IJRESM
International Journal of Research in Engineering, Science and Management
Volume 5, Issue 4, April 2022
https://www.ijresm.com | ISSN (Online): 2581-5792

Phishing URL Detection using Artificial Neural Network

Jay Patel*

UG Student, Department of Information Technology, Birla Vishvakarma Mahavidyalaya, Anand, India

***Abstract*: URL phishing is a developing problem in which fraudsters create fake websites to entice victims into giving up vital information. These bogus websites frequently resemble the actual thing, so looking for telltale signals might help protect you from URL phishing. Organizations can reduce their danger by training users and implementing automatic email-screening measures. We provided a method to categorize URLs as real or phishing URLs in this study. The data was collected and the selective features were extracted from the URLs. We constructed a dataset with a mix of phishing and authentic URLs after extracting features based on three criteria. From a total of 10,000 URLs, we were able to extract 18 features, with 5000 phishing and 5000 genuine URLs. Naley Decision Tree, Random Forest, Support Vector Machine, and Artificial Neural Network were used as machine learning models. ANN has a maximum accuracy of 84.35 percent on these models.**

***Keywords*: Phishing URL, ANN, Decision Tree, Random Forest, SVM.**

1. Introduction

The Phishing URLs are used by cybercriminals to gain delicate information such as passwords, usernames, and banking information for harmful purposes. They deliver phishing emails to their targets, instructing them to submit important information on a false website that appears to be authentic. Phishing sites are created by hackers to acquire private and or qualitative information. Thieves send people email messages in an endeavor to get them to visit the phishing site. When a victim clicks the link to a site and submits the necessary information, the operation is completed. Such URLs are usually posed as real service logins or identification verification. The webpage is also modified so the victim is unaware that it is a scam [8], [9].

Hackers have begun to utilize social engineering strategies to avoid identification and lure consumers into clicking on dangerous websites in recent times. They mix URL hacking with impersonating methods, employ recently registered high-reputation domains, or even use referrals or Url services to capture a genuine company's site for their phishing operation. Ensure our cybersecurity includes URL screening or link prevention. By matching the URLs of sites users want to access to a supports advanced or list of known harmful domains, these techniques will prohibit access to particular URLs. Link security revises these URLs so that they can be examined by our security feature whenever they are clicked, preventing harmful links. Because attackers are altering their tactics to get beyond email access points and spam protection, having a solid spear-phishing system that guards versus phishing URLs are essential. Ai-powered defense can detect and prevent phishing URLs that are unusual or impersonating [12], [14].

We study a proper solution to phishing URL identification in this research, which follows the rise of internet learning. Our approach is based on the result of a strong content inspection-based technique and is mostly lexical. In contrast to earlier work, we developed a new network that predicts the fraud URL accurately and practically tested through the website we created. There it shows whether the given URL was safe or harmful based on the model we proposed. We performed various techniques like SVM, Random Forest, Decision Tree, and Artificial Neural Network [13].

This paper is subdivided into five sections, each with its own set of visuals. In section 2, review and details about the recently published work. Section 3, details a description of the proposed work. In section 4, the results indicate the performance of various algorithms by using websites to show the output. Followed by the conclusion in section 5.

2. Literature Survey

In this, they proposed an artificial neural network to classify Nonphishing and phishing. The focus is mainly on parameters respectively neurons in the hidden layer, a number of hidden layers, momentum value, and learning rate that are used to improve the accuracy to analyze the URLs. The Errors were reduced by using more networks. It shows the results for several numbers of neurons (8,5,4,3,2), but the hidden neurons layers set to 2 yields a superior outcome. To categorize and train the NN, they used 18 features [1].

This paper aims to find the phishing email and find out the potency & organization of the multilayer insight neural network of this proposed method. They evaluated a number of classification methods, including NN, SVM, decision trees, and Naive Bayes, but found that Neural Network had the best recognition rate with 95% precision, indicating that neural networks are the finest at detection of phishing emails [2].

They use a separate artificial neural network, a continuous solution for detecting phishing sites has been developed. In this study, the first part of the technique uses an algorithm to

*Corresponding author: jaypatel15082@gmail.com

compute the worth of six optimistic traits objectively. The neural network was trained on a dataset of 11,660 locations, and two testing datasets were used to ensure accuracy. The best result is that our heuristic technique detects 98.43 percent of fraudulent sites. For a huge dataset, this strategy will not provide a superior outcome [3].

Many researchers have examined how online training can be used in URL-based identification. Ma et al work studying several forms of online classification algorithms in this scenario are particularly pertinent to this paper. Despite the inclusion of lexical features, there was no clear effort to distinguish between lexical and host-based features. We aim to put the possibility of depending solely on features generated from URLs for classifying up for review [4], [5].

In the detection of phishing emails, researchers presented an intelligent model for categorization based on extracting knowledge, data mining methods, and text processing methods. The very first step of the model is feature extraction, after which the score of features is evaluated using the Mutual Information criterion, and only certain features with a mutual information impact are incorporated into the model. A comparison of classification techniques was offered in the paper; the RF classification algorithm obtained the highest accuracy of 99.1 percent, while J48 obtained 98.4 percent [6].

In this research, we present a method for categorizing URLs as either phishing or non-phishing. To enhance the capacity of the artificial neural network, PSO was used to train it to identify URLs. The suggested model was tested with various learning ratios and transfer functions on the number of hidden layers and output layers. The ANN with based optimization method was assessed using RMSE and accuracy criteria. In comparison to BPNN, the ANN PSO model shows the best performance of training accuracy [7].

3. Proposed Method

A phishing site is a frequent sort of social networking that imitates reliable URL and online pages. The objective of this work is to collect data & extract the selective features from the URLs. Due to the obvious open-source service Phish-Tank, collecting phishing URLs is rather simple. This site gives a list of phishing URLs in a variety of forms, including CSV, json, and others. This is updated every hour. I found information that includes a variety of spam, benign, phishing, malware, and obliteration URLs for the legitimate ones. The dataset comes from the University of New Brunswick. The number legitimate URLs in this collection are 35,300. The URL collection is downloaded. This work is uploaded to the Colab for feature extraction. The Figure 1 shows the flow of the proposed model.

Random Forest, Support Vector Machine, Decision Tree, and ANN are among the categorization methods used in this study. Figure 1 depicts the proposed technique. Google Colab Server was used to create this model. ANN is a well-known algorithm for resolving challenging real-world issues. Accounting equations to path planning are examples of possible uses. It is a very simplified version of the human nervous system. An ANN is made up of computing units that are similar to neurons. In general, the ANN model has three layers: input, output, and hidden. A particular node to process data, it employs a non-linear activation method to the input linear combinations. The node output, which is subsequently used as an input by next node in the next tier. From left to right, the signal travels, as well as the ultimate output is derived by repeating the process for all nodes. Obtaining the data associated with all of the edges is the first step in creating this deep neural network [10], [11].

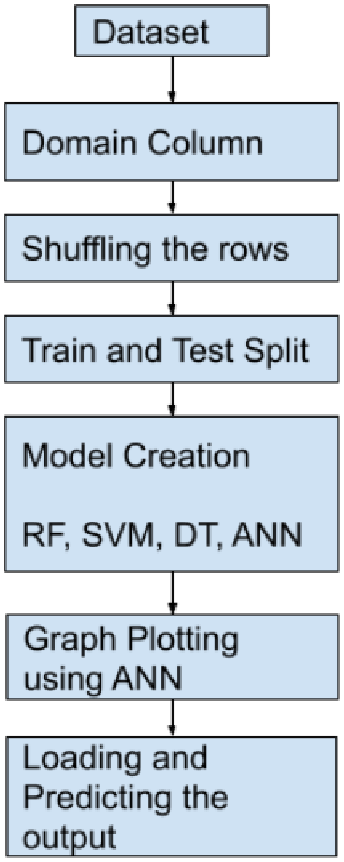


Fig. 1. Block diagram of the proposed model

The phishing URLs are collected from the PhishTank from the reference. The CSV file of phishing URLs is obtained by using wget command. After downloading the dataset, it is loaded into a Dataframe. So, the data has thousands of phishing URLs. But the problem here is, this data gets updated hourly. Without getting into the risk of data imbalance, we considering a margin value of 10,000 phishing URLs & 5000 legitimate URLs. Thereby, picking up 5000 samples from the above data frame randomly. As of now collected 5000 phishing URLs. Now, we need to collect the legitimate URLs.

In the next step, features are extracted from the URLs dataset. The extracted features are categorized into three features based on data respectively Domain-based Features, Address Bar based Features, and HTML & Javascript-based features. [15]

Several data that can be called address strip base attributes can be obtained. The following must be considered for this prototype out of all of them. The URL's domain, IP Address, "@" Symbol, Size of URL, Deep of URL, Reconfiguration "//" in URL, "HTTP/HTTPS" in Web Domain, Utilizing URL Shortening Programs "TinyURL," and Prefix or Sufix "-" in

Domain, we're only taking the domain from the URL in this case. This capability isn't really useful in learning. It's possible that it'll be eliminated although the model is being trained. The existence of an IP address in the URL is checked. An IP address may be something other than a domain name in URLs. If such an IP address is being used instead of a domain name in a URL, one can be certain that the URL is being used to collect sensitive information. The value attached to this characteristic is 1 (Spoofing) if the domains part of the URL contains an IP address, or 0 elsewhere (Legitimate). Furthermore, it examines the URL for the existence of the '@' symbol. When you use the '@' symbol in a URL, the browser ignores anything before the "@" symbol, and the actual address usually comes after the "@" symbol. If the URL contains the '@' symbol, the value given to this characteristic is 1 (phishing) or 0 (legitimate) [16].

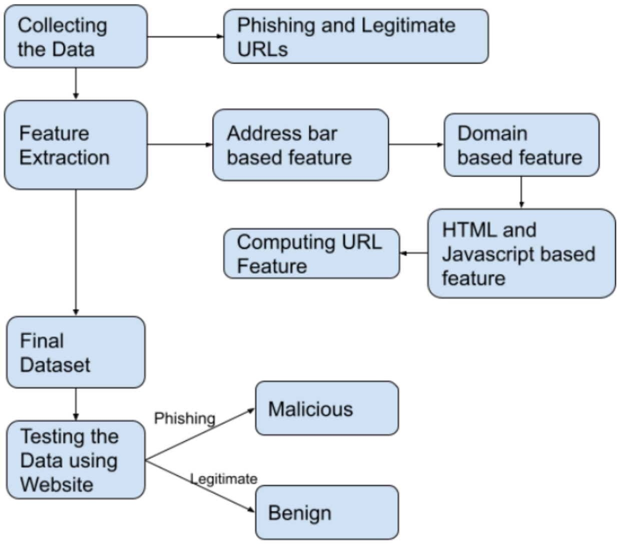


Fig. 2. Detection of flow of malicious and benign

Calculates the URL's length. Phishers can disguise the suspicious element of a URL in the search box by using a lengthy URL. If the URL is longer than or equivalent to 54 bits, it is regarded as a fraud in this application; alternatively, it is considered genuine. If the URL length is greater than 54 characters, the value attached to this attribute is 1 (phishing) or 0 (legitimate). Calculates the URL's depth. Based on the '/', this feature decides the number of sub-pages in the specified URL. The feature's value is number and is determined by the URL. The URL is checked for the presence of the "//" character. If the URL path contains the character "//," redirect the user to some other site. The "//" in a URL's location is calculated. We discovered that if the URL begins with "HTTP," the "//" should be placed in the 6th position. If the URL uses "HTTPS," though, the "//" must be in the 7th position. The value attached to this characteristic is 1 (phishing) if the "//" appears anywhere in the URL other than after the protocol, or 0 otherwise (legitimate).

The existence of "HTTP/HTTPS" in the domain portion of the URL is checked. To deceive users, phishers may append the "HTTPS" tag to the domain section of a URL. URL reduction is a means of reducing the length of a URL while still directing to the desired webpage on the "WWW." This is performed by using an "HTTP Redirect" on a short domain name to a page with a long URL. Checking for the existence of a '-' in the URL's domain part. In genuine URLs, the dash symbol is rarely used. Phishers frequently append prefixes or suffixes to domain names, split by (-), to give the impression that they are interacting with a legitimate website.

This group contains a lot of characteristics that can be retrieved. The following factors were taken into account for this study: DNS records, web traffic, and domain age. In the case of phishing websites, the stated identification is either not recognized by the WHOIS database or no entries for the domain are located. The value attached to this characteristic is 1 (phishing) if the DNS record is blank or not discovered, or 0 otherwise (legitimate). This function determines the visitor numbers and the page number they visit to determine the appeal of the website. However, because phishing websites only exist for a brief time, the Alexa database may not recognise them. By looking at our statistics, we can see that in the worst-case scenarios.

Furthermore, it is categorized as "Phishing" if the site has no activity or is not recognised by the Alexa database. The WHOIS database can be used to extract this characteristic. The majority of phishing websites are only active for a short time. For this project, the age limit of a legal domain is deemed to be 12 months. The gap between the moment of conception and the time of expiry is what we refer to as age. The WHOIS database can be used to extract this characteristic. The available domain calculation is done for this feature by subtracting the end date from the current time. For this project, the end term evaluated for the genuine site is 6 months or fewer. If the domain's end term is greater than 6 months, the quantity of this characteristic is 1 (phishing), otherwise it is 0. (legitimate) [17], [18].

This category contains a lot of characteristics that can be retrieved. The following were taken into account for this study: IFrame redirecting, Toolbar Personalization, Disable Right-Click, and Website Transferring. The IFrame is an HTML tag that allows you to insert another webpage into the one you're now viewing. Phishers can utilise the "iframe" tag to create the frame invisible, i.e., without frame boundaries. Phishers employ the "frame border" feature in this case, which enables the browser to create a visual demarcation. If the screen is vacant or no answer is detected, this feature's value is set to 1 (phishing) or 0 (no response).

Phishers may utilize JavaScript to trick visitors into seeing a false URL in the taskbar. To get this feature, we'll need to delve into the webpage software, specifically the "onMouseOver" event, and see if it affects the status bar. If the reply is blank or onmouseover is discovered, this feature's value is set to 1 (phishing) or 0. (legitimate). Phishers employ JavaScript to block the right-click feature, preventing users from viewing and saving the source code of a webpage. This functionality is handled in the same way as "Hiding the Link with onMouseOver." Nonetheless, we'll look for the event "event.button==2" in the website code and see if the right-click is disabled for this functionality. If the reply is blank or onmouseover is not discovered, this feature's value is set to 1 (phishing) or 0. (legitimate). The amount of times a site has been rerouted is a narrow line that separates phishing sites from authentic ones. We discovered that authentic websites were only routed once in our sample. Phishing sites with this functionality, on either hand, have been directed at least four times.

4. Result and Discussion

Computing URL Features and creating a list and a function that calls the other functions and stores all the features of the URL in the list. We will extract the features of each URL and append them to this list. Firstly, feature extraction is done on legitimate URLs. In that extraction of features and storing them in a list and converting the list to a data frame. Then it stores the extracted legitimate URLs features in a CSV file. Secondly, feature extraction is performed on phishing URLs. The extraction of features & storing them in a list and converting the list into a data frame. Then it stores the extracted legitimate URLs features in a CSV file. In the above process, we formed two data frames of legitimate & phishing URL features. Now, we will combine them into a single data frame and export the data to a CSV file for the machine learning training done in a separate notebook.

The experimental results are described as follows. During the study, authors have carried outset of approach on different data and experimented with Decision Tree, Random Forest, Support vector machine, and Artificial neural network. By using count plot it calculates the number of output labels present in the data shown in Figure3. The creation of algorithms with their performance is mentioned below in table1. Among all the models' ANN achieved the highest accuracy 84.35%.

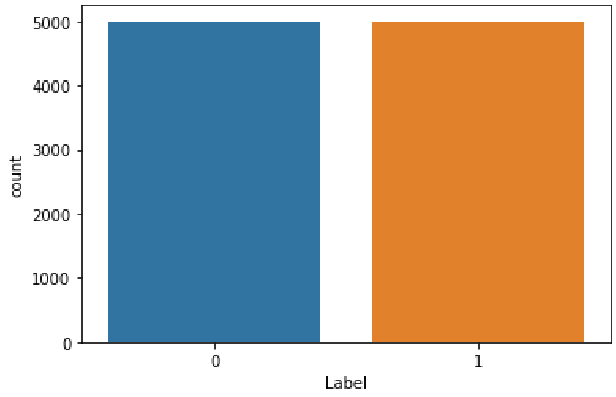


Fig. 3. Label count plot

Table 1
Proposed model performance

Model	Accuracy	F1 Score	Precision	Recall
Decision tree	0.8085	0.777	0.9751	0.6460
Random Forest	0.8135	0.7822	0.9867	0.6479
Support Vector Machine	0.796	0.7591	0.9742	0.6218
Artificial Neural Network	0.8435	0.8314	0.9380	0.7466

The generation of an ANN model is based on the construction of a structure. First, I built two layers to the model using a sequential function from the Keras layer dense with input 16 and the activation 'relu' function. However, once the layers have been added to the model, the model must be compiled using the Adam optimizer, binary cross-entropy loss, and accuracy metrics. The first ten observations are predicted using the model that has been tested. The confusion matrix was created using the test and precision of y split data and displayed in Figure 4 as a heatmap. Figure 5 shows an overview of the history of accuracy based on epoch with accuracy. Figure 6 shows an overview of Loss's history plotted based on loss vs. epochs.

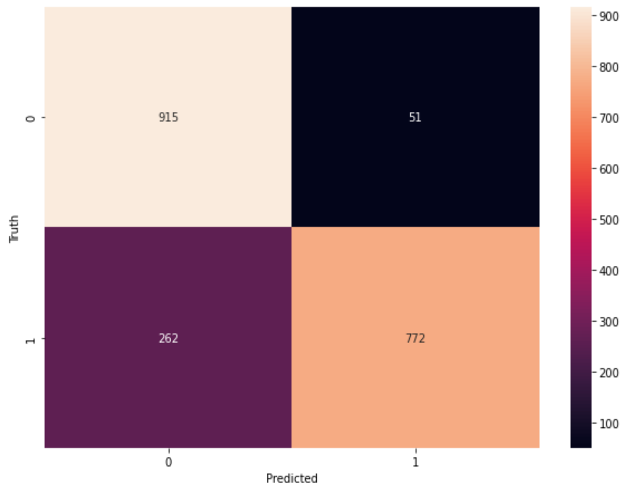


Fig. 4. Heatmap of confusion matrix on ANN model

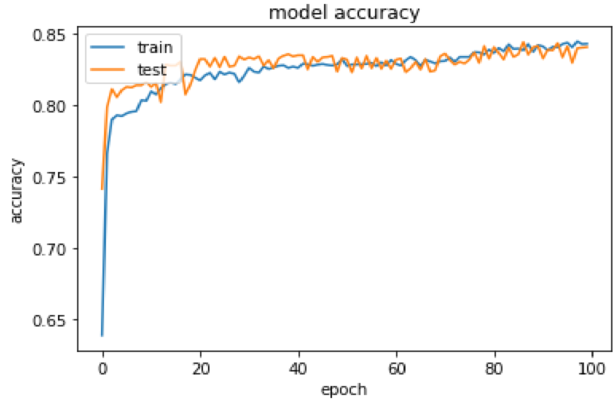


Fig. 5. Model accuracy (Acc vs. Epoch)

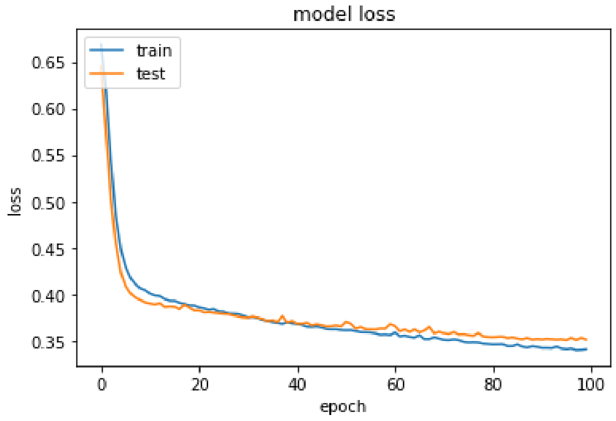
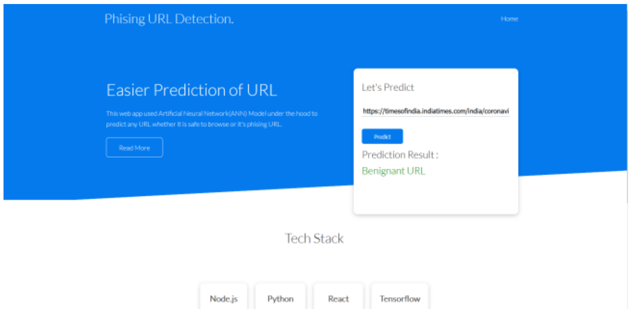


Fig. 6. Model loss (Loss vs. Epoch)

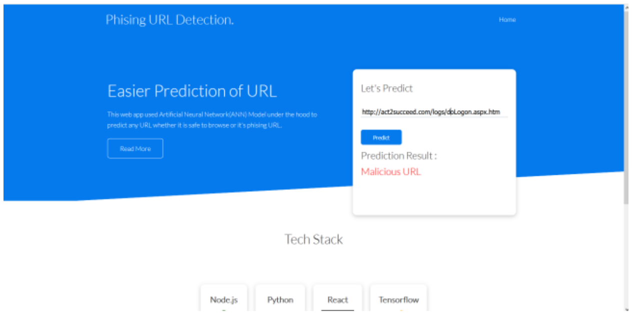


This web app used Artificial Neural Network(ANN) Model under the hood to predict any URL whether it is safe to browse or it's phising URL.

https://timesofindia.indiatimes.com/india/coronavi

Prediction Result : Benignant URL

Fig. 7. Benignant URL detection



This web app used Artificial Neural Network(ANN) Model under the hood to predict any URL whether it is safe to browse or it's phising URL.

http://act2succeed.com/logs/dpLogon.aspx.htm

Prediction Result : Malicious URL

Fig. 8. Malicious URL detection

5. Conclusion

The Phishing URLs are the subject of research. This is used to determine whether a URL is authentic or not. The prediction of URL phishing has been the subject of several learnings. Simply put, it means efficiently detecting it and improving people's security and company privacy. When using machine learning algorithms to the selection of URL features, the goal of this research is to study and compare the performance of various categorization algorithms using classification statistics. The goal of this notebook has been accomplished. Finally, 18 features were retrieved from 10,000 URLs, with 5000 phishing and 5000 authentic URLs.

References

[1] R. M. Mohammad, F. Thabtah, and L. M. Cluskey, “Predicting phishing websites based on self-structuring neural network”, Neural Computing and Applications, vol. 25, pp. 443-458, 2014.

[2] N. Zhang and Y. Yuan, “Phishing detection using neural network,” CS229 lecture notes.

[3] L. A. T. Nguyen, B. L. To, H. K. Nguyen and M. H. Nguyen “An efficient approach for phishing detection using single-layer neural network,” in Advanced Technologies for Communications (ATC), pp. 435-440, 2014.

[4] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, Beyond blacklists: Learning to detect malicious websites from suspicious urls. In Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 1245–1253, Paris, France, June–July 2009.

[5] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Identifying suspicious urls: An application of large-scale online learning. In Proceedings of the 26th International Conference on Machine Learning (ICML-2009), pages 681–688, Montreal, Quebec, Canada, 2009.

[6] G A. Yasin and A. Abuhasan “An intelligent classification model for phishing email detection,” vol. 8, pp. 55-72, 2016.

[7] Gupta, S., & Singhal, A. (2017, August). Phishing URL detection by using artificial neural network with PSO. In 2017 2nd International Conference on Telecommunication and Networks (TEL-NET) (pp. 1-6). IEEE.

[8] Jeswal, S. K., & Chakraverty, S. (2021). Fuzzy eigenvalue problems of structural dynamics using ANN. In New Paradigms in Computational Modeling and Its Applications (pp. 145-161). Academic Press.

[9] J. L. Lee, D. H. Kim and L. Chang-Hoon “Heuristic-based Approach for Phishing Site Detection Using URL Features.” 2015.

[10] C. J. Chandan, H. P. Chhed, D. M. Gosar and H. R. Shah “A Machine Learning Approach for Detection of Phished Websites Using Neural Networks” International Journal on Recent and Innovation Trends in Computing and Communication, vol. 2, Issue 12, pp 4205-4209, 2014.

[11] A. Almomani, T. C. Wan, A. Altaher, A. Manasrah,and E. ALmomani, M. Anbar and S. Ramadass, “Evolving fuzzy neural network for phishing emails detection” Journal of Computer Science, vol. 8, pp. 1099-1107, 2012.

[12] L. A. T. Nguyen, B. L. To, H. K. Nguyen and M. H. Nguyen “An efficient approach for phishing detection using single-layer neural network.”In Advanced Technologies for Communications (ATC), pp. 435-440, 2014.

[13] R. M. Mohammad, F. Thabtah, and L. M. Cluskey, “Predicting phishing websites based on self-structuring neural network” Journal of Neural Computing and Applications vol. 3, pp. 1-16, 2014.

[14] N. Zhang and Y. Yuan, “Phishing detection using neural network,” CS229 lecture notes.

[15] N. G. M. Jameel and L. E. George “Detection of phishing emails using feed forward neural network” International Journal of Computer Applications, vol. 77 issue 7, 2013.

[16] Y. Li, R. Xiao, J. Feng and L. Zhao (2013) “A semi-supervised learning approach for detection of phishing WebPages” Optik-International Journal for Light and Electron Optics, vol. 124, issue 23, pp. 6027-6033, 2013.

[17] X. Zheng, Z. Zeng, Z. chen, Y. Yu and C. Rong, “Detecting spammers on social networks,” Neurocomputing, vol. 159, pp. 27-34, 2015.

[18] M. Moghimi and A. Y. Varja “New rule-based phishing detection method” Expert Systems with Applications, vol. 53, pp. 231–242, 2016.