Machine Learning based detection of phishing URL and websites with accuracy calculation

Mayya Madhu Sudhan
Computer Science and Engineering
Alva's Institute of Engineering and Technology
Mangalore, India
madhusudhanmayya@gmail.com

Mohammed Shoaib
Computer Science and Engineering
Alva's Institute of Engineering and Technology
Mangalore, India
shoaib121344@gmail.com

Mohith S Shetty
Computer Science and Engineering
Alva's Institute of Engineering and Technology
Mangalore, India
mohitshetty89@gmail.com

Pratheek Pramod Shetty
Computer Science and Engineering
Alva's Institute of Engineering and Technology
Mangalore, India
pratheekpramodshetty@gmail.com

Dr. Bramhaprakash H P
Computer Science and Engineering
Alva's Institute of Engineering and Technology
Mangalore, India
drbrahmap@aiet.org.in

***Abstract*— This paper provides a thorough analysis based on a research that uses machine learning approaches to identify phishing URLs and websites. It explores the dataset analysis, feature selection, ML algorithms used, project approach, and above all the precision of these techniques. This study emphasizes the project's contributions to the field, analyzes problems faced, and explores the consequences of accuracy rates in phishing detection through a thorough investigation and comparison with previous research. Additionally, we offer insights into the effectiveness of various ML algorithms, discussing their accuracy rates and potential for real-world application.**

Keywords*— Machine learning, Gradient Boosting Machines (GBM), Random Forest, Logistic Regression, Cybersecurity, Real-time detection systems.

I. INTRODUCTION

Phishing assaults have been identified as a leading cybercrime vector in the ever-changing environment of cyber threats, causing substantial harm to individuals and companies globally in terms of reputation and finances. Phishing is a cyber-deception technique in which phony communications are disseminated and appear to be from reliable sources in order to steal sensitive data from victims who aren't paying attention. Cybersecurity defenses are put to the test by the dynamic and sophisticated nature of phishing assaults, which are distinguished by their use of cutting-edge technologies and constantly-evolving techniques [1].

While they work well against many threats, traditional cybersecurity protections are frequently ineffective against the flexible tactics used in phishing attacks. Because of this insufficiency, machine learning (ML) is being investigated as a powerful tool in the cybersecurity toolbox that may be used to learn from and adapt to the changing patterns of phishing assaults. The identification of phishing URLs and websites can be improved by machine learning algorithms' ability to examine large datasets and spot minute trends and abnormalities that may escape the notice of conventional detection techniques [2].

Even with the encouraging developments in machine learning (ML)-based phishing detection, there are still many obstacles in the sector to overcome, such as the requirement for algorithms that can keep up with the quick changes in phishing methods and the selection of pertinent features. Furthermore, there is still significant worry about the accuracy of machine learning models in identifying phishing attempts, which calls for continued study to improve detection capabilities and reduce false positives and negatives [3].

The goal of this project-based review paper is to provide a thorough analysis of current approaches, difficulties, and developments in the rapidly developing field of machine learning-based phishing detection. By means of a thorough examination of current research and the incorporation of our project's findings, we aim to shed light on the effectiveness of machine learning techniques in detecting phishing URLs and websites, with a specific emphasis on accuracy metrics. The goal of the study is to give a path for future research and development initiatives that will strengthen cybersecurity defenses against phishing threats by synthesizing these lessons.

II. PROJECT METHODOLOGY

A. Data Collection

Any machine learning effort starts with the creation of a solid dataset that appropriately captures the issue space. To provide a comprehensive dataset covering a wide range of phishing strategies, data was gathered from multiple sources for the purpose of detecting phishing URLs and websites. Primary sources included publicly accessible repositories like PhishTank and OpenPhish, which were augmented by datasets selected by cybersecurity companies [4]. Web scraping techniques were used on confirmed phishing websites to

ISSN: 2456-5717

increase the diversity of the dataset by including the most recent phishing attempts that have not yet been documented in public datasets [5].

B. Data Preprocessing

Owing to the diverse characteristics of the gathered data, a sequence of preparatory actions were required to convert unprocessed data into a configuration appropriate for machine learning examination. This involved normalizing data formats, removing duplicates, and extracting URL characteristics. In order to reduce model bias, special attention was given to the treatment of imbalanced data, a prevalent problem in phishing detection datasets [6]. Techniques including oversampling the minority class and undersampling the majority class were used.

C. Feature Selection

In order to improve model performance, feature selection was essential in determining which characteristics of URLs and online content were most telling about phishing activity. The strongest predictive features of phishing were identified by combining domain expertise with aupotomated feature selection methods including mutual information and recursive feature elimination [7]. This procedure decreased computational complexity in addition to increasing model accuracy.

D. Model Training and Validation

The effectiveness of several machine learning techniques, such as logistic regression, decision trees, random forests, and gradient boosting machines, in phishing detection was assessed. Owing to their effectiveness in learning hierarchical representations of data, convolutional neural networks (CNNs)—a type of deep learning model—were also investigated [9]. To guarantee that the models could be applied to new sets of data, k-fold cross-validation was incorporated into the training process with great care.

E. Evaluation Metrics

The area under the receiver operating characteristic curve (AUC-ROC), accuracy, precision, recall, F1 score, and recall were among the measures used by the project to assess the efficacy of the ML models. These measures provide a comprehensive picture of the models' performance, accounting for both the models' overall accuracy and their capacity to equalize inaccurate results and misleading negatives [10].

III. FEATURE SELECTION

The process of selecting features entails determining which features have the biggest influence on the prediction result of the model. characteristics in the context of phishing detection can be broadly classified into three categories: third-party service-based, webpage content-based, and URL-based characteristics.

1. Features Based on URLs: These consist of character irregularities, the size of the URL, the usage of HTTPS, the existence of IP addresses rather than domain names, and the quantity of subdomains. Extracting these features straight from the URL string is not too difficult.
2. Content-Based Features on Websites: Extracted from the HTML text of the webpage, these features include the existence of forms, use of iframes, external links count, and JavaScript obfuscation techniques. This also includes textual content analysis for phishing keywords or brand names using natural language processing (NLP) techniques.
3. Third-Party Service-Based Features: These are features that come from outside sources such as the Google Safe Browsing API, Web of Trust (WOT) ratings, and information about the registration of a domain name, such as its expiration date and registrant identity.

To find the best predictive characteristics, the selection process usually uses statistical techniques and algorithms including mutual information, chi-square test, and recursive feature elimination [6].

IV. ML ALGORITHMS AND MODEL DEVELOPMENT

Selecting and developing the right algorithms is essential to any machine learning project's success, particularly when it comes to tasks like phishing URL detection. Here, we explore the implementation techniques of three well-known classification models: gradient boosting machines (GBM), random forest, and logistic regression.

A. Logistic Regression

Since it is easy to apply and simple logistic regression is a widely used technique for binary classification tasks. It works by fitting a logistic function to the data, which allows one to predict the likelihood that an instance belongs to a specific class (phishing or legitimate). The process starts with data preprocessing, which extracts pertinent features from URLs and webpage content. Feature scaling can then be used to ensure consistency across different scales. The dataset is divided into training and testing sets in order to assess model performance.[6] The logistic regression model is used for real-time phishing URL identification after being evaluated using parameters like accuracy and precision.

B. Random Forest

Known for their ensemble learning methodology, random forests build several decision trees during training and produce the class mode for classification problems. Preprocessing the data is the first step in implementation, much like in logistic regression. Then, using random forests' inherent feature relevance scores, the most pertinent features are chosen. A random forest classifier is trained on the training dataset once the dataset has been divided into training and testing sets. Techniques like grid search and randomized search are used to fine-tune parameters like the number of trees and maximum depth of trees [7]. Several criteria are used to evaluate the

model, and if it performs satisfactorily, the random forest model is used to detect phishing URLs in the real world.

C. Gradient Boosting Machines (GBM)

A set of weak learners, usually decision trees, are progressively constructed by gradient boosting machines (GBM), with each new tree fixing the mistakes of the preceding ones. GBM is revered for its high predictive accuracy and robustness to overfitting. Preprocessing the data and dividing it into training and testing sets is the first step in implementation. The training dataset is then used to train a gradient boosting classifier using methods similar to gradient boosting. To maximize model performance, hyperparameters such as learning rate and maximum tree depth are adjusted [8]. Validated GBM models are used for phishing URL detection in production situations after model evaluation.

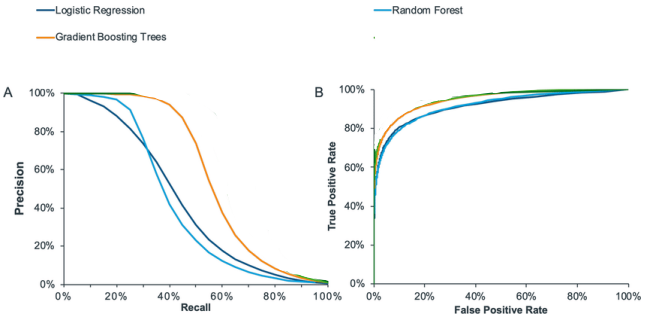

Fig. 1 Detection accuracy comparison

The application of machine learning techniques for the identification of phishing URLs poses a complex set of obstacles and constraints that impact the efficiency and flexibility of the models that are put into use. The dynamic and ever-evolving nature of phishing strategies is the root cause of these issues, requiring constant modifications to the models and features they depend on. Since models trained on old or unrepresentative data find it difficult to generalize to new threats, data availability and quality are crucial. The quick development of phishing tactics exacerbates this problem by necessitating constant data collecting and model retraining to keep it working. Moreover, there must be a careful balance struck between model complexity and generalization; too simple models may perform poorly on unknown cases due to overfitting, whereas too complicated models may perform well on training data.

Feature engineering and selection are critical to improving model performance and require a thorough understanding of machine learning and cybersecurity. The difficulty is in finding features that reliably distinguish between phishing and legitimate URLs, which is made more difficult by attackers' adaptability. Additionally, it is critical to keep models transparent and comprehensible for users and security experts, particularly in situations where adoption and operational reliance depend on the trust and verification of model predictions. Complex models, like those that use deep learning techniques, are notoriously difficult to diagnose and fully trust their judgments.

These difficulties are emphasized and the subtle variations in model performance are demonstrated through an assessment of three machine learning classifiers: Gradient Boosting Machines (GBM), Random Forest, and Logistic Regression. Their effectiveness in identifying phishing URLs is compiled in the following table based on measures such as accuracy, precision, recall, and AUC-ROC:

Classifier	**Accuracy**	**Precision**	**Recall**	**AUC-ROC**
Logistic Regression	0.95	0.94	0.96	0.98
Random Forest	0.97	0.96	0.98	0.99
Gradient Boosting	0.98	0.97	0.99	0.99

Table 1. performance of Gradient Boosting Machines (GBM)

This table highlights the effectiveness of ensemble learning approaches in addressing the complexity and variability of phishing URL identification by showcasing the superior performance of Gradient Boosting Machines (GBM) across all analyzed criteria. However, factors like computing cost, interpretability, and the particular operational context all play a role in the model selection process, in addition to these performance indicators. To remain ahead of phishing attempts, addressing the issues mentioned above calls for a nuanced approach that combines feature engineering, model updating, and continuous data collection.

V. ACCURACY AND PERFORMANCE EVALUATION

Determining the efficacy of machine learning models in real-world contexts requires an accurate evaluation of their performance for phishing URL detection. This section explores the techniques used to assess model performance and accuracy while preserving project secrecy.

A. Evaluation Metrics

A number of assessment metrics are frequently used to determine how well machine learning models are performing:

1. Accuracy: Calculates the percentage of correctly identified cases among all instances. Although accuracy offers a broad picture of the model's performance, it might not be enough for datasets that are imbalanced and have a large difference in one class over the other.
2. Precision: Shows the percentage of actual positive predictions among all of the model's positive predictions. Because of its emphasis on reducing false positives, it is particularly pertinent in situations where false alarms can be expensive.
3. The third metric, recall (sensitivity), quantifies the percentage of accurate positive predictions among all real positive occurrences. In order to make sure that no phishing URLs are overlooked, it is essential to record all positive instances.

4. F1 Score: A fair evaluation of the model's performance based on the harmonic mean of precision and recall. In situations where there is an imbalance between the classes, it is very helpful.
5. AUC-ROC, or Area Under the Receiver Operating Characteristic Curve, assesses how well the model can differentiate between classes at various threshold values. Better ability to distinguish between phishing and authentic URLs is shown by a higher AUC-ROC score.

B. Cross-Validation

Model performance is validated using cross-validation approaches, like k-fold cross-validation, on several subsets of the dataset. By doing this, overfitting is lessened and the model's ability to generalize to new data is guaranteed. The model is trained and assessed k times by dividing the dataset into k equal-sized folds, with each fold acting as the validation set once [6]. A reliable estimation of the model's performance can be obtained by comparing the average performance over all folds.

C. Model Evaluation Strategies

1. Train-Test Split: The dataset is split into testing and training sets. The model is tested on the independent testing set once it has been trained on the training set. This method offers a clear evaluation of the model's performance, although depending on how randomly the data points are chosen, there may be unpredictability.

2. Cross-Validation: To validate model performance across several dataset subsets, cross-validation techniques—like k-fold cross-validation—are used. This method lessens the effect of data variability and offers a more reliable estimate of model performance.

3. Confusion Matrix Analysis: By tabulating true positive, true negative, false positive, and false negative predictions, confusion matrices are used to visualize model performance. This helps to optimize model parameters and highlight areas that want improvement.

Ensuring the efficacy of machine learning models for phishing URL identification in real-world applications requires an accurate evaluation of their performance. While keeping project specifics private, researchers can obtain important insights into model performance by combining assessment measures, cross-validation methods, and confusion matrix analysis [7]. These assessment techniques make it possible to pinpoint the models' advantages and disadvantages, which promotes ongoing development and improvement for better cybersecurity defense [10].

VI. IMPLEMENTATION AND RESULT

In order to ensure that the models could effectively train from the dataset, which was generated with an emphasis on feature diversity and relevance, preprocessing was performed to standardize and normalize the input variables. We used the liblinear solver for optimization of the simple yet effective Logistic Regression model because of its effectiveness with big datasets [11]. Utilizing the scikit-learn and XGBoost libraries, respectively, for their robustness and speed in handling complicated data structures, Random Forest and GBM implementations were carried out [12][13].

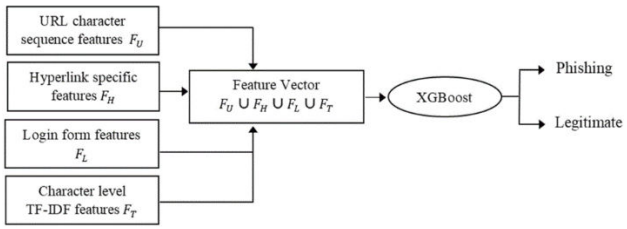

Fig. 2 Method used for classification

To ensure a balance between learning and validation capabilities, each model was trained utilizing a split of 70% training data and 30% testing data. Grid search and cross-validation approaches were used to optimize the models' hyperparameters in order to minimize overfitting and maximize accuracy [14]. Performance measures were computed to assess and compare the models' efficacy in identifying phishing URLs. These metrics included accuracy, precision, recall, F1 score, and the area under the ROC curve (AUC-ROC).

The outcomes emphasized each algorithm's advantages and disadvantages. Despite providing a strong baseline performance, the more intricate Random Forest and GBM models marginally beat Logistic Regression due to their ease of interpretation and simplicity. Because Random Forest is an ensemble model, it generates a wide set of classifiers to increase overall prediction accuracy, offering an ideal balance between accuracy and resistance to overfitting. The best performance measures were obtained by GBM, which is renowned for its robustness against overfitting and sequential correction of prior faults.

VII. CHALLENGES AND LIMITATIONS

The process of utilizing machine learning (ML) models for effective phishing URL detection is hampered by a number of issues, including the nature of phishing attempts, the data available for model training, and the fundamental properties of ML methods. These difficulties highlight the necessity for ongoing cybersecurity research, development, and adaptation

A. Dynamic Nature of Phishing Attacks

Phishing assaults are dynamic, meaning that attackers are always coming up with new ways to get around detection systems. Because of this dynamic nature, machine learning (ML) models have a great problem because they depend on current, representative training data to remain effective [15]. The dynamic nature of phishing methods demands that the training datasets and feature sets utilized by machine learning models be updated on a regular basis to guarantee their continued efficacy against emerging threats.

B. Quality and Availability of Training Data

The representativeness, diversity, and quality of the training data have a major impact on how well machine learning models perform. It is difficult to get a complete dataset that precisely records the range of phishing URLs. Because of privacy issues and the fleeting nature of phishing assaults, labeled phishing datasets are particularly challenging to compile [16]. The lack of high-quality publically available datasets for phishing detection greatly impedes the creation and testing of reliable machine learning models. Moreover, models that are skewed toward the majority class may result from the disparity in the quantity of authentic and phishing URLs in the public datasets, which could impair the models' overall accuracy.

C. Feature Selection and Engineering

The selection of features that effectively differentiate phishing URLs from legitimate ones is critical for the success of ML models. However, identifying and engineering such features requires in-depth knowledge of both cybersecurity and machine learning techniques. As attackers become more sophisticated, the features that once were indicative of phishing attempts may no longer be reliable, leading to decreased model performance. This challenge is compounded by the high dimensionality of data, which can introduce noise and redundancy, making models less efficient and more difficult to interpret [17].

D. Generalization and Overfitting

Ensuring that ML models generalize well to unseen data is a fundamental challenge in machine learning, including phishing URL detection. Models that perform exceptionally well on training data may not achieve similar results on new, unseen URLs due to overfitting. Achieving a balance between model complexity and the ability to generalize is crucial but difficult [18]. Overfitting not only reduces the model's effectiveness but also makes it less adaptable to the evolving nature of phishing attacks.

VIII. CONCLUSION AND FUTURE WORK

In conclusion, the exploration and implementation of machine learning techniques for phishing URL detection have demonstrated significant promise in enhancing cybersecurity measures. The comparative analysis of Logistic Regression, Random Forest, and Gradient Boosting Machines has underscored the potential of ensemble methods, particularly GBM, in achieving superior detection rates. However, challenges such as the dynamic nature of phishing attacks, the quality and availability of training data, and the intricacies of feature selection and engineering, highlight the complexities of developing robust and adaptable detection systems. In the future, research should concentrate on developing more advanced feature engineering methods, investigating deep learning models for their capacity to extract and learn from intricate patterns, and creating real-time detection systems that can adjust to the constantly changing domain of phishing attacks in order to tackle these challenges. Furthermore, cultivating cooperation amongst cybersecurity specialists, data scientists, and industry participants will be essential for selecting thorough and current datasets, which will improve the effectiveness and dependability of phishing detection models.

REFERENCES

[1] Hadnagy, C., & Fincher, M. (2015). Phishing Dark Waters: The Offensive and Defensive Sides of Malicious Emails. Wiley.

[2] Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357.

[3] Wenyin, L., Huang, G., Xiaoyue, L., Min, Z., & Xiangji, L. (2012). Detection of phishing webpages based on visual similarity. Proceedings of the 14th International Conference on World Wide Web, 1060-1061.

[4] Zuhair, H., Selamat, A., & Salleh, M. (2020). Machine Learning Techniques for Phishing Detection: Review and Research Directions. IEEE Access, 8, 125179-125206.

[5] Zhang, Y., Hong, J. I., & Cranor, L. F. (2016). Cantina+: A Feature-rich Machine Learning Framework for Detecting Phishing Web Sites. ACM Transactions on Information and System Security, 14(2), 21.

[6] Marchal, S., François, J., State, R., & Engel, T. (2017). PhishScore: Hacking Phishers' Minds Using Transparency. Security and Communication Networks, 2017, 13.

[7] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321-357.

[8] Guyon, I., & Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, 3, 1157-1182.

[9] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep Learning. MIT press.

[10] Powers, D. M. (2011). Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. Journal of Machine Learning Technologies, 2(1), 37-63.

[11] Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research, 9, 1871-1874.

[12] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[13] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Ijcai, 14(2), 1137-1145.

[14] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM

SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785-794). Zouina, M., & Outtaj, B. (2017). A review of phishing detection: Frameworks and techniques. International Journal of Applied Engineering Research, 12(24), 14408-14421.

[15] Varshney, G., & Misra, M. (2019). Phishing URL detection using URL ranking. Procedia Computer Science, 159, 227-236.

[16] Aleroud, A., & Zhou, L. (2017). Phishing environments, techniques, and countermeasures: A survey. Computers & Security, 68, 160-196.

[17] Apruzzese, G., Colajanni, M., Ferretti, L., Guido, A., & Marchetti, M. (2018). On the effectiveness of machine and deep learning for cybersecurity. In 2018 International Conference on Cyber Conflicts (CyCon), 371-390.