INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING & COMPUTER SYSTEMS (IJSECS)
ISSN: 2289-8522 e-ISSN: 2180-0650
VOL. 7, ISSUE 2, 1 – 8
DOI: https://doi.org/10.15282/ijsecs.7.2.2021.1.0084

ORIGINAL ARTICLE

Model for Phishing Websites Classification Using Artificial Neural Network

N.H. Hassan[1,*], A.S. Fakharudin[1]

[1]Faculty of Computing, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia.

ABSTRACT – Internet users might be exposed to various forms of threats that can create economic harm, identity fraud, and lack of faith in e-commerce and online banking by consumers as the internet has become a necessary part of everyday activities. Phishing can be regarded as a type of web extortions described as the skill of imitating an honest company's website aimed at obtaining private information for example usernames, passwords, and bank information. The accuracy of classification is very significant in order to produce high accuracy results and least error rate in classification of phishing websites. The objective of this research is to model a suitable neural network classifier and then use the model to class the phishing website data set and evaluate the performance of the classifier. This research will use a phishing website data set which was retrieved from UCI repository and will be experimented using Encog Workbench tool. The main expected outcome from this study is the preliminary ANN classifier which classifies the target class of the phishing websites data set accurately, either phishy, suspicious or legitimate ones. The results indicate that ANN (9-5-1) model outperforms other models by achieving the highest accuracy and the least MSE value which is 0.04745.

ARTICLE HISTORY
Received: 3 Dec 2020
Revised: 14 Jan 2021
Accepted: 26 April 2021

KEYWORDS
Artificial Neural Network
Classification
Phishing Websites

INTRODUCTION

Data Mining is defined as extracting information from huge sets of data. It is a process for extracting and converting information from a data set into a comprehensible structure. It is a mathematical method of identifying patterns involving techniques at the intersection of artificial intelligence, machine learning, statistics, and database systems in broad data sets [1]. It is also one of the most inspiring study fields to find valuable data from huge data sets. The information or knowledge extracted can be used for market analysis, fraud detection, customer retention, and production control. The examples of techniques in data mining are classification, regression, association rule, cluster analysis, text mining, and link analysis [2]. In data mining, classification is a main method and commonly used in different fields. Therefore, this research will be focused on classification techniques.

Classification is a supervised learning technique that assigns objects in a group to target classes [3]. It is a data mining role to allocate objects in a group to target classes or categories. The objective of classification is to correctly classify the target class in the data for each case. As an example, to identify loan applicants as having low, medium or high credit risks, a classification model may be used [4]. Binary classification is the simplest form of classification problem. The target attribute has only two possible values in the binary classification, such as a high credit rating or a low credit rating [5]. Multiclass targets have more than two values for example, low, medium, high, or unknown credit rating. The methods widely used for classification are statistical, discriminant analysis, decision tree, Markov based, swarm intelligence, k-nearest neighbor, genetic classifiers, artificial neural network, support vector and association rule [6].

Artificial neural network (ANN) is a mathematical model based on biological neural networks [7]. It is constructed from basic components known as neurons that take a real value, multiply it by a weight, and run it through a function of non-linear activation. The network can learn very complex functions through building multiple layers of neurons, which obtains part of the input variables, and then transfers the output to the next layers. Theoretically, a neural network able to learn the shape of any function, provided sufficient computational power [8]. One of the advantages of neural networks is high tolerance to noisy data. In order to handle high dimensional data, it is simple and convenient to implement and requires some knowledge or parameter settings. It can easily interpret the findings obtained from this algorithm. Another benefit of using ANN is that it produces probability based output so that the accuracy will be high when bigger data volumes are given as input [9]. Hence, this research will focus on classification using an artificial neural network algorithm.

According to phishing activity trends report 3rd quarter 2020 by Anti-Phishing Working Group (APWG), the number of phishing attacks has grown since March 2020 globally. Number of unique phishing websites detected in July 2020 is 171040, in August 2020 is 201591 and in September 2020 is 199133 [10]. From this data, we can see the number of phishing websites increase from July to August but slightly decrease in September but it is still worrying. In terms of cyber threats, 2020 is a year that has been record breaking. These attacks have resulted in the loss of millions of dollars at a global level, negatively impacting many well-known organisations around the world [11]. For this reason, it is

*CORRESPONDING AUTHOR | N.H. Hassan | ✉ noorhazirahassan@gmail.com

becoming increasingly vital to address phishing websites classification problems for keeping individual and corporate data safe.

While several approaches and methods have been introduced, the phishers are still able to resolve countermeasures that have been applied. Usually, a blacklist-based method is used to classify phishing websites. Blacklist-based detection method is where the requested uniform resource locator (URL) is compared with a list of predefined phishing URLs. One of the disadvantages of this method is it usually fails to discover all phishing sites as a recently created forged website takes a significant time before it can be added to the list. The blacklist-based approach is also inefficient in reacting to emanating phishing attacks as it has become easier to register fresh domain names; no robust blacklist will guarantee a perfect up-to-date database. There is still the opportunity for "Zero Day Attacks". Therefore, it is more desirable to use web page features via machine learning techniques, since this technique does not have the above described blacklist approach problems and does not rely on any human databases. Furthermore, the classification technique for machine learning can work with consistent accuracy[12]. In order to avoid the drawbacks from the blacklist-based method, we decided to use ANN to classify phishing websites as ANN has many advantages that were discussed in the previous paragraph.

This research will use a phishing websites dataset obtained from University of California, Irvine (UCI) machine learning repository. Phishing websites dataset contains 10 attributes and 1352 instances will be used in the experiment. Neural network algorithms will be used to classify the phishing websites dataset into a correct class either phishy, suspicious or legitimate ones. These experiments will take place in Encog Workbench tools and mean square error (MSE) will be used to evaluate the neural network performance.

The rest of this research paper is organized as follows. Related work section discusses some existing work in this domain while methodology section comprises the methodology used in this study. Experiments and preliminary results section presents the discussion and comparison of results achieved followed by the conclusion section which concludes the study and presents directions for future work.

RELATED WORK

Research indicates that there are a number of approaches to classify phishing websites. One study was done by [13] investigated the issue of phishing websites to seek its applicability to the phishing issue by using an established AC method known as Multi-label Classifier based Associative Classification (MCAC). The authors would like to determine characteristics that can differentiate between phishing websites and the genuine ones. The findings by employing real data gathered from various sources indicated that MCAC was able to identify phishing websites with greater accuracy than other intelligent algorithms. It also produces new rules that cannot be discovered by other algorithms, and this has enhanced the efficiency of its classifiers. The classification accuracy of MCAC is around 94% compared to Multiclass Classification based Association Rules (MCAR) and Classification Based on Associations (CBA) which are around 92% each and PART is around 90%.

A study presented by [14] suggested Fast Associative Classification Algorithm (FACA) which is a new Associative classification (AC) algorithm. The authors examined the suggested algorithm against four well-known AC algorithms (CBA, CMAR, MCAR, and ECAR) on real world phishing datasets. This study revealed that FACA outperforms the other four AC algorithms in classification accuracy and F1 evaluation measure.

Another study was done by [15] proposed a novel classification model consisting of website URL and content features to identify Chinese phishing e-Business websites automatically. The model contains a number of special domain-specific characteristics of Chinese websites for e-Business. The authors applied around three thousands of Chinese e-Business websites and 4 approaches for classification to assess the proposed model. This study showed that the Sequential Minimal Optimization (SMO) algorithm accomplishes the best with accuracy of 95.83% compared to Naïve Bayes which is 92.94% and Random Forest which is 93.75%. The outcomes of a sensitivity analysis confirmed that domain-specific features are the greatest major effect on the identification of Chinese e-Business phishing websites.

A new rule-based approach has been proposed by [16] for detecting phishing attacks in internet banking. The authors used a support vector machine (SVM) algorithm in order to classify web pages. The experiments showed that the proposed model can detect phishing web pages with accuracy of 99.14% true positive and 0.86% false negative. Sensitivity analysis performance indicated the important influence of their proposed features over conventional features. They inserted the extracted rules into a browser extension called PhishDetector to make the proposed method more practical and easier to use.

Next, an intelligent model was suggested by [17] to predict phishing attacks using Artificial Neural Network (ANN) particularly self-structuring neural networks. In order to handle the issue where important features in defining the category of websites are frequently evolving, the network structure needs to be continuously improved. By automating the process of structuring the network, the proposed model resolves this issue and demonstrates high accuracy of prediction and high acknowledgment for fault tolerance and noisy data. The result showed the training set accuracy is 91.32% for the experiment with the epochs number is 50 and optimal number of hidden neurons are four. Meanwhile, the experiment with epochs number 1000 and three optimal numbers of hidden neurons resulted in 94.07% accuracy.

Although there were many studies done to solve phishing websites classification issues, there is still a space for a lot of improvements in this view. Therefore, classification using an artificial neural network will be implemented in this research.

METHODOLOGY

In this section, we will describe our approach which is an artificial neural network to classify phishing websites. The process is summarized in Figure 1. This research proposes the following process specified in Figure 1 to classify the phishing website using ANN algorithm, which includes data collection, normalization, neural network design, data training, validation and data testing. The proposed work is implemented using Encog Workbench 3.3.0. Encog Workbench is a GUI application that was introduced back in 2008, provided to help model and train neural networks efficiently [18]. It visually works with neural networks and other machine learning methods such as Support Vector Machines, Neural Networks, Bayesian Networks, Hidden Markov Models, Genetic Programming and Genetic Algorithms, as well as support classes to normalize and process data. The workbench is a Java application that produces data that it works across any Encog platforms [19].

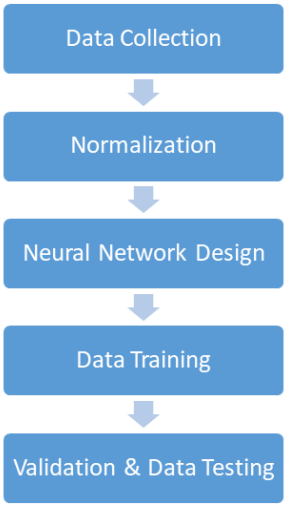

Figure 1. Steps in Artificial Neural Network algorithm.

Figure 1 shows all the five steps that take part in ANN algorithm. Datasets of Phishing Websites were collected from UCI repository. There are 10 attributes in which the last attribute is the class. Remaining 9 attributes namely Server Form Handler (SFH), using pop-up window, SSL final state, request URL, URL of anchor, website traffic, URL length, age of domain and IP address.

The dataset has a classification attribute with multiple classes -1, 0 and 1. Some of the attributes hold categorical values such as “Legitimate”, “Suspicious” and “Phishy”. These values have been replaced with the numerical values 1, 0 and -1 respectively.

Normalization is a process of reducing redundancies of data in a database. It is a process or set of guidelines used to optimally design a database to reduce redundant data. In this research, the data already go through the normalization process. In this step, data pre-processing occurred along with splitting the data to 70% for training and 30% for testing purposes.

There are three different layers in the neural network which are input layer, hidden layer and output layer. All the inputs are fed via the input layer into the model. The inputs are passed to the hidden layers. A few self-reliant variables that have an effect on the performance of the neural network should be represented by each input neuron.

The hidden layer is the group of neurons that has an activation function added to it and is a middle layer between the layer of input and the layer of output. The task of the hidden layer is to process the inputs received by the previous layer. Therefore, the layer is accountable for extracting the required features from the input data [20]. There can be more than one hidden layer in a neural network. The difficulty of determining the hidden layer structure that optimally learns the problem is to avoid creating a hidden structure that is either too complicated or too simple. It would take too long to train if the hidden layer structure is too complex. It will not learn the problem if the hidden layer structure is too simple [19].

The neural network output layer gathers and transfers the information appropriately in the manner it has been intended to provide. It is able to straight away trace the pattern given by the output layer back to the input layer [21]. The data after processing is made accessible at the output layer.

Encog Workbench 3.3.0 is used to design the neural network model. It has been decided to start the experiment using neural network design of (9-5-1) which means 9 input neurons, 5 hidden neurons, and 1 output neuron as shown in Figure 2. Another ANN architecture used in this research are (9-5-5-1) as shown in Figure 3 while (9-10-5-1) neural network design as shown in Figure 4. The second and third architecture have 2 hidden layers. This is a preliminary run to find the best model for classification of phishing websites. The purpose of the hidden layers is to allow the neural network to better produce the expected output for the given input [19]. The reason why we use different numbers of hidden layers and neurons for each neural network design is to compare the result of MSE between them.

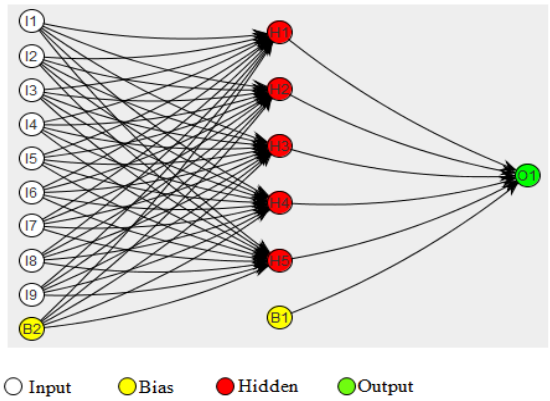

Figure 2. Network structure of ANN 9-5-1.

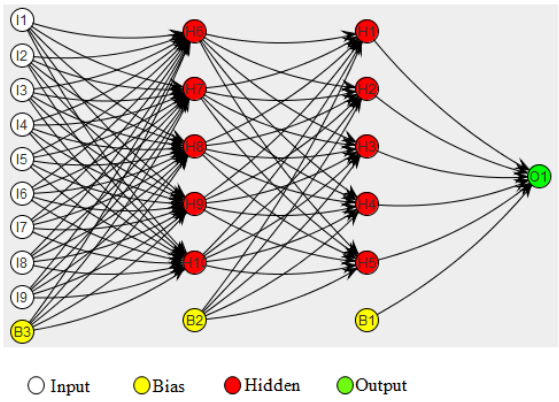

Figure 3. Network structure of ANN 9-5-5-1.

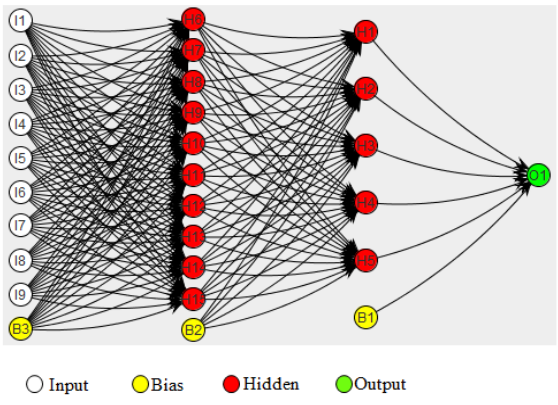

Figure 4. Network structure of ANN 9-10-5-1.

Training a neural network is a procedure by which the weights and biases' values are identified. This process is carried out by train-test technique in most cases. The data set is divided into a training and test data set with 70% and 30% of the data respectively.

The data set for training is used to train the neural network. In order to find the set of values, different weights and biases' values are tested so that the calculated output values most closely match the correct output values. Training is also the method of finding values for the weights and biases to reduced error.

The data set for testing is not used at all in the training stage. The accuracy of the weights and biases of the resulting neural network model is used on the test data only once after the training process is completed. It gives an irregular approximation of how precise the model will be when new and formerly unknown data is presented [22].

A data set for testing is a data set that is used to provide a fair assessment of a final model that matches the training dataset. Test data set is a data set that is independent of the data set for training, but that follows the same distribution of probability as the dataset for training. If a model also suits the test dataset well with the training dataset, minimal overfitting has occurred. As opposed to the test data set, a better fitting of the training data set typically points to

overfitting. Therefore, a test set is a set of examples that are only used to evaluate output, which is the generalization of a completely defined classifier.

The test data set is run as usual, which is used on the test data once the value of weights and bias of the model have already been decided, and the resulting accuracy is an approximation of the neural network model's overall accuracy. Then, the neural network performance will be evaluated using mean squared error (MSE). The details of MSE will be discussed in the next section.

EXPERIMENTS AND PRELIMINARY RESULTS

A preliminary training was done to find the best network architecture with the lowest error [23]. The experiments implemented a feed forward neural network along with Levenberg Marquardt training method in Encog Workbench. The Levenberg Marquardt (LM) training algorithm is one of the fastest training algorithms available for Encog and a supervised training method. It is based on the LevenbergMarquardt method for minimizing a function. This training algorithm can only be used for neural networks that contain a single output neuron. Maximum error percent used in this training algorithm was 1.0.

In order to intelligently figure out the complicated interconnections of neurons, ANN has an input layer, a hidden layer and an output layer. The artificial neural network's fundamental units are neurons. One or more inputs are obtained and summed up through an activation function to produce an output. In this research study, the hyperbolic tangent activation function was used as a range of output values (-1, 1) in the hidden layer, which is fit for this data set. A number of inputs and the targeted output for training are accepted by the network. There are two steps which are training and testing executed by the network [24]. The network performance was evaluated by an error indicator, known as the mean squared error (MSE), that can be determined by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \tag{1}$$

where y_i denotes the *i* th target value, and x_i denotes the *i* th output value. The closer the value of MSE to zero, the better the performance of the neural network.

There are three neural network architectures designed in this experiment as mentioned in methodology's section. The training process was repeated 3 times for each neural network architecture and all the training and testing results will be collected. In total, 3 models were produced by each training algorithm and a total of 9 models from a phishing dataset. Table 1 shows all the collected results for MSE.

Table 1. MSE for ANN architectures.

ANN Architecture	MSE					
	Training			Testing		
	1st	2nd	3rd	1st	2nd	3rd
9-5-1	0.204774	0.218691	0.383222	0.275076	0.277105	0.396843
9-5-5-1	0.153556	0.2185	0.159004	0.344001	0.364929	0.361756
9-10-5-1	0.082981	0.122026	0.120996	0.421854	0.369432	0.265928

From results in Table 1, we plotted three graphs which are Figure 5 for ANN of (9-5-1), Figure 6 for ANN of (9-5-5-1) and Figure 7 for ANN of (9-10-5-1).

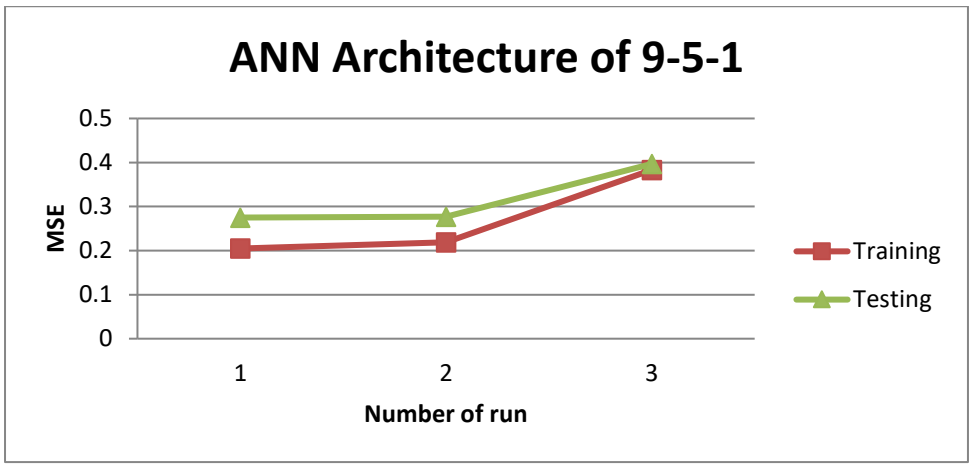

Figure 5. MSEs of ANN 9-5-1.

Figure 5 shows the MSE results for ANN architecture of (9-5-1) for the training and testing set. The training set had a high MSE compared to Figure 6 and Figure 7 with average training MSE of (9-5-1) model was 0.268896. The testing set had the lowest MSE compared to Figure 6 and Figure 7 with average testing MSE was 0.316341.

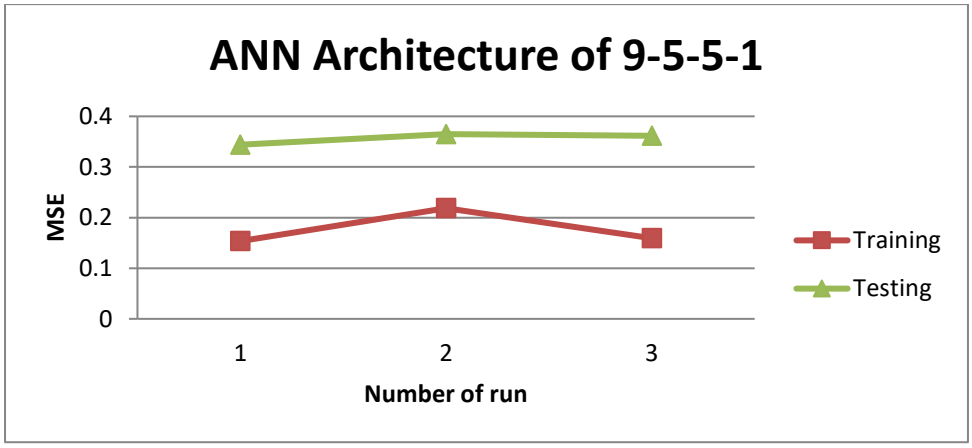

Figure 6. MSEs of ANN 9-5-5-1.

The training set in Figure 6 had a lower MSE compared to Figure 5 with average training MSE was 0.17702. The testing set had a high MSE compared to Figure 6 and Figure 7 with average testing MSE was 0.356895.

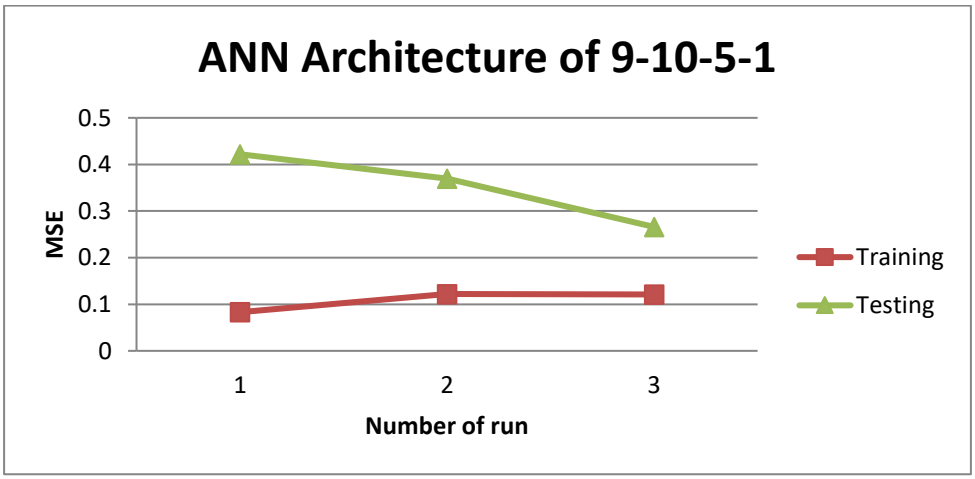

Figure 7. MSEs of ANN 9-10-5-1.

The training set in Figure 7 had the lowest MSE compared to Figure 5 and Figure 6 with average training MSE was 0.108668. The testing set had a lower MSE than Figure 6 with average testing MSE was 0.352405.

The best ANN architecture's output is measured by the lowest MSE. It indicates a lower deviation between the output and the target data if the network error indicator values are small. At the end of the training process, a well-trained ANN needs to have a very small MSE. The definition of having very small MSE, which is the value of MSE close to zero, is that the target outputs and the outputs of ANN for the training set have become very close to each other.

Table 2. MSE difference.

ANN Architecture	Average Training MSE	Average Testing MSE	Average difference between training and testing MSE
9-5-1	0.268896	0.316341	-0.04745
9-5-5-1	0.17702	0.356895	-0.17988
9-10-5-1	0.108668	0.352405	-0.24374

The average MSE difference between training and testing for the LM algorithm can be referred to Table 2. The MSE difference between training and testing for (9-10-5-1) architecture was the highest with 0.24374. This network architecture generally yielded right output when the training set was being inputted into the network. However, the ability to predict correct output from unknown input was less accurate and far from the intended target. (9-5-1) had the smallest difference between training and testing MSE with 0.04745 compared to other architectures in this study. Therefore, training set output and testing set output produced ANN's output much similar to the target output. Thus, (9-5-1) model yielded the highest classification accuracy as having the lowest MSE value amongst the others. This model had significant results to classify phishing websites into a correct class.

CONCLUSION

Artificial neural network algorithm was used in this paper for the phishing websites classification problem. This study shows that classification is done on the basis of some features which demonstrate the characteristics of phishing websites. It is seen from the preliminary result that ANN (9-5-1) model achieved the best results with the average training MSE of 0.268896 and average testing MSE of 0.316341 as it produced the smallest difference between training and testing MSE which is 0.04745. This result shows that ANN (9-5-1) model can classify phishing websites with the highest accuracy compared to (9-5-5-1) and (9-10-5-1) models. This is preliminary results for comparing different ANN model architecture. The work can be extended by testing the same algorithm on other larger instances than before to see whether it is still suitable to be used for a larger dataset. Further studies will refine the ANN model with additional implementation of deep learning to improve the classification.

REFERENCES

[1] B. Makhija, "'REVIEW ON CLASSIFICATION TECHNIQUES IN DATA MINING,'" *Int. J. Res. Comput. Inf. Technol.*, vol. 1, 2016.

[2] V. Paramasivam, T. Sing, S. K. Dhillon, and A. S. Sidhu, "ScienceDirect A methodological review of data mining techniques in predictive medicine : An application in hemodynamic prediction for abdominal aortic aneurysm disease," *Integr. Med. Res.*, vol. 34, no. 3, pp. 139–145, 2014.

[3] V. Kunwar, K. Chandel, A. S. Sabitha, and A. Bansal, "Chronic Kidney Disease Analysis Using Data Mining Classification," *Cloud Syst. Big Data Eng. (Confluence), 2016 6th Int. Conf. IEEE*, pp. 300–305, 2016.

[4] N. Satyanarayana, C. H. Ramalingaswamy, and Y. Ramadevi, "Survey of Classification Techniques in Data Mining," 2014.

[5] K. L. Taylor, *Oracle Data Mining Concepts*. 2010.

[6] N. Jothi, N. Aini, A. Rashid, and W. Husain, "Data Mining in Healthcare – A Review," *Procedia - Procedia Comput. Sci.*, vol. 72, pp. 306–313, 2015.

[7] V. Krishnaiah, G. Narsimha, and N. S. Chandra, "Diagnosis of Lung Cancer Prediction System Using Data Mining Classification Techniques," *Int. J. Comput. Sci. Inf. Technol.*, vol. 4, no. 1, pp. 39–45, 2013.

[8] MissingLink.ai, "Classification with Neural Networks: Is it the Right Choice? - MissingLink.ai," 2016. [Online]. Available: https://missinglink.ai/guides/neural-network-concepts/classification-neural-networks-neural-network-right-choice/. [Accessed: 18-Nov-2020].

[9] K. Balasaravanan and M. Prakash, "Detection of dengue disease using artificial neural network based classification technique," vol. 7, pp. 13–15, 2018.

[10] Anti-Phishing Working Group, "Phishing Activity Trends Report 3rd Quarter 2020," *Apwg*, no. November, pp. 1–12, 2020.

[11] D. Meharchandani, "Staggering Phishing Statistics in 2020 - Security Boulevard," 2020. [Online]. Available: https://securityboulevard.com/2020/12/staggering-phishing-statistics-in-2020/. [Accessed: 22-Jan-2021].

[12] A. Kalybayev, "COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS IN WEBSITE PHISHING DETECTION," 2013.

[13] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based Associative Classification data mining," vol. 41, no. 13, pp. 5948–5959, 2014.

[14] W. Hadi, F. Aburub, and S. Alhawari, "A new fast associative classification algorithm for detecting phishing websites," *Appl. Soft Comput. J.*, vol. 48, pp. 729–734, 2016.

[15] D. Zhang, Z. Yan, H. Jiang, and T. Kim, "A domain-feature enhanced classification model for the detection of Chinese phishing e-Business websites," *Inf. Manag.*, vol. 51, no. 7, pp. 845–853, 2014.

[16] M. Moghimi and A. Y. Varjani, "New rule-based phishing detection method," *Expert Syst. Appl.*, vol. 53, pp. 231–242, 2016.
[17] F. Abdeljaber, R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting Phishing Websites based on Self-Structuring Neural Network," 2014.
[18] J. Heaton, "Encog Machine Learning Framework | Heaton Research." [Online]. Available: https://www.heatonresearch.com/encog/. [Accessed: 21-Jan-2021].
[19] J. Heaton, *Programming Neural Networks with Encog3 in Java*. 2011.
[20] A. C. del Castillo, *Digital Twin Systems Modelling to Improve Real Time Assets Operation and Maintenance. 2*. 2018.
[21] S. Saxena, "Artificial Neuron Networks(Basics) | Introduction to Neural Networks | by Shubh Saxena | Becoming Human: Artificial Intelligence Magazine," 2017. [Online]. Available: https://becominghuman.ai/artificial-neuron-networks-basics-introduction-to-neural-networks-3082f1dcca8c. [Accessed: 20-Nov-2020].
[22] J. McCaffrey, "Neural Network Train-Validate-Test Stopping -- Visual Studio Magazine," 13-May-2015. [Online]. Available: https://visualstudiomagazine.com/Articles/2015/05/01/Train-Validate-Test-Stopping.aspx?Page=1. [Accessed: 17-Nov-2020].
[23] A. S. Fakharudin, N. Sulaiman, and N. Mustapha, "Artificial Neural Network Modelling of Biogas Production Processes," 2017.
[24] K. Chun, S. King, P. Chiong, and K. Ho, "Modeling electrostatic separation process using artificial neural network (ANN)," *Procedia - Procedia Comput. Sci.*, vol. 91, pp. 372–381, 2016.