	0.00005	0.00005	0.99995	0.99995	0.99995	0.99995	0.99995	0.99995	0.00005	0.00005	0.00005

Fig. 8. Fuzzy values in the second layer

0.9. RMSE (Root Mean Square Error) is a good measure of identifying accuracy. RMSE is calculated by (12)

$$RMSE = \sqrt{\frac{\sum(A_i - I_i)^2}{N}} \tag{12}$$

Where I_i is the number of identifying sites, A_i is the number of actual sites and N is the number of samples in the testing dataset. Accuracy ratio is calculated as follows: Accuracy_Ratio = 100 - RMSE. The results of the test with learning rate of 0.1, 0.2, 0.3, 0.4, 0.5, 0.5, 0.6, 0.7, 0.8, 0.9 will be shown in Table I. From the obtained results, RMSE and accuracy are shown in Table II. We have found It shows the best ratio of 99.31% with learning rate of 0.7 and the worst ratio of 98.32% with learning rate of 0.2.

C. Comparing to technique [10]

We experimented with the technique [10] and compared to the result of our proposed technique. First, we collect 10 testing datasets, each of which contains 1,000 phishing sites or 1,000 legitimate sites. Second, we experiment the technique [10] and the results will be shown in Table III. From the obtained result and using RMSE, we have found that the technique [10] with the accuracy of 86.06%.

D. Comparing to technique [11]

We experimented with the technique [11] using 8 hidden nodes and hyperbolic tangent activation function. First, we collect 2 testing datasets, each of which contains 5,000 phishing sites or 5,000 legitimate sites. Second, we experiment the technique [11] and the results will be shown in Table IV. Then, the obtained results of RMSE and accuracy are shown in Table V. By using the technique in [11], we obtained the best accuracy of 94.68%.

V. CONCLUSIONS

We have proposed a new technique to identify phishing sites efficiently. In the technique, the system model is built to identify phishing sites by using neuro-fuzzy network with five layers and six heuristics (PrimaryDomain, SubDomain, PathDomain, PageRank, BackLink, GoogleIndex). The technique is experimented with the training dataset containing 11,660 sites and 2 testing datasets that each dataset contains 5,000 phishing sites or 5,000 legitimate sites. The best accuracy results can obtain 99.31%. We also make a comparison about the accuracy identification with [10], [11], our work shows that it is more efficient and accurate. In the future, the neuro-fuzzy model will be improved to enhance the identification ratio. Besides, the system could be furthermore enhanced by using larger datasets and more heuristic parameters.

TABLE I
RESULT OF TESTING WITH PROPOSED TECHNIQUE

Learning Rate	Testing dataset	A_i	I_i
0.1	No.1	5,000	4,928
0.1	No.2	5,000	4,921
0.2	No.1	5,000	4,914
0.2	No.2	5,000	4,918
0.3	No.1	5,000	4,932
0.3	No.2	5,000	4,935
0.4	No.1	5,000	4,949
0.4	No.2	5,000	4,939
0.5	No.1	5,000	4,938
0.5	No.2	5,000	4,930
0.6	No.1	5,000	4,935
0.6	No.2	5,000	4,935
0.7	No.1	5,000	4,969
0.7	No.2	5,000	4,962
0.8	No.1	5,000	4,917
0.8	No.2	5,000	4,917
0.9	No.1	5,000	4,922
0.9	No.2	5,000	4,917

TABLE II
RMSE AND ACCURACY WITH PROPOSED TECHNIQUE

Learning rate	RMSE	Accuracy
0.1	1.51	98.49%
0.2	1.68	98.32%
0.3	1.33	98.67%
0.4	1.12	98.88%
0.5	1.32	98.68%
0.6	1.30	98.70%
0.7	0.69	99.31%
0.8	1.66	98.34%
0.9	1.61	98.39%

TABLE III
RESULT OF TESTING WITH TECHNIQUE [10]
(1):VERY PHISHY AND PHISHY (2) : VERY LEGITIMATE AND LEGITIMATE (3) : SUSPICIOUS

Testing dataset	(1)	(2)	(3)
No.1	867 (86.7%)	82 (8.2%)	51 (5.1%)
No.2	865 (86.5%)	76 (7.6%)	59 (5.9%)
No.3	847 (84.7%)	90 (9.0%)	63 (6.3%)
No.4	902 (90.2%)	172 (17.2%)	26 (2.6%)
No.5	841 (84.1%)	109 (10.9%)	50 (5.0%)
No.6	64 (6.4%)	873 (87.3%)	63 (6.3%)
No.7	50 (5.0%)	911 (91.1%)	39 (3.9%)
No.8	39 (3.9%)	895 (89.5%)	66 (6.6%)
No.9	97 (9.7%)	871 (87.1%)	32 (3.2%)
No.10	85 (8.5%)	863 (86.3%)	52 (5.2%)

TABLE IV
RESULT OF TESTING WITH TECHNIQUE [11]

Learning Rate	Testing dataset	A_i	I_i
0.1	No.1	5,000	4,612
0.1	No.2	5,000	4,520
0.2	No.1	5,000	4,624
0.2	No.2	5,000	4,478
0.3	No.1	5,000	4,689
0.3	No.2	5,000	4,735
0.4	No.1	5,000	4,456
0.4	No.2	5,000	4,792
0.5	No.1	5,000	4,732
0.5	No.2	5,000	4,736
0.6	No.1	5,000	4,721
0.6	No.2	5,000	4,678
0.7	No.1	5,000	4,599
0.7	No.2	5,000	4,725
0.8	No.1	5,000	4,772
0.8	No.2	5,000	4,697
0.9	No.1	5,000	4,719
0.9	No.2	5,000	4,699

TABLE V
RMSE AND ACCURACY WITH TECHNIQUE [11]

Learing rate	RMSE	Accuracy
0.1	8.73	91.27%
0.2	9.10	90.90%
0.3	5.78	94.22%
0.4	8.24	91.76%
0.5	5.32	94.68%
0.6	6.03	93.97%
0.7	6.88	93.12%
0.8	5.36	94.64%
0.9	5.82	94.18%

REFERENCES

[1] PhishTank. (2013, Nov.) [Online]. Available: http://www.phishtank.com

[2] D. Goodin. (2012) Google bots detect 9,500 new malicious websites every day. [Online]. Available: http://arstechnica.com/security/2012/06/

[3] S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang. (2009) An empirical analysis of phishing blacklists. [Online]. Available: http://ceas.cc/2009/papers/ceas2009-paper-32.pdf

[4] McAfee. (2011, July) Mcafee site advisor. [Online]. Available: http://www.siteadvisor.com

[5] Y. Zhang, J. I. Hong, and L. F. Cranor, "Cantina: a content-based approach to detecting phishing web sites," in The 16th international conference on World Wide Web, 2007, pp. 639—648

[6] G. Xiang, J. Hong, C. P. Rose, and L. Cranor, "Cantina+: a feature-rich machine learning framework for detecting phishing web sites, "ACM Transactions on Information and System Security", vol.14, no.2 .pp. 1—28, Sept. 2011.

[7] M. E. Maurer and D. Herzner, "Using visual website similarity for phishing detection and reporting," in CHI '12 Extended Abstracts on Human Factors in Computing Systems, 2012, pp. 1625—1630.

[8] A. Sunil and A. Sardana, "A pagerank based detection technique for phishing web sites," in IEEE Symposium on Computers & Informatics, 2012, pp. 58—63.

[9] M. G. Alkhozae and O. A. Batarfi, "Phishing websites detected based on phishing characteristic in the webpage source code," in International Journal of Information and Communication Technology Research, vol. 1, no. 6, Oct. 2011, pp. 283—291

[10] M. Aburrous, M. Hossain, F. Thabatah, and K. Dahal, "Intelligent phishing website detection system using fuzzy techniques," in Third International Conference on Information and Communication Technologies: From Theory to Applications, 2008, pp. 1—6.

[11] N. Zhang and Y. Yuan, "Phishing Detection Using Neural Network", CS229 lecture notes, http://cs229.stanford.edu/proj2012/ZhangYuan-PhishingDetectionUsingNeuralNetwork.pdf, 2012

[12] L. A. T. Nguyen , B. L. To , H. K. Nguyen , C. Pham , C. S. Hong, "A novel neuro-fuzzy approach for phishing identification", 2014 International Conference on Control, Automation and Information Sciences (ICCAIS), pp. 188—193, Dec. 2014.

[13] Wikipedia. [Online]. Available (2014) : http://en.wikipedia.org/wiki/Uniformresourcelocator

[14] Levenshtein. [Online]. Available (2014) : http://en.wikipedia.org/wiki/Levenshteindistance

[15] G. Inc. [Online]. Available (2014) : http://toolbarqueries.google.com

[16] DMOZ. [Online]. Available (2014) : http://rdf.dmoz.org/rdf/