Web Phishing Classification Model using Artificial Neural Network and Deep Learning Neural Network

Noor Hazirah Hassan, Abdul Sahli Fakharudin
Faculty of Computing, Universiti Malaysia Pahang, Pekan, Pahang, Malaysia

***Abstract*—Phishing is an online crime in which a cybercriminal tries to persuade internet users to reveal important and sensitive personal information, such as bank account details, usernames, passwords, and social security numbers, to the phisher, usually for mean purposes. The target victim of the fraud suffers a financial loss, as well as the loss of personal information and reputation. Therefore, it is essential to identify an effective approach for phishing website classification. Machine learning approaches have been applied in the classification of phishing websites in recent years. The objectives of this research are to classify phishing websites using artificial neural network (ANN) and convolutional neural network (CNN) and then compare the results of the models. This study uses a phishing website dataset collected from the machine learning database, University of California, Irvine (UCI). There were nine input attributes and three output classes that represent types of websites either legitimate, suspicious, or phishing. The data was split into 70% and 30% for training and testing purposes, respectively. The results indicate that the modified ANN with Rectified Linear Unit (ReLU) activation function model outperforms other models by achieving the least average of root mean square error (RMSE) value for testing which is 0.2703, while the CNN model produced the least average RMSE for training which is 0.2631. ANN with Sigmoid activation function model obtained the highest average RMSE of 0.3516 for training and 0.3585 for testing.**

Keywords*—Phishing website; classification; artificial neural network; convolutional neural network; machine learning

I. Introduction

Phishing is a form of social engineering attack that is regularly used to get individuals to provide confidential data, like credit card information and login credentials. This happens when a phisher pretends to be a reliable organization to get a targeted victim to open a text message or an email. Then, the victim is tricked into clicking a malicious link that leads to a bogus website where private and sensitive information such as account numbers and Internet banking passwords can be obtained. A cyber-attack might have disastrous consequences, such as unlawful transactions, money theft, or identity theft for users [1].

A non-profit association known as the Anti-Phishing Working Group (APWG) investigates phishing assaults that have been informed by its fellow corporations, including MarkMonitor, iThreat Cyber Group, Forcepoint, Internet Identity (IID), and Panda Security. It evaluates the attacks and releases quarterly and half-yearly reports regularly. Additionally, it offers statistics data on phishing attacks and malicious domains that are active worldwide. The most recent phishing activity trends report states that in the third quarter of 2022, APWG detected a total of 1,270,883 phishing attacks. This quarter's phishing activity was the worst that the APWG has ever recorded. 23.2% of all phishing attacks targeted the financial sector. In the third quarter, email-based scams involving advance fee payments grew by 1,000% [2]. Globally, millions of dollars have been lost as a result of these attacks, which had a severe effect on numerous renowned organizations worldwide. For protecting personal and business information as well as financial assets, addressing the problem of phishing website classification is getting more important. Hence, classifying and minimizing the impact of phishing attacks are the motivations to conduct this study.

A common countermeasure of phishing websites involves checking the websites against blacklists of known phishing websites, which are traditionally compiled, based on manual verification but this method is inefficient as it usually fails to discover all phishing sites because a recently created forged website takes a significant time before it can be added to the list. Moreover, there is no robust blacklist that will guarantee a perfect up-to-date database as nowadays, it has become easier to register new domain names. As the Internet scale grows, advanced website classification is increasingly important to provide timely protection to end users. Thus, this study aims to classify phishing websites using artificial neural network and convolutional neural network and then compare the models' classification performance. Biological neural networks are the foundation of the artificial neural network (ANN). It is made up of fundamental building blocks called neurons, which multiply weight with a real value and later put the result via a nonlinear activation function. By constructing multiple layers of neurons, where each of them takes the input variables and then passes the output to the following layers, the network can learn complicated functions. Supposedly, an ANN with adequate computational power could learn the shape of any function [3]. Neural networks have a high tolerance for noisy data, which is one of the advantages of using them. Although it is simple and convenient to use, handling huge dimensional data requires some understanding of parameter settings. It is simple to interpret the outcomes of this method. Another advantage of employing ANN is it can generate probability-based output, which ensures that the model will be more accurate as larger volumes of input data are provided [4].

A deep learning network architecture known as a convolutional neural network (CNN) automatically extracts features from the data rather than having to manually extract them first. The primary benefit of CNN over its predecessors is that it automatically recognizes the significant attributes

without human intervention [5]. CNN is mainly suitable for recognizing faces, scenes, and objects by discovering patterns in input images. It is also excellent in non-image data classification for example signal data, audio, and time series. The dataset was procured from a reputable source which is University of California, Irvine (UCI) repository, an open-source machine learning database platform frequently used by researchers in machine learning studies. We also used this dataset for our previous research article [6].

There are two contributions from this study. Firstly, the study employed two machine learning algorithms, which are an artificial neural network and a convolutional neural network to classify types of websites. These techniques enabled to build of robust classification models to recognize different types of websites either phishing, suspicious, or legitimate ones. Secondly, the research aimed to improve the accuracy of phishing classification. By leveraging the proposed methodology, the study contributes to reducing the error rate, thereby enhancing the overall performance of phishing classification. The findings from this research have practical implications for the development of cybersecurity tools or applications to detect phishing websites. By improving the accuracy of phishing classification, organizations or individuals can enhance their defenses against cyber threats, ultimately safeguarding themselves from being a victim of website scams more effectively.

The other components of this research article are structured in Sections II, III, IV, and V. Section II provides a discussion of related work, while Section III discusses thoroughly the methodology of ANN and CNN. Section IV presents the results and discussion of the experimental work, and Section V gives a conclusion, limitation, and direction for future works of research.

II. Related Work

There are several machine learning techniques for phishing website classification. A study conducted by Verma et al. [7] utilized deep belief network and artificial neural network approaches in which ANN achieved 89.95% accuracy with five hidden layers and five nodes in hidden layers, while the deep belief network's accuracy was 96.32% using similar settings. It was observed that the deep belief network technique performs better than ANN. Next, a study by Zamir et al. [8] presented a comparison of supervised learning approaches (NB, k-NN, SVM, RF, bagging, and NN) and stacking models to classify phishing websites. Stacking1 (RF + NN + bagging) outperformed all other classifiers with proposed features N1 (combined weakest features) and N2 (combined strongest features) and achieved 97.4% accuracy. Another study by Sharivari et al. [9] implemented XGBoost, Support Vector Machine, KNN, Logistic Regression, Ada Boost, Decision Tree, Random Forest, Gradient Boosting, and Neural Networks. XGBoost outperformed other methods with an accuracy of 98.32% followed by random forest and neural network.

A recent study conducted by Somesha et al. [10] applied Convolution Neural Network (CNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM) to classify phishing websites. The results indicated that the accuracy of the proposed method for LSTM, DNN, and CNN is 99.57%, 99.52%, and 99.43% respectively. Besides, Yerima & Alzaylaee [11] employed convolutional neural networks (CNN) for high-accuracy classification to differentiate fraudulent websites from legitimate websites. The result showed that the CNN technique, which achieved 97.3% accuracy with an F1-score of 97.6%, outperformed conventional machine learning classifiers evaluated on the same dataset. Another study by Geyik et al. [12] adopted decision tree, naive Bayes, random forest, and logistic regression algorithms. According to the findings, the Random Forest method outperforms the others, with an 83% of accuracy rate.

Afterward, Nadar et al. [13] proposed a hybrid Stacking model. Then, it was compared with Naïve Bayes, Random Forest, and XGboost approaches. The research outcome showed that the proposed Stacking Classifier outperforms other methods with results of 85.6% of accuracy. A study by [6] applied an artificial neural network and the findings show that the ANN model with (9-5-1) architecture design gave the best result by obtaining the lowest difference between average training and testing MSE, which is 0.04745. Later, research by [14] focused on using multilayer perceptron (MLP), a type of neural network concept, to classify phishing websites. MLP is compared with other machine learning approaches like logistic regression, random forest, and support vector machine (SVM) for result evaluation, and it was found that MLP achieved the highest accuracy of 96.80%.

Machine learning approaches were also being employed in other fields such as healthcare, for dengue prediction [15], diabetes classification [16], and depression prediction [17]. Machine learning was also applied in agriculture for classifying broccoli leaf disease [18] and types of coral reef fish [19]. It was also used for classifying types of traffic violations [20], carbon monoxide concentration prediction [21], and air traffic communication system [22]. Deep learning approaches were also utilized in sound recognition [23], prognosis of Covid-19 [24], and glioma classification [25]. After reviewing previous studies in this field, it is evident that the classification accuracy should be enhanced. It is crucial to decrease the error rate to attain a high level of classification accuracy leading to producing more reliable and robust classification models.

III. Methodology

The modelling of website phishing classification was divided into three main parts which represent three types of classification models. All classification processes followed the same methodology which started with pre-processing of the dataset where the phishing dataset [26] was normalized according to the selected model. There were nine input features and three output classes: Legitimate, Suspicious, and Phishy. It was split into a training set and a testing set with a ratio of 70:30. Next, the model was designed based on a selected algorithm to model the classifier. The model was trained using a gradient descent training algorithm with a learning rate set to 0.01 and momentum set to 0.1 to ensure slow convergence without skipping any possible best solution or overfitting. Finally, the model was evaluated using RMSE to measure the

model output and targeted output was minimized and the model accurately classified the phishing classes. The flow of the classification process using ANN is represented in Fig. 1.

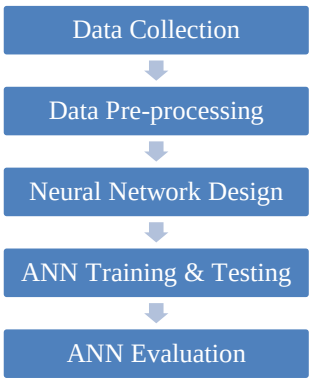

Fig. 1. ANN classification methodology.

A. *Artificial Neural Network Model with Sigmoid Activation Function*

The phishing website dataset was collected from the UCI repository with a total number of instances is 1353. Each instance has nine attributes and must be categorized into one of the three classes, the first class is Legitimate, the second class is Suspicious, and the third class is Phishy. Numbers had been used to replace these categories, with 1 denoting legitimate, 0 for suspicious, and -1 for phishing. There are 103 suspicious, 548 legitimate, and 702 phishing websites from the total number of 1353 websites. The data were saved using a comma-separated values (.csv) format. When a website is classified as suspicious, it might be either phishy or legitimate, implying that it has both phishy and legit attributes. The attributes in this dataset are including Server Form Handler (SFH), using a pop-up window, Secure Socket Layer (SSL) final state, request URL, anchor's URL, traffic of a website, length of URL, domain's age, and using an IP address. Out of 1353 instances, 70% of the instances were given for the training set which was 948 instances and the remaining 30% were given for the testing set which was 405 instances. Usually, in data pre-processing, normalization is used to scale the values from different ranges to a common range such as -1 to 1 or 0 to 1. Data must be scaled into the range used by the input neurons in the neural network. The dataset was scaled from -1 to 1 into 0 to 1 for the research purpose.

In a neural network, there are three layers, which are an input layer, a hidden layer, and an output layer. The model's input layer receives all of the inputs to be fed into the model. Then, these inputs are transmitted to the hidden layers. Each input neuron should represent some independent variable that has an impact on the neural network's output. There is a layer called the hidden layer which is located between the input layer and the output layer that is made up of a group of neurons that have activation functions applied to them. Its responsibility is to process the inputs obtained from the input layer. This layer is in charge to extract the required features from the input data. A neural network could contain multiple hidden layers. The output layer of ANN collects and passes the data in a way that it was designed to do. At the output layer, the processed data is made available. The neural network design used in this study was (9-10-10-1) which means 9 input neurons, two layers of hidden neurons with 10 neurons for each, and 1 output neuron as presented in Fig. 2.

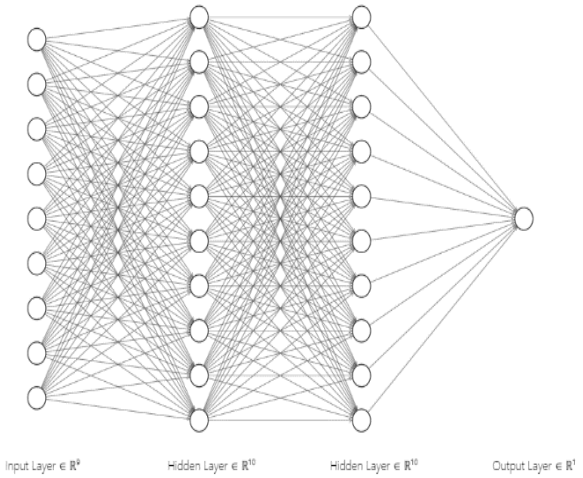

Fig. 2. ANN structure 9-10-10-1.

The activation function employed in this model was Sigmoid. Any real value can be used as an input for this function, which returns output values between zero and one. The output value will be closer to one if the input value is greater or more positive, whereas if it is smaller or more negative, the output value will be closer to zero. It can be mathematically formulated as depicted in (1).

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

The total number of epochs and batch size used in this experiment were set to 200 and 50, respectively. Training is a process of determining the values of weights and biases for a neural network. The train-test technique is the most common way used to conduct training. The phishing dataset was split into training and testing sets. The neural network was trained using the training dataset. For determining the set of weights and biases values which have a small difference between actual output and expected output values, a few weights and biases values are tested. The process of selecting appropriate weights and biases values for minimizing error was also a part of the training. The test dataset was not used throughout this phase. After the training was complete, the final ANN model's weights and biases were applied to the test dataset. The model's accuracy on the test dataset provides a rough estimation of the model's accurateness will be once provided with previously unseen and new data.

An unbiased evaluation of the final model fit on the training dataset was made using the test dataset. A distinct dataset with a similar probability distribution as the training dataset is referred to as a test dataset. Minimal overfitting has occurred if a model that fits the training dataset also fits the test dataset well. Overfitting is typically shown by the training dataset fitting the model better than the test dataset. A test dataset is therefore a set of instances utilized just to evaluate the performance of a model which is a generalization of a fully specified classifier.

B. Convolutional Neural Network Model

Fig. 3 shows a complete process involved in numerical data classification using CNN, which includes input data, pre-processing, CNN feature extraction, classification, and performance evaluation.

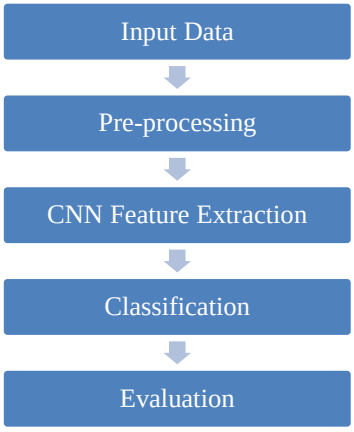

Fig. 3. CNN classification methodology.

The input data used in this process were the same phishing dataset used for ANN classification. All the data are numerical data that must be saved in .dat format for pre-processing purposes. In the pre-processing stage, this study used one of the pre-processing methods proposed by [27], called Equidistant Bar Graphs. It is a technique of data wrangling to transform data in numerical form into image form. In order to represent a particular class, the transformed image must reflect some patterns. Equidistant bar graphs show the measurement of each attribute of a particular dataset. The phishing dataset was first normalized to 0 and 1, and then each attribute was drawn by using its measured value. The image's width in pixels was given by a formula, $wx + y(x + 1)$, where x represents the total number of attributes, w represents the bar width, and the distance between two consecutive bars represents by y. The image's height was normalized to create a square image. In the experiments, the authors used 1−pixel and 2-pixel lengths for w and y, respectively. This creates an approximately 3d×3d size of the square image. A few data samples of the phishing website dataset that had already been transformed into bar graphs images are presented in Fig. 4. The images tagged along with the name of the respective class were labelled as Legitimate, Suspicious, and Phishy. CNN can only utilize these images if they represent a pattern in a convolved image. In this stage, all experiments were performed using Matlab 2020a software.

A convolutional neural network (CNN) is a deep learning algorithm that had been used in this research. Back-propagation artificial neural networks are the foundation of its architecture. It starts with an input image where each pixel represents input data that goes through a sequence of the feature selection process through convolution before being passed to the weighted perceptron where learning occurs through backpropagation. The main benefit of CNN is that it can learn the features on its own, as opposed to traditional neural networks where feature selection is a separate process and the model's accuracy depends on the selection of pre-processing and feature selection methods used. Fig. 5 shows the architecture of CNN used in this study.

There are two main layers of CNN architecture which are feature extraction or it can be called as feature learning and classification layers. In the feature extraction layer, there are three sublayers which are convolutional, activation, and pooling layers. The convolutional layer takes in the images directly as the input and a set of small filters is convolved over the image to produce one or more feature maps. The process of convolution involves moving the filter over the image while computing the dot product of the filter and image elements. Certain features are extracted from the image as a result of this process. Then, a bounded output is created by passing the convolutional layer's results via an activation function. CNN frequently employs Rectified linear units (ReLU) that turn any negative values into zero. Additionally, it trains the network far more quickly than other activation functions such as the hyperbolic tangent activation function (tanh). The down-sampling is carried out by the pooling layer, which also decreases the input size along each dimension. Average pooling and max pooling are two common pooling techniques that are usually used where the received image is divided into a collection of non-overlapping rectangles. Only the maximum and average values of each sub-region are obtained using max pooling and average pooling, respectively. The image is down-sampled in this process. In this study, only the max pooling technique was used in the pooling layer.

The architecture of CNN moves to the classification layer after learning features in the feature extraction layer. The fully connected network in a traditional neural network is similar to this fully connected layer. The classification output is produced by a classification layer, such as softmax, in the CNN architecture's final layer. For example, an image of a car goes through all the layers in CNN architecture which is then classified among the possible vehicles such as car, truck, van, or bicycle as the output of classification [28]. Meanwhile, for this study, the input images were classified as either phishy, suspicious, or legitimate. This experiment was performed using Visual Studio Code IDE with Python programming language.

Fig. 4. Bar graph of phishing website dataset.

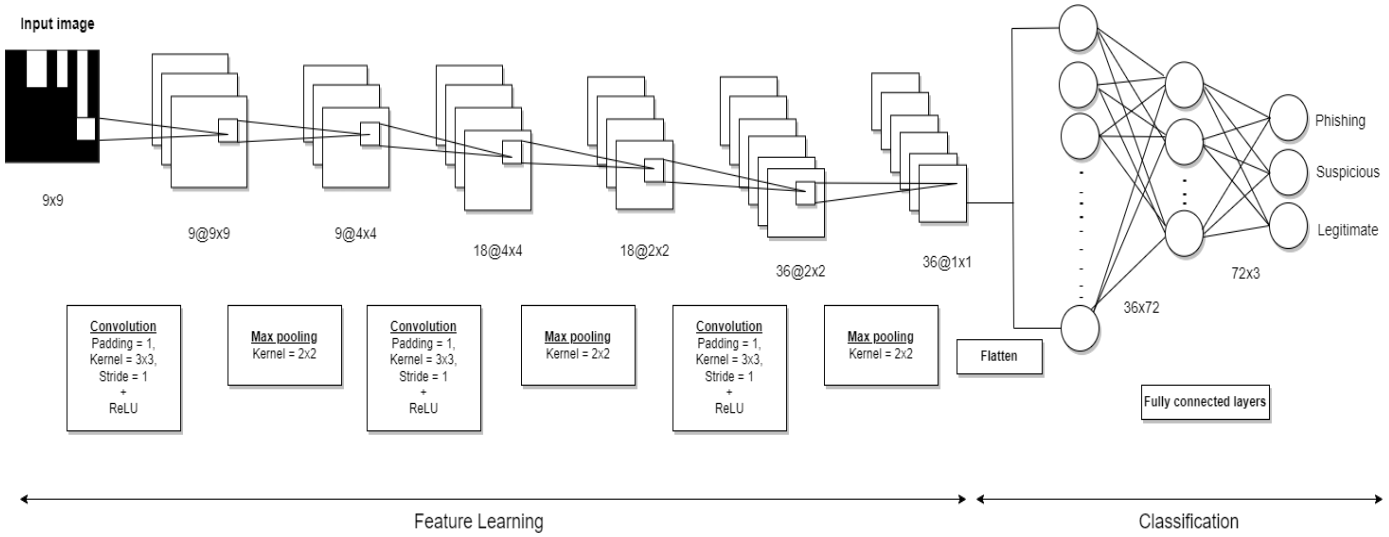

Fig. 5. Architecture of CNN for phishing website dataset.

C. Modified Artificial Neural Network Model with Rectified Linear Unit Activation Function

The modified ANN model had gone through all the stages in ANN classification methodology as shown in Fig. 1 which were collecting data, pre-processing the data, designing a suitable ANN model, training and testing the model, and lastly evaluating it. It was the same process as detailed in subsection *A*. The only difference was that this model employed rectified linear unit (ReLU) as an activation function which is usually utilized in the CNN model. Mathematically, ReLU is defined as shown in (2).

$$f(x)= \max(0,x) \tag{2}$$

The function indicates that if it is given any negative input value, the output of the function will be zero, but if it is given any positive value of x, the result will be the value itself. The equation gives the outcome within a range value from zero to infinity [29]. It has turned out to be the default activation function for several types of neural networks since a lot of classification models that employ it are easier to train and often produce better results [30]. Since the function involved a simple mathematic calculation, training also takes a lesser amount of time to complete [31].

D. Performance Evaluation

The ANN and CNN performances were evaluated by using root mean square error (RMSE), which is a well-known error indicator. It is a commonly employed metric for determining the differences between a model's predicted values and the actual values. RMSE is a metric of accuracy used to compare the forecasting errors of several models for a certain dataset. It is always a positive value, and a value of 0, which is rarely achieved in practice, would mean that a perfect fit for the data. Generally, a smaller RMSE is preferable to a greater one. RMSE can be calculated as shown in (3).

$$\text{RMSE}= \sqrt{\left[\sum_{i=1}^{n}(y_i-x_i)^2/n\right]} \tag{3}$$

The formula showed that y_i represents the ith expected value, x_i indicates the ith actual value, and n is the total number of data instances. In mathematics, the Greek sigma symbol that resembles a strange E is known as summation. It is the total of the sequence numbers from i=1 to n. The performance of the classification models improves with decreasing RMSE values. The best performance of the ANN and CNN was determined by the lowest RMSE.

All the experiments were performed in a 3.20 GHz, 8 GB RAM, CPU Intel Core i7 processor system, and the operating system was Windows 10 64-bit. ANN and CNN models were implemented on Visual Studio Code IDE (Integrated Development Environment) using Python programming language and used the Keras library with TensorFlow backend. Numpy, Seaborn, Scikit Learn, and Pandas were among the additional libraries that were imported and used for the experiments.

IV. Results and Discussion

There were three classification models designed in this research as stated in the method's section. Each model was trained and tested 20 times to ensure the consistency of the results of the experiment. All training and testing RMSE values were recorded. Fig. 6 shows a graph plotted from the results of training and testing RMSE with 20 runs of experiments using an artificial neural network (ANN) with a sigmoid activation function. The average training RMSE was 0.3516, while the average testing RMSE was 0.3585. The lowest training RMSE was 0.3036, while the highest was 0.4148. The lowest testing RMSE was 0.3179, while the highest was 0.4164. The testing RMSE usually gave a higher value than the training RMSE as it used new data that need to be classified. This also can be observed in [27] which the accuracy of the test set is lower than the validation set. It can be seen in Fig. 6 that there was a very small difference between training and testing RMSE for each run. 0.1 is considered a good value of RMSE while 0.5 and above is considered high which is not good for accuracy.

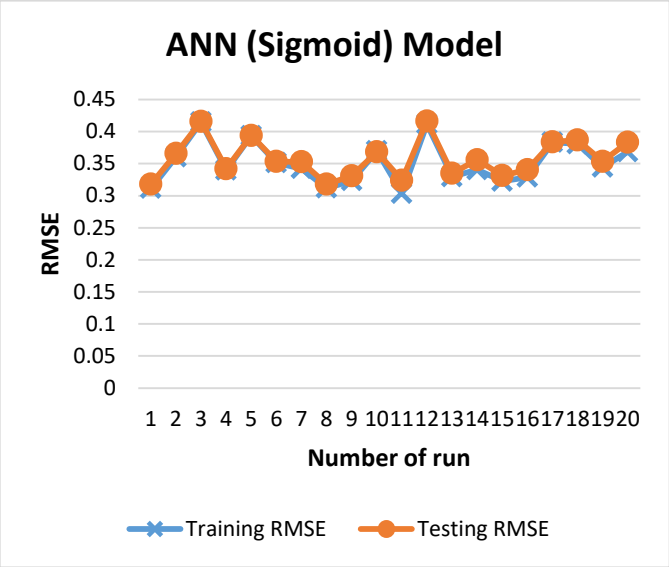

Fig. 6. Training and testing RMSE of ANN (Sigmoid) model.

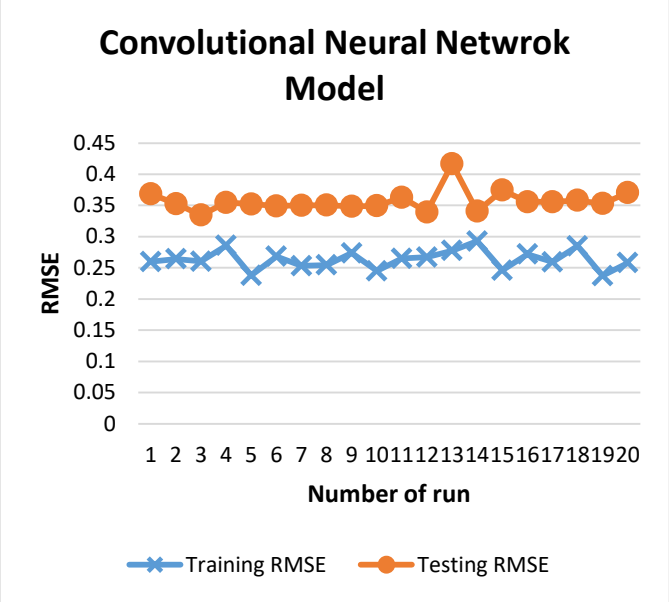

Fig. 7. Training and testing RMSE of CNN model.

The ANN (Sigmoid) model shows a high degree of variability in both training and testing RMSE values, with the range of values being relatively large. This variability may indicate that the ANN (Sigmoid) model is not consistent in its performance and may produce unreliable results.

Fig. 7 shows the results of training and testing RMSE of a convolutional neural network (CNN) model. The training RMSE values range from 0.2374 to 0.2926, while the testing RMSE values range from 0.3345 to 0.4169. The average training RMSE was 0.2631 while the average testing RMSE was 0.3569.

Compared to the ANN (Sigmoid) model, the CNN model showed less variation in the RMSE values across the 20 runs, indicating that it is more reliable in producing consistent results. The average training and testing RMSE of the CNN model were also lower than the average training and testing RMSE of the ANN (Sigmoid) model, suggesting that the CNN model has better performance in capturing the patterns in the dataset which produces better accuracy. Similarly, [7] shows an increase in accuracy by using the deep belief network technique compared to the neural network algorithm. Although the CNN method improves the performance and accuracy, the original pattern of the data may have changed due to the alteration of the data during the pre-processing stage. Furthermore, the networks used in the CNN model were very large compared to ANN (Sigmoid) model. Therefore, this study proposed to use the advantage of CNN which is Rectified Linear Unit (ReLU) activation function in the artificial neural network.

The modified ANN model using the ReLU activation function achieved a training RMSE between 0.2492 and 0.2817, with a mean training RMSE of 0.2703 as shown in Fig. 8. The testing RMSE was between 0.2976 and 0.3084, with a mean testing RMSE of 0.3033. The model showed consistent and reliable performance in terms of producing low training and testing RMSE values with small fluctuations between each run. The average training and testing RMSE values of the model were relatively low which indicated that the model was able to fit the training data well and also generalized well to new and unseen data, respectively. In addition, the model's performance was relatively consistent across the 20 runs, as the range of the RMSE values for both the training and testing sets is relatively narrow. This suggests that the model is not overfitting or underfitting the data, but rather finding a good balance between the two.

Compared to the ANN (Sigmoid) model, the modified ANN (ReLU) model showed slightly better performance in terms of producing more consistent results and more stable. The ReLU activation function is known for its ability to handle vanishing gradients and reduce the likelihood of overfitting. The results from the modified ANN (ReLU) model appear to support this claim, as the model produced more consistent results than the ANN (Sigmoid) model.

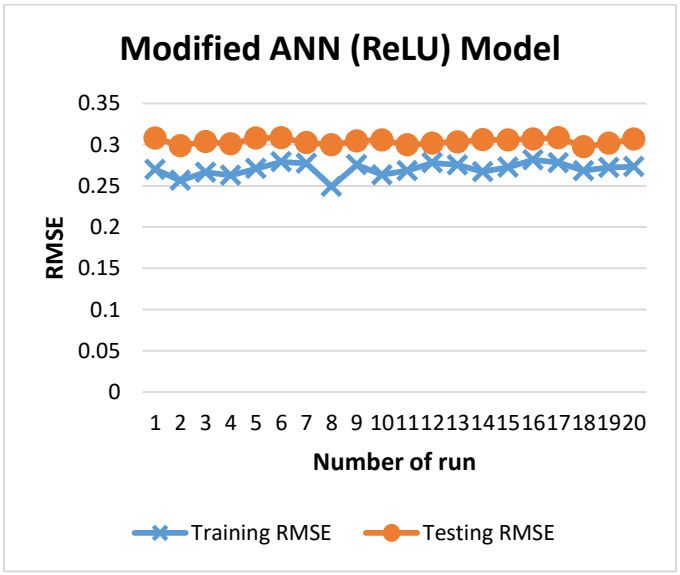

Fig. 8. Training and testing RMSE of ANN (ReLU) model.

Compared to the CNN model, the modified ANN (ReLU) model has a slightly higher training RMSE, but the difference between average testing and training RMSE is smaller for the modified ANN (ReLU) but higher for the CNN model, indicating that the CNN model tends to overfit the data [32]. Moreover, the modified ANN (ReLU) model requires fewer networks than the CNN model, which can be an advantage in terms of computational efficiency.

Table I shows the average RMSE comparison between ANN (Sigmoid), CNN, and modified ANN (ReLU) models. Fig. 9 shows the comparison of all the training RMSE trained using ANN (Sigmoid), CNN, and modified ANN (ReLU) models. The CNN model has the lowest average RMSE which was 0.2631 followed by the modified ANN (ReLU) model with an average of 0.2703 RMSE. ANN (Sigmoid) has the highest RMSE average which was 0.3516. The testing RMSE has a different pattern, where modified ANN (ReLU) produced the best average RMSE which was 0.3039, followed by CNN with an average RMSE of 0.3569. The highest RMSE was produced by ANN (Sigmoid) model with an average of 0.3585 as shown in Fig. 10.

Overall, the results of this study suggest that the modified (ReLU) model is an effective approach for classifying phishing, legitimate, and suspicious websites. The modified ANN (ReLU) model was selected as the best model because it was able to generalize well to new data and the number of neural networks used in this model was also smaller than the CNN model. The modified (ReLU) model may have outperformed the CNN model because the original patterns of data were preserved, whereas, in the CNN model, some original patterns may have been changed during the pre-processing stage when converting numerical data to image data.

TABLE I. COMPARISON OF CLASSIFICATION MODELS' PERFORMANCE

Phase	ANN (Sigmoid)	CNN	Modified ANN (ReLU)
Training	0.3516	0.2631	0. 2703
Testing	0.3585	0.3569	0. 3039

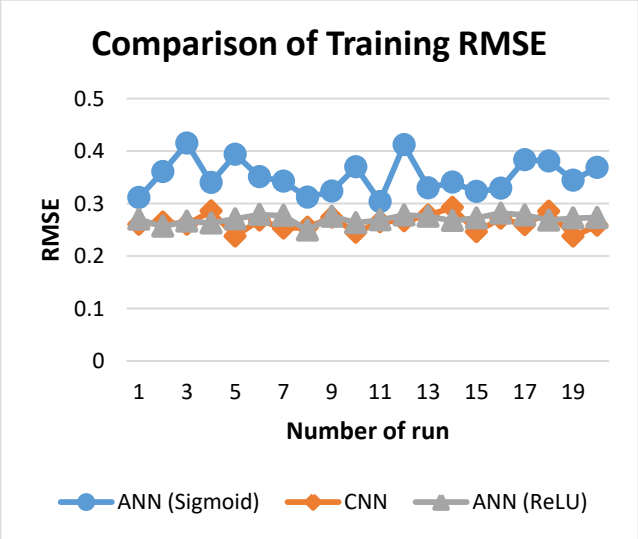

Fig. 9. Training RMSE comparison.

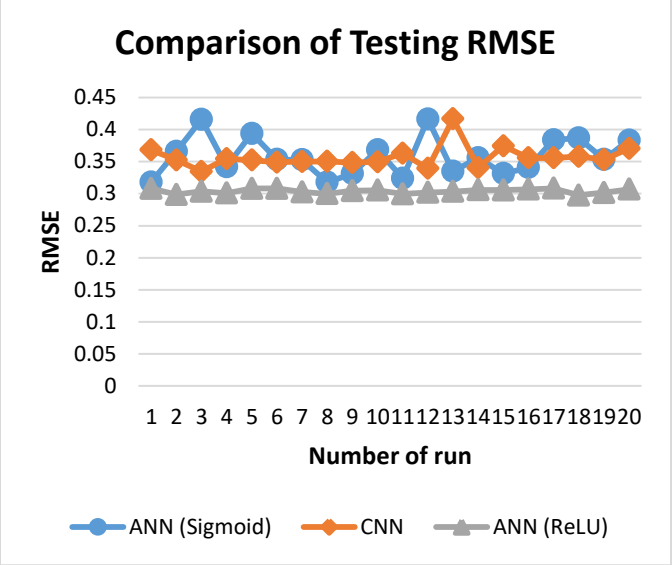

Fig. 10. Testing RMSE comparison.

V. CONCLUSION

Artificial neural networks and convolutional neural network classifiers were implemented in this study for classifying phishing websites. The models produced from this study were ANN (Sigmoid), CNN, and modified ANN (ReLU). It can be seen from the experimental result that the modified ANN (ReLU) model demonstrated the best performance for phishing website classification followed closely by the CNN model. On the other hand, the ANN (Sigmoid) model had the poorest performance of the three models. In the future, more methods such as hybrid or other deep learning techniques can be implemented to produce much more reliable results and improve performance and accuracy. The dataset used for training and testing the models in this study may be relatively small, which could impact the model's ability to capture the full complexity and diversity of phishing websites. A larger dataset might be needed to improve the model's robustness and generalization. Further research on other datasets is also needed to validate the effectiveness of the three models produced by this study.

REFERENCES

[1] "Phishing attacks." https://www.imperva.com/learn/application-security/phishing-attack-scam/ (accessed Jun. 24, 2021).

[2] APWG, "Phishing Activity Trends Report 3rd Quarter 2022," 2022. [Online]. Available: http://www.apwg.org,

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

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

[5] A. Dertat, "Applied Deep Learning - Part 4: Convolutional Neural Networks | by Arden Dertat | Towards Data Science," 2017. https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2 (accessed Jun. 10, 2022).

[6] N. H. Hassan and A. S. Fakharudin, "Model for phishing websites classification using artificial neural network," International Journal of

Software Engineering & Computer Systems (IJSECS), vol. 7, no. 2, pp. 1–8, 2021.

[7] M. K. Verma, S. Yadav, B. K. Goyal, B. R. Prasad, and S. Agarawal, "Phishing Website Detection Using Neural Network and Deep Belief Network," Springer Singapore, 2019, pp. 293–300. doi: 10.1007/978-981-10-8639-7.

[8] A. Zamir et al., "Phishing web site detection using diverse machine learning algorithms," Electronic Library, vol. 38, no. 1, pp. 65–80, 2020, doi: 10.1108/EL-05-2019-0118.

[9] V. Sharivari, M. M. Darabi, and M. Izadi, "Phishing Detection Using Machine Learning Technique," Proceedings - 2020 1st International Conference of Smart Systems and Emerging Technologies, SMART-TECH 2020, 2020, doi: 10.1109/SMART-TECH49988.2020.00026.

[10] M. Somesha, A. R. Pais, and R. S. Rao, "Efficient deep learning techniques for the detection of phishing websites," Sādhanā, vol. 0123456789, 2020, doi: 10.1007/s12046-020-01392-4.

[11] S. Y. Yerima and M. K. Alzaylaee, "High Accuracy Phishing Detection Based on Convolutional Neural Networks," in ICCAIS 2020 - 3rd International Conference on Computer Applications and Information Security, Institute of Electrical and Electronics Engineers Inc., Mar. 2020. doi: 10.1109/ICCAIS48893.2020.9096869.

[12] B. Geyik, K. Erensoy, and E. Kocyigit, "Detection of Phishing Websites from URLs by using Classification Techniques on WEKA," Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021, pp. 120–125, 2021, doi: 10.1109/ICICT50816.2021.9358642.

[13] V. K. Nadar, B. Patel, V. Devmane, and U. Bhave, "Detection of Phishing Websites Using Machine Learning Approach," 2021 2nd Global Conference for Advancement in Technology, GCAT 2021, pp. 1–8, 2021, doi: 10.1109/GCAT52182.2021.9587682.

[14] A. Dev and V. Jain, Identifying Phished Website Using Multilayer Perceptron, vol. 127. 2021. doi: 10.1007/978-981-15-4218-3_15.

[15] N. Farisha, M. Krishnan, Z. A. Zukarnain, A. Ahmad, and M. Jamaludin, "Predicting Dengue Outbreak based on Meteorological Data Using Artificial Neural Network and Decision Tree Models," International Journal on Informatics Visualization, vol. 6, no. 3, pp. 597–603, 2022, [Online]. Available: www.joiv.org/index.php/joiv

[16] B. S. Bahnam and S. A. Dawwod, "A proposed model for diabetes mellitus classification using coyote optimization algorithm and least squares support vector machine," IAES International Journal of Artificial Intelligence (IJ-AI), vol. 11, no. 3, pp. 1164–1174, 2022, doi: 10.11591/ijai.v11.i3.pp1164-1174.

[17] H. Diyana, A. Rahimapandi, R. Maskat, R. Musa, and N. Ardi, "Depression prediction using machine learning : a review," IAES International Journal of Artificial Intelligence (IJ-AI), vol. 11, no. 3, pp. 1108–1118, 2022, doi: 10.11591/ijai.v11.i3.pp1108-1118.

[18] Y. Ferdinand and W. F. al Maki, "Broccoli leaf diseases classification using support vector machine with particle swarm optimization based on feature selection," International Journal of Advances in Intelligent Informatics, vol. 8, no. 3, pp. 337–348, Nov. 2022, doi: 10.26555/ijain.v8i3.951.

[19] L. A. Latumakulita et al., "Combination of Feature Extractions for Classification of Coral Reef Fish Types Using Backpropagation Neural Network," International Journal on Informatics Visualization, vol. 6, no. 3, pp. 643–649, 2022, [Online]. Available: www.joiv.org/index.php/joiv

[20] N. A. Othman, C. F. M. Foozy, A. Mustapha, S. A. Mostafa, S. Palaniappan, and S. A. Kashinath, "A data mining approach for classification of traffic violations types," International Journal of Advances in Intelligent Informatics, vol. 7, no. 3, pp. 282–291, Nov. 2021, doi: 10.26555/ijain.v7i3.708.

[21] R. K. Angatha and A. Mehar, "MODELING OF CARBON MONOXIDE CONCENTRATIONS AT URBAN SIGNALIZED INTERSECTIONS USING MULITPLE LINEAR REGRESSION AND ARTIFICIAL NEURAL NETWORKS," Suranaree J. Sci. Technol., vol. 29, no. 1, pp. 1–7, 2022.

[22] Y. Mnaoui, A. Najoua, and H. Ouajji, "Artificial intelligence in a communication system for air traffic controllers ' emergency training," IAES International Journal of Artificial Intelligence (IJ-AI), vol. 11, no. 3, pp. 986–994, 2022, doi: 10.11591/ijai.v11.i3.pp986-994.

[23] K. M. O. Nahar, F. Al-Omari, N. Alhindawi, and M. Banikhalaf, "Sounds Recognition in the Battlefield Using Convolutional Neural Network," International Journal of Computing and Digital Systems, vol. 11, no. 1, pp. 1177–1185, 2022, doi: 10.12785/ijcds/110196.

[24] A. W. Reza, J. F. Sorna, M. M. U. Rashel, and M. M. A. Shibly, "Modcovnn: A convolutional neural network approach in covid-19 prognosis," International Journal of Advances in Intelligent Informatics, vol. 7, no. 2, pp. 125–136, 2021, doi: 10.26555/ijain.v7i2.604.

[25] A. E. Minarno, Y. Sasongko, Y. Munarko, H. A. Nugroho, and Z. Ibrahim, "Convolutional Neural Network featuring VGG-16 Model for Glioma Classification," International Journal on Informatics Visualization, vol. 6, no. 3, pp. 660–666, 2022, [Online]. Available: www.joiv.org/index.php/joiv

[26] "UCI Machine Learning Repository: Website Phishing Data Set," 2016. https://archive.ics.uci.edu/ml/datasets/Website+Phishing# (accessed Jul. 04, 2022).

[27] A. Sharma and D. Kumar, "Non-image Data Classification with Convolutional Neural Networks," 2020, [Online]. Available: http://arxiv.org/abs/2007.03218

[28] S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," 2018. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed Mar. 04, 2022).

[29] Great Learning Team, "An Introduction to Rectified Linear Unit (ReLU) | Great Learning," 2020. https://www.mygreatlearning.com/blog/relu-activation-function/ (accessed Jun. 14, 2022).

[30] J. Brownlee, "A Gentle Introduction to the Rectified Linear Unit (ReLU)," 2019. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/ (accessed Jun. 14, 2022).

[31] D. Liu, "A Practical Guide to ReLU. Start using and understanding ReLU… | by Danqing Liu | Medium," 2017. https://medium.com/@danqing/a-practical-guide-to-relu-b83ca804f1f7 (accessed Jun. 14, 2022).

[32] M. A. Mohd Yusof, Z. Abdullah, F. A. Hamid Ali, K. A. Mohamad Sukri, and H. S. Hussain, "Detecting Malware with Classification Machine Learning Techniques," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 14, no. 6, pp. 167–172, 2023, [Online]. Available: www.ijacsa.thesai.org.