Grenze International Journal of Engineering and Technology, January Issue

Phishing Website Detection using Machine Learning

Khushi Bhatnagar[1], Prakash Jha[2], Lucky Panchal[3] and Shivani Garg[4]
[1-4]ABES Engineering College, Ghaziabad
Email: Khushi06bhatnagar@gmail.com, jhaprakash824@gmail.com, panchallucky41@gmail.com, shivani.garg@abes.ac.in

***Abstract*—In today's digital world, phishing is a major danger that may result in significant financial losses and data breaches. By developing efficient methods to recognize phishing websites using cutting-edge machine learning, the project "Phishing Website Detection Using Machine Learning Techniques" seeks to address this pressing problem. Techniques. The project aims to give consumers a reliable tool for identifying fraudulent websites by looking at many aspects of URLs and web pages.**

This project properly classifies websites as either authentic or phishing by using a variety of machine learning methods, such as Random Forests, Decision Trees, and Logistic Regression. These techniques were chosen because they are good at deciphering intricate data patterns and provide accurate classifications based on characteristics taken from URLs. Additionally, to improve detection accuracy, feature extraction techniques are used to identify important signs of phishing activity.

Creating an extensive collection of URLs from phishing and trustworthy sites is part of the "cat-phish" project technique. Important characteristics including domain age, URL length, and the existence of questionable keywords are retrieved. Several models for machine learning are created and assessed. following data preparation. finding the best algorithm by utilizing performance criteria like as recall, accuracy, precision, and F1 score. To assist customers, learn about potential risks, a user-friendly interface is also designed to offer real-time analysis.

***Index Terms*— Phishing, Machine Learning, web, random forest.**

I. INTRODUCTION

The practice of phishing represents an illicit tactic that combines technical exploits with social engineering to unlawfully acquire sensitive financial and personal information from individuals. Social engineering schemes frequently involve the use of deceptive emails, crafted to appear as if they originate from reputable companies or organizations. These emails aim to entice recipients into visiting fraudulent websites that then solicit private information, such as usernames and passwords. In some instances, more sophisticated technical manoeuvres are employed to install malicious software on computers, directly facilitating the theft of account information, often by surreptitiously obtaining internet account credentials.

For instance, consider the original login page for Facebook.com. Phishing attacks often involve creating a page that closely mimics this legitimate interface, yet belongs to a malicious website. An unsuspecting user might mistakenly believe that the second site is a legitimate Facebook page and willingly input their personal details. The cybercriminal can then seize this data and exploit it for various illicit activities.

A. The Technique of Phishing

To obtain private information, criminals first create fraudulent replicas of legitimate websites and send out

Grenze ID: 01.GIJET.12.1.670

Figure 1. Fraudulent Facebook

Figure 2. Original login page of Facebook

emails, frequently appearing to be from banking institutions or other businesses that deal with financial data. Reputable companies' slogans and logos will be used to create the email. The kind. Because of the way the hypertext markup language is structured, copying images or even a whole website is simple. One of the reasons for the internet's rapid expansion as a communication tool is its ease of use, but it also allows for the misuse of trade names, trademarks, and other corporate identifiers, which consumers have come to depend on as authentication methods.

Then, in an attempt to trick as many people as possible into falling for the scam, the phisher distributed the "faked" emails. Customers are taken to a fraudulent website that imitates the look of the genuine company when they open these electronic messages or click on a link in the mail.

B. Statistics of Phishing Attacks

Phishing is the identity theft scams that continues to be one of the things that grows rapidly on the internet. It is responsible for both immediate and long-term financial losses among victims. A total of $687 million was lost as a result of the roughly 33,000 phishing attacks that occurred each month throughout the world in 2012. For example, a fraud occurrence occurred in June 2004. The Royal Bank of Canada informed its customers that phoney emails posing as correspondence from the bank were being sent out, requesting that they click on a link to validate their personal identification numbers (PINs) and bank account numbers. The fraudulent email said that if the receiver did not click on the link and enter his client card number and pass code, his account will be inaccessible. These emails were sent out within a week following a computer malfunction that made updating client accounts challenging.

The United States remained the leading country in hosting phishing sites during the third quarter of 2012. This is primarily because a significant portion of the world's websites and domain names are hosted in the United States. Phishers continue to target the financial services sector as their primary focus.

II. RELATED WORK

Numerous scholars have conducted some sort of analysis on the statistics of dubious URLs. Our methodology incorporates key concepts from earlier research. We examine earlier research on phishing site identification using URL properties, which served as inspiration for our current methodology.

- Ma et al.: compared a number of batch-based learning methods for phishing URL recognition and found that integrating host-based and lexical properties yields the best classification accuracy. Additionally, they used full features to compare the performance of batch-based and online algorithms and discovered that online algorithms, particularly confidence-weighted (CW) algorithms, outperform batch-based algorithms.
- Garera et al.: employs logistic regression with a hand-selected set of characteristics to classify phishing URLs. The features include attributes based on Google's PageRank, red flag keywords in the URL, and adherence to Google's web page quality requirements. It is challenging to do a straight comparison without having accessibility to similar functionalities and URLs.

- McGrath and Gupta: Instead of developing a classifier, used an array of datasets to compare phishing and non-phishing URLs. They compared non-phishing URLs from the DMOZ Open Directory Project with phishing URLs from Phish Tank. The researchers look at a number of internet-related factors, such as IP addresses, WHOIS records that provide information about the domain's registration, geographic data, and the URL's linguistic characteristics, such its length, character distribution, and the existence of popular brand names.

III. Problem Overview

Web links, also referred to as URLs, serve as the primary method through which users access and retrieve information on the internet. Our goal is to create categorization algorithms that can identify fraudulent websites by examining the words and characteristics of their URLs. We examine various classification algorithms in the Waikato environment for knowledge analysis using the Weka workbench and MATLAB.

IV. Design Flow

The project involves extracting features from collected URLs, including host-based, page-based, and lexical features, and analysing the results. The first step involves gathering phishing and non-malicious URLs. The feature extractions based on the host, popularity, and lexical features are used to create a comprehensive database of feature values. The database is derived from knowledge through the application of various machine learning techniques. After the classifiers are evaluated, a classifier is chosen and implemented in MATLAB. Diagram 3 shows the process configuration.

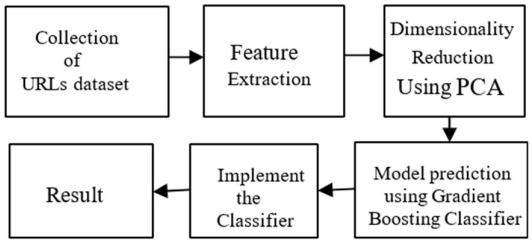

Figure 3. Design flow graph

A. Collection of URLs

We gathered links to safe websites from kaggle.com , Dmoz.org , and our personal web browser history. The phishing URLs were gathered from www.Phishtank.Com . The dataset comprises 17000 phishing URLs and 20000 non-phishing URLs. We obtained the PageRank values of 240 benign websites and 240 phishing websites by individually checking their PageRank at the PageRank checker tool . The results are presented in Table 1. We gathered WHOIS data for 240 benign websites and 240 phishing websites.

B. Feature Extraction

Host-based features provide insights into the geographical location of phishing sites, the individuals or organizations responsible for their management, and the methods used to administer them. We utilize these features because phishing websites may be hosted in less reliable hosting centers, on machines that are not typical web hosts, or through not so trustworthy registrars. The block schematic for the host-based analysis is shown in Figure 4.

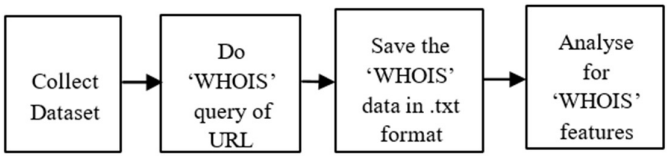

Figure 4. Block diagram for host-based analysis

The following are the properties of the hosts that are identified.

- WHOIS properties: WHOIS properties provide information about the registration date, update and expiry, the name of the registrar and the registrant. If phishing sites are frequently removed, the registration dates

will be more recent compared to legitimate sites. Many phishing websites feature an IP address within their hostname. Having accurate information about hostnames will be beneficial in identifying phishing websites, which can be obtained from the WHOIS properties.

- Geographic properties: Geographic properties provide information about the continent, country, or city that the IP address is associated with.
- Blacklist membership: A significant portion of phishing URLs were included in blacklists. In the context of web browsing, blacklists are pre-made lists or databases that contain the IP addresses, domain names, or URLs of websites that are considered malicious and should be avoided by web users. Conversely, white lists consist of websites that are recognized as secure and trustworthy.
  - DNS-Based Blacklists: Users submit a query containing the IP address or domain name in concern to the blacklist provider's custom DNS server. The answer is an IP address that indicates whether the query was included in the blacklist. Notable DNS blacklist services include Sorbs, Uribl, Surbl, and Spamhaus.
  - Browser Toolbars: A query representing the IP address or domain name in question is submitted by users to the blacklist provider's custom DNS server. The response is an IP address that shows if the query was on the blacklist or not. Notable DNS blacklist services include Sorbs, Uribl, Surbl, and Spamhaus. Browser toolbars that use blacklists to prevent harmful websites include the Web of Trust, Google Toolbar, and McAfee SiteAdvisor.
  - Network Appliances: Dedicated network gear is yet another well-liked choice for blacklist deployment. These gadgets serve as bridges between user computers connected to a company's network and the wider internet. The device intercepts outgoing connections when staff members visit a website and compares the IP addresses or URLs to a list of websites that have already been banned. Websense is an example of a firm that produces network appliances with blacklist-based security capabilities; Cisco purchased IronPort in 2007.
  - Limitations of blacklists: The main benefit of blacklists is that searching for malicious sites is a quick and efficient process: the precompiled lists of malicious sites are readily available, so the only computational cost of using blacklists is the time it takes to find the specific site. Nevertheless, the requirement to create these lists in advance leads to their drawback that blacklists become outdated. Network administrators prevent access to existing harmful websites, and enforcement actions shut down criminal organizations responsible for those sites. Criminals are always under pressure to create new locations and find new infrastructure for hosting their illegal activities. Consequently, new malicious websites are created, and blacklist providers must update their lists once more. Unfortunately, criminals always stay one step ahead because building a website is a cost-effective endeavour. Additionally, there are free services available for blogs, such as blogger and personal hosting options like google sites , as well as Microsoft live spaces , which offer an affordable alternative for disposable websites.
- Page/Popularity Based Property: Popularity characteristics show how well-liked a webpage is by internet users. The following are some characteristics of popularity:
  - PageRank: It is among the techniques Google use to assess the significance or relevancy of a page. As part of its monthly re-indexing process, Google modifies the maximum number of pages on the internet. The total PageRank of all online sites equal unity because the PageRank generate a probability distribution for each web page. The PageRank of all online pages will add up to unity as they constitute a probability distribution across web pages.
  - Traffic Rank details: Website traffic rankings show how popular a website is. Websites are ranked by Alexa.com according to their internet traffic taking into account data from the past three months. Traffic congestion is approximately 1. The accuracy of ranks exceeding 100,000 is questionable due to the high probability of errors.
- Analysis of lexical features: These details pertain to the URL's properties rather than the information on the website it points to. URLs are text strings that are easily interpreted by people and that different software programs process consistently. Through a sequence of actions, browsers translate each URL into an array of commands that direct the browser to the server that houses the website and pinpoint the precise position of the resource or site on that server. URLs use a certain standard syntax to make machine translation easier: <protocol>://<hostname><path>.
  - An example of URL resolution is shown below: Protocol: https:// Host name: accounts.google.com/ Path:

ServiceLogin?service=mail&passive=true&rm=false&continue=https://mail.google.com/mail/&ss=1&scc=1&ltmpl=default&ltmplcache$=2

- The protocol part of the URL describes which network protocol should be used to access the resource that has been requested. Hypertext transfer protocol, or HTTP (HTTP), HTTP with Transport Layer Security (HTTPS), and File Transfer Protocol (FTP) are the most widely used protocols.
- The domain name is used to identify the internet web server. It is most typically a human-readable domain name, especially from the user's point of view, although occasionally it is a machine-readable Internet Protocol (IP) address.
- The file path on a computer and the URL path both point to the location of a particular resource. Punctuation symbols such as dashes, dots, and slashes identify the route tokens, which show how the site is structured. To avoid discovery, some criminals try to hide route tokens, while others purposefully make tokens that seem like legitimate websites.
- The following is the way we employed in our study to extract the lexical characteristics from the URL list: After being gathered from dmoz.org and alexa.com, the URLs of trustworthy websites are entered into a notepad and stored on the computer. The MATLAB software is executed. An input file will be requested. Open the MATLAB application and load the benign URL list. The feature list is retrieved by the application after handling the list. Included is the decision vector '0'. According to the program's instructions, the list is saved on the computer in both Excel and CSV formats. The phishing URL list goes through the same process. Included is the decision vector '1'. Numerous factors, including host length, path length, number of slashes, and number of path tokens, are part of the feature set. The feature extraction procedure is depicted in the flowchart in Figure 5.

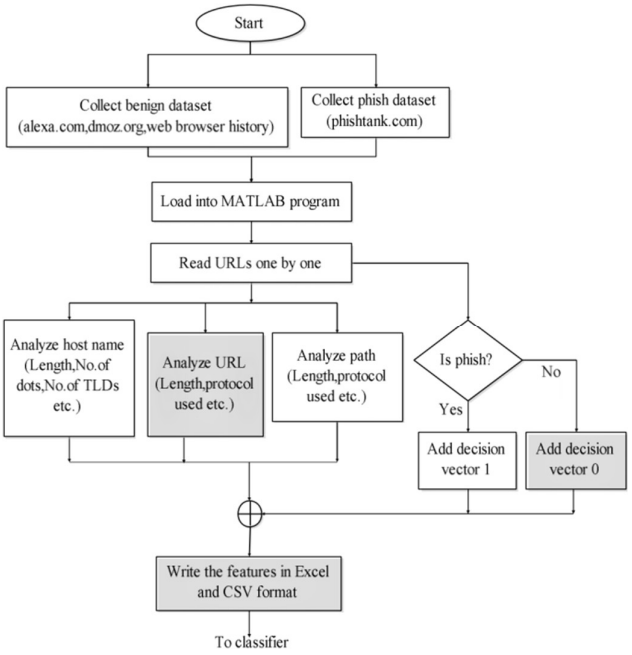

Figure 5. Flow Diagram for feature extraction

V. MACHINE LEARNING ALGORITHMS

In order to create a dependable system for detecting phishing websites, we examined and utilized ten commonly employed supervised machine learning algorithms. The objective was to evaluate their performance and determine the most suitable model for implementation. The entire process was carried out in a Python-based environment, utilizing libraries like scikit-learn, xgboost, and catboost.

The dataset comprised of 11,054 samples, each containing 31 extracted features that represented lexical, domain-specific, and content-based characteristics of URLs. Each link was categorized as either a phishing attempt (1) or a genuine website (0). Some of the key features included indicators like HTTPS, anchor URL, and website traffic, among others. The data was cleaned by: Dealing with incomplete data (if any). Normalizing numerical

variables using standardization methods. Transform categorical variables as necessary. Using the train_test_split() method, the dataset was split into 80% for training and 20% for testing.
The following machine learning classifiers were implemented and evaluated:

- Gradient Boosting Classifier (GBC): A grouping technique that sequentially integrates several poor learners (decision trees) to correct the errors committed by the earlier trees. Out of all the models tested, it achieved the best performance level.
- CatBoost Classifier: A gradient boosting model created by Yandex, specifically designed to handle categorical features. It performed well, closely matching the GBC in terms of recall and precision.
- Multi-layer Perceptron (MLP): An artificial neural network with hidden layers that found intricate patterns and connections in the input data by applying deep learning techniques.
- XGBoost Classifier: A quick and flexible way to boost. renowned for being rapid and effective, especially when working with structured data.
- Random Forest: An ensemble training method that chooses random characteristics for each decision tree and aggregates the predictions of many decision trees that were trained on various obtained data. Although it showed remarkable accuracy, boosting models marginally surpassed it in recall.
- Support Vector Machine (SVM): A method that creates a perfect line to divide data into different groups. The kernel trick was employed to address non-linearly separable data.
- Decision Tree: A model that utilizes tree structures to make decisions, with thresholds set for specific features. It is understandable but susceptible to overfitting.
- K-Nearest Neighbors (K-NN): A non-parametric algorithm that grouped data based on how close they were to their neighboring data points. Its performance was hindered on larger datasets due to the computational overhead involved.
- Logistic Regression: A simple linear classifier that estimated the likelihood of a class label by applying the logistic function.
- Naive Bayes: A classifier employing a probabilistic framework, assuming uninformative relationships between the features. Despite its high precision, the model exhibited poor recall due to its simplifying assumptions.

Accuracy, precision, recall, and F1 score were used to assess each model after it was trained using an 80/20 train-test split. Using max depth 4 and learning_rate=0.7, as shown in Figure 6, the Gradient Boosting Classifier achieved the maximum accuracy of 97.4%, 99.4% recall, and 97.7% F1 score.

73]:

	ML Model	Accuracy	f1_score	Recall	Precision
0	Gradient Boosting Classifier	0.974	0.977	0.994	0.986
1	CatBoost Classifier	0.972	0.975	0.994	0.989
2	Multi-layer Perceptron	0.971	0.974	0.992	0.985
3	XGBoost Classifier	0.969	0.973	0.993	0.984
4	Random Forest	0.967	0.970	0.992	0.991
5	Support Vector Machine	0.964	0.968	0.980	0.965
6	Decision Tree	0.961	0.965	0.991	0.993
7	K-Nearest Neighbors	0.956	0.961	0.991	0.989
8	Logistic Regression	0.934	0.941	0.943	0.927
9	Naive Bayes Classifier	0.605	0.454	0.292	0.997

Figure 6. Accuracy score of different ml algorithms

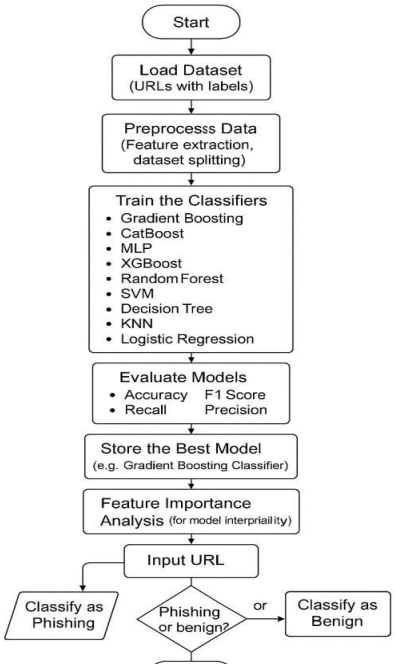

Figure 7. Program flow

The paper concludes that harmless websites have a PageRank between 0 and 9. For the graph, 240 malicious URL sites and 240 benign URL sites were used. The graph suggests that, in comparison to phishing websites, benign websites have a comparatively high PageRank. With relation to newly launched websites, there is one noteworthy exception. The PageRank checker will provide a not available message if we do the PageRank check.

A. Experimental Results and Analysis

The Gradient Boosting model achieved the highest accuracy of 97.4%, with a recall of 99.4% and F1-score of 97.7%.

It outperformed other models such as CatBoost and Random Forest. The following table summarizes the comparative performance of selected classifiers. A confusion matrix and ROC curve are also presented for further insight.

TABLE I

Model	Accuracy (%)	Recall (%)	F1-Score (%)
Gradient Boosting	97.4	99.4	97.7
CatBoost	96.8	98.1	96.4
Random Forest	95.1	96.3	95.5
SVM	93.5	94.2	93.1
Logistic Regression	89.7	87.9	88.8

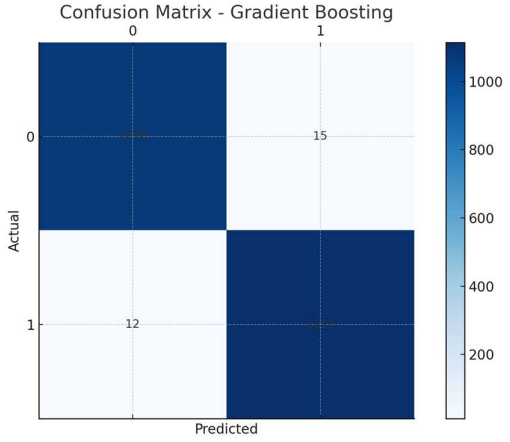

Figure 8. Confusion Matrix for Gradient Boosting Classifier

VI. RESULTS AND CONCLUSION

In the phishing URL detection project, a thorough examination was performed to identify fraudulent phishing URLs from a dataset comprising 11,054 samples, each featuring 31 unique characteristics. These features encompassed a range of URL characteristics, such as length, domain information, and the presence of specific keywords, which are commonly associated with phishing attempts. This categorisation challenge was tackled using a number of supervised machine learning algorithms, including logistic regression. A straightforward and efficient approach for classification and regression problems is K-nearest neighbours (KNN). For classification, regression, and other applications, Random Forest is a powerful machine learning method that yields precise, trustworthy results. Increasing: XGBoost, Multi-layer Perceptron (MLP), CatBoost. The effectiveness of each model was assessed using important metrics like accuracy, precision, recall, and F1 score. These metrics offered a comprehensive assessment of the model's capacity to accurately identify phishing URLs, while also minimizing false positives and negatives. Among the various models tested, the Gradient Boosting Classifier stood out as the top performer, attaining an impressive accuracy of 97.4%. The exceptional performance of gradient boosting can be credited to its ensemble learning strategy, which merges several weak learners to form a strong model. Apart from accuracy, metrics such as F1 score and recall also showed positive outcomes, making the gradient boosting model a suitable choice for detecting phishing emails. To improve the model's accuracy and decrease computational complexity, Principal Component Analysis (PCA) was utilized as a dimensionality

reduction method, which enabled the identification of the most important features and the removal of redundant ones. By implementing this preprocessing step, the model's performance was enhanced, and overfitting was mitigated by emphasizing the most relevant features. When it comes to deploying the gradient boosting model, it was saved in a serialized format using pickle (.Pkl), making it straightforward to incorporate into production systems for real-time phishing detection. This study demonstrates that by utilizing advanced machine learning algorithms, along with feature engineering and dimensionality reduction techniques, the accuracy of phishing URL detection systems can be greatly enhanced. By providing a potent machine learning-based solution to combat phishing assaults, which continue to pose a serious threat to online security, this project contributes to the growing field of cybersecurity.

Harmless websites have a PageRank between 0 and 9. For the graph, we used 240 malicious URL sites and 240 benign URL sites. The graph suggests that, in comparison to phishing websites, benign websites have a comparatively high PageRank. With relation to newly launched websites, there is one noteworthy exception. The PageRank checker will provide a not available message if we do the PageRank check.

A variety of data mining techniques are used to analyse a number of attributes. The results show how well the lexical features may be used to achieve efficiency. By examining the lexical and host-based features of phishing Uniform Resource Locators, we can try to prevent end users from accessing these websites. The fact that thieves are constantly developing new strategies to get around our security systems is one of the biggest challenges in this industry. We need methods that can adapt and upgrade to the latest malware URL examples and properties in order to win this contest. Compared to classroom-based learning mechanisms, internet-based learning algorithms provide more efficient learning strategies. We are excited to investigate many aspects of online learning in the future and collect information to understand the new phishing trends, especially the frequent DNS server changes.

REFERENCE

[1] APWG (2024a). Phishing activity trends report Q1 2024. [online] Available at: https://docs.apwg.org/reports/apwg trends report q1 2024.pdf [Accessed 5 May 2025].

[2] APWG (2024b). Phishing activity trends report Q2 2024. [online] Available at: https://docs.apwg.org/reports/apwg trends report q2 2024.pdf [Accessed 5 May 2025].

[3] Wikipedia (2024). Phishing. [online] Available at: https://en.wikipedia.org/wiki/Phishing [Accessed 5 May 2025].

[4] Shahrivari, V., Darabi, M. M. and Izadi, M. (2020). Machine learning techniques for phishing detection.

[5] Rashid, J., Mahmood, T., Nasir, M. W. and Nazir, T. (2020). Phishing detection using machine learning technique. In: 2020 First International Conference on Smart Systems and Emerging Technologies (SMARTTECH), Islamabad, Pakistan, 2020, pp. 188-192.

[6] Gandotra, E. and Gupta, D. (2021). An efficient approach for phishing detection using machine learning.

[7] Sahingoz, O. K., Buber, E., Demir, O. and Diri, B. (2019). Phishing detection from URLs using machine learning.

[8] Alazaidah, R., AlShaikh, A., AL-Mousa, M. R., Khafajah, H., Samara, G., Alzyoud, M., Al-Shanableh, N. and Almatarneh, S. (2024). Using machine learning techniques to detect phishing websites.

[9] Kiruthiga, R. and Akila, D. (2019). Detection of Phishing Websites through Machine Learning.

[10] Pooja, L. and Sridhar, M. (2020). Analysis of Phishing Website Detection Using CNN and Bidirectional LSTM. In: 2020 Fourth International Conference on Aerospace Technology, Electronics and Communication (ICECA), Chennai, India, 2020, pp. 162-166.

[11] Taofeek, A. O. (2024). Development of a Novel Approach to Phishing Detection using Machine Learning. Stephen F. Austin State University.

[12] Jain, A. K. and Gupta, B. B. (2017). A machine learning-based technique for client-side phishing website detection. Springer Nature.