RESEARCH ARTICLE

SPECIAL ISSUE

International Research Journal on Advanced Science Hub
2582-4376
Vol. 05, Issue 05S May
www.rspsciencehub.com

Check for updates

RSP Science Hub

http://dx.doi.org/10.47392/irjash.2023.S041

International Conference on intelligent COMPUting TEchnologies and Research (i-COMPUTER) 2023

Detection of Phreaking Website Using Various Algorithms

Maneesha K [1], Rajasekhar K [2], Prema Latha K [1], Venkata Prasad N [1]

[1]Department of Information Technology, Laki reddy Bali Reddy College of Engineering, Mylavaram, Andhra Pradesh, India
[2]Sr Assistant Professor, Department of Information Technology Laki reddy Bali Reddy College of Engineering, Mylavaram, Andhra Pradesh, India

Email: maneeshakillampalli15@gmail.com

Article History

Received: 1 March 2023
Accepted: 21 March 2023

***Keywords**:*

Phreaking Website;
Phishing attack;
URLs;
Convolutional Neural Network (CNN);
Long Short-Term Memory (LSTM);
Machine Learning

Abstract

A big concern to the Internet nowadays is phishing, a crime that involves exploiting technological tools to steal sensitive consumer data. Phishing losses are also rising quickly. The importance of feature engineering in solutions for detection of phishing websites, however the precision of detection is crucial and it depends on the features you know already. Additionally, although features retrieved from multiple dimensions are more thorough, extracting these characteristics has the downside of taking a long time. To address these, we proposed a new approach in which dataset contains millions of URLs by this approach we can identify the URL which is attacked by the phisher. To determine whether the URL has been targeted by the phisher, some of the Convolutional Neural Network algorithms like CNN-LSTM, CNN BI-LSTM, Logistic Regression, and XG Boost are utilized and resulting in the correctness of the graph between the two machine learning methods by using trained dataset and more likely to produce sensitivity, specificity, precision, recall, and f1-score along with accuracy graph, confusion matrices and also along with ROC-AUC curves.

1. Introduction

The development of the Internet as a vital infrastructure that profoundly aids human society Internet users' economies have already been seriously threatened by phishing, harmful software, and privacy revelations, which are unavoidable security challenges. The APWG (Anti-Phishing Working Group) (Yang, Zhao, and Zeng) describes phishing as a criminal tactic that combines technological and social engineering deception to get users' personal information and login credentials for bank accounts.

Phishing is the practice of attempting to get sensitive data (Bhavani et al.), such as usernames, passwords, and payment card details. By posing as a trustworthy party in an electronic contact (often for malicious motives and, indirectly, for money). Because utilizing bait to try to catch a victim is analogous to fishing, this word was formed as a homophone of fishing. The two most common phishing techniques are email spoofing and instant messaging, which regularly persuade people to divulge personal information on a false website that looks and functions exactly like the real one. Victims are frequently duped by communications posing as coming from websites for social networking, auctions, banks, processors of online payments, or IT managers. Links in phishing emails could lead to sites that have been infected with malware. Phishing is a type of social engineering approach that deceives consumers by taking advantage of flaws in current online security (Krishna et al.). Legislation, user education, public awareness campaigns, technical security measures, and other methods are being put into place to combat the rise in reported phishing instances. But they should be clearly branded as to who made them and users shouldn't use them without permission. Numerous websites have created additional tools for programmers, including game maps.

In general, phishing is a type of cyberattack (Anil et al. Kumar et al.) that has a detrimental effect on people by deceiving them into exposing private information such account passwords, bank details, ATM pin numbers, etc. Protecting recent sensitive documents while guarding against malware or web phishing is risky. Techniques for detecting phreaking websites can mainly be categorized into four groups, which are shown in Figure 1 below.

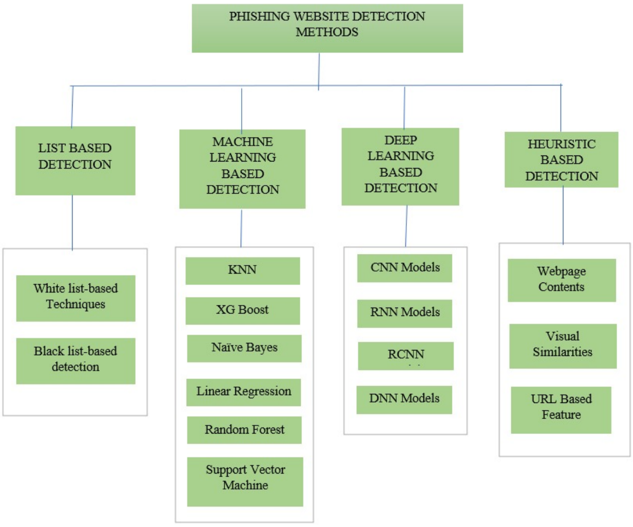

FIGURE 1. **Types of Phishing Website Detection Methods**

Figure 2 demonstrates the block diagram of phreaking websites such that six steps are involved to attack the user credentials in that first step is attacker prepares a server to host phishing script and page and it is stored in the attacker's server and then attacker compromises website and injects malicious script in the website. In third step user accesses compromises websites and runs the injected script and then injected script loads and phishing script from attacker's server and next step phishing script shows fake login credentials [9]to the user and last step victim enters credentials on the fake form and then the credentials are sent to the attacker.

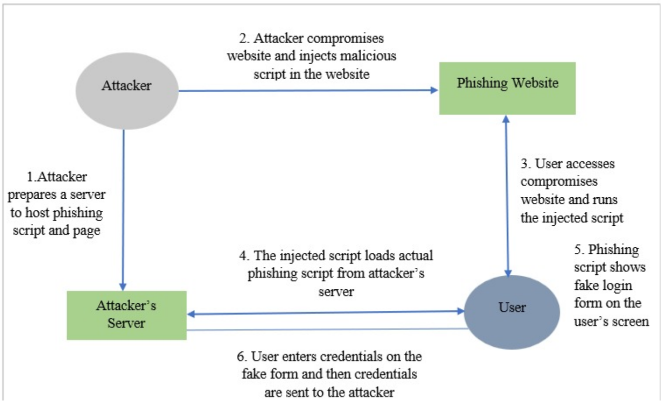
Attacker
2. Attacker compromises website and injects malicious script in the website
Phishing Website
1.Attacker prepares a server to host phishing script and page
3. User accesses compromises website and runs the injected script
4. The injected script loads actual phishing script from attacker's server
5. Phishing script shows fake login form on the user's screen
Attacker's Server
User
6. User enters credentials on the fake form and then credentials are sent to the attacker

FIGURE 2. **Block Diagram of Phishing Process**

The search space dimension has increased as a result of the classification model frequently being trained using a high number of features. Hughe's effect (Dharani et al.), also referred to as the Curse of Dimensionality, asserts that a classifier's performance only shows a steady increase up to a particular threshold dimensionality before falling. To resolve this problem, a feature selection approach must be used.

Despite the fact that traditional machine learning algorithms are extremely prone to under fitting and overfitting, they may not necessarily produce the best results. This issue might be solved using ensemble machine learning approaches and deep learning techniques.

Deep learning is built on machine learning, a branch of artificial intelligence. Deep learning will succeed because neural networks reproduce how the human brain functions. In deep learning, nothing is explicitly coded.

A specific kind of neural network called a convolutional neural network is frequently employed in the fields of object recognition, image classification, and image clustering. DNNs enable the construction of hierarchical visual representations. More than any other neural network, deep convolutional neural networks are advised for achieving the best accuracy.

2. Literature Survey

To understand the reviews regarding whether or not a website has been attacked by phishers, the previous study may be discovered in the literary survey.

- Lizhen Tang And Qusay H. Mahmoud (2022)

The framework which has been proposed by the Lizhen Tang and Qusay Mahmoud has put into place as a browser plug-in that can identify phishing risks in real time when a user visits a website and issue a warning. The real-time prediction service integrates several techniques, such as whitelist filtering, blacklist interception, and machine learning (ML) prediction. They have compared various machine learning models utilizing various datasets in the ML prediction module. The RNN-GRU model has the highest accuracy of 99.18% according to the trial findings, proving the viability of the suggested approach. (Tang and Mahmoud)

• Peng Yang, Guangzhen Zhao, And Peng Zeng (2019)

They suggested a multidimensional feature phishing detection methodology based on a quick detection method by utilizing deep learning to address the constraints. In this, they mix deep learning's rapid classification output with URL statistical data, webpage code features, webpage text features, and multidimensional features. Test results on a dataset with millions of legitimate and phishing URLs show that the accuracy is 98.99% and the false positive rate is only 0.59%. (Yang, Zhao, and Zeng)

• Rishikesh Mahajan, Irfan Siddavatam (2018)

In order to distinguish between legal and phishing URLs, this article uses machine learning technology. It extracts and analyses many aspects of both types of URLs. Algorithms such as Support Vector Machine, Decision Tree, and Random Forest are used to identify phishing websites. By evaluating each algorithm's accuracy rate, false positive and false negative rates, the study aims to identify phishing URLs and identify the best machine learning method with the highest accuracy of 97.4% of Random Forest algorithm. (Mahajan and Siddavatam)

• Arathi Krishna V, Anusree A, Blessy Jose, Karthika Anilkumar, Ojus Thomas Lee (2021)

In this paper they done the work on the identification of phishing URLs, or to categorise a URL as phishing or legitimate, various machine learning techniques are used. our goal in this work is to review several machine learning techniques utilised for this purpose. The objective is to establish a survey resource for academics to learn about recent advancements in the industry and help develop phishing detection models that produce more reliable findings. (Krishna et al.)

• Dr Anil Gn, G Om Prakash, K Harsha Manoj, M Lokesh, Madhusudhan K (2020)

They created resource descriptions, in which they use combination of methods to detect phishing websites. In order to train their programme, we use supervised learning approaches. This approach has a very encouraging score, which is commendable. Also, they employed a software programme to remove features that allow quantifying the frequency of each job within the dataset in addition to a random forest classification to handle incomplete data sets. In order to test the effectiveness of the Random Forest Algorithm and ensemble learning techniques, the accuracy is impressive. (Anil et al.)

• Naresh Kumar D, Nemala Sai Rama Hemanth, Premnath S, Nishanth Kumar V, Uma S (2020)

This study proposes a unique machine learning-based classification technique with heuristic features, where feature selection may be taken from properties such as Uniform Resource Locator, Source Code, Session, and so on. Five machine learning techniques, including random forest, K Nearest Neighbour, decision tree, support vector machine, and logistic regression, were used to assess the suggested model. The random forest approach outperforms existing models, detecting attacks with an accuracy of 91.4%. Moreover, the Random Forest Model chooses the best data using orthogonal and oblique classifiers. (Kumar et al.)

3. Methodology

3.1. Dataset Description:

The model's training data set was obtained from Kaggle.com. This has more than 25 thousand (Prabakaran, Chandrasekar, and Sundaram Elsadig et al.) data entries and 48 attributes. 80 percent of them are regarded as training data, whereas 20 percent are test data.

3.2. Data Preprocessing:

The process of translating raw data into information that a machine learning model can utilize is known as data preparation. It is both the initial and most critical phase in the building of a machine learning model. Preprocessing data entails the following actions:

3.2.1. Getting the Dataset:

Data is the foundation of all machine learning models; Thus, the first thing needed to develop one is a dataset. The dataset is the bundle of data that has been properly arranged for a certain issue. For instance, the dataset needed for someone to build a business-oriented machine learning model will be distinct than the collection needed for another purpose (Mahajan and Siddavatam). Datasets can take on several forms and serve a variety of functions. Here, we're using a dataset in the.csv format for this project. The project is carried out utilizing this dataset.

3.2.2. Importing Libraries:

To done data preparation in Python, we must import a number of predefined Python packages. Several of the libraries are used to complete some particular tasks.

3.2.3. Importing the Datasets:

In order to achieve the required results for our research, we must import the dataset in this phase. But first, we must make the current directory the working directory in order to import a dataset.

3.2.4. Splitting the Dataset into Training set and Testing Set:

our data into a test set and a training set throughout the machine learning data preparation stage. One of the most important steps in data preparation because it allows us to improve the performance of our machine learning model.

Dataset
Training Set
Test Set

FIGURE 3. Splitting the Dataset

3.3. Various Algorithms Used

Here, we employ a variety of techniques, including two machine learning algorithms and two deep learning models, such as convolutional neural network algorithms, for another two.

3.3.1. Machine Learning Algorithms

Logistic Regression:: A categorical dependent variable is used in the regression model known as logistic regression (DV). A mathematical technique called logistic regression can be used to calculate a binary response's likelihood given one or more independent variables. Logistic regression is used to forecast outcomes with two alternative values, such as 0 or 1, pass or fail, yes or no, and so forth. The logistic regression is a type of regression model. is a prognostic study (Choudhary et al.). It is widely employed in data visualization to emphasize the connection between a binary a nominal, ordinal, interval, or ratio-level one or more independent factors and the dependent variable. It also necessitates a cost function that is trickier. in place of being a linear. The term "sigmoid function" or "logistic function" is used to describe this cost function. Equation demonstrates that the algorithm's hypothesis holds between 0 and 1 for the cost function limit. tends to hold. The only possible values for the binary dependent variable included in this logistic regression are "0" and "1," which represent outcomes such as "Yes/No," "True," "False," "High," and "Low," among others.

$0<h(x)<1$

XG Boost:: XG Boost is the abbreviation for Extreme Gradient Boosting. The application for gradient-boosted decision trees was developed with efficiency and speed in mind. Boosting is a type of ensemble learning that incorporates extra techniques to correct flaws in previous models. The number of models is gradually increased until there is no more opportunity for improvement (Bhavani et al.). To reduce the loss when incorporating new models, it employs a gradient descent technique. This approach provides quick memory and computing time. This approach aimed to train the model with the least number of resources possible. Model performance and execution speed are XG Boost's two key benefits.

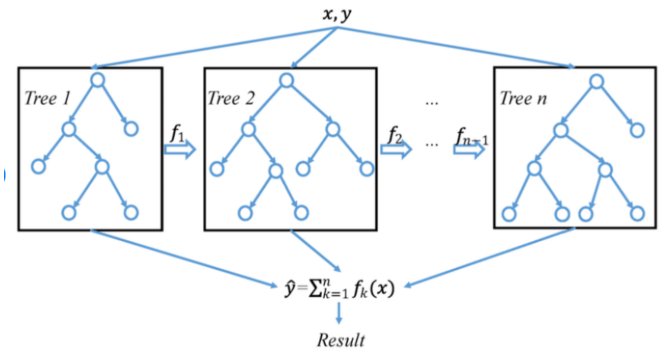

FIGURE 4. Architecture of XGboost

3.3.2. Deep Learning Algorithms:

Deep neural networks, such as convolutional neural networks (CNN), are frequently used to evaluate mental images When we think of neural networks, we frequently think of matrix multiplications. It uses uses it uses, it uses it using the convolution. In mathematics, convolution shows how the shape of one function is altered by another by fusing two functions to produce a third function.

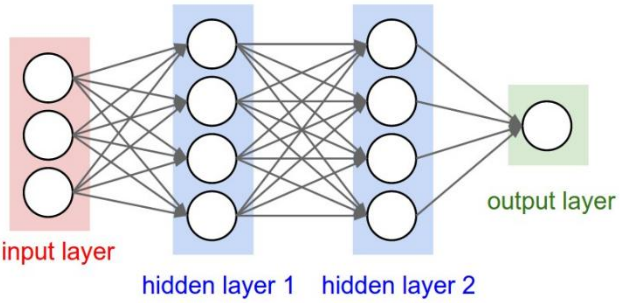

FIGURE 5. Architecture of CNN

CNN-LSTM (Convolutional Neural Network-Long Short--Term Memory: Due to CNN and LSTM's accessibility, their combination is a common idea for merging benefits. This work integrated CNN and LSTM to provide the idea for an unique deep learning scheme. To make sure the multidimensional data was appropriately correlated and collected, two layers of CNN were used (Tang and Mahmoud Yang, Zhao, and Zeng). The LSTM algorithm received a set of feature series from the CNN layer as input. The layer LSTM extracted time dependencies in greater detail. The URL input matrix is insufficient to appropriately reflect the data on the phishing website. In this section, multidimensional features that thoroughly explain the entire flow are generated by combining the CNNLSTM URL, a web page code, a text function, and a rapid grading result. Phishers typically create phishing URLs by replicating the URL of your website in an effort to confound consumers.

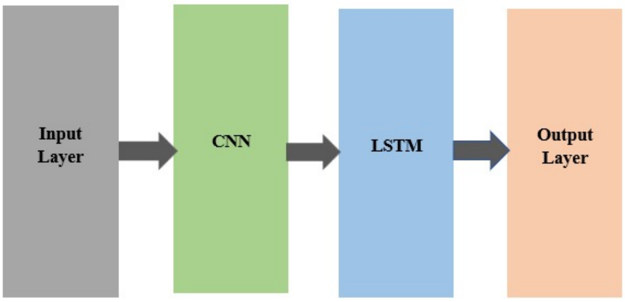

FIGURE 6. CNN-LSTM

CNN BI-LSTM (CNN Bidirectional Long Short-Term Memory: Bidirectional long short-term memory is a kind of recurrent neural networks. It consists of two concealed layers that combine forward and reverse data processing, enabling the structure to remember information from earlier input (Tang and Mahmoud Yang, Zhao, and Zeng). In our suggested architecture, it is the second layer, and its purpose is to keep track of previous transactions that are useful for forecasting the output y, which may be stated as follows.

y^t= g (w_y [h^t, c^t]+b_y)

where w is the weight value given to the concatenation of the hidden and current states produced by the Bi-LSTM, h and c are the hidden and current states, and t = transaction.

3.4. Proposed Framework

The two steps of the suggested system are the classification phase and the phishing detection phase and its proposed framework is depicted as follows:

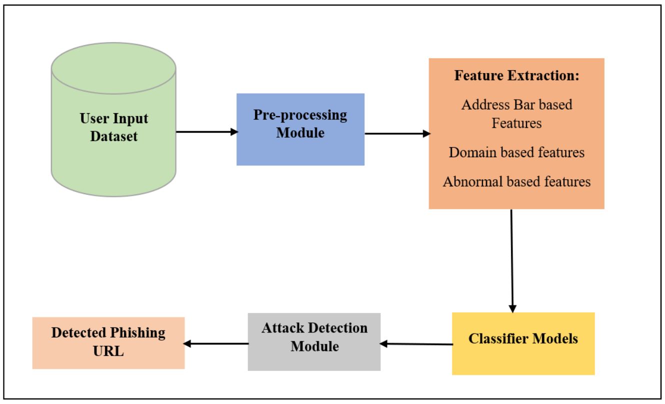

FIGURE 7. Proposed Framework for Phishing Attack

3.4.1. Classification Phase:

Regular URLs and suspect URLs for phishing websites make up the input for the categorization step. These inputs are sent to three submodules: the Data Collecting module, the Feature Selection module, and the Classification module. The feature extraction module takes into consideration the Address Bar, features with an anomalous basis, and features with a domain basis. These attributes are provided as input to the categorization module. By contrasting their URLs with those of real websites, the classification module's main goal is to accurately detect phishing websites. In order for the classifier to successfully identify phishing Sites, feature selection's primary goal is to extract the true and necessary features from the attributes offered by the feature selection module. Two classifiers that use machine learning and two that use deep learning make up the proposed study.

3.4.2. Detection Phase:

The primary goal of this module is to identify phishing URLs from a dataset that includes a large number of URLs using information collected from feature extraction module's characteristics.

4. Performance Metrics

4.1. Metrics considered for performance evaluation

The category or categories of data are found while using training data to solve a classification problem. The model gains knowledge from the previously provided dataset classifying the groups or classes of fresh data in accordance with the training. In the response, it makes predictions about whether a class will be Yes or No, 0 or 1, spam or not, etc (Elsadig et al.). A categorization model's effectiveness is measured using a range of measures, some of which are as follows:

4.1.1. Accuracy

The proportion of correct predictions a model makes out of all feasible ones when doing categorization tasks.

$$Accuracy \ = (TP+TN)/(TP+FP+FN+TN) \quad (1)$$

4.1.2. Precision

Precision is the measure of the proportion of correct positive forecasts (Buber, Demir, and Sahingoz). It can be calculated as the True Positive, or the percentage of all accurate positive forecasts (True Positive and False Positive).

$$Precision = TP/(TP+FP) \quad (2)$$

4.1.3. Recall

It aims to quantify the proportion of false positives that were in fact real positives. You can compute it using the TP formula, that contrasts the total number of correctly expected positives or incorrectly predicted negatives with the number of accurate forecasts (TP and FN).

$$Recall(R) = TP/(TP+FN) \quad (3)$$

4.1.4. F1-Score

Considering the forecasts offered for the positive class, the F-score or F1 Score measure is used to assess a binary classification model. It is calculated by using Precision and Recall (Al-Ahmadi, Alotaibi, and Alsaleh). As a result, the F1 Score can be deliberated with equal weights for each variable using the harmonic means of recall and precision.

$$F1 - Score = 2 * (Precision \ * \ Recall)/(Precision \ + \ Recall) \quad (4)$$

4.1.5. Confusion Matrix

A confusion matrix, a tabular representation of the anticipated results is used to demonstrate how well a binary classifier performed on a set of test data when true values were known.

4.1.6. Specificity

The proportion of true negatives to true negatives and false positives that the model correctly picks up is known as specificity.

$$Specificity = TN/(TN \ + \ FP) \quad (5)$$

4.1.7. Sensitivity

In machine learning, the metric known as sensitivity is used to evaluate the capacity of a model to forecast each available category's true positives.

$$Sensitivity = TP/(TP \ + \ FN) \quad (6)$$

5. Output of Experiments and Analysis

The training and testing portions of the dataset for the model are split 80:20. This is a summary of the machine learning models that were applied to our suggested framework. Table 1 displays the accuracy of the various algorithms we used, and Table 2 displays the matrices of the two algorithms that were compared using the metrics of logistic regression and XG boost. Figure 8 shows the accuracy of the all algorithms used in this project where x-axis shows the algorithms and y-axis shows the accuracy percentage achieved by the algorithms. Figure 9, Figure 10 shows the confusion matrix and ROC curve achieved by logistic regression algorithm. Figure 11, Figure 12 shows the Confusion matrix and Roc curve by XG Boost algorithm and figure 13 shows that the comparison of roc curves.

TABLE 1. Accuracy of different algorithms

Algorithms	Accuracy
CNN LSTM	56.4%
CNN BI-LSTM	55.9%
Logistic Regression	91.6%
XG Boost	92%

TABLE 2. Metrics of two algorithms

Metrics	Logistic Regression	XG Boost
Precision	0.95	1.00
Recall	0.93	1.00
F1-Score	0.93	1.00
Sensitivity	0.9	1.00
Specificity	0.09	1.00
ROC-AUC	0.96	0.95

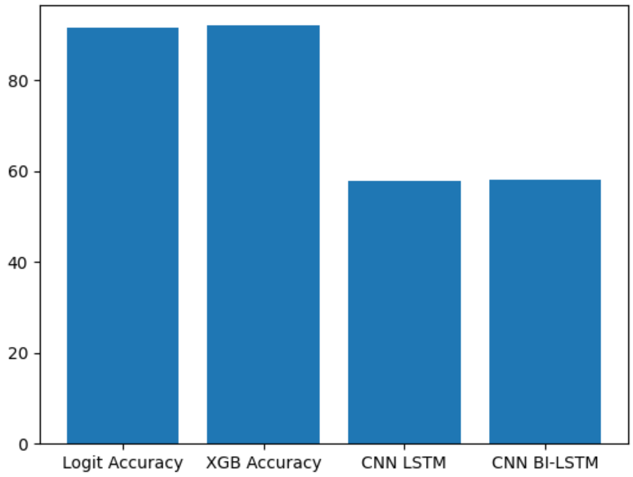

FIGURE 8. Comparison of four different Algorithms

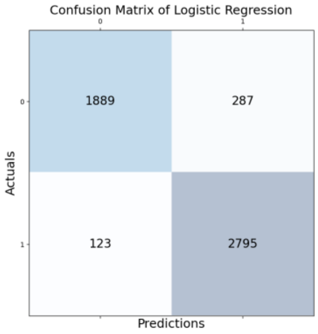

FIGURE 9. Confusion Matrix of Logistic Regression

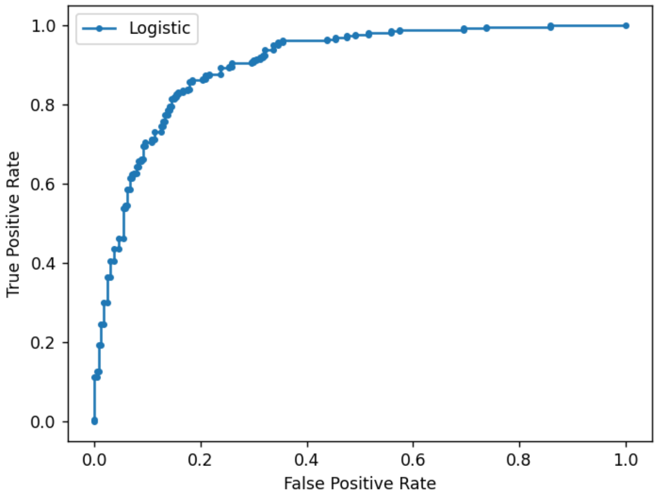

FIGURE 10. ROC Curve for Logistic Regression

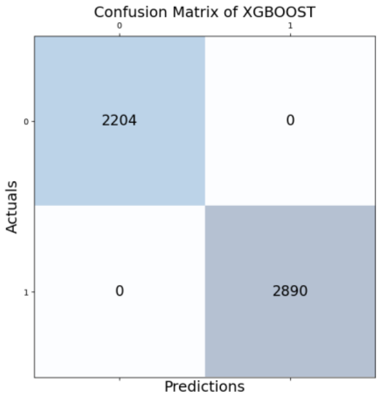

FIGURE 11. Confusion Matrix of XG Boost

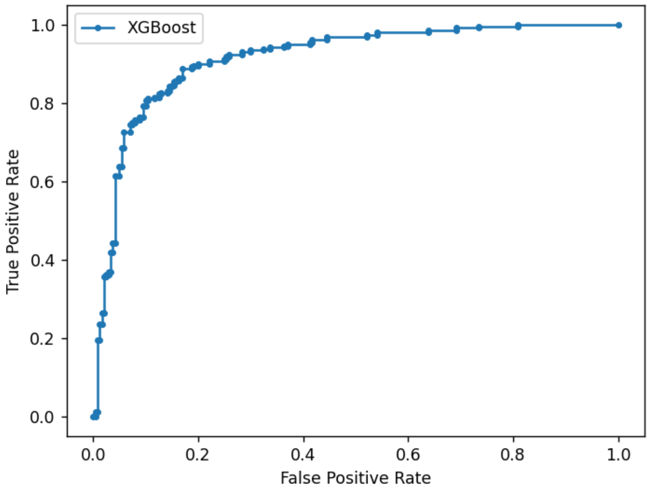

FIGURE 12. ROC Curve for XG Boost

6. Conclusion

In this study, the problem of phishing assaults is taken into consideration, and a useful model is given

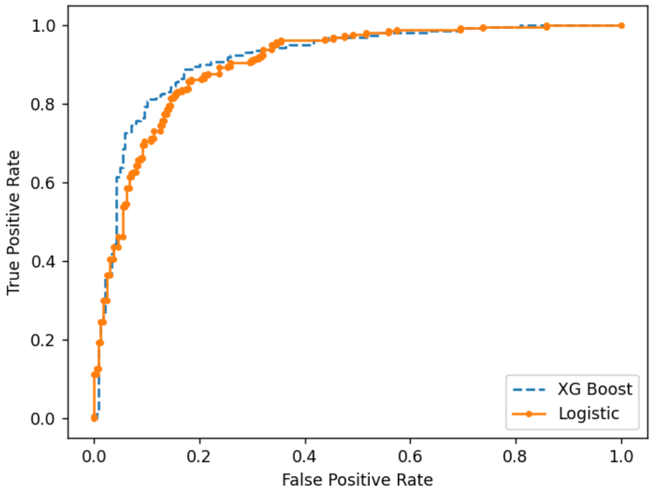

FIGURE 13. **Comparison of Logistic Regression and XG Boost ROC Curves**

using the CNN LSTM, CNN-Bi-LSTM, Logistic regression, and XG Boost algorithms, which integrate deep neural networks and machine learning to identify and categorise malicious URLs. The Logistic regression and XG Boost algorithm model produce an excellent accuracy in detecting the comparison of phreaking URLs to the most widely used LSTM model. The model's suitability is demonstrated by the analysis, which yields 92% accuracy along with performance data. To make this application accessible to everyone, we can further expand it as a website.

References

Al-Ahmadi, Saad, Afrah Alotaibi, and Omar Alsaleh. "PDGAN: Phishing Detection With Generative Adversarial Networks". (2022): 10–10.

Anil, Dr, et al. "Detection of Phishing Websites based on Feature Extraction Using Machine Learning". (2020): 7–7.

Bhavani, P Amba, et al. "Phishing Websites Detection Using Machine Learning". (2021): 2021–2021.

Buber, Ebubekir, Onder Demir, and Ozgur Koray Sahingoz. "Feature selections for the machine learning based detection of phishing websites". *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)* (2017): 1–5.

Choudhary, Abu Saad, et al. "Detection and Prevention of Phishing Attacks". 1 (2021).

Dharani, M, et al. "Detection of Phishing Websites Using Ensemble Machine Learning Approach". (2021).

Elsadig, Muna, et al. "Intelligent Deep Machine Learning Cyber Phishing URL Detection Based on BERT Features Extraction". *Electronics* 11.22 (2022): 3647–3647.

Krishna, Arathi, et al. "Phishing Detection using Machine Learning based URL Analysis: A Survey". (2021): 2278–0181.

Kumar, Naresh, et al. "Detection of Phishing Websites using an Efficient Machine Learning Framework". 9 (2020): 2278–0181.

Mahajan, Rishikesh and Irfan Siddavatam. "Phishing Website Detection using Machine Learning Algorithms". *International Journal of Computer Applications* 181.23 (2018): 45–47.

Prabakaran, Manoj Kumar, Abinaya Devi Chandrasekar, and Parvathy Meenakshi Sundaram. "An enhanced deep learning-based phishing detection mechanism to effectively identify malicious URLs using variational autoencoders". *IET* (2022): 2022–2022.

Tang, Lizhen and Qusay H Mahmoud. "A Deep Learning-Based Framework for Phishing Website Detection". *IEEE Access* 10 (2022): 1509–1521.

Yang, Peng, Guangzhen Zhao, and Peng Zeng. "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning". *IEEE Access* 7 (2019): 15196–15209.

Embargo period: The article has no embargo period.

To cite this Article: , Maneesha K, Rajasekhar K , Prema Latha K , and Venkata Prasad N . "**Detection of Phreaking Website Using Various Algorithms** ." International Research Journal on Advanced Science Hub 05.05S May (2023): 305–313. http://dx.doi.org/10.47392/irjash.2023.S041