IRJEMS International Research Journal of Economics and Management Studies
Published by Eternal Scientific Publications
ISSN: 2583 – 5238 / Volume 2 Issue 1 March 2023 / Pg. No: 345-355
Paper Id: IRJEMS- V2I1P144, Doi: 10.56472/25835238/IRJEMS-V2I1P144

Original Article

Real-Time Identification of Phishing Websites Using Advanced Machine Learning Methods

Vetrivelan Tamilmani[1], Venkata Deepak Namburi[2], Aniruddha Arjun Singh Singh[3], Vaibhav Maniar[4], Rami Reddy Kothamaram[5], Dinesh Rajendran[6]

[1]Principal Service Architect, SAP America.
[2]University of Central Missouri, Department of Computer Science.
[3]ADP, Sr. Implementation Project Manager.
[4]Oklahoma City University, MBA / Product Management.
[5]California University of management and science, MS in Computer Information systems.
[6]Coimbatore Institute of Technology, MSC. Software Engineering.

Received Date: 26 March 2023 Revised Date: 05 April 2023 Accepted Date: 18 April 2023

Abstract: *Phishing is a method of social engineering that exploits the trust that users have in online services to obtain sensitive information, including financial data and login credentials. Fake emails that impersonate legitimate businesses and agencies are used to redirect users to fraudulent websites, where they are required to submit confidential financial information, such as login credentials, for social media systems to function successfully. The method presented in this research is effective for identifying fraudulent websites by utilizing two ensemble learning models, XGBoost and AdaBoost. More than 11,000 websites, along with 30 parameters and class labels, are utilised in the Kaggle Phishing Website Detector dataset. To ensure high-quality input for model training, thorough preparation is performed, including data cleaning, data normalization, label encoding, and feature extraction. The two models are compared using the following metrics: F1-score (97.77%), AUC-ROC (97.21%), recall (98.58%), accuracy (98.17%), and precision (97.21%). XGBoost outperforms AdaBoost in all four categories. AdaBoost achieves only 95.73%. Further test the robustness of the suggested models by ROC and confusion matrix analysis. A study comparing ensemble approaches to standard classifiers, such as Naïve Bayes, SVM, and Neural Networks, shows that ensemble methods are more effective. According to the results, XGBoost and AdaBoost are the most effective options for detecting phishing websites in the real world, as they are accurate, scalable, and dependable.*

Keywords: *Phishing Website Detection, Ensemble Learning, Machine Learning, Cybersecurity, Website security, Real-time detection.*

I. INTRODUCTION

The purpose of the online fraud known as "phishing" is to persuade victims to disclose their personal information by creating the illusion that they are communicating with a legitimate company or organization [1]. To get sensitive user information, fraudsters use this gadget [2] [3]. The criminals construct a phony website mimicking the appearance of legitimate ones. Passwords [4], bank information, and account credentials were among the sensitive details that users fell prey to when they divulged them to fraudulent websites. Phishing websites impersonate legitimate websites to deceive users into disclosing sensitive information, including passwords, account details, and credit card numbers, through social engineering [5]. The phishing webpage is identical to the official Facebook page; however, it retains the victim's username and password and transmits them to the perpetrators. The issue of fraudulent websites is becoming more severe [6][7].

Phishing website detection using intelligent algorithms based on feature analysis has been the subject of numerous studies. Approaches to identify online spoofing with two-factor authentication [8]. According to the authors, recurrent neural networks outperform previous methods in the classification of fraud attempts based on URLs [9][10]. There has been a steady rise in the use of AI algorithms for URL threat detection and prevention, and their importance has only grown in recent years. Although the concept of AI was first introduced in the 1950s, it has experienced substantial growth in recent years and is now affecting all facets of communities and professions [11] [12]. The use of ML to identify the evolving nature of the issue has shown considerable potential in recent years. Newer technologies are presently being employed in conjunction with machine learning methodologies [13]. Machine learning-based filters can change to fit the ever-changing style and habits of spammers. This is a critical advantage, as spammers consistently introduce subtle changes and devise new methods to disseminate spam URLs as broadly as possible [14][15]. Deep learning is a more sophisticated and modern approach to learning than the traditional method, even though it is a subset of machine learning. Consequently, the DL approaches are thoroughly described and cited, with a focus on the most important ones [16].

A) Motivation and Contribution

Phishing websites use deceptive tactics to get users to reveal sensitive information, such as login credentials, passwords, bank details, and credit card numbers. Concerns about this have grown in the cybersecurity sector. Detection using conventional rule-based or signature-based methods is becoming more challenging because these fraudulent websites are sometimes made to appear identical to verified ones. The pressing need for more intelligent, adaptive, and scalable detection methods is emphasized by the large-scale increase in reported attacks and the ongoing evolution of phishing techniques. This encourages the implementation of sophisticated machine learning and ensemble learning methodologies, which are capable of analyzing intricate patterns in website features and adapting to novel fraud strategies. This research offers several key contributions as listed below:

- Research on phishing detection relied heavily on the Kaggle Phishing Website Detector dataset.
- Applied essential steps such as data cleaning, standard scaling, normalization, label encoding, and feature extraction to ensure high-quality input for modelling.
- Implemented and compared advanced ensemble learning algorithms, namely XGBoost and AdaBoost, to handle complex feature interactions effectively.
- Used a variety of metrics to objectively assess the models' performance, including recall, accuracy, precision, F1-score, and AUC-ROC.

B) Novelty and Justification of the Study

The proposed study is innovative because it applies the strong ensemble learning algorithms, XGBoost and AdaBoost, to a significant issue of phishing webpage detection. In this component, conventional classifiers are frequently ineffective in terms of accuracy and scalability. The advantage of this method is that the models can represent complex, non-linear trends that often appear in phishing data and are therefore more resistant to changing attack patterns. The justification stems from the increasing frequency and sophistication of phishing threats, which necessitate detection systems that surpass basic rule-based methods. By focusing on robust and intelligent learning techniques, the study addresses a pressing cybersecurity challenge and contributes toward building more reliable, real-world solutions for safeguarding users against online fraud.

C) Organization of the Paper

This paper follows the following format: Studying relevant literature on phishing website identification is covered in Section II. In Section III, detail the dataset, the procedures for pre-processing, and the model's execution. Section IV showcases the outcomes of the experiments and the analysis of comparisons. Final thoughts and recommendations for further research are presented in Section V.

II. LITERATURE REVIEW

A comprehensive evaluation and critical critique of previous research on phishing website identification laid the groundwork for this study's objective and guided its overall development. Kumar et al. (2020) One possible solution to identify bogus websites is to use blacklists. PhisTank is one of the numerous prominent websites that maintains a list of blacklisted websites. There are two deficiencies in the blacklisting technique: blacklists may not be comprehensive and are incapable of identifying newly generated fraudulent websites. Malicious intent on the web has been previously detected and categorised using ML algorithms. This research aims to compare and analyse several ML algorithms that can be used to detect fraud. Categorisation of web addresses. Accuracy, recall, and F1-score were all top performers for the Naïve Bayes Classifier, which reached 98% [17].

Su (2020) A new method for detecting fraudulent websites using LSTM, RNN was created and presented in this research. One benefit of LSTM is that it can record data timeliness and long-term dependencies. LSTM has a strong learning capacity, can automatically gain data characterization without complex feature extraction by hand, and may potentially thrive with enormous, complex, and high-dimensional data. With an estimated accuracy of 99.1%, the testing findings show that this model outperforms existing neural network algorithms [18]. Zaman et al. (2019) analyze the Filter method's procedures in a new manual feature selection methodology and offer different approaches. The dataset utilised for this study is sourced from the UCI ML repository and comprises 2670 instances and structure attributes of 30 websites. Based on the empirical results, the feature group that uses the address bar to identify phishing websites is the most effective. Not only that, but two state-of-the-art algorithms, J48 and HNB, were developed to enable an integrated multi-classified algorithm. According to the data, a 96.25% accuracy rate in identifying fraudulent websites for all applications is achieved by combining approaches [19].

Yang, Zhao and Zeng (2019) suggested a DL-enabled fast detection method for multi-dimensional feature phishing detection. The detection time for establishing the threshold can be decreased by employing this methodology. The accuracy rate was 98.99% and the false positive rate was 0.59% when tested on a dataset that included millions of legitimate and malicious URLs. Through a rational shift of the threshold, the results of the experiment prove that the detection efficiency is optimizable [20]. Shyni, Sundar and Ebby (2018) provided as a method to identify legitimate websites from fraudulent ones; this method is called parse tree validation. This innovative method for identifying phishing websites is based on constructing a parse tree from the hyperlinks obtained from a given page using the Google API. Using 1000 phishing pages and 1000 authentic pages, this strategy has been applied and tested. A 7.3% FN rate and a 5.2% FP rate were the results [21].

Subasi et al. (2017) applied various data mining methods to make decisions about the types of sites: phishing or legitimate. A powerful intelligent system to identify phishing websites was developed by applying several classifiers. Metrics like classification accuracy, F-measure, and ROC curve area define evaluations of data mining methods. Thanks to its top accuracy rate of 97.36%, RF emerged as the clear winner among the ranking methods. RF can detect phishing attempts on a wide variety of websites and has reasonably fast run times [22]. A summarized overview of the latest research on phishing website detection is presented in Table I below, covering the models that are used, the datasets that are used, the main findings of the research, the challenges that have been identified, and the future research directions.

Table 1: Recent Studies on Phishing Website Detection Using Machine Learning

Author & Year	Proposed Work	Dataset	Key Findings	Challenges & Future Work
Kumar et al. 2020	A comparison of blacklists and ML techniques for identifying phishing websites	Used publicly available blacklists like PhishTank	Using Naïve Bayes, we were able to attain a recall of 0.95, an F1-score of 0.97, and a maximum accuracy of 98%.	Blacklists are not exhaustive; cannot detect newly generated phishing websites; need for adaptive detection methods
Su 2020	Phishing detection system using LSTM RNN	Likely publicly collected URLs	LSTM captures long-term dependencies and complex features; achieved accuracy of 99.1%	High computational cost; further optimization for real-time detection and large-scale deployment
Zaman et al. 2019	Manual feature selection approach with comparative study to Filter methods; HNB and J48 integration	29 properties, 2670 instances in the UCI ML Repository	The combination of HNB and J48 yielded an accuracy rate of 96.25%, with address bar-based features providing the best accuracy.	Manual feature selection is time-consuming; need to automate feature engineering; scalability for larger datasets
Yang, Zhao & Zeng 2019	Features with multiple dimensions ML for phishing attack detection	Personalized database including millions of malicious and safe URLs	Two-step approach: character sequence features for fast classification, combined with URL/webpage features; achieved 98.99% accuracy, 0.59% FP rate	Threshold adjustment needed for efficiency; handling extremely large datasets; real-time implementation
Shyni, Sundar & Ebby 2018	Parse tree validation for phishing detection	1000 malicious websites and 1000 genuine ones	There was a 7.3% FN rate and a 5.2% FP rate when a parse tree with hyperlinks was created.	Limited dataset; could improve detection for more diverse websites; integration with other methods
Subasi et al. 2017	Phishing detection using various data mining classifiers	Likely publicly available phishing datasets	RF works well for a variety of websites, has a short execution time, and reached an accuracy of 97.36%.	Model performance on dynamic phishing techniques; adapting to evolving phishing strategies

III. RESEARCH METHODOLOGY

The methodology for Phishing Website Detection begins with the data collection stage on Kaggle. Then it proceeds to the data preprocessing stage, which includes cleaning, standard scaling, normalization, label encoding, and feature extraction to model the data. The modified data is divided into two parts: the training set, which comprises 80% of the total, and the testing set, which comprises 20% of the total. Execute the prediction models using the training data in conjunction with two ML classifiers, XGBoost and AdaBoost. To evaluate these classifiers, we employ practical metrics such as accuracy, precision, recall, F1-score, and AUC-ROC. Finally, find the model that is the most effective at detecting phishing websites by looking at the results. See the overall process flow of the suggested methodology in Figure 1.

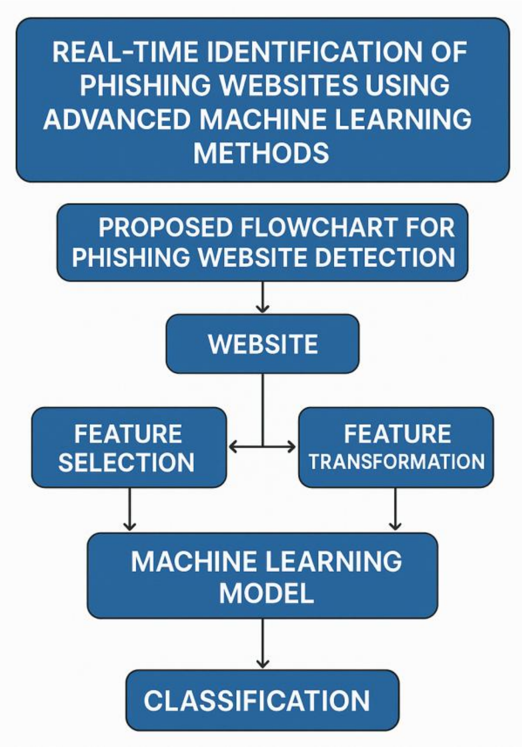
REAL-TIME IDENTIFICATION OF PHISHING WEBSITES USING ADVANCED MACHINE LEARNING METHODS
PROPOSED FLOWCHART FOR PHISHING WEBSITE DETECTION
WEBSITE
FEATURE SELECTION
FEATURE TRANSFORMATION
MACHINE LEARNING MODEL
CLASSIFICATION

Fig. 1 Proposed Flowchart for Phishing Website Detection

The following section details the steps outlined in the proposed data aggregation flowchart for Phishing Website Detection.

A) Data Collection and Visualization

The Kaggle Phishing Website Detector Dataset is used in this investigation and can get the data set in two different formats: text and csv. The list contains the addresses of more than 11,000 different websites. There are 30 criteria for each sample's website, plus a class label that indicates if the website is a phishing one or not (1 or -1). Better model interpretation is made possible through the use of data visualizations for analyzing phishing patterns, attribute correlation, and feature distributions. The visualizations are presented below:

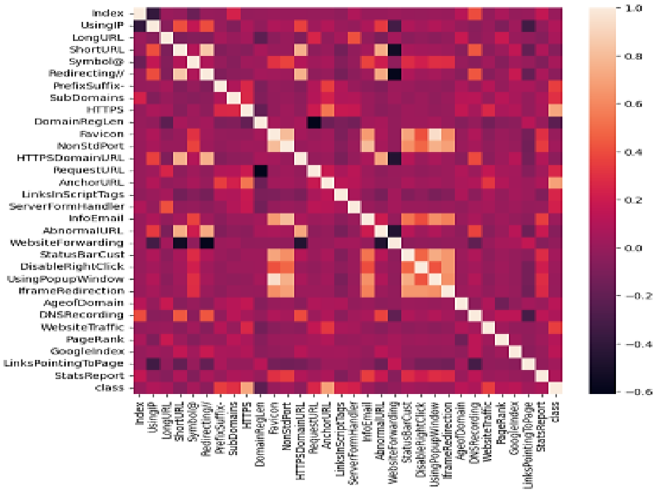

Fig. 2 Correlation Heatmap

Figure 2 shows a correlation matrix, likely generated using a heatmap, displaying pairwise correlations between website-related features such as *HTTPS*, *DomainRegLen*, *Favicon*, and *class*. Variables are listed on both axes, with the diagonal representing self-correlation (always 1). A lighter shade of colour shows a stronger positive correlation, whilst a deeper shade of colour indicates a stronger negative correlation. For instance, *PageRank* and *GoogleIndex* show a strong positive correlation. This visualization helps quickly identify relationships between features.

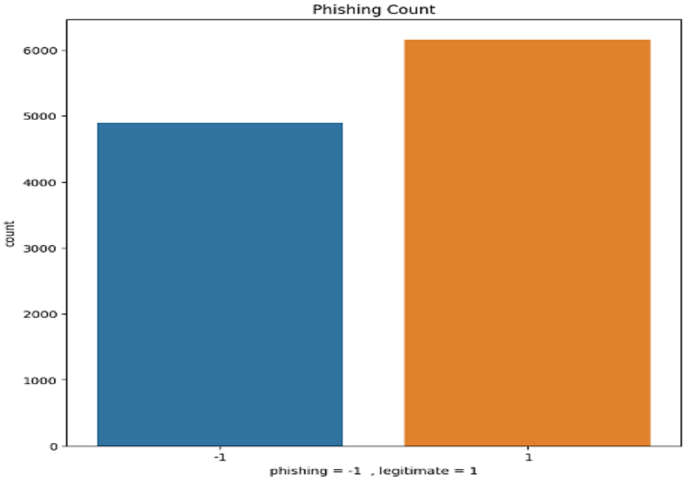

Fig. 3 Phishing Count

Figure 3 shows the distribution of website categories. The x-axis shows numerical labels (*phishing = -1*, *legitimate = 1*), and the y-axis indicates the number of instances. Legitimate websites slightly outnumber phishing sites, with just over 6,000 versus just under 5,000, indicating a relatively balanced dataset.

B) Data Pre-processing

Pre-processing is a critical phase in determining the quality of the ML model, making it a necessary stage before deployment. Several procedures have been carried out in this instance, as detailed later on: standard scaling normalization, label encoding, feature selection, removal of HTML tags, and removal of infrequent words:

- **Removing HTML Tags:** A parser or regular expressions can be used to remove HTML tags from text, which are codes used to format and structure web content. Nevertheless, the "form" tags utilised to construct a false login page and other HTML tags may provide information useful to identifying phishing websites. It is feasible to retain some tags that are helpful while removing others.
- **Remove Infrequent Words:** Eliminating words with a low frequency in the dataset can help the model train more effectively. The result is a smaller vocabulary with improved generalizability in the model.
- **Stem Words:** A smaller vocabulary and more generalizability can be achieved through stemming, which entails reducing words to their base form.
- **Normalization:** Standard Scaling is used to normalize the data and make sure it's consistent. This helps the model function better. Equation (1) was used to standardise each feature value and obtain this:

$$X_{std} = \frac{x - \mu}{\sigma}$$

The standardised value X_{std}, The original value X, the mean of the feature values μ, and the standard deviation of those values σ.

- **Label Encoding:** The dataset's category labels—"phishing" and "legitimate"—are transformed into numerical representations using Label Encoding.
- **Feature Extraction:** This module collects a variety of URL information, including domain name age, number of subdomains, presence of special characters, and URL length. Using the features, the URLs are then categorised as either authentic or phishing.

C) Data Splitting

Tests and training Building a reliable phishing detection model requires 80% of the data and 20% of the time. The training dataset (80%) helps the model learn the trends in phishing and genuine URLs, enabling it to identify significant features that distinguish between them.

D) Proposed Models

The following section of the paper proposes the framework of the efficient and effective use of XGBoost and AdaBoost classifiers to identify phishing websites. These models are explained in the following details:

a. XGBoost Classifier

Many data challenge competitions, including those on Kaggle, have been won with the XGBoost algorithm. Improved computing speed and performance are the goals of the gradient boosting framework it offers [23]. The boosting technique enables it to manage missing values and conduct regularization more effectively [24]. The ability to minimize loss is a major reason why XGBoost is utilised so often. The yield can be predicted using S additive functions in a tree ensemble model, as illustrated in Equation (2), for a dataset with n models and k features $F = (a_i, b_i)(|F| = n, a_i \in Z k, b_i \in Z)$:

$$b_i = \emptyset(a_i) = \sum_{s=1}^{S} m_s(a_i), m_s \in M$$

M is equal to $\{m(a) = \omega_l(a)\}$. The space of the regression trees is denoted as $(l: Z^k \rightarrow N, \omega \in Z^N)$.

b. AdaBoost Classifier

Similar to RF, Ada-Boost joins together weak classification models to create a powerful classifier; this is one way in which the two algorithms are similar. A single model might not classify things very well. On the other hand, total classification performance can improve when many classifiers are combined, with each iteration picking a different set of samples and the final vote given sufficient weight. Weak learners sequentially build trees, giving more weight to previously erroneously predicted samples after each prediction round in an effort to fix them [25]. As a result of its mistakes, the model is getting better. The outcome that is ultimately predicted is determined by the weighted majority vote, or weighted median if there are regression issues. So, here is the last AdaBoost Equation. (3):

$$Z(a) = sign = \left(\sum_{p=1}^{P} \Omega_p z_p(a)\right)$$

The $p - th$Fragile classifier is denoted by z_p, And the comparative weight is called Ω_p. Weak classifiers PEp the weighted mix.

E) Evaluation Metrics

The effectiveness of the model in detecting spoofing is one of the metrics used by the framework to assess its performance. A few examples of metrics include Accuracy, F1 Score, Precision, Recall, and ROC-AUC. Part, one stands for true positives, part two for true negatives, part three for false positives, and part four for false negatives. Some of the important metrics obtained from this are recall, accuracy, and the F1-score, which will be addressed later.

- **True Positive (TP):** the classifier's effective identification of a URL as phishing, as a sum of all instances.
- **True Negative (TN):** the total number of cases in which the classifier consistently ascertains that a specific URL does not contain malicious code.
- **False Positive (FP):** the sum of all instances that the classifier incorrectly labels a URL as a phishing URL.
- **False Negative (FN):** included all cases when the classifier made a FP or negative determination regarding the phishing status of a URL.

a. Accuracy

As a percentage, accuracy measures how many forecasts were right out of all the ones made. As can be shown in Equation (4):

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

b. Precision

The accuracy of the model's positive predictions is referred to as precision. The number of positive instances that were actually predicted but turned out to be negative is what precision tells us. Using the provided Equation. (5), have determined the model's accuracy:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

c. Recall

A possible approach to determine it is to divide the overall count of true positives by their proportion. With a high price tag for FN, this becomes critically important. Equation. (6) shows that:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

d. F1-Score

A model's accuracy on a dataset can be measured by its F1 score. Its primary use is to assess the efficacy of categorization algorithms that use "positive" or "negative" examples. As can be seen in Equation (7):

$$\mathrm{F1-Score} = 2 \times \frac{\mathrm{Precision} \times Recall}{\mathrm{Precision} + \mathrm{Recall}}$$

e. AUC-ROC

Visualizing the diagnostic capabilities of binary classifiers—which strive to create a balance between the trade-offs of TP and FP rates—ROC curves are an essential part of phishing detection. The AUC is a measure of the model's ability to distinguish between positive and negative categories, as shown in Equation (8):

$$AUC = \int_0^1 TPR(t)dt$$

IV. RESULTS AND DISCUSSION

A Core (TM) i7-1065G7 CPU-based processor running at 1.30GHz and 1.50 GHz is utilised for the execution of the tests in this paper. Python 3.7.1 is also utilised because of the extensive collection of classification models and packages it provides. Table II displays the results of the XGBoost and AdaBoost classifiers for phishing website identification in the suggested models. For detecting phishing websites, the XGBoost model performs admirably, with an F1-score of 97.77%, accuracy of 98.17%, precision of 97.21%, recall of 98.58%, and a harmony between the two metrics. XGBoost's accuracy, precision, recall, and F1-score were all lower than AdaBoost's, at 95.73% and 95.26%, respectively. All in all, the findings indicate that both of the models are quite effective in detecting phishing websites.

Table 2: Performance Results of the Proposed Models for Phishing Website Detection

Performance matrix	XGBoost	AdaBoost
Accuracy	98.17	95.73
Precision	97.21	95.26
Recall	98.58	94.72
F1-Score	97.77	94.88

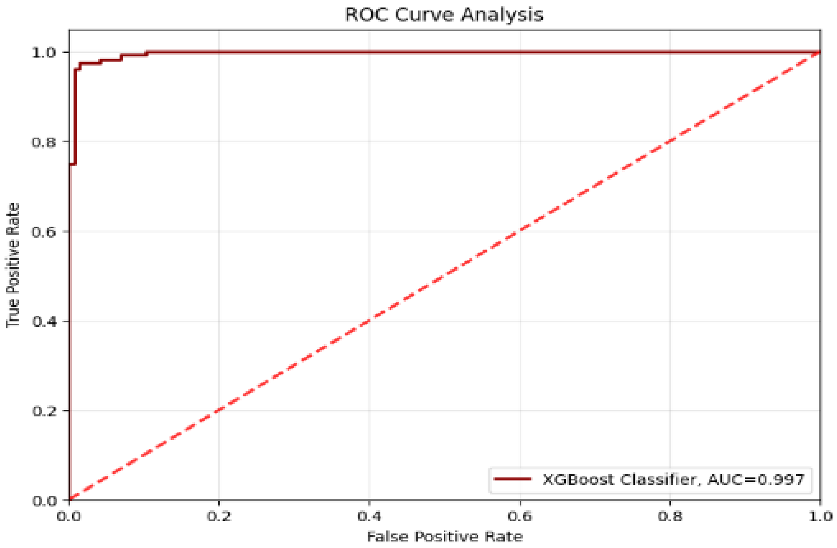

Fig. 4 ROC Analysis of the XGBoost Classifier

Figure 4 shows the XGBoost classifier ROC curve by overlaying a TPR versus FPR line. A solid maroon line shows the classifier's performance, sharply rising and hugging the top-left corner, while a dashed red line represents a random classifier. With an AUC of **0.997**, the model demonstrates exceptional discriminatory power and high effectiveness.

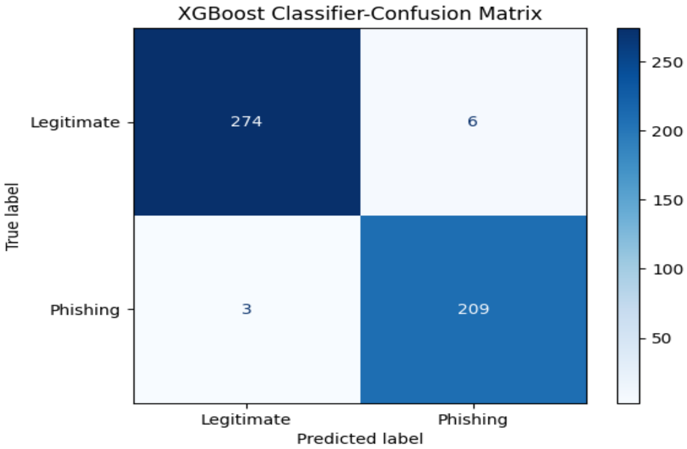

Fig. 5 Confusion Matrix of XGBoost Classifier

Figure 5 confusion matrix for an XGBoost Classifier model, which is used to evaluate its performance in classifying legitimate and phishing attempts. The matrix shows that the model correctly identified 274 legitimate cases (TN) and 209 phishing cases (TP). However, it also made some errors: it incorrectly classified 6 legitimate cases as phishing (FP) and, more critically, missed 3 phishing cases by classifying them as legitimate (FN).

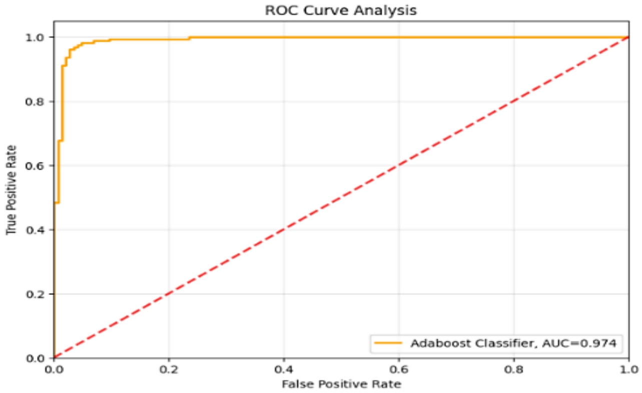

Fig. 6 ROC Analysis of the AdaBoost Classifier

The Adaboost classifier's ROC curve is shown in Figure 6. The dashed red line represents a random classifier, while a solid orange line indicates high TPR at low FPR. The model's high AUC of 0.974 demonstrates its ability to effectively differentiate between classes.

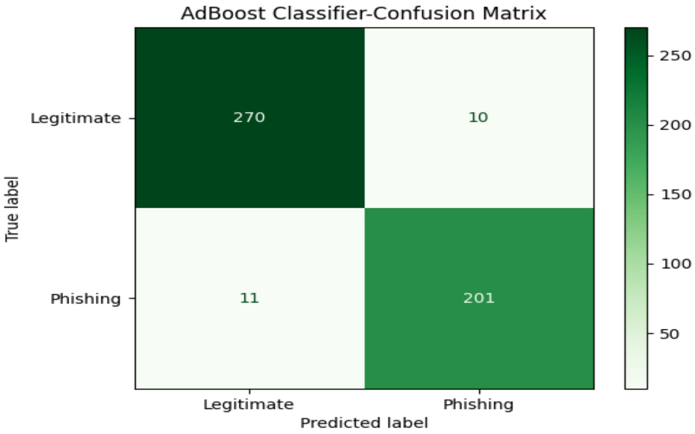

Fig. 7 Confusion Matrix of AdaBoost Classifier

Figure 7 shows a confusion matrix for an Adaboost Classifier, which is used to evaluate its performance in classifying legitimate and phishing attempts. The matrix shows that the model correctly identified 270 legitimate cases (TN) and 201 phishing cases (TP). However, it also made some errors: it incorrectly classified 10 legitimate cases as phishing (FP) and, more critically, missed 11 phishing cases by classifying them as legitimate (FN).

A) Comparative Analysis

The accuracy-based performance comparison of different phishing site detection methods is presented in Table III. An NN model achieved 85.61% accuracy, whereas more conventional ML models like NB and SVM reached 67.04% and 90%, respectively. XGBoost and AdaBoost were the most successful proposed ML models with an accuracy of 98.17 and 95.73, respectively, which proves their better performance in detecting phishing sites and indicates the relevance of ensemble learning methods in the context.

Table 3: Performance Comparison of Different Models for Phishing Website Detection

Models	Accuracy
NN [26]	85.61
NB [27]	67.04
SVM [28]	90
XGBoost	98.17
AdaBoost	95.73

The proposed XGBoost and AdaBoost models offer several benefits for detecting phishing websites. The two models can benefit from ensemble learning methods that enable them to capture complex patterns more effectively and minimize the risk of overfitting relative to single classifiers. XGBoost uses gradient boosting, which means it can correct the errors of the previous models a few times. In contrast, AdaBoost can adaptively concentrate on the misclassified cases, which enhances the overall accuracy and strength. The models can also process high-dimensional data and diverse types of features, making them suitable for real-world phishing detection scenarios.

V. CONCLUSION AND FUTURE STUDY

Phishing is a well-known attack that deceives users into visiting malicious content to extract their information. The URL and webpage interface of the majority of fraudulent webpages are similar to those of genuine webpages. To sum up, the current paper has demonstrated that ensemble learning frameworks, specifically XGBoost and AdaBoost, can be effectively utilized to identify phishing websites. By utilizing the Phishing Website Detector dataset, the proposed framework demonstrated very promising results, with XGBoost achieving a performance accuracy of 98.17, surpassing the AdaBoost performance accuracy of 95.73. The findings, which have been corroborated by ROC and confusion matrices, bring out the strength and dependability of these models in separating legitimate and phishing websites. In addition, the comparative analysis against the traditional classifiers like Naive Bayes, SVM, and Neural Networks verifies the excellence of ensemble methods in processing the complicated patterns and reducing the false classifications. All in all, the results highlight the potential of XGBoost and AdaBoost as scalable, efficient, and practical solutions for real-world phishing detection systems, as they can make a significant contribution to the mechanism of cybersecurity defense. The future of the field will involve the integration of real-time phishing information, hybrid systems, and DL models such as CNNs and RNNs to improve flexibility. Growing datasets of changing phishing trends and countering concept drift or adversarial attacks will enhance resilience and make detection systems more robust and practical in real-world applications.

VI. REFERENCES

[1] M. F. A. Razak, N. B. Anuar, F. Othman, A. Firdaus, F. Afifi, and R. Salleh, “Bio-inspired for Features Optimization and Malware Detection,” *Arab. J. Sci. Eng.*, vol. 43, no. 12, pp. 6963–6979, 2018, doi: 10.1007/s13369-017-2951-y.

[2] M. T. Suleman and S. M. Awan, “Optimization of URL-Based Phishing Websites Detection through Genetic Algorithms,” *Autom. Control Comput. Sci.*, vol. 53, no. 4, pp. 333–341, Jul. 2019, doi: 10.3103/S0146411619040102.

[3] D. D. Rao, “Multimedia-Based Intelligent Content Networking for Future Internet,” in *2009 Third UKSim European Symposium on Computer Modeling and Simulation*, 2009, pp. 55–59. doi: 10.1109/EMS.2009.108.

[4] N. S. Zaini *et al.*, “Phishing detection system using machine learning classifiers,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 17, no. 3, pp. 1165–1171, 2020, doi: 10.11591/ijeecs.v17.i3.pp1165-1171.

[5] A. Thapliyal, P. S. Bhagavathi, T. Arunan, and D. D. Rao, “Realizing Zones Using UPnP,” in *2009 6th IEEE Consumer Communications and Networking Conference*, 2009, pp. 1–5. doi: 10.1109/CCNC.2009.4784867.

[6] B. Wei *et al.*, “A Deep-Learning-Driven Light-Weight Phishing Detection Sensor,” *Sensors*, vol. 19, no. 19, Sep. 2019, doi: 10.3390/s19194258.

[7] A. Kushwaha, P. Pathak, and S. Gupta, “Review of optimize load balancing algorithms in cloud,” *Int. J. Distrib. Cloud Comput.*, vol. 4, no. 2, pp. 1–9, 2016.

[8] A. K. Jain and B. B. Gupta, “Two-level authentication approach to protect from phishing attacks in real time,” *J. Ambient Intell. Humaniz. Comput.*, vol. 9, no. 6, pp. 1783–1796, 2018, doi: 10.1007/s12652-017-0616-z.

[9] J.-L. Chen, Y.-W. Ma, and K.-L. Huang, “Intelligent Visual Similarity-Based Phishing Websites Detection,” *Symmetry (Basel).*, vol. 12, no. 10, p. 1681, Oct. 2020, doi: 10.3390/sym12101681.

[10] S. S. S. Neeli, "Decentralized Databases Leveraging Blockchain Technology," vol. 8, no. 1, pp. 1–8, 2020.

[11] T. C. Truong, Q. B. Diep, and I. Zelinka, "Artificial Intelligence in the Cyber Domain: Offense and Defense," *Symmetry (Basel).*, vol. 12, no. 3, p. 410, Mar. 2020, doi: 10.3390/sym12030410.

[12] S. S. S. Neeli, "Real-Time Data Management with In-Memory Databases : A Performance-Centric Approach," p. 49, 2020.

[13] H. P. Kapadia, "Cross-Platform UI/UX Adaptions Engine for Hybrid Mobile Apps," *Int. J. Nov. Res. Dev.*, vol. 5, no. 9, pp. 30–37, 2020.

[14] A. J. Saleh *et al.*, "An Intelligent Spam Detection Model Based on Artificial Immune System," *Information*, vol. 10, no. 6, p. 209, Jun. 2019, doi: 10.3390/info10060209.

[15] Gopi, "Zero Trust Security Architectures for Large-Scale Cloud Workloads," *Int. J. Res. Anal. Rev.*, vol. 5, no. 2, pp. 960–965, 2018.

[16] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A Survey of Deep Learning Methods for Cyber Security," *Information*, vol. 10, no. 4, 2019, doi: 10.3390/info10040122.

[17] J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran, and B. S. Bindhumadhava, "Phishing website classification and detection using machine learning," in *2020 International Conference on Computer Communication and Informatics, ICCCI 2020*, 2020. doi: 10.1109/ICCCI48352.2020.9104161.

[18] Y. Su, "Research on Website Phishing Detection Based on LSTM RNN," in *Proceedings of 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2020*, 2020. doi: 10.1109/ITNEC48623.2020.9084799.

[19] S. Zaman, S. M. Uddin Deep, Z. Kawsar, M. Ashaduzzaman, and A. I. Pritom, "Phishing Website Detection Using Effective Classifiers and Feature Selection Techniques," in *ICIET 2019 - 2nd International Conference on Innovation in Engineering and Technology*, 2019. doi: 10.1109/ICIET48527.2019.9290554.

[20] P. Yang, G. Zhao, and P. Zeng, "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning," *IEEE Access*, vol. 7, pp. 15196–15209, 2019, doi: 10.1109/ACCESS.2019.2892066.

[21] C. E. Shyni, A. D. Sundar, and G. S. E. Ebby, "Phishing Detection in Websites using Parse Tree Validation," in *2018 Recent Advances on Engineering, Technology and Computational Sciences (RAETCS)*, 2018, pp. 1–4. doi: 10.1109/RAETCS.2018.8443961.

[22] A. Subasi, E. Molah, F. Almkallawi, and T. J. Chaudhery, "Intelligent phishing website detection using random forest classifier," in *2017 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2017*, 2017. doi: 10.1109/ICECTA.2017.8252051.

[23] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[24] T. Manyumwa, P. F. Chapita, H. Wu, and S. Ji, "Towards Fighting Cybercrime : Malicious URL Attack Type Detection using Multiclass Classification," pp. 1813–1822, 2020, doi: 10.1109/BigData50022.2020.9378029.

[25] V. Shahrivari, M. M. Darabi, and M. Izadi, "Phishing Detection Using Machine Learning Techniques," 2020.

[26] A. D. Kulkarni and L. L. Brown, "Phishing Websites Detection using Machine Learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 7, pp. 8–13, 2019.

[27] M. Korkmaz, O. K. Sahingoz, and B. DIri, "Detection of Phishing Websites by Using Machine Learning-Based URL Analysis," *2020 11th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2020*, no. July 2020, 2020, doi: 10.1109/ICCCNT49239.2020.9225561.

[28] J. Mao *et al.*, "Detecting Phishing Websites via Aggregation Analysis of Page Layouts," *Procedia Comput. Sci.*, vol. 129, pp. 224–230, 2018, doi: 10.1016/j.procs.2018.03.053.

[29] Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Vangala, S. R. (2021). Big Text Data Analysis for Sentiment Classification in Product Reviews Using Advanced Large Language Models. *International Journal of AI, BigData, Computational and Management Studies*, *2*(2), 55-65.

[30] Gangineni, V. N., Tyagadurgam, M. S. V., Chalasani, R., Bhumireddy, J. R., & Penmetsa, M. (2021). Strengthening Cybersecurity Governance: The Impact of Firewalls on Risk Management. *International Journal of AI, BigData, Computational and Management Studies*, *2*, 10-63282.

[31] Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., & Gangineni, V. N. (2021). An Advanced Machine Learning Models Design for Fraud Identification in Healthcare Insurance. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(1), 26-34.

[32] Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., & Polam, R. M. (2021). Advanced Machine Learning Models for Detecting and Classifying Financial Fraud in Big Data-Driven. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(3), 39-46.

[33] Tyagadurgam, M. S. V., Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., & Chalasani, R. (2021). Enhancing IoT (Internet of Things) Security Through Intelligent Intrusion Detection Using ML Models. *International Journal of Emerging Research in Engineering and Technology*, *2*(1), 27-36.

[34] Vangala, S. R., Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., & Chundru, S. K. (2021). Smart Healthcare: Machine Learning-Based Classification of Epileptic Seizure Disease Using EEG Signal Analysis. *International Journal of Emerging Research in Engineering and Technology*, *2*(3), 61-70.

[35] Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., Polam, R. M., & Kamarthapu, B. (2021). Big Data and Predictive Analytics for Customer Retention: Exploring the Role of Machine Learning in E-Commerce. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(2), 26-34.

[36] Penmetsa, M., Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., Gangineni, V. N., & Pabbineedi, S. (2021). Next-Generation Cybersecurity: The Role of AI and Quantum Computing in Threat Detection. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(4), 54-61.

[37] Polu, A. R., Vattikonda, N., Gupta, A., Patchipulusu, H., Buddula, D. V. K. R., & Narra, B. (2021). Enhancing Marketing Analytics in Online Retailing through Machine Learning Classification Techniques. *Available at SSRN 5297803*.

[38] Polu, A. R., Buddula, D. V. K. R., Narra, B., Gupta, A., Vattikonda, N., & Patchipulusu, H. (2021). Evolution of AI in Software Development and Cybersecurity: Unifying Automation, Innovation, and Protection in the Digital Age. *Available at SSRN 5266517*.

[39] Polu, A. R., Vattikonda, N., Buddula, D. V. K. R., Narra, B., Patchipulusu, H., & Gupta, A. (2021). Integrating AI-Based Sentiment Analysis With Social Media Data For Enhanced Marketing Insights. *Available at SSRN 5266555*.

[40] Buddula, D. V. K. R., Patchipulusu, H. H. S., Polu, A. R., Vattikonda, N., & Gupta, A. K. (2021). INTEGRATING AI-BASED SENTIMENT ANALYSIS WITH SOCIAL MEDIA DATA FOR ENHANCED MARKETING INSIGHTS. *Journal Homepage: http://www. ijesm. co. in*, *10*(2).

[41] Gupta, A. K., Buddula, D. V. K. R., Patchipulusu, H. H. S., Polu, A. R., Narra, B., & Vattikonda, N. (2021). An Analysis of Crime Prediction and Classification Using Data Mining Techniques.

[42] Rajiv, C., Mukund Sai, V. T., Venkataswamy Naidu, G., Sriram, P., & Mitra, P. (2022). Leveraging Big Datasets for Machine Learning-Based Anomaly Detection in Cybersecurity Network Traffic. *J Contemp Edu Theo Artific Intel: JCETAI/102*.

[43] Sandeep Kumar, C., Srikanth Reddy, V., Ram Mohan, P., Bhavana, K., & Ajay Babu, K. (2022). Efficient Machine Learning Approaches for Intrusion Identification of DDoS Attacks in Cloud Networks. *J Contemp Edu Theo Artific Intel: JCETAI/101*.

[44] Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., Gangineni, V. N., Pabbineedi, S., & Penmetsa, M. (2020). Big Data-Driven Time Series Forecasting for Financial Market Prediction: Deep Learning Models. *Journal of Artificial Intelligence and Big Data*, *2*(1), 153–164.DOI: 10.31586/jaibd.2022.1341

[45] Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., Polam, R. M., Kamarthapu, B., & Kakani, A. B. (2022). Advance of AI-Based Predictive Models for Diagnosis of Alzheimer's Disease (AD) in Healthcare. *Journal of Artificial Intelligence and Big Data*, *2*(1), 141–152.DOI: 10.31586/jaibd.2022.1340

[46] Tyagadurgam, M. S. V., Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., & Chalasani, R. (2022). Designing an Intelligent Cybersecurity Intrusion Identify Framework Using Advanced Machine Learning Models in Cloud Computing. *Universal Library of Engineering Technology*, (Issue).

[47] Vangala, S. R., Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., & Chundru, S. K. (2022). Leveraging Artificial Intelligence Algorithms for Risk Prediction in Life Insurance Service Industry. *Available at SSRN 5459694*.

[48] Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Vangala, S. R. (2021). Data Security in Cloud Computing: Encryption, Zero Trust, and Homomorphic Encryption. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(3), 70-80.

[49] Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., Chalasani, R., & Tyagadurgam, M. S. V. Efficient Framework for Forecasting Auto Insurance Claims Utilizing Machine Learning Based Data-Driven Methodologies. *International Research Journal of Economics and Management Studies IRJEMS*, *1*(2).

[50] Vattikonda, N., Gupta, A. K., Polu, A. R., Narra, B., Buddula, D. V. K. R., & Patchipulusu, H. H. S. (2022). Blockchain Technology in Supply Chain and Logistics: A Comprehensive Review of Applications, Challenges, and Innovations. *International Journal of Emerging Research in Engineering and Technology*, *3*(3), 99-107.

[51] Narra, B., Vattikonda, N., Gupta, A. K., Buddula, D. V. K. R., Patchipulusu, H. H. S., & Polu, A. R. (2022). Revolutionizing Marketing Analytics: A Data-Driven Machine Learning Framework for Churn Prediction. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *3*(2), 112-121.

[52] Polu, A. R., Narra, B., Buddula, D. V. K. R., Patchipulusu, H. H. S., Vattikonda, N., & Gupta, A. K. BLOCKCHAIN TECHNOLOGY AS A TOOL FOR CYBERSECURITY: STRENGTHS, WEAKNESSES, AND POTENTIAL APPLICATIONS.

[53] Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., Gangineni, V. N., Pabbineedi, S., & Penmetsa, M. (2022). Big Data-Driven Time Series Forecasting for Financial Market Prediction: Deep Learning Models. *Journal of Artificial Intelligence and Big Data*, *2*(1), 153–164.DOI: 10.31586/jaibd.2022.1341

[54] Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., Polam, R. M., Kamarthapu, B., & Kakani, A. B. (2022). Advance of AI-Based Predictive Models for Diagnosis of Alzheimer's Disease (AD) in Healthcare. *Journal of Artificial Intelligence and Big Data*, *2*(1), 141–152.DOI: 10.31586/jaibd.2022.134