MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

URL-Driven Machine Learning for Phishing Website Detection

Eswar Patnala,
Assistant Professor,
Department of IoT,
Koneru Lakshamaiah Education Foundation,
peswar@kluniversity.in

Dornala Prathyusha,
Department of IoT,
Koneru Lakshamaiah Education Foundation,
2200100032iot@gmail.com

MD. Sahila Sana,
Department of IoT,
Koneru Lakshamaiah Education Foundation,
2200100013iot@gmail.com

P. Sai Prabhas
Department of IoT,
Koneru Lakshamaiah Education Foundation,
2200100005iot@gmail.com

Abstract**—**Phishing is a type of cybercrime that has been growing more common in recent years because more people are using the Internet and its tools. It is one of the most frequent forms of social engineering, where attackers try to get users to share their private or sensitive information. The machine learning based framework analyses URL features, Domain Names System features, and website content features to distinguish between legitimate and fraudulent websites in near real-time. The various supervised learning models including Random Forest, Logistic Regression, XG Boost perform. Researchers are always working to make these models better and more accurate. The proposed framework allows for a scalable and reliable method for addressing cybersecurity issues while protecting users from online fraud.

Keywords**—** *Phishing Detection, Machine Learning, Cybersecurity, URL Analysis, Feature Extraction, Web Security*

I. INTRODUCTION

The rise of the internet has changed the way we and organizations communicate, do business, and share information, however, as a result, there is an increase of cybercrime, especially phishing attacks. Phishing means creating a fake site that looks like a legitimate site, then getting users to enter sensitive information such as usernames, passwords, and credit card details (the attackers take over users' identities).[1] The financial loss is not only a problem but also the lack of trust that users have and organizations lose their good reputation.

Traditional methods of finding phishing sites such as black lists and rule-based filters are often not successful because phishing sites change constantly [2]. Criminals modify their site structures and change their domains, making it difficult for traditional methods of identifying phishing sites to detect them. Therefore, machine learning (ML) has been proven to be an effective way to detect phishing websites by automatically identifying similarities in sites to differentiate between a phishing site and a legitimate site [5].

This study presents a machine learning system for detecting phishing websites. This system is composed of URL-based features, content-based features, and host-based features that can be used to provide confidence to identify a phishing site. We will train and evaluate a number of classifiers (Logistic Regression, Random Forest, and XG Boost) to identify which model is the best for detecting phishing sites.

II. BACKGROUND

A. *Phishing Detection*

The most prevalent type of cyberattack is phishing and it is done by a hacker using a fake website to deceive users into submitting their confidential information (user name, password or other private data) [27]. Phishing websites are typically sent as emails or as advertisements. When you click on one of these links, you will be directed to a bogus login page that is an exact duplicate of the legitimate website. Once you submit your confidential information, this data is collected and used for identity theft and financial fraud.

Phishing websites are difficult to detect because of the constant changing of URLs, domain names and appearance of phishing sites by hackers who want to remain undetected. Existing technologies to detect phishing websites consist of blacklist, whitelist and rule based filtering systems that only detect previously known phishing websites [6]. However, as the acceptance of machine learning based detection technologies is increasing, it is becoming more common for businesses to employ these technologies to assist in detecting phishing websites. Machine learning (ML) based models are able to automatically analyse the URL, domain and design of a website [8].

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

B. Phishing Detection Approaches

The Machine Learning-Based Approach provides a supervised learning model through which to distinguish between phishing and legitimate domains, based upon a labelled dataset that consists solely of URL-based feature sets [9]. This new method overcomes the limitations associated with the more conventional techniques of blacklists, whitelists and heuristics, whilst allowing for complete independence of manual input from the model during its pattern learning process [10].

The current research study makes use of the UCI Phishing Website Dataset, which contains 31 characteristics for each domain, collected from various sources of information including URL features, domain features and webpage content characteristics [19]. As described above, in order to conduct this current research, it was necessary to carry out an initial step of preprocessing the dataset before then splitting the data into both training and testing datasets.

To realize the full benefits of machine learning technology, three separate supervised machine learning algorithms were deployed, and their accuracy and classification performance were compared. These three algorithms were Logistic Regression, Random Forest and Extreme Gradient Boosting (XG Boost).

Logistic Regression is a linear classifier that predicts the probability of a website being a phishing website using a sigmoid function [3]. In this study, Logistic Regression serves as a straightforward baseline model and is a useful method because the model itself has specific validity measurements, that is, overall accuracy and explained variation in the prediction.

Random Forest is an ensemble-based learning algorithm that employs many decision trees to improve prediction accuracy compared to being overfitting. Extreme Gradient Boosting (XG Boost) is a more sophisticated boosting algorithm optimized for speed and performance compared to the above methods. Next, decision trees are built in a sequential mechanism, where each new tree attempts to compensate for the way in which the prior trees make mistakes [4].

C. Machine Learning Algorithms

Machine learning algorithms have proven to be a valuable means for identifying the patterns of phishing attacks, since instead of relying on predefined rules, they can learn complex patterns from the data supplied to them. In the context of this framework, the designed algorithms utilize feature sets derived from the URL, domain, and web content to help classify websites as either legitimate or phishing. The feature set includes features such as URL length, the presence or absence of special characters, the number of subdomains, whether the URL has an SSL certificate, or redirection. Three supervised learning algorithms, specifically Logistic Regression, Random Forest, and Extreme Gradient Boosting (XG Boost), were used and evaluated in this study [17].

III. RELATED WORK

The most common types of cybercrime are phishing. Phishing takes advantage of the way people think, and does not try to take advantage of the way that software works, unlike other cybercrimes. Since more people are now using the Internet for things like banking, shopping, and communicating digitally, the amount of phishing attack attempts is increasing. Many phishing detection techniques that have existed over the years (e.g., blacklists, whitelists and heuristic techniques) are not very effective at identifying phishing attempts that are new or "zero days old". Therefore, using machine learning (ML) has become an important way to detect phishing websites based on how humans interact with computers by teaching themselves through the automated collection of large quantities of data about both URLs and website features and finding relationships and patterns in this data.

According to Krishna et al. (2021), there are five main types of virus detection mechanisms: list-based, heuristic-based, visual similarity-based, content-based, and machine learning- based techniques [2]. Currently, ML techniques have received the most attention because of the complexity of the information they can detect and their ability to change and develop as people become aware of potential phishing schemes. The authors of this article reviewed and summarized a few different types of ML algorithms that have been utilized in the area of phishing and found that the models based on URL features consistently perform better than most standard ones. Some URL features that were analyzed were length, domain age, SSL state, number of subdomains, amount of redirections attempted to the URL, and the presence of specific characters, such as the "@", and "- " symbols.

Many researchers have analyzed and compared multiple machine learning algorithms for phishing detection. For example, in 2020, Mahajan and Siddavatam analyzed Decision Trees (dt), random forests (rf), and Support Vector Machine classifiers (svm) using more than 36,000 URLs from both Alexa and Phish Tank databases [4]. Their findings indicated that the random forest classifier outperformed all other algorithms with 97.14% accuracy and the lowest false-negative rate when classifying phished URLs. Similarly, Kumar et al. developed an imbalanced dataset of phishing/legitimate URLs and created/used many classifiers (logistic regression, Naïve Bayes, random forest, decision tree, K-nearest neighbor). Their analysis indicated that the best performer was Naïve Bayes with 98% accuracy; therefore, for accurate phishing detection, it is important to use both balanced data and feature selection for building machine-learning algorithms.

In addition, an analysis published in 2024 by R. Alazaidah et al. [14] compared 24 classifiers from six major learning strategies on datasets from the UCI repository. The results showed that the Filtered Classifier, J48 (c4.5 decision tree), and Random Forest had the best overall performance with an accuracy of > 90 %. Additionally, the authors identified the Info Gain Attribute Eval feature selection method to be the most effective feature selection strategy; using this method can significantly improve the efficiency and accuracy of phishing detection by selecting only the most relevant features for the classification problem

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

Authors Manoj S.V. & Rishi Kumar V. [3] performed an implementation-based study wherein they designed a model to detect phishing websites; this model was created during a university undergraduate project. They utilized various machine learning algorithms for their experiments, including decision tree, random forest, multilayer perceptron (MLP), and XG Boost, and trained these algorithms on data obtained from Phish Tank. Their experimental results showed that the MLP model outperformed the other models with an accuracy of 86.3%, whereas the other three algorithms (XG Boost and Random Forest) had similar performances.

Furthermore, the authors discuss multiple methods for extracting features from URLs, domains, and HTML/JavaScript, emphasizing the importance of pre- processing and feature engineering when implementing an effective phishing detection system. Challenges still exist today regarding the detection of sophisticated and zero-day phishing attacks, including most of the current models being trained with static datasets and thus, being unable to adapt quickly to new types of attacks. In addition, research areas focusing on detecting phishing sites in real time and across multiple platforms still require additional work.

IV. DATASETS

This research study used a dataset provided by UCI, titled "UCI Machine Learning Phishing Websites Dataset". This dataset contains 11,055 records and 32 attributes, with records representing individual web pages. The primary purpose of this dataset is to establish whether the web pages are valid/legitimate or phishing. Each attribute will have its numerical value assigned based on the characteristics of the web page. These attributes can be categorized into four main categories: URL Attribute (presence of "@" or "IP address"), Security Attribute (total number of words in the page/document and whether or not the page has an SSL Certificate), Content Attribute (links, popups, e-mails), and Traffic Attribute (page rank, web traffic) [20] [21]. This UCI Machine Learning dataset was selected from the UCI Machine Learning repository and was used by researchers to conduct experiments or develop machine learning models to determine whether a web page is a phishing website. This dataset consists of unstructured data that can help researchers learn how the attributes of a web page may indicate that the web page could be a phishing site [22].

V. METHODOLOGY

This research project focuses on creating an automated process for identifying phishing sites using machine learning algorithms. It utilizes information gathered from the URL, Domain Name, and Content of web pages to determine whether a web page is authentic (legitimate) or deceptive (phishing) [8]. This study presents a multi-step workflow approach that includes collecting data, cleaning data, extracting features, training and testing machine learning models, and finally deploying the best performing model(s) into a production environment. This multi-step approach should increase the accuracy and reliability of detecting phishing attempts. It also provides a framework for other organizations interested in developing similar types of systems.

The UCI Phishing Websites Dataset is made up of 11,055 samples and 31 features. The dataset's features consist of numerical indicators of URL structure, domain registration details, and webpage behaviour. The dataset's target variable (or variable to be predicted) "Result" was converted from the negative and positive classes (-1 and 1) to binary 0 and 1 to represent phishing and legitimate websites respectively. To ensure that a good proportion of examples from each of the two classes were retained within each half of the data, a stratified sampling approach was adopted to split the data into 80% training and 20% testing subsets.

During data preprocessing steps, missing values and inconsistencies were handled, and feature scaling using Standard Scaler was applied to enhance performance of models highly affected by the magnitude of features (Logistic Regression). Additionally, several features were extracted as indicators of phishing behaviour, including the presence of "@" symbols, URL length, the use of HTTPS, hyphenated domain names, and double "//" redirections, all of which have been cited in the literature as having high predictive power for identifying phishing activity.

Three machine learning models— XG Boost beat the rest of the models (Random Forest and Logistic Regression) as the best performer because it was able to identify the non-linear characteristics of data and reduce overfitting through regularization. Random Forest performed exceptionally well in terms of consistency and transparency because it is an ensemble model. Logistic Regression was used as a comparison for the other two models because it is the most basic linear model. The models were evaluated on multiple metrics including accuracy, precision, recall, F1 score, and confusion matrices. The evaluation of the importance of the features produced by the XG Boost model showed that structural URL features are among the top determinants for the prediction of phishing websites.

A. Logistic Regression

The Logistic Regression model is a type of learning that machines use. Machines can learn to put things into categories with the Logistic Regression model. It is not like models, such as linear regression. Linear regression gives us numbers. The Logistic Regression model is different it gives us a probability. This model tells us how likely something is to be in a group. We use the Logistic Regression model to answer questions. These questions are simple they usually have two possible answers. For example the answer can be Yes or No. It can also be True or False. Sometimes the answer is 0 or 1..The sigmoid function is defined as:

$$P(y = 1 \mid X) = 1 / 1+e-z$$

$$Z = w0 + \sum_{i=1}^{n} wi\ xi$$

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

where x_i represents the input features and $w_i$denotes the corresponding model parameters. A threshold value of 0.5 is typically used to assign class labels. Logistic Regression minimizes the binary cross-entropy loss function during training to achieve optimal parameter estimation. Feature scaling is often required to ensure stable and efficient convergence of the model. Due to its probabilistic output, the model also provides confidence estimates for each prediction.
In this study, Logistic Regression is used as a baseline model for performance comparison with more advanced ensemble- based classifiers.

B. Random Forest

Random Forest can be viewed as a supervised learning algorithm that can be systematically categorized under the umbrella of Ensembles. This process, known as ensembling, consists of many decision trees trained differently on varied subsets of the same data set, whose predictions are then aggregated into a single prediction. The class assigned to a target object for a classification problem can be obtained by performing a majority vote of all trees, while for the regression problem, it can be obtained by calculating an average of all predictions provided by all trees.

This algorithm begins with the creation of `n` decision trees via bootstrap sampling, whereby each of these `n` decision trees gets trained for a randomly chosen subset of the same dataset. This randomness or lack of correlation among tress increases their diversity. Individual decision trees are generated through recursive splitting of data depending upon some measure of impurities or purity, generally `Gini` Index, which can be explained by the following equation:

$$\text{Gini} = 1 - \sum_{i=1}^{c} \text{pi}\hat{}2$$

where $p_i$represents the probability of class $i$at a given node and $c$is the total number of classes. The feature and split that result in the minimum Gini impurity are selected to partition the data. After all trees generate their individual predictions T_k (X), the final classification output is determined using a majority voting mechanism:

$$\hat{y} = \text{mode}\{T1(X), T2(X), \ldots, TM(X)\}$$

where Mdenotes the total number of trees in the forest. This aggregation strategy helps mitigate overfitting and enhances generalization performance. Due to its ability to model complex non-linear relationships and its robustness against noisy features, Random Forest is well suited for phishing website detection using URL-based feature sets.

C. XG Boost (Extreme Gradient Boosting)

XG Boost or Extreme Gradient Boosting: This is the advanced form of the ensemble learning algorithm 'Gradient Boost Machines'. Unlike Random Forests, in this algorithm, decision trees are generated one by one, and each new tree is learned using the errors of all the previous trees.

This boosting strategy enables the model to learn complex non- linear relationships among features and significantly improve classification performance. The prediction of the XG Boost model is represented as an additive function of multiple regression trees:

$$\hat{y}i = \sum_{k=1}^{K} fk\,(xi)$$

where f_k denotes the k^{th} decision tree, $x_i$represents the input feature vector, and $K$is the total number of trees. Each tree contributes incrementally to the final prediction. XG Boost optimizes a regularized objective function defined as:

$$L = \sum_{i=1}^{N} l\,(yi, \hat{y}i) + \sum_{k=1}^{K} \Omega(fk)$$

where $l(\cdot)$is the loss function (logistic loss for binary classification) and $\Omega(f_k)$is a regularization term used to control model complexity. To efficiently minimize the objective function, XG Boost employs both first-order and second-order gradient information of the loss function using a Taylor series approximation. This optimization strategy enables faster convergence and leads to improved classification accuracy compared to traditional boosting methods. Additionally, XG Boost supports parallel tree construction and optimized memory management, making it computationally efficient and scalable for large-scale datasets. The algorithm also provides feature importance scores, which help identify the most influential URL features contributing to phishing detection, thereby improving model interpretability. Furthermore, XG Boost incorporates built-in handling of missing values and regularization mechanisms that enhance robustness against noisy and imbalanced data. Due to its strong generalization capability, scalability, and superior predictive performance, XG Boost achieved the highest accuracy among the evaluated models. Consequently, it was selected as the final model for real-time phishing website detection and deployed within the web-based application framework. In addition to its predictive accuracy, XG Boost offers flexibility through hyperparameter tuning, allowing control over learning rate, tree depth, and regularization strength to balance bias and variance effectively.

To provide value to users in the real world, a simple web application was developed using Flask which will allow users to enter a URL and receive a real-time prediction of the URL categorizing it as either Phishing or Legitimate with a confidence level associated with the classification. The web app will take the URL entered, parse it for the relevant features, carry out all the same preprocessing and scaling on the parsed features as was done during the training of the model, and then submit the parsed feature set to the trained XG Boost model for prediction. The web application is user-friendly and easy to navigate, built with HTML and CSS.

Overall, the proposed system has demonstrated a solid combination of machine learning and web-based technologies for phishing detection. The use of a standardized dataset provides an objective source for model evaluation, and both

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

ensemble-based models as well as baseline linear models provide an exhaustive way to assess both linear and non-linear model development. By implementing the model as a web- based application, the system is able to provide an efficient, helpful and extendable way to be able to improve your security on the web and protect users from phishing scams. Possible future studies may investigate integrating real-time threat intelligence, dynamic feature-update implementations, as well as advanced deep-learning techniques to further enhance model performance.

The process of phishing URL detection is illustrated in this flow diagram. The feature extraction stage is initiated from the input of the URL. After extracting relevant features, they are input into machine learning algorithms including Random Forest, Logistic Regression, and XG Boost, which provide an output indicative of whether the URL is legitimate or phishing, and display this information to the user. The system allows for the precise identification of phishing attempts in real-time, acting as a highly dependable and further providing users with an effective means of counteracting phishing attacks.

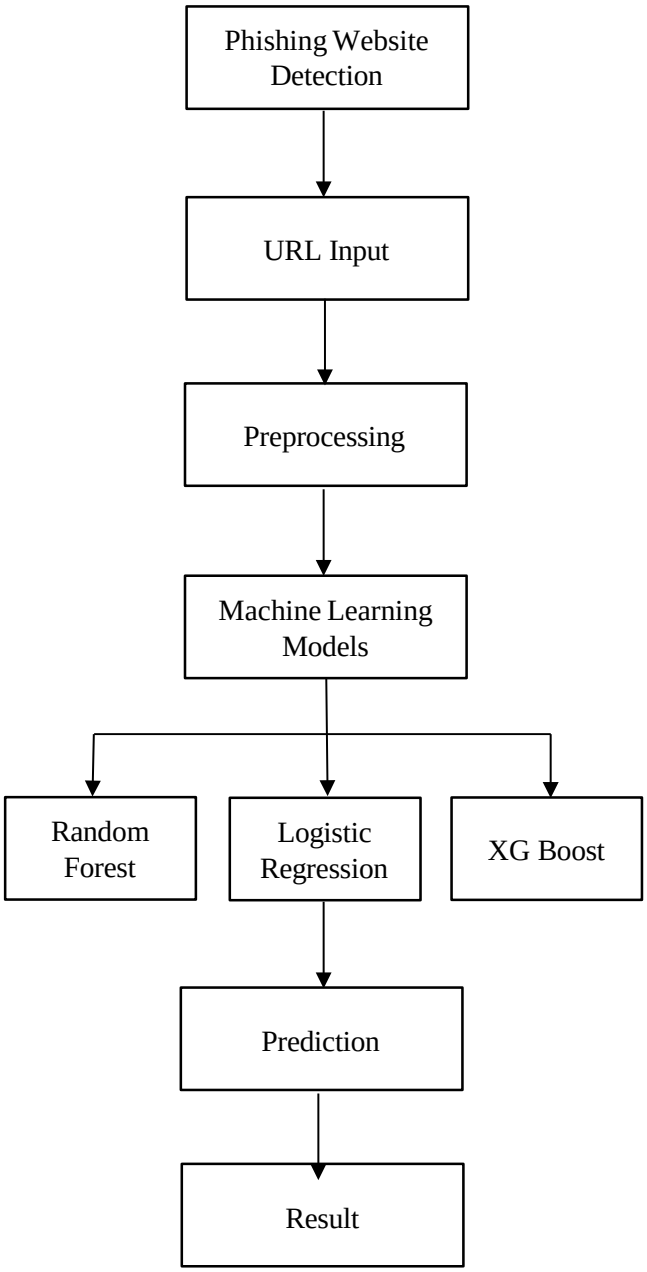

Fig 1. Architectural Diagram of URL Phishing Detection

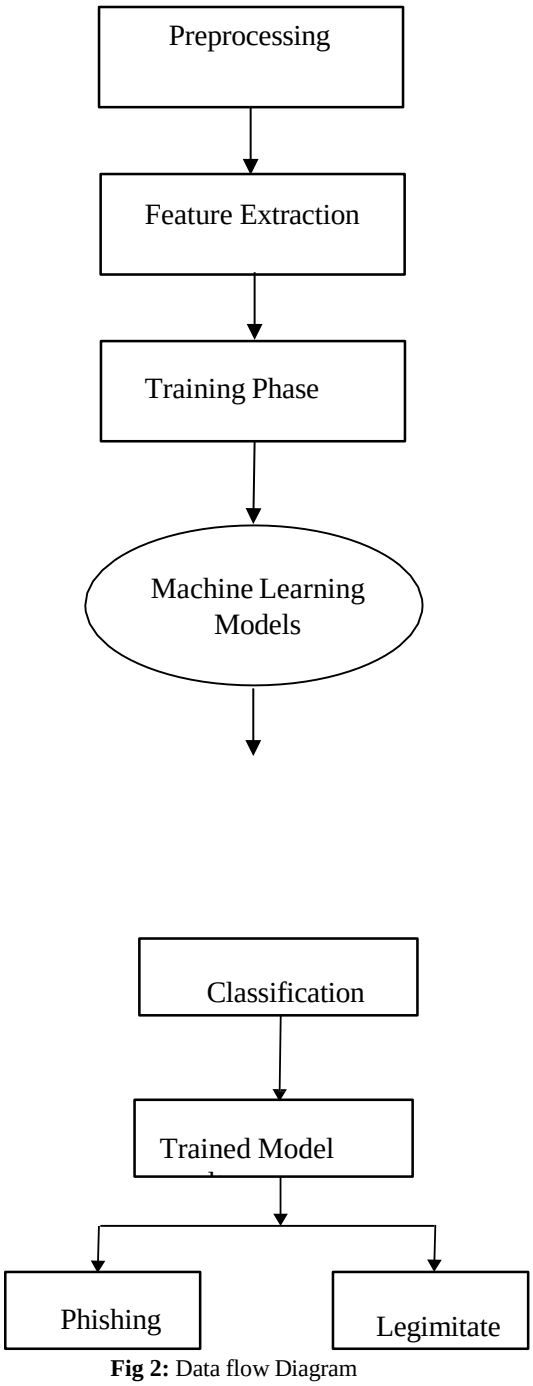

Fig 2: Data flow Diagram

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

VI. WEB APP IMPLEMENTATION

Utilizing the Flask framework, the phishing URL detection system has been developed as a web application which is an efficient and easy way to deploy machine learning models in a real-time program. When it is run, the Flask server is hosted locally at http://127.0.0.1:5000 so that a user can utilize a simple web form to input a URL to detect phishing. When the system is initiated, the backend retrieves the pre-trained XG Boost model, scaler and feature mappings using Joblib, ensuring the model is consistent with the training phase. During inference, once a user has entered the URL, the application extracts structural features of importance, including whether or not the URL contains an '@' symbol, hyphens, double slashes, the URL length, and if confidential information is present. This new data is then pre-processed with the saved scaler from the model training phase, and the pre-processed data is passed to the trained XG Boost model to classify the URL to 'Phishing' or 'Legitimate', while also producing a confidence level in the prediction. Then the final output is presented to the user in an interactive interface created with HTML and CSS, ensuring efficiency and clarity in use by all users. During development, Flask runs in debug mode, allowing for many features such as automatic reload when changes are made to the code, as well as an interactive debugger for effective testing and debugging. However, it is worth noting that the Flask development server is not intended for use in production due to limitations in scaling and security. A production WSGI server, such as Gunicorn, is highly recommended for deployment of production systems.

VII. RESULTS AND DISCUSSIONS

Three supervised Machine Learning algorithms, namely: XG Boost, Random Forest & Logistic Regression were implemented and evaluated for URL classification into Valid or Phishing. These ML algorithms were evaluated using industry standard evaluation metrics (Accuracy, Precision, Recall and F1-Score), which were computed based upon the performance of these ML algorithms on the test data set of n=2211 samples. The classification performance results from XG Boost show that it achieved overall the highest accuracy at 97.15% and average Precision, Recall and F1 score of approximately 0.97 for both classes. This suggests that XG Boost was able to learn the underlying complex interactions between features present in the URL dataset and has shown a high degree of generalization ability when dealing with previously unseen URLs.

The Random Forest Classifier outperformed the other classifications with 96.61% accuracy. That's only slightly lower (by ~0.17%) than the XG Boost model at 96.78% accuracy, so they have very similar performance. Both models produce ensemble types of models and employ the same methods for preventing overfitting, which results in recall and precision for the various types of label legitimacy.

As a result, Logistic Regression can also be a good tool for developing binary classification methods because it is relatively straightforward to use and offers very fast computation time when building the models. However, since it is a purely linear type of model, Logistic Regression cannot accurately reflect the nonlinearity of the correlations between its predictors.

Methods	Accuracy	Precision	Recall
Logistic Regression	92.81%	0.92	0.95
Random Forest	96.61%	0.97	0.97
XG Boost	97.15%	0.97	0.98

Table 1: Accuracy Comparison

The enhanced performance of ensemble model types such as XG Boost is due to their ability to iteratively train weak learners while optimising the boundaries of classification using gradient boosting methods to achieve better results than other methods. Both Random Forest, which is based on bag data sampling and randomising the features, also performed well in this task; however, XG Boost demonstrated better precision and recall consistency than Random Forest.

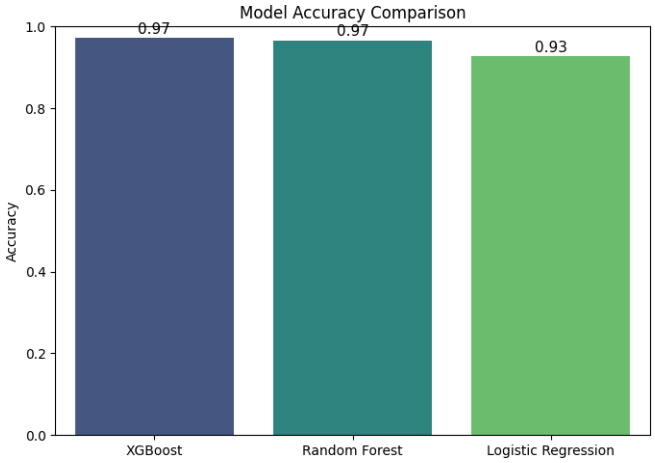

Fig 3. Analysis of model Accuracies

Ultimately, the overall results from our experiments demonstrate that ensemble learning methods can provide a high level of accuracy when detecting phishing URLs. Additionally, the resulting models have demonstrated the ability to perform at or near 100% predictive accuracy and have provided a balanced output in terms of accuracy and overall classification metrics, making them suitable for use in phishing detection products for real-time applications. The algorithms we tested demonstrated that XG Boost is the most reliable and accurate of the methods, making it a good option for implementation in real-world applications.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

The Line Chart depicts the model accuracy for the three ML classifier models: XG Boost, Random Forest & Logistic Regression. On the X-Axis items are Model names on the Y- Axis are their Model Accuracy. The XG Boost Model had 97.15% highest Accuracy (2nd highest was Random Forest 96.61%, Lowest was Logistic Regression 92.81%) The Results indicate that the ensemble-based models (XG Boost & RF), which are designed to detect and classify complex patterns (phishing URLs) outperform linear models (Logistic Regression). The trends & numerical data were sufficient to validate the choice of the XG Boost Classifier as the 1st model selected for Real-Time detection and deployment in a phishing detection system. In Detail, The Graph indicates The Performance of the XG Boost model is sufficient compared to Random Forest & Logistic Regression, indicating that XG Boost can reliably provide excellent performance in a production environment for phishing detection systems. Overall, the graphic enables a rapid understanding of the model's performance, validating the ability of the proposed machine-learning methodology to accurately identify and safeguard against phishing attacks.

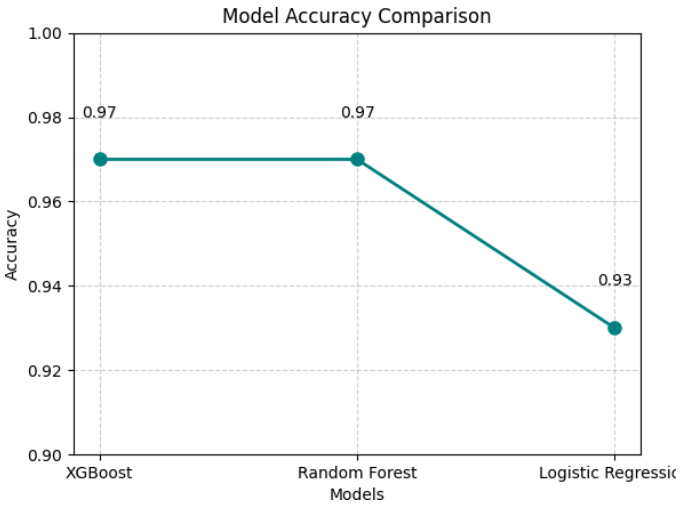

Fig.4 Graph: Analysis of model accuracies

To assess the classifier performance in greater detail, confusion matrices were produced for XG Boost, Random Forest, and Logistic Regression classifiers. The data shown in the confusion matrices displays, visually, how correct and incorrect predictions are divided between a phishing and legitimate class. The results from XG Boost were the most balanced and accurate, with many true positives and true negatives, which indicates its ability to accurately classify phishing versus legitimate URLs. The Random Forest model came close in performance, but it had a slightly greater number of false negatives than XG Boost. Finally, Logistic Regression showed a greater number of misclassifications than the other models, which indicates its reduced capability to account for the complex, non-linear structures in the dataset. Overall, the confusion matrices demonstrate the ensemble models, particularly XG Boost, provided a more accurate and robust predictive capability in the area of phishing detection.

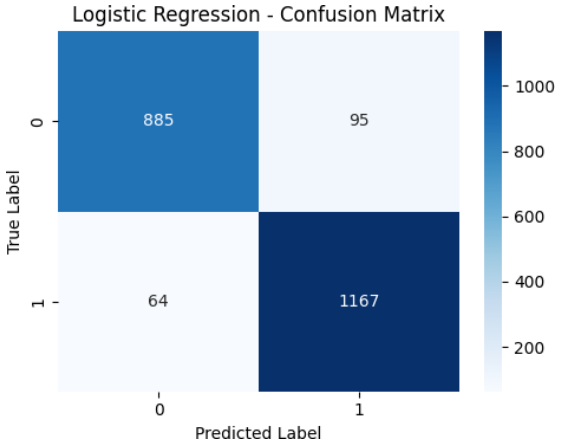

Fig 5. Confusion matrix of Logistic Regression

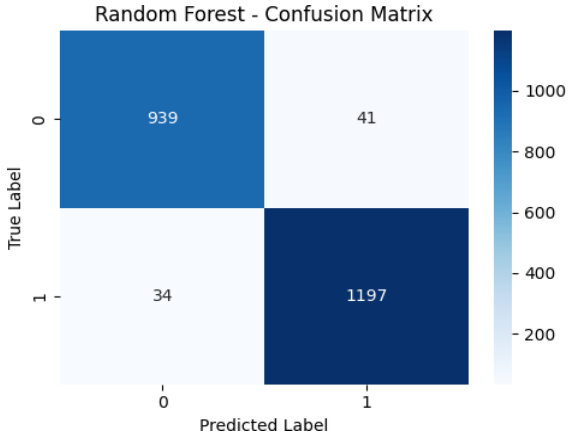

Fig 6. Confusion matrix of Random Forest

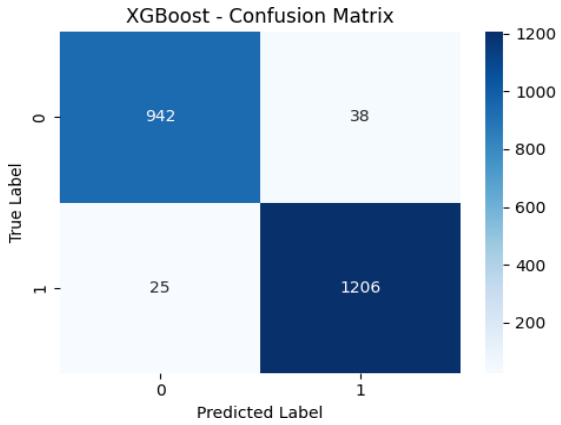

Fig 7. Confusion matrix of XG Boost

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

Model	TN	FP	FN	TP
Logistic Regression	885	95	64	1167
Random Forest	939	41	34	1197
XG Boost	942	38	25	1206

Table 2: Confusion matrix

XG Boost shows the best performance, with the fewest misclassifications (lowest FP and FN). Random Forest performs slightly worse but still better than Logistic Regression. Logistic Regression, while simpler, has the lowest accuracy and higher errors.

Fig 8. Web Page of Phishing

VIII. CONCLUSION

Phishing has become one of the most significant and harmful categories of cybercrime, reliant upon the relationship of trust and increasing dependency on digital services. Conventional security mechanisms - at the extreme are based on blacklisting phishing web addresses or heuristic analysis, do not effectively analyze and recognize newly generated or zero-day phishing websites. This paper illustrates the value of improving the detection of phishing websites through the utilization of machine learning models, based on URL feature extraction, page content, and domains. Using algorithms based on Decision Tree, Random Forest, Support Vector Machine and Gradient Boosting methods, the models produce effective and reliable detection of usually difficult to identify phishing websites from legitimate websites.

Overall, machine learning not only improves automatic identification but it shortens time consuming human effort and response time in security operations reaction. The Random Forest and ensembles showed improved performance with reduced overfitting and inefficiency in a large, multi- dimensional search space. Additionally, real-time URL analysis and feature selection methods lead to the development of a responsive detection system capable of detecting phishing scams because the system adapted to up-to-the-minute methods of phishers.

Machine learning has proven to be an effective phishing detection system, while showing scalability in detecting new phishing modalities.

REFERENCES

[1] Sakhare, N. N., Bangare, J. L., Purandare, R. G., Wankhede, D. S., & Dehankar, P. (2024). Phishing website detection using advanced machine learning techniques. *International Journal of Intelligent Systems and Applications in Engineering*, *12*(12s), 329-346.

[2] Krishna, V. A., Jose, B., Anilkumar, K., & Lee, O. (2021). Phishing detection using Machine Learning based URL analysis: A survey. *International Journal of Engineering Research & Technology*, *9*, 156-161.

[3] Mandadi, A., Boppana, S., Ravella, V., & Kavitha, R. (2022, April). Phishing website detection using machine learning. In *2022 IEEE 7th International conference for Convergence in Technology (I2CT)* (pp. 1-4). IEEE.

[4] Yang, R., Zheng, K., Wu, B., Wu, C., & Wang, X. (2021). Phishing website detection based on deep convolutional neural network and random forest ensemble learning. *Sensors*, *21*(24), 8281.

[5] Stobbs, J., Issac, B., & Jacob, S. M. (2020, December). Phishing web page detection using optimised machine learning. In *2020 IEEE 19th international conference on trust, security and privacy in computing and communications (TrustCom)* (pp. 483-490). IEEE.

[6] Alshdadi, A. A. (2024, July). LSTM-PSO: NLP-based model for detecting phishing attacks. In *Proceedings of the First International Conference on Natural Language Processing and Artificial Intelligence for Cyber Security* (pp. 70-79).

[7] Pawar, B. R., & Pawar, N. R. (2021). *Detection of Phishing URL using Machine Learning* (Doctoral dissertation, Dublin, National College of Ireland).

[8] Sahoo, D., Liu, C., & Hoi, S. C. (2017). Malicious URL detection using machine learning: A survey. *arXiv preprint arXiv:1701.07179*.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 4436-4444

[9] Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. (2018). URLNet: Learning a URL representation with deep learning for malicious URL detection. *arXiv preprint arXiv:1802.03162*.

[10] Jha, A. K., Muthalagu, R., & Pawar, P. M. (2023). Intelligent phishing website detection using machine learning. *Multimedia Tools and Applications*, *82*(19), 29431-29456.

[11] Maneriker, P., Stokes, J. W., Lazo, E. G., Carutasu, D., Tajaddodianfar, F., & Gururajan, A. (2021, November). Urltran: Improving phishing url detection using transformers. In *MILCOM 2021- 2021 IEEE Military Communications Conference (MILCOM)* (pp. 197-204). IEEE.

[12] Aljabri, M., Altamimi, H. S., Albelali, S. A., Al-Harbi, M., Alhuraib, H. T., Alotaibi, N. K., ... & Salah, K. (2022). Detecting malicious URLs using machine learning techniques: review and research directions. *IEEE Access*, *10*, 121395-121417.

[13] Das Guptta, S., Shahriar, K. T., Alqahtani, H., Alsalman, D., & Sarker, I. H. (2024). Modeling hybrid feature-based phishing websites detection using machine learning techniques. *Annals of Data Science*, *11*(1), 217-242.

[14] Bahaghighat, M., Ghasemi, M., & Ozen, F. (2023). A high-accuracy phishing website detection method based on machine learning. *Journal of Information Security and Applications*, *77*, 103553.

[15] Safi, A., & Singh, S. (2023). A systematic literature review on phishing website detection techniques. *Journal of King Saud University- Computer and Information Sciences*, *35*(2), 590-611.

[16] Opara, C., Chen, Y., & Wei, B. (2024). Look before you leap: Detecting phishing web pages by exploiting raw URL and HTML characteristics. *Expert Systems with Applications*, *236*, 121183.

[17] Haq, Q. E. U., Faheem, M. H., & Ahmad, I. (2024). Detecting phishing URLs based on a deep learning approach to prevent cyber-attacks. *Applied Sciences*, *14*(22), 10086.

[18] Fajar, A., Yazid, S., & Budi, I. (2024). Enhancing phishing detection through feature importance analysis and explainable ai: A comparative study of catboost, xgboost, and ebm models. *arXiv preprint arXiv:2411.06860*.

[19] Kulkarni, A. D., & Brown III, L. L. (2019). Phishing websites detection using machine learning.

[20] Alam, M. N., Sarma, D., Lima, F. F., Saha, I., & Hossain, S. (2020, August). Phishing attacks detection using machine learning approach. In *2020 third international conference on smart systems and inventive technology (ICSSIT)* (pp. 1173-1179). IEEE.

[21] Phadke, P., & Thorpe, C. (2021, June). Analysis of API driven application to detect smishing attacks. In *European Conference on Cyber Warfare and Security* (pp. 588-XV). Academic Conferences

[22] Dobolyi, D. G., & Abbasi, A. (2016, September). Phishmonger: A free and open source public archive of real-world phishing websites. In *2016 IEEE conference on intelligence and security informatics (ISI)* (pp. 31-36). IEEE.

[23] Tanti, R. (2024). Study of Phishing Attack and their Prevention Techniques. *International Journal of Scientific Research in Engineering and Management*, *8*(10), 1-8.

[24] Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009, June). Identifying suspicious URLs: an application of large-scale online learning. In *Proceedings of the 26th annual international conference on machine learning* (pp. 681-688).

[25] Tuoyo, O. S., Hossain, A., Habibur, H. B., Rahman, M. A. A. M., Hussein, L., Khan, M. A., ... & Shah, S. (2024). The Role of Machine Learning and Deep Learning in Shaping Modern Computer Science: Challenges, Opportunities, and Future Directions. *ResearchGate, September*.

[26] Rehman, A. U., Imtiaz, I., Javaid, S., & Muslih, M. (2025). *Real-Time Phishing URL Detection Using Machine Learning. Eng. Proc. 2025, 107, 108*.

[27] Sruthi, K. (2025). A novel framework for effective phishing URL detection using an LSTM-based Siamese network. *Knowledge-Based Systems*, 11