ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 10, Issue 5, May 2023

A Hybrid Machine Learning Model for Classifying Phishing Uniform Resource Locators

[1] Anietie Ekong, [2] Godwin Ansa, [3] Odikwa Ndubuisi Henry, [4] Emmanuel Adabra

[1] [2] [4] Department of Computer Science, Akwa Ibom State University, Akwa Ibom State, Nigeria
[3] Department of Computer Science, Abia State University, Abia State, Nigeria
Corresponding Author Email: [1] anietieekong@aksu.edu.ng, [2] godwinansa@aksu.edu.ng,
[3] ndubuisi.odikwa@abiastateuniversity.edu.ng

Abstract— ***Phishing remains a major concern for security specialists over the years. Phishing attack are aimed at tricking people into giving out sensitive or confidential information using social engineering method. So far, machine learning (ML) algorithms like Artificial Neural Network (ANN), Decision Tree (DT), Support Vector Machines (SVM) Logistic Regression (LR) etc, have offered the most effective means of classifying scam Uniform Resource Locators (URLs). The main focus of this work is to classify URLs or sites into two classes: legitimate (0) or phishing (1). A total number of 8,391 legitimate URLs and 7,727 phishing URLs were sourced from phishTank. After data preprocessing, the dataset was split into training set and testing set in the ratio of 8:2. The trained dataset was fit into the LR and SVM algorithm. The performance of the SVM and LR algorithms was tested using the test dataset and their outcomes were used as an input to the Stacking Model in order to improve the classification accuracy. This model was trained and tested using tools developed from Python programing language, Jupyter notebook IDE and Python external libraries. A classification accuracy of 90% and 95% were recorded for LR and SVM respectively. The hybrid Model which is an enhanced model has an accuracy of 97%. Based on the above metrices, the Stacked Model can be used to effectively detect scam URLs with high accuracy.***

Keywords: Phishing, Scam, Uniform Resource Locator, Stacked Model, Logistic Regression, Support Vector Machine, Machine Learning.

I. INTRODUCTION

Phishing has been of serious concern for security specialists over the years especially with the increasingly ease of developing phony websites that seem to be identical to the real one. Information security needs to be prioritized by every organization that is growth oriented and attackers have devised different strategies to access organizations' data irrespective of where they are stored [1]. Although experts can spot fake websites, many online users cannot do same. As a result, they are vulnerable and potential victims of phishing scams. The attacker's goal is to steal the user's bank account, passwords and other personal information. Updating of blacklisted URLs and Internet Protocol (IP) to an antivirus database, otherwise called the "blacklist" approach, is one of the most used methods of identifying phishing websites. To get around blacklists, attackers utilize obfuscation and other basic tactics like fast-flux, where proxies automatically automate development of new URLs among other means. This method's main flaw is that it cannot identify zero-hour phishing attacks. Heuristic-based detection attempts to detect phishing URLs by using common features seen in phishing attacks. While this approach can detect zero-hour phishing assaults, all attacks may not poses these traits and detection's false positive rate is high [5].

II. LITERATURE REVIEW

According to Internet records, the phrase "phishing" was first used in January 2, 1996. This word was used in the AOHell Usenet newsgroup. Since 2012, phishers have been increasingly using HTTPs on their websites. A user clicking on a phishing link will be directed to a website that tries to deceive visitors into giving over valuable credentials or personal information. According to Ekong (2023) [1], machine learning models make it possible for variables to be classified with or without human intervention. Despite the existence of machine learning (ML) algorithms, the existing systems were unable to detect accurately the increasing number of new-born phishing URLs. Raju et al. [7] adopted an approach that focused on identifying phishing attempts by utilizing the Blacklist and the WHOIS database. A few website parameters that were picked to aid in detecting phishing sites were domain identification, URLs, source code, security/encryption, page layout and contents, web address bar and social human component. Rao et al. [8] developed a system that divided websites into three groups: benign (websites that are safe or legitimate and provide customers with basic services), spam (these are the websites that bombard users with spammy surveys, advertising and malware) and malicious (these are phishing websites, which are run by attackers to steal personal information while appearing to be regular websites). Garje et al. [5] proposed the model, Machine learning methods like K-Nearest

Neighbour (KNN), Naive Bayes, Decision trees, and Gradient Boosting used to detect phishing websites. Tubyte et al. [11] demonstrated the categorization of phishing and authentic URLs using supervised machine learning algorithms and defined the categorization of phishing URLs as a two-class issue. Five distinct algorithms—Random Forest, Support Vector Machine, Logistic Regression, Linear Discriminant Analysis, and Decision Trees—were used with two datasets. Despite the dataset, each model's accuracy was greater than 91%. On both datasets, the RF algorithm delivered the greatest overall accuracy result. Sanchez-Paniagua et al. [9] put out an improved blacklist approach for phishing detection that would shield users from bogus login forms. A large dataset containing URLs which are is assumed to be a good representation of real-world phishing attacks is used and then applied deep learning and machine learning tools trained on phishing and real home URLs. Dutta [4] focused on the classification aspect of the phishing approach, where phishing websites are thought to automatically classify websites into a preset range of class values based on a variety of attributes and the class variable.

Aljofey et al. [3] developed a machine learning-based method that uses the URL and HTML properties of a given webpage to identify phishing websites that is is entirely client-sided with zero reliance on outside services.

Vaneeta et al. [12] developed a phishing websites detection mechanism based on ML classifiers with wrapper features selection to address the problem of phishing.

The overall performance of the existing system models are not adequately efficient and not encouraging in terms of detection accuracy and speed, hence the proposed LR and SVM in classifying URL's.

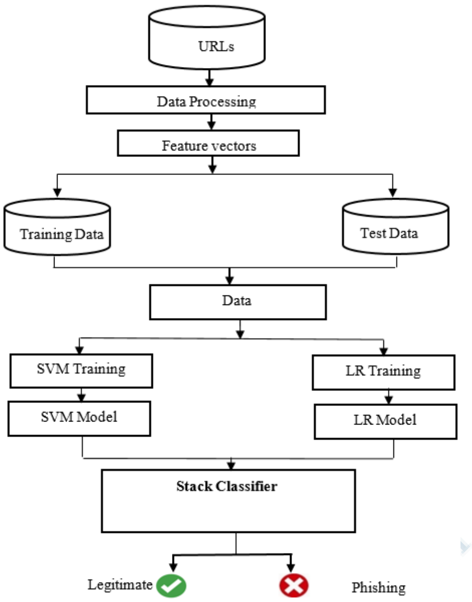


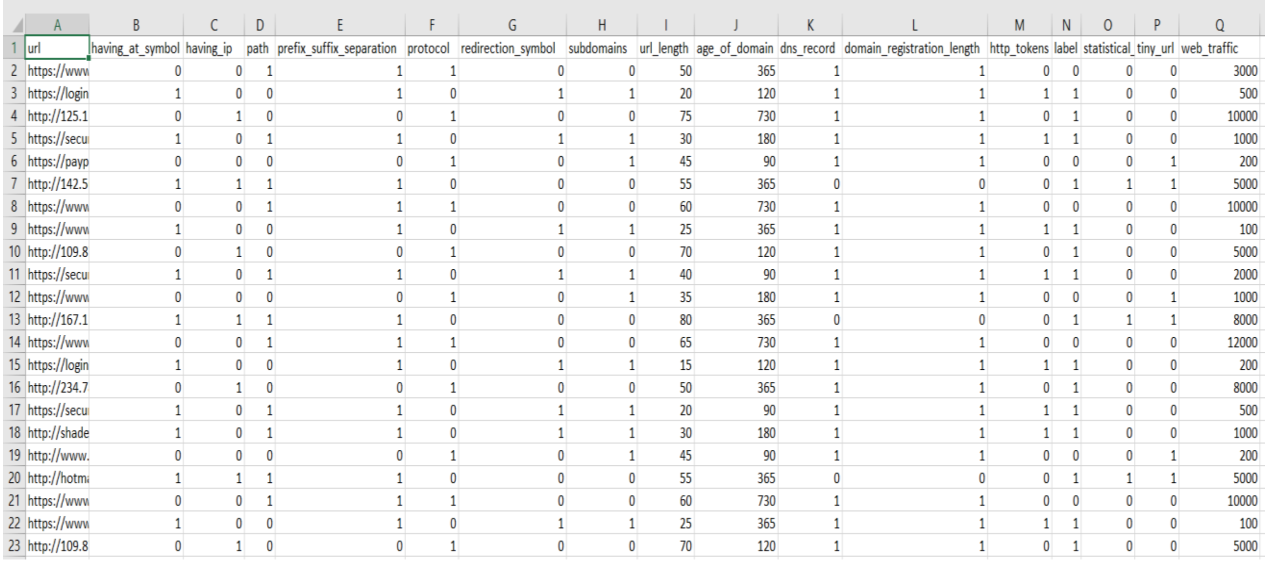
Fig. 1: Architecture of the Proposed System

III. METHODOLOGY

This work employed the ML algorithms of logistic regression and support vector machines. The system architecture is as shown in Figure 1. Data is collected from the phishing tank and then pre-processed before being fed into the different system's phases. Python programming Language (Spyder Integrated Development Environment) will then be used as a tool in each steps with each of the algorithm applied as due.

A. Dataset from Phishtank

The dataset (phishing and non-phishing URLs) was gotten from phishtank.com, an anti-phishing site, which is a collaborative repository of phishing data with thousands of phishing URLs.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	url	having_at_symbol	having_ip	path	prefix_suffix_separation	protocol	redirection_symbol	subdomains	url_length	age_of_domain	dns_record	domain_registration_length	http_tokens	label	statistical_	tiny_url	web_traffic
2	https://www	0	0	1	1	1	0	0	50	365	1	1	0	0	0	0	3000
3	https://login	1	0	0	1	0	1	1	20	120	1	1	1	1	0	0	500
4	http://125.1	0	1	0	0	1	0	0	75	730	1	1	0	1	0	0	10000
5	https://secu	1	0	1	1	0	1	1	30	180	1	1	1	1	0	0	1000
6	https://payp	0	0	0	0	1	0	1	45	90	1	1	0	0	0	1	200
7	http://142.5	1	1	1	1	0	0	0	55	365	0	0	0	1	1	1	5000
8	https://www	0	0	1	1	1	0	0	60	730	1	1	0	0	0	0	10000
9	https://www	1	0	0	1	0	1	1	25	365	1	1	1	1	0	0	100
10	http://109.8	0	1	0	0	1	0	0	70	120	1	1	0	1	0	0	5000
11	https://secu	1	0	1	1	0	1	1	40	90	1	1	1	1	0	0	2000
12	https://www	0	0	0	0	1	0	1	35	180	1	1	0	0	0	1	1000
13	http://167.1	1	1	1	1	0	0	0	80	365	0	0	0	1	1	1	8000
14	https://www	0	0	1	1	1	0	0	65	730	1	1	0	0	0	0	12000
15	https://login	1	0	0	1	0	1	1	15	120	1	1	1	1	0	0	200
16	http://234.7	0	1	0	0	1	0	0	50	365	1	1	0	1	0	0	8000
17	https://secu	1	0	1	1	0	1	1	20	90	1	1	1	1	0	0	500
18	http://shade	1	0	1	1	0	1	1	30	180	1	1	1	1	0	0	1000
19	http://www.	0	0	0	0	1	0	1	45	90	1	1	0	0	0	1	200
20	http://hotm	1	1	1	1	0	0	0	55	365	0	0	0	1	1	1	5000
21	https://www	0	0	1	1	1	0	0	60	730	1	1	0	0	0	0	10000
22	https://www	1	0	0	1	0	1	1	25	365	1	1	1	1	0	0	100
23	http://109.8	0	1	0	0	1	0	0	70	120	1	1	0	1	0	0	5000

Fig. 2: Snapshot of the Dataset

B. Data Preprocessing

Preprocessing data is a crucial step in achieving higher accuracy. Data from the outside world is often noisy, with missing and erroneous values. Data cleaning identifies elements of datasets that are incomplete, erroneous, imprecise, or unsuitable [10]. The data then becomes consistent with the system's other data sets.

Here, the dataset shown in figure 2 is separated into properties that are dependent and independent variables. Missing values are replaced with the mean value of that specific characteristic.

C. Training and Test Dataset

The total dataset is divided into two parts; the training set is 80% of data and also the test set is 20% of data. The training set and the testing set are uses to train and evaluate the model respectively. The preprocessed data is fed to the LR and DT classifiers for training using the training dataset.

D. Logistic Regression (LR)

LR is a supervised Machine learning approach that is used to predict the likelihood of an event happening based on dataset of independent variables. The dependent variable is of binary nature, with data represented as 1 or 0 which stand for success/yes or 0 failure/no). In our example, 1 (phishing) and 0 (no phishing).

Given Odds $=\frac{p(events)}{1-p(events)}$ this defines the ratio that the probability occurs by the ratio that the probability does not occur

Let $Pr(y=1|x)=p(x)$ (1)

where $X \varepsilon R$

$P(X) \varepsilon [0,1]$ where X is the predictor domain

Using the logistic function to get the s-shaped curve that lies between 0 or 1

$$P(X)= \frac{1}{1+e^{-\beta X}} \text{where } \beta x= \log\left(\frac{p(x)}{1-p(x)}\right) \quad (2)$$

Logistic Regression (LR) Algorithm

```
Step 1: Start [Logistic regression (LR)]
Step 2: Load dataset
Step 3: divide data into training and test set
Step 4:  Split data into 2 (0 and 1, not-phishy or phishy)
Step 5: For Each in the split of 2-fold
    (a). take the K-th fold of the data as the test data, and the rest as training in the range (0,1)
    (b). for j=1....n
    (c). do
    (d). Linearly scale the features for all i in the training set in the range (1,0)
    (e). end
Step 6: Cluster the training data for k categories.
Step 7: Train a new logistic regression model.
Step 8: Use the trained regression model to predict on the test set and evaluate the performance.
Step 9: Repeat Step 5-8 until suitable accuracy is reached.
Step 10: Stop
```

E. The Logistic Model

The process of identifying the coefficients that correspond to the optimal value of the cost function is known as model fitting. Using the logistic regression () function in the sklearn Python library, the logistic regression model was run and utilized to construct an LR classifier object.

F. The SVM Model

Selection of SVM models establishing the SVM hyper parameters, which includes a kernel function and its parameters—is a crucial, yet computationally intensive operation because badly adjusted parameters might influence SVM performance, automatic model selection is critical. The SVM training procedure entails determining a hyperplane to divide the training data into two groups. Support vectors are a typically small subset of vectors from the training set that specify its location (SVs). Knowing which vectors are chosen as SVs makes the SVM decisions more understandable. After the model has been trained, the accuracy of the SVM is determined by classifying using the test data sample.

The SVM Algorithm

```
Step 1: Start
Step 2: Load dataset (phishing and non-phishing)
Step 3: Split dataset into two (training and test)
Step 4: Select training dataset for learning.
Step 5: Split training data into 2 classes (phishing and non-phishing)
Step 6: Find mapping between every URL to classes
Step 7: Find all possible values for every URL and that corresponding possible classes.
Step 8: Count values of each URL which belongs to unique class
Step 9: Similarly select other URL for next level in from remaining URL on the basis of minimum number of  values having unique class.
Step 10: Use the trained model to predict on the test set evaluate the performance
Step 11: Stop
```

G. The hybrid (stack) model

After training the LR and SVM classifiers using the extracted feature vectors (x), the output(y) of LR and SVM classifiers can either be 0(legitimate) or1(phishy), 1 and 0, 0 and 0, 1 and 1 respectively depending on the target URL features. After this stage new feature vectors are formed using both classifiers' outputs (y1 and y2) and they serve as input into the stack (hybrid) classifier. Figure 3 shows the stacking process.

A uniform assessment procedure is necessary to compare the varied outcomes to each other. Table 1 shows the possible classification scenarios based on the input received.

Table 1: Evaluation of Algorithms

Scenarios	Y	X	Classification
True Positive	1	1	True
True negative	0	0	False
False Positive	1	0	False
False Negative	0	1	False

The error rate can be calculated using the general equation:

$$\text{Error rate} = \frac{TP+TN}{TP+FP+TN+FN} \qquad (3)$$

The sensitivity or also called the True Positive Rate (TPR) is given by:

$$\text{TPR} = \frac{TP}{TP+FN} \qquad (4)$$

The False Positive Rate (FPR) or also called Fall-Out is given by:

$$\text{FPR} = \frac{FP}{FP+TN} \qquad (5)$$

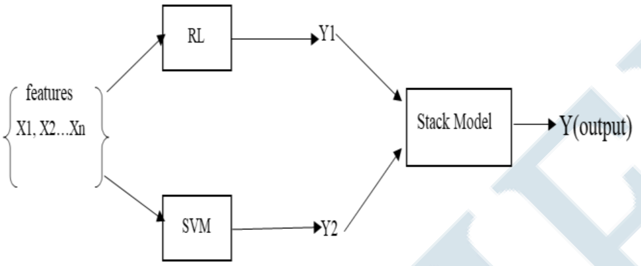


Fig. 3: The Stack Model

IV. RESULTS AND DISCUSION

To determine genuine and fraudulent URLs, LR, SVM and stack model were used. This was done after the data preprocessing and training. The amount of data used are as shown below

$$\text{Training dataset= 80\% of total dataset} = \frac{80*16118}{100} = 12{,}894$$

$$\text{Testing dataset= 20\% of total dataset} = \frac{20*16118}{100} = 3{,}223$$

Figures 4, 5 and 6 together with tables 2 , 3 and 4 shows for results of LR, SVM and Stacked model used to evaluate the classification accuracy of the models.

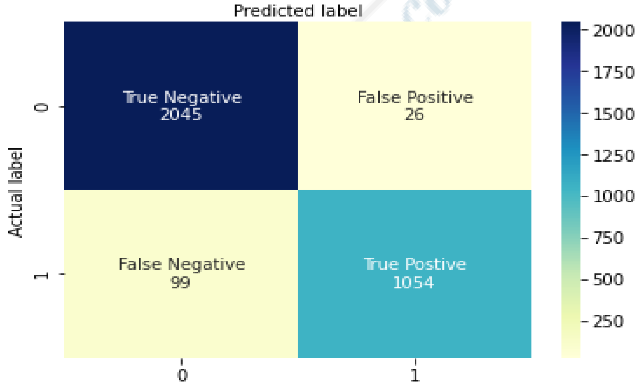


Fig. 4: The SVM Confusion Matrix

Table 3: Matrix table of LR

	precision	Recall	F1-score	Support
Legitimate (0)	0.91	0.95	0.93	2071
Phishing (1)	090	0.82	0.86	1153
Accuracy			0.91	3224
Macro avg	0.90	0.89	0.89	3224
Weighted avg	0.91	0.91	0.90	3224

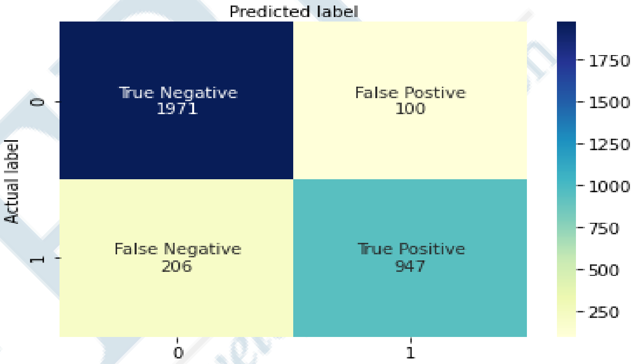


Fig. 5: The LR Confusion Matrix

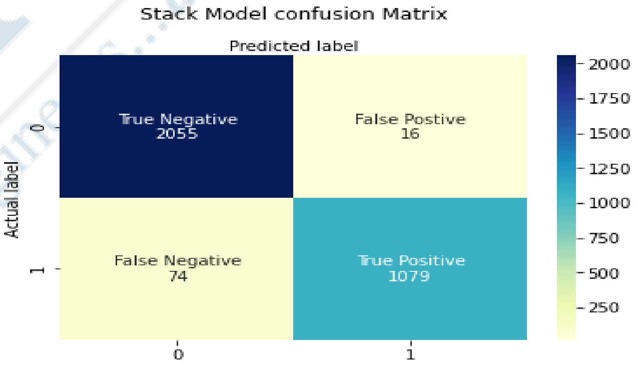


Fig. 6: The Stack Model Confusion Matrix

Table 2: Matrix Table of SVM

	Precision	Recall	F1-score	Support
Legitimate (0)	0.95	0.99	0.97	2071
Phishing (1)	0.98	0.91	0.94	1153
Accuracy			0.96	3224
Macro Avg.	0.96	0.95	0.96	3224
Weighted Avg.	0.96	0.96	0.96	3224

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \qquad (6)$$

$$\text{Recall} = \frac{1054}{1054+26} = 0.99$$

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TOTAL}} \qquad (7)$$

$$\text{Accuracy} = \frac{2045+1054}{2045+26+99+1054} = 0.96$$

$$\text{Recall} = \frac{1079}{1079+16} = 0.99$$

$$\text{Accuracy} = \frac{2055+1079}{2055+74+1079+16} = 0.97$$

Table 4: Matrix table for Stack Model

	Precision	Recall	F1-score	Support
Legitimate (0)	0.97	0.99	0.98	2071
Phishing (1)	0.98	0.93	0.96	1153
Accuracy			0.97	3224
Macro Avg.	0.98	0.96	0.97	3224
Weighted Avg.	0.97	0.97	0.97	3224

$$\text{Recall} = \frac{1079}{1079+16} = 0.99$$

$$\text{Accuracy} = \frac{2055+1079}{2055+74+1079+16} = 0.97$$

This work is centered on detecting phishing URLs using the SVM and LR data classification model. The system model is trained to classify URLs that are phishy loaded from python sklearn library using the PhishTank dataset. The results in this research work shows how the hybrid SVM - LR models were able to recognize phishing URLs with greater accuracy.

From the result shown in, it can be seen that the model actually classified 3067 URLs to be positive and 134 URLs to be negative.

V. CONCLUSION

This work was aimed at classifying URLs as either phishing or not towards securing the majority of web users from cyber attackers using machine learning algorithms. A Machine Learning (ML) Stacked Model is train from the classification outcome of SVM and LR ML algorithms to classify URLs to be legitimate or Phishing based on certain features extracted from the URL. This model, to a significant extent, has addressed the issue encountered by many internet users which is the issue of URL's trust. The prior methods used by phishing detection systems had not proven to be as effective. The results derived from using SVM and LR model have an accuracy of 91% and 96% respectively while the hybrid model have an accuracy of 97%. Based on these results, the developed model performed better than the LR and SVM and other individual algorithm classification in terms of accuracy in detecting phishing URLs.

REFERENCES

[1] Anietie Ekong (2023). Evaluation of Machine Learning Techniques towards Early Detection of Cardiovascular Diseases, *American Journal of Artificial Intelligence*, 7(1): 6-16.

[2] Anietie Ekong, Odikwa, Abasiama Silas, Imou Douglas(2022). Hybridized Cryptography and Cloud Folder Model (CFM) for Secure Cloud-Based Storage, *American Journal of Computer Science and Technology*, 5(3): 178-183

[3] Aljofey, A., Jiang, Q., Rasoo, A., Chen, H., Liu, W., Qu, Q. and Wang, Y. (2022). An Effective Detection Approach for Phishing Websites Using Url And Html Features. *Scientific Report*, *12*(8842), 1-19.

[4] Dutta Ashit (2021) Detecting phishing websites using machine learning technique. *PLOS ONE* 16(10).

[5] Garje, A., Tanwani, N., Kandale., S., Zope, T. and Gore, S. (2021). Detecting Phishing Websites Using Machine Learning. *International Journal Of Creative Research Thoughts (Ijcrt)*, *9*(11), 243-246.

[6] Mahmoud Khonji, & Youssef Iraqi. (2013). Phishing Detection: A Literature Survey IEEE.

[7] Raju, R., Likhitha, S., Deepa, N. and Sushma, S. (2022). Survey on Phishing Websites Detection using Machine Learning. *International Journal for Research in Applied Science & Engineering Technology (IJRASET*), *7.538*(10), 2379-2381.

[8] Rao, K.V., Reddy, J. M., and Prasad, G. L. (2020). An Approach For Detecting Phishing Attacks Using Machine Learning Techniques. *Journal of Critical Reviews*, *7*(18), 321-324.

[9] Sanchez-Paniagua, M., Fernandez, E., Alegre E., Al-Nabki, W. and Gonzalez-Castro, V. (2022). Phishing URL Detection: A Real-Case ScenarioThrough Login URLs. *IEEE Access, 10*, 42949 – 42960.

[10] Theiler, J. & Cai, D. M. (2003). Resampling approach for anomaly detection in multispectral images. In Proceedings of SPIE, 5093, 230-240.

[11] Tubyte, M., and Paulauskaite-Taraseviciene, A. (2021). Research on phishing email detection based on URL parameters using machine learning algorithms. *CEUR Workshop Proceedings*, *2915*(3), 18-26.

[12] Vaneeta, M., Pratik, N., Prajwal D., Pradeep, S. and Suhas, K. (2020). Detection of Phishing Websites Using Machine Learning Techniques. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *7*(6), 117-123.