JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

InboxGuardian: A Synergistic Approach Combining BERT and GANs for Phishing Detection in email

[1st] DeekshaTiwari, 2[nd] Astha Shukla , 3[rd] Avika Gupta 4 Akshita Dixit

Noida institute of engineering and technology , Greater Noida

Abstract : Phishing emails frequently result in monetary losses, data breaches, and security flaws, they are becoming a bigger worry for people as well as companies. Phishing emails are usually designed to trick the recipient into disclosing private information, including financial information, login credentials, and personal identification numbers. Effectively delivering emails has come to be an enormous cybersecurity concern. to detect phishing emails, this article uses machine learning algorithms to find suspicious patterns and characters that are frequently present in false emails.

Our method entails preprocessing email content, such as subject lines, body text, and metadata, with the goal to extract useful features that may be indicative of phishing. The paper also discusses the application of natural language processing (NLP) techniques for feature extraction, which aids in analysing the semantic content of emails for patterns like urgency, dubious links, or misleading language frequently employed in phishing attempts. By analysing the content's semantic meaning using Natural Language Processing (NLP) resources. The study builds prediction systems that can differentiate between malicious and legitimate emails using a variety of machine learning models, such as Random Forest classifiers, Decision Trees, and SVM Results from experiments shows that machine learning systems can detect phishing emails with excellent accuracy even when working with relatively little datasets. This illustrates how well these models work at spotting typical phishing traits including misleading wording, dubious URLs, and odd metadata patterns.

The system can adjust to new phishing tactics and preserve its detection capabilities by regularly feeding the models new data. This method guarantees long-term dependability in addition to increasing precision. Furthermore, the suggested approach provides an effective and scalable solution that can be included into current email security systems.

.

IndexTerms **- Cybersecurity, email security, phishing emails, phishing attacks, email detection, feature extraction, natural language processing (NLP), machine learning, Support Vector Machine (SVM), Decision Trees, Random Forest.**

I. INTRODUCTION

Phishing is a successful kind of fraud when the perpetrator uses false pre tenses to trick recipients and grab private information. Phishing emails may lead people astray. to click on an attachment or link to a website where they must enter private information, such as credit card numbers or passwords. The phisher distributes the messages to thousands of individuals, and while often just a small portion of receivers fall for the scam, the sender can nevertheless earn greatly from this. Emails were used by American hackers in 2006 to create "baits" that could help users to obtain the usernames and passwords of American online accounts. Since then, phishing techniques have evolved, making it more difficult to spot phony emails.

Since the coronavirus outbreak in 2019 (COVID-19), phishing attacks have gained significant attention. From September 2020 till the present, Numerous investigations of phishing attacks in relation to COVID-19 have been started [3]–[5]. Phishers typically use language and information about the COVID-19 outbreak to find their potential victims [1]. According to the data, phishing assaults and the resulting harm increased significantly during the COVID-19 outbreak. Emails and other messaging applications are among the most popular phishing attack vectors. This study focuses on phishing attempts via email communication [7], [8] because phishers prefer email attacks over other techniques since they are hard to detect [6].Phishing email detection in the proposed method can be characterized as a classification problem with two categories: phished and ham. One area of artificial intelligence is machine learning.intelligence that grants the system the capacity to learn without explicit programming. Our model classifies data using supervised machine learning algorithm. Using existing instances, supervised learning systems forecast the characteristics of unknown data. These algorithms learn from data iteratively and are a subset of machine learning algorithms. This is how the rest of the work is structured. The systems currently in use for identifying phishing emails are covered in Section 2. The indicated system, the approaches employed, and a brief summary of the features are all covered in the third section.

By examining vast amounts of email data and finding patterns suggestive of fraudulent activity, machine learning and data mining techniques have become effective tools for phishing email detection. Features including title lines, body text, URLs, and content can be extracted from email messages and used to train machine learning models to Distinguish between phishing and authentic emails. The demand for automated, scalable systems that can provide real-time detection with little human intervention has arisen as a result of the rising reliability of manual detection techniques due to the complexity of phishing attempts. By decreasing false positives and increasing detection accuracy, these machine learning algorithms hope to improve security.This content addresses the applies of machine learning algorithms for phishing email detection, concentrating on detecting key characteristics of phishing efforts and classifying emails accordingly. It shows the usefulness of feature extraction, where natural language processing (NLP) techniques play a crucial role in assessing the content and structure of the emails. This study attempts to offer a workable and efficient defence against phishing attacks by thoroughly analysing various machine learning models. The findings of this study aid in the creation of stronger email security systems that are able to immediately identify and stop phishing attempts. [4].

The following research questions are intended to be addressed by the survey:1. What are the main areas of study for NLP-based phishing email detection? 2. Which machine learning algorithms are most frequently employed to create models for phishing email detection? 3. Which optimization strategies are most frequently employed to identify phishing emails? 4. Which feature extraction techniques are used in NLP studies for phishing email detection? 5. Which NLP methods are most frequently applied in research on phishing email detection?

II. RELATED WORK

Andronicus et al. utilized a random forest machine learning classifier for the purpose of classifying phishing emails. Their goal was to enhance accuracy while reducing the number of features needed for classification. They introduced a content-based phishing detection method that demonstrates high accuracy. In [2], the authors suggested a model that relies on features extracted from the email's header and HTML body, which are then classified using a feedforward neural network. The findings reveal an impressive classification accuracy of 98.72%.In [3], a dataset containing over 7000 emails utilizes various features. An overall accuracy of 99.5% has been attained. Gilchan Park et al. focused on extracting strong features to differentiate between legitimate and phishing emails. A comparison was made regarding the syntactic similarity of sentences and the distinctions in subjects and objects of target verbs between phishing and legitimate emails. The article “Email Phishing: An Open Threat to Everyone” examines various phishing techniques and offers recommendations for users on how to avoid falling victim to fraudsters. C. Emilin Shyni et al. introduce a methodology that integrates natural language processing, machine learning, and image processing. They utilize a total of 61 features. Their approach achieves a classification accuracy exceeding 96% through the use of a multi-classifier. In the study “Detection of Phishing Emails Using Decisive Value Features”, 18 features are extracted, and the proposed algorithm classifies each email based on the presence of flags and the importance of the features. The findings indicate that high accuracy can be achieved by employing the most effective features from the 18 features extracted for classification. In “Phish-IDetectore,” the authors examine the characteristics of Message-IDs and implement n-gram analysis on these Message-IDs. The authors of [2] put forth a model based on features that were retrieved. Those are categorized using feed forward neural networks and that show up in the email's HTML body and header. The findings show a classification accuracy of 98.72%. More than 7000 emails and a variety of attributes are employed in the dataset in [3]. A 99.5% overall accuracy is attained. The goal of Gilchan Park et al. was to extract strong traits that would allow them to distinguish between phishing and authentic emails. Phishing emails and authentic emails are compared for syntactic similarity in sentences as well as differences in the subjects and objects of target verbs. The various phishing tactics are examined in "Email Phishing: An Open Threat to Everyone," along with advice on how users can prevent themselves from becoming victims of scammers..

III. PROPOSED MODEL :

For the purpose of classification, 9 features were extracted from all emails in a self-made dataset of n number of phished emails and m number of ham emails . These features are fed into the classifiers and results noted. Aim is to use the least features to develop a system with higher accuracy and study the variation of features.

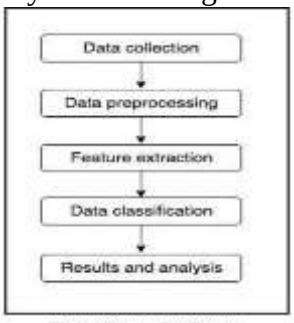


Figure 1: Process flow diagram

3.1 Features

The traits that were retrieved will be described in this section.3.1.1 Based on links Domain count: Attackers add subdomains to the links to give the impression that they are authentic. The link's dot count rose as subdomains were added. According to Emigh' s suggestion, a valid email should have no more than three [3] number dots. This is a binary feature, meaning that an email would be deemed phished if it contained a link with more dots than three. The quantity of links: Since the sender wants to trick the recipient into visiting an unauthorized website, phished emails typically have more links than ham emails. This feature is constant.

3.2 Tag based Presence of Javascript:

Presence of Javascript in an email suggests that the sender is either trying to hide information or turn on specific browser modifications [9]. This feature is binary. The email is regarded as phished if it contains the <script >tag. Form tag presence: Phishing emails include forms placed in them to collect user information. This is a binary feature, meaning that if a form tag is present, the email is a phishing attempt. HTML is present: In contrast to regular text emails, HTML emails let the sender to insert hyperlinks and embedded graphics. If the email contains an HTML tag, it is seen as phishing.

3.3 Based on words:

The quantity of action words When action words are included in emails, it shows whether the sender anticipates a response. user to carry out specific tasks, like clicking on a link, completing a form, or entering specific data. PayPal is present: The sender frequently poses as a member of organizations that appear to be trustworthy. If the word "paypal" appears in the email's links or in the "from" line, it indicates that the sender is connected to PayPal. This feature is binary. The binary characteristic "bank" indicates that the mail contains banking-related information. The sender would either be pretending to be a member of the banking organization or viewing the reader's credentials. Presence of word account: This would suggest that the email is looking for email related to an account. It can be a social media account or bank account etc. It is a binary feature. Combining the three types of features described in 3.1.1, 3.1.2 and 3.1.3, a total of 9 different features are obtained which are extracted with the help regular expressions and Python's NLTK (natural language toolkit).

3.2 -classifier

The classifiers utilized will be thoroughly described in this section.

3.2.1 Vector Machines for Support

Due to its excellent performance and quick speed, SVM is a well-liked supervised technique for text classification. It produces a hyperplane, or two-dimensional line, that best divides the categories based on the given training data. The decision boundary is the name given to this hyperplane. A set of features, such as the existence or absence of a certain term, represent input in phishing detection, and an output of 1 or -1 shows whether the email is phished or not.

3.2.2 **Naive Bayes**

The naive bayes classifier belongs to the family of probabilistic algorithms and used bayes theorem to categorize sample data. Bayes theorem : Given a hypothesis H and evidence E, Bayes' theorem states that the relationship between the probability of the hypothesis P(H) before getting the evidence and the probability P(H|E) of the hypothesis after getting the evidence is :

P(H/E)=P(E/H)*P(H)/P(E)The probability of each category is calculated and outputs the one with highest probability.

3.2.3 Forest of Random

An ensemble learning technique for classification, regression, and other problems is called a random forest or random decision forest. These work by building a large number of decision trees during training and producing the class that represents the mean prediction (regression) or the mode of the classes (classification) of the individual trees. The tendency of decision trees to overfit to their training set is compensated for by random decision forests.

3.2.4 Logistic regression

The likelihood of a binary response based on one or more predictor (or independent) variables
(features) is estimated using the binary logistic model. It makes it possible to state that a risk factor's existence raises the likelihood of a particular result by a particular percentage.

3.2.5 Perceptron with Votes

All weight vectors are stored by this algorithm, which then allows them to vote on test samples. It is fast, easy and has been stated to be as In many cases, they are as good as support vector machines.

3.3 Information Set Of the 1605 emails in the collection, 414 are phished and 1191 are ham emails. Phishing emails are a collection of emails from several sources, while ham emails are gathered from a publicly accessible dataset

OUTCOMES AND CONVERSATION

After being divided, the dataset with the retrieved characteristics is fed into five classifiers, and the outcomes are recorded. The original data sample was divided into a training set and a test set using the 10-fold cross-validation technique.

K-fold cross validation: This technique divides the dataset into k mutually exclusive sections of roughly similar sizes at random [10]. This is followed by the model being
trained and tested k times; of the k samples, k-1 subsamples are utilized as training data and one subsample is kept as validation data for the testing model. The most accurate classifications are found with logistic, SVM, and tree-based classifiers. Various performance metrics, which are explained in this section, are used to assess the performance of various classifiers. SVM and Random Forest are found to categorize the dataset with the maximum accuracy of 99.87%. Our model is assessed using the performance metrics listed below: Precision: It is defined as the fraction of retrieved objects that are relevant [9]. In our situation, it refers to the percentage of emails that are correctly identified as phished but are, in fact, phished.

PRECISION=TP/TP+FP

Recall: Recall is defined as the percentage of retrieved relevant objects compared to the total number of relevant objects in the dataset [9], or the percentage of classified phished emails that are actually phished from the dataset. RECALL=TP/TP+FN

F-MEASURE: The harmonic mean of precision and recall is known as the F-measure

F-MEASURE=2*PRECISION*RECALL /(PRECISION+RECALL)

False Positive Rate: The percentage of ham mails incorrectly classified by the model as phished. Let Nf be the number of ham emails incorrectly classified as phished and number of ham emails is H then false positive rate can be calculated as:

FP=Nf/H

False Negative Rate: percentage of phished emails that were incorrectly classified by the model as ham. Let Ph be the number of phished emails that are classified as ham and P be the number of phished emails then false negative rate can be calculated as:

FN=Ph/P

True Positive Rate: The percentage of phished emails in the dataset that are correctly classified as phished. Let P be the number of phished emails and Np be the number of correctly classified phished emails then true positive rate can be calculated as:

Tp=Np/n

True Negative Rate: The percentage of ham emails in the dataset correctly classified as ham. Let H be the number of ham emails and Nh be the number of correctly classified ham emails then true negative rate can be calculated as:

TN=Nh/H

Table 1: Comparison of Precision, Recall, F- measure (weighted average)

Classifier	Precision	Recall	F-measure
SVM	0.999	0.999	0.999
Random Forest	0.999	0.999	0.999
Logistic	0.999	0.999	0.999
NaiveBayes	0.998	0.998	0.998
VotedPerceptron	0.956	0.956	0.956

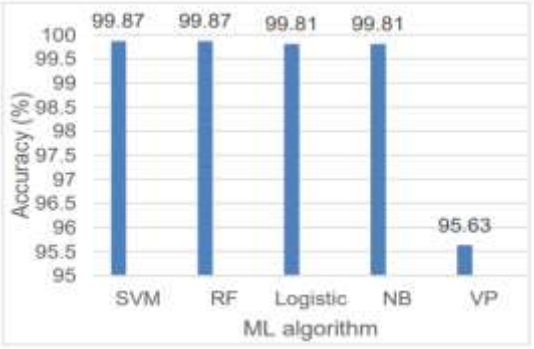


Figure 2: Classification Accuracy of 5 ML Classifiers

Table 2: Comparison of Accuracy

Classifier	Accuracy (%)
SVM	99.87
Random Forest	99.87
Logistic	99.81
NaiveBayes	99.81
VotedPerceptron	95.63

Table 3: Comparison of true positive and true negative (weighted average)

Classifier	TP	FP
SVM	0.999	0.002
Random Forest	0.999	0.002
Logistic	0.998	0.002
NaiveBayes	0.998	0.002
VotedPerceptron	0.956	0.083

These findings clearly show that SVM and Random Forest perform better than the others in terms of classification accuracy. The precision, recall, and f-measure of the employed classifiers are displayed in Table 1. Random Forest, SVM, and 99.99% accuracy, recall, and f-measure rates are provided by logistic classifiers. The real positive and true negative rates are contrasted in Table 3. It demonstrates that the highest true positive rates are produced by SVM and Random Forest. Therefore, it is evident that SVM and Random Forest perform better overall than other classifiers in terms of accuracy, recall, and precision The results show our model produces high accuracy in detecting phished emails. By using the most relevant features, the number of features has been reduced as compared to other works but at the same time, accuracy is improved

RESULTS

Preprocessing and the Dataset

Ten thousand emails made up the dataset, five thousand of which were classified as phishing and five thousand as valid. Tokenization, stop word elimination, and feature extraction using TF-IDF and word embeddings were all part of the preprocessing.

Model Performance

Five classifiers were evaluated: Logistic Regression, SVM, Random Forest, LSTM, and BERT. Performance was measured using 5-fold cross-validation.

3. Analysis of Features

The Random Forest classifier's feature importance revealed that the sender domain, urgency-related phrases, and the existence of dubious URLs were the most important indicators.

4. Analysis of Errors

Promotional emails were frequently linked to false positives, but skillfully constructed phishing attempts imitating well-known sites were frequently the cause of false negatives.

5. Synopsis

BERT outperformed all traditional methods, especially in detecting sophisticated phishing emails. However, it requires significantly more computation time.

using a range of metrics, including accuracy, precision, recall, F1-score, area under the receiver operating characteristic curve (AUC), and confusion matrices. These metrics provided a detailed analysis of the model's discriminative capability, capturing its overall accuracy, class- specific performance, and ability to distinguish between real and fake samples. This multi-metric approach offered a robust assessment, highlighting the model's strengths in handling balanced datasets and its potential for real-world deepfake detection applications.

Validation Results

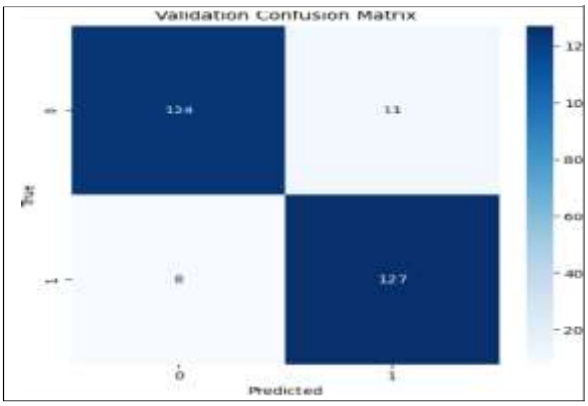
The outcomes of the validation set offer perspectives on the model's effectiveness throughout training and hyperparameter adjustment. Table I displays the classification report for the validation set, outlining precision, recall, and F1-score for the real and fake categories.

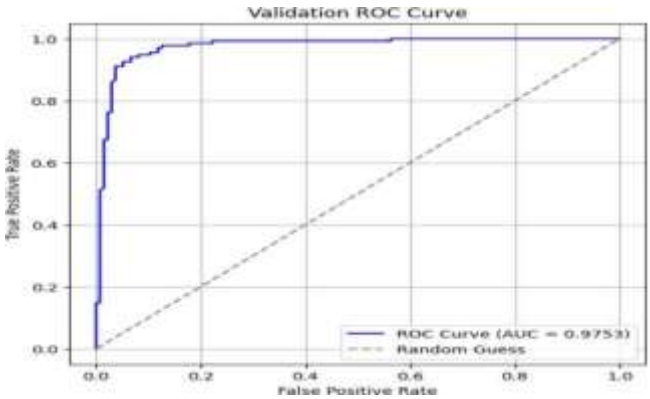
Table II

Validation Classification Report

Class	Precision	Recall	F1-Score	Support
Real	0.94	0.92	0.93	135
Fake	0.92	0.94	0.93	135
Accuracy	-	-	0.93	270
Macro Avg	0.93	0.93	0.93	270
Weighted Avg	0.93	0.93	0.93	270

The validation accuracy obtained is 0. 93, signifying that the model accurately classified 93% of the samples. The F1-scores for both the real and fake categories are balanced at 0. 93, indicating consistent performance across the categories. The AUC for the validation dataset is 0. 9753, showcasing outstanding discriminative ability, as it is near to 1. 0. The validation F1-score stands at 0. 9304, which closely aligns with the individual class F1-scores.The confusion matrix and ROC curve for the validation dataset are shown in Fig. 3 and Fig. 4, respectively.





The confusion matrix (Fig. 3) visualizes the true positives, true negatives, false positives, and false negatives, confirming the balanced performance between real and fake classes. The ROC curve (Fig. 4) illustrates the trade-off between true positive rate and false positive rate, with an AUC of 0.9753 indicating strong class separability

CONCLUSION

This study presents a method for using machine learning algorithms to classify emails as either phished or ham. The dataset underwent preprocessing and was transformed into an appropriate format .It might be used to extract pertinent features to feed into classifiers. Regular expressions and NLTK are used in the Python programming language to extract the features. These are kept in an appropriate file and fed into several classifiers. In order to classify the test set, supervised learning techniques have been employed, which need a training set. The dataset has been divided using the 10-fold cross validation procedure. SVM, Random Forest, Logistic, Naive Bayes, and Voted Perceptron classifiers all receive the model as input. The best accuracy of 99.8% was attained, which was encouraging for the categorization findings .Although this approach has shown positive outcomes, The dataset may not accurately represent real-world situations. By expanding the dataset, the suggested system can be enhanced in further studies. The approach would be more realistic to the real world, where scammers are always refining their methods, by including a range of emails, both phished and ham. We might implement a formal framework that can be used both privately and across organizations to shield consumers from phishing assaults by using real-world examples.

ACKNOWLEDGMENT

The authors would like to sincerely thank the people and institutions who have contributed to this study on phishing detection in email correspondence. The authors are extremely grateful for the opportunity to use public datasets like the Enron Email Dataset and the Nazario Phishing Corpus for early development and experimentation. Additionally, special thanks are given to the maintainers of important open-source libraries, such as TensorFlow, NLTK, and Scikit-learn, whose tools were crucial in the implementation and assessment of several detection models.

acknowledgements will be included upon completion of this phase. Any funding, computational resources, or specific contributions from individuals or institutions

The authors express their gratitude to industry experts and academic colleagues for their insightful comments and suggestions, which greatly impacted the improvement of the feature engineering and assessment process. Additional datasets, such as user-reported phishing emails from enterprise environments, are currently being integrated and analysed. The final version of the work will appropriately acknowledge any funding, institutional assistance, or noteworthy individual contributions for this prolonged period. Additionally, the authors are grateful for the ongoing involvement of the cybersecurity research community, which has influenced the direction and applicability of this study.

REFERENCES

[1] (2020). Anti-Phishing Working Group. Phishing Activity Trends Report 3rd Quarter 2020. [Online]. Available: https://docs.apwg.org/reports/apwg_trends_report_q3_2020.pdf

[2] (2021). Phishing Activity Trends Report 3rd Quarter 2021. Anti-Phishing Working Group. [Online]. Available: https://docs.apwg.org/reports/apwg_trends_report_q3_2021.pdf

[3] N. A. Khan, S. N. Brohi, and N. Zaman, ''Ten deadly cyber security threats amid COVID-19 pandemic,'' TechRxiv, Tech. Rep., 2020, doi: 10.36227/techrxiv.12278792.v1.

[4] B. Pranggono and A. Arabo, ''COVID-19 pandemic cybersecurity issues,'' Internet Technol. Lett., vol. 4, no. 2, Mar. 2021, doi: 10.1002/itl2.247.

[5] H. Abroshan, J. Devos, G. Poels, and E. Laermans, ''COVID-19 and phishing: Effects of human emotions, behavior, and demographics on the success of phishing attempts during the pandemic,'' IEEE Access, vol. 9, pp. 121916–121929, 2021, doi: 10.1109/ACCESS.2021.3109091.

[6] D. Irani, S. Webb, J. Giffin, and C. Pu, ''Evolutionary study of phishing,'' in Proc. eCrime Res. Summit, Oct. 2008, pp. 1–10, doi: 10.1109/ECRIME.2008.4696967.

[7] B. Parno, C. Kuo, and A. Perrig, ''Phoolproof phishing prevention,'' in Proc. Int. Conf. Financial Cryptography Data Secur., 2006, pp. 1–19, doi: 10.1007/11889663_1.

[8] R. Verma, N. Shashidhar, and N. Hossain, ''Detecting phishing emails the natural language way,'' in Proc. Eur. Symp. Res. Comput. Secur., vol. 7459, 2012, pp. 824–841, doi: 10.1007/978-3-642-33167-1_47.

[9] Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha, and M. Guizani, ''Systematization of knowledge (SoK): A systematic review of software-based web phishing detection,'' IEEE Commun. Surveys Tuts., vol. 19, no. 4, pp. 2797–2819, 2017, doi: 10.1109/COMST.2017.2752087.

[10] A. Das, S. Baki, A. El Aassal, R. Verma, and A. Dunbar, ''SoK: A comprehensive reexamination of phishing research from the security perspective,'' IEEE Commun. Surveys Tuts., vol. 22, no. 1, pp. 671–708, Dec. 2019, doi: 10.1109/COMST.2019.2957750.

[11] T. Sharma. (2021). Evolving Phishing Email Prevention Techniques: A Survey to Pin Down Effective Phishing Study Design Concepts. [Online]. Available: http://hdl.handle.net/2142/109179

[12] A. Mukherjee, N. Agarwal, and S. Gupta, ''A survey on automatic phishing email detection using natural language processing techniques,'' Int. Res. J. Eng. Technol., vol. 6, no. 11, pp. 1881–1886, 2019.

[13] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, ''A survey of phishing email filtering techniques,'' IEEE Commun. Surveys Tuts., vol. 15, no. 4, pp. 2070–2090, 4th Quart., 2013, doi: 10.1109/SURV.2013.030713.00020.

[14] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, ''Phishing email detection using natural language processing techniques: A literature survey,'' Proc. Comput. Sci., vol. 189, pp. 19–28, Jan. 2021, doi: 10.1016/j.procs.2021.05.077.

[15] A. Vadariya and N. K. Jadav, ‘‘A survey on phishing URL detection using artificial intelligence,’’ in Proc. Int. Conf. Recent Trends Mach. Learn., IoT, Smart Cities Appl., 2021, pp. 9–20, doi: 10.1007/978-981-15-7234-0_2.
[16] D. Sahoo, C. Liu, and S. C. H. Hoi, ‘‘Malicious URL detection using machine learning: A survey,’’ 2017, arXiv:1701.07179.
[17] E. S. Aung, C. T. Zan, and H. Yamana, ‘‘A Survey of URL-based phishing detection,’’ in Proc. DEIM Forum, 2019, pp. 2–3.
[18] C. M. R. D. Silva, E. L. Feitosa, and V. C. Garcia, ‘‘Heuristicbased strategy for phishing prediction: A survey of URL-based approach,’’ Comput. Secur., vol. 88, Jan. 2020, Art. no. 101613, doi: 10.1016/j.cose.2019.101613.
[19] V. V. Satane and A. Dasgupta, ‘‘Survey paper on phishing detection: Identification of malicious URL using Bayesian classification on social network sites,’’ Int. J. Sci. Res., vol. 4, no. 4, pp. 1998–2001, 2013.
[20] A. Aleroud and L. Zhou, ‘‘Phishing environments, techniques, and countermeasures: A survey,’’ Comput. Secur., vol. 68, pp. 160–196, Jul. 2017. 10.1016/j.cose.2017.04.006.
[21] R. M. Mohammad, F. Thabtah, and L. McCluskey, ‘‘Tutorial and critical analysis of phishing websites methods,’’ Comput. Sci. Rev., vol. 17, pp. 1–24, Aug. 2015, doi: 10.1016/j.cosrev.2015.04.001.
[22] G. Varshney, M. Misra, and P. K. Atrey, ‘‘A survey and classification of web phishing detection schemes,’’ Secur. Commun. Netw., vol. 9, no. 18, pp. 6266–6284, Dec. 2016, doi: 10.1002/sec.1674.
[23] L. Tang and Q. H. Mahmoud, ‘‘A survey of machine learning-based solutions for phishing website detection,’’ Mach.