Kashf Journal of Multidisciplinary Research
Vol: 02 - Issue 3 (2025)
P-ISSN: 3007-1992 E-ISSN: 3007-200X
https://kjmr.com.pk

# DETECTING PHISHING ATTACKS IN CYBERSECURITY USING MACHINE LEARNING WITH DATA PREPROCESSING AND FEATURE ENGINEERING

***Sohaib Latif****
*Department of Computer Science and Software Engineering, Grand Asian University, Sialkot, 51310, Pakistan.*
***Saher Pervaiz***
*Department of Computer Science, The University of Chenab, Gujrat, 50700, Pakistan.*

**Corresponding author: Sohaib Latif (sohaiblatif095@gmail.com)*

***DOI : https://doi.org/10.71146/kjmr335***

Article Info

## Abstract

Phishing attacks are one of the most persistent cybersecurity threats, evolving rapidly to bypass traditional security measures. Given the widespread use of email for sensitive communications, detecting phishing attempts has become more critical than ever. This study explores the effectiveness of multiple machine learning models in classifying phishing emails using a dataset of 39,000 samples. To enhance accuracy, we employ preprocessing techniques such as feature engineering, vectorization, and class balancing with SMOTE (Synthetic Minority Over-sampling Technique). Our analysis compares various models, including Random Forest, XGBoost, Logistic Regression, Naïve Bayes, and AdaBoost, evaluating their performance using precision, recall, F1-score, and accuracy metrics. The results demonstrate that ensemble learning techniques, particularly XGBoost and Random Forest, significantly outperform other models, achieving accuracy rates as high as 99.00%. These findings reinforce the importance of advanced classification techniques and data preprocessing in phishing detection. Beyond academic implications, our research contributes to strengthening email security, mitigating financial losses, and protecting personal data from cyber threats. Future work could focus on integrating deep learning models and real-time detection systems to further improve accuracy and adaptability.

**Keywords:**
*Phishing Detection; Email Security; Ensemble Learning; Fraud Detection; Spam Filtering*

## Introduction

Phishing attacks have become a significant concern in cybersecurity due to their increasing sophistication and prevalence. One cannot stress the value of cybersecurity in the digital age particularly when it comes to email, but it also poses a lot of vulnerabilities. Sensitive information is frequently sent by email, hence protecting it against breaches is necessary to maintain integrity and privacy. The objective of this study is to evaluate the performance of multiple machine learning models on an unbalanced dataset of 39,000 samples on Jupyter Notebook, employing preprocessing techniques to enhance accuracy. The study draws comparisons with prior research, such as the phishing detection model by Fares et al., to identify patterns in achieving high classification accuracy across various domains, including phishing email detection. Studies indicate that preprocessing methods, including Feature Engineering, Vectorization and class balancing techniques like SMOTE, significantly enhance model accuracy. Ensemble methods, such as Random Forest and XGBoost, have gained prominence due to their ability to capture complex relationships within data. The study by [1] on phishing detection underscores the value of ensemble learning combined with feature selection in achieving high accuracy, which serves as an inspiration for this research.

Phishing techniques are constantly evolving which make it serious threat [2]. In [3], researchers proposed a phishing email detection method leveraging deep semantic analysis and machine learning algorithms. They employed algorithms like NB, SVM, DT, LSTM, KNN, and Embedding on a dataset containing labeled emails. The study achieved accuracy rates ranging from 75.72% to 95.97%, highlighting the importance of in-depth analysis and machine learning in detecting phishing emails. Detecting phishing emails requires robust classification techniques that can handle high-dimensional and often imbalanced datasets.

In [4], researchers tackled the problem of detecting phishing and spam emails using deep learning and natural language processing techniques. They utilized algorithms like LSTM and MLP on a dataset containing labeled messages. The study achieved accuracy rates of 99% for LSTM and 94% for MLP, showcasing the power of deep learning in enhancing email security. Researchers in [5] addressed the challenge of data imbalance between phishing and benign emails. They proposed algorithms like DT, RF, GND, MLP, KNN, SEL, SVEL, FMPED, and FMMPED to achieve accurate detection. The study achieved impressive accuracy rates ranging from 90.25% to 99.45%, highlighting the importance of addressing data imbalance for effective detection.

Researchers in [6] propose a solution named RAIDER (Reinforcement Aided Spear Phishing Detector) to address the challenges posed by spear phishing. These challenges include the difficulty of detection, susceptibility of machine learning to zero-day attacks, issues with email address spoofing, and scalability concerns. They conducted their study using a dataset comprising over 11,000 emails from three different attack scenarios. The proposed algorithm, RAIDER, is a reinforcement learning-based feature evaluation system that autonomously selects significant features to detect various spear phishing attacks. Notably, RAIDER achieves an enhancement in spoofing attack detection accuracy from 90% to 94%, and in Known Sender attack detection accuracy from 49% to 62%. Its strengths lie in the autonomous feature selection process and a remarkable 55% reduction in the required features' dimensions. However, its weaknesses include reliance on historical data access and potential limitations in identifying sophisticated attacks.

The importance of accurate classification extends beyond academic research, impacting areas such as fraud detection, patient diagnosis, and secure digital communication. In cybersecurity, detecting phishing emails is critical to prevent data breaches and financial loss. By systematically comparing models and techniques, this study bridges the gap between theoretical advancements and practical implementation, providing a replicable framework for future research.

Rest of the study is organized as: In section 2 review the recent related works, while section 3 provides the methodology detailed of our approach. Followed by the experiment result, the discussion, and the limitations of our study in section 4. Finally, section 5 concludes our contribution.

## 2. Related Work

Studies highlight those preprocessing steps, such as feature selection and class balancing, are critical for improving model performance. The paper [1] emphasizes the use of feature extraction and SMOTE for phishing detection, achieving an SVM accuracy of 97.61%. These findings align with the current study's approach of employing similar preprocessing techniques to enhance classification across various domains, including phishing detection and fraud analysis. Ensemble models, including Random Forest and XGBoost, are renowned for their ability to handle high-dimensional and noisy data. In the context of phishing detection, the base paper compares SVM, Random Forest, and XGBoost, with XGBoost outperforming other models, achieving an accuracy of 96.61%. This study extends this analysis by evaluating additional models, such as AdaBoost and Logistic Regression, to explore their applicability in fields like cybersecurity and e-commerce.

In [7], authors super covenant analyzed the threats of phishing and spam emails with the help of deep learning and natural language processing tools. By training LSTM and MLP using a dataset with labeled messages they deployed it. This research reached an accuracy of 99% for LSTM and 94% for MLP, which demonstrates possibilities for improving repeated email security using deep learning.

Researchers in [8] proposed algorithms to solve the problem of distinctive quantities of phishing and all normal messages. The analyzed data used methods such as DT, RF, MLP, KNN, SEL, SVEL, FMPED, and FMMPED. It has been reported to work with accuracies ranging from 88.50% to 99.45%. They expressed the need to address issues of data imbalance that this study has laid emphasis on in the fight against phishing.

Another work [9] that uses GCN in conjunction with NLP algorithms is the work that is specializing in the detection of phishing in the body texts of emails. The model has tested satisfactory, yielding to an accuracy of 98% by using a self-generated email body text dataset. The novelty of the proposed approach is because GCN is used for text classification for the first time, and it is applied to an important and practical problem of phishing emails detection. Nevertheless, the main drawback of the study is that it was undertaken in an English setting, and the results might not be applicable to other languages or work-related emails.

Work [10] helps to use the Multi-Layer Perceptron (MLP) model with the participation of Spam Base, Spam Assassin, and, finally, the UK.2011 Web spam datasets with the given accuracies 96.9%, 98.1%, and 95.6%, respectively. This work is merit worthy because of the several dataset and feature sets used in the research, thus the assessment of spam detection is more holistic. However, the main drawback of the study is the fact that it has focused squarely in spam detection and not thoroughly on the special features of phishing, which may be a problematic area regarding the efficiency against phishing spam emails.

Shaukat Muhammed Waqas et al. [11] have proposed a solution for phishing websites classification. They have made use of a sizable collection of website URLs to achieve their goal. A variety of learning models have been employed, such as the multilayer perceptron, random forest, optimizer gradient-boosting decision trees (XGBoost), and support vector machine (SVM). The XGBoost algorithm beat other applicable models, according to the performance evaluation, with a maximum accuracy and precision of 94.

Phishing attacks have progressed and enhanced their techniques as the technology progress and research does not stop at developing or improving its techniques in cybersecurity to detect and classify phishing attacks, by developing several techniques. Many researches have demonstrated that approaches used artificial intelligence (AI) specifically, machine learning with the goal to strengthen system security and prevent any network intrusion [12]. In recent years, most studies have concentrated on applying machine learning and deep learning methods in cybersecurity and developing new feature selection techniques. As a result, several machine learning techniques have proven to be effective in accurately identifying phishing attacks such as: Decision Tree, Random Forest, SVM, Naïve Bayes, Neural Network and so on [13].

**Table 1: Research Studies on Phishing Email Detection**

| Reference | Title | Year | Methodology | Dataset | Accuracy |
|---|---|---|---|---|---|
| [14] | Neural Networks by Enhance Grasshopper Optimization Algorithm for Spam Detection System, | 2021 | Neural Networks | Enron, Spam Assassin | 96.9%, 98.1%, and 95.6% |
| [15] | phishing email detection using natural language processing techniques | 2022 | NLP, Deep Learning | Nazario phishing corpus | |
| [16] | A Study of Large Language Models for Phishing Detection | 2024 | Rule-based + ML (SVM, Decision Trees) | Custom Dataset | 97.46% |
| [17] | Classifying phishing email using machine learning and deep learning. | 2019 | NB, SVM, and DT | Private Dataset | 98.89% |
| [18] | Phishing detection tool for financial emails. | 2024 | SVC, Random Forest | URL dataset | |
| [19] | An ensemble model for detecting phishing attack | 2018 | Feature selection technique (RRFST) | Private Datasets | 99.27% |
| [20] | Feature selection for email phishing detection using machine learning | 2022 | decision trees (J48), random forest, and logistic regression | Spam Assassin corpus | 95.6% and 99.4% |
| [21] | Phishing Emails Detection in Cyber Security | 2024 | Distil BERT-based model | Private Dataset | 95% |

## 3. Methodology

### 3.1 Dataset Description

The dataset used in this study comprises 39,000 samples, each containing multiple features relevant to phishing email classification. It is a well-preprocessed dataset that, like the one used by Fares et al., requires handling of class imbalance and enhancement of data quality. The features were extracted using domain-specific analysis, ensuring their relevance to phishing detection. Though not originally designed exclusively for phishing detection, the preprocessing pipeline and classification approach applied to this dataset make it highly applicable for cybersecurity tasks, particularly phishing email detection.

### 3.2 Preprocessing Steps

#### 3.2.1 Data Cleaning:

Missing values were handled by imputing the most frequent value or removing instances with significant gaps in essential features. Duplicate records were removed to ensure data consistency and reduce bias. Outliers were detected using statistical methods and either capped or removed to minimize their effect on model training.

#### 3.2.2 Feature Extraction:

Relevant features were identified through domain knowledge and statistical analysis. This helped reduce dimensionality and increase the interpretability of the data. For text-based features (e.g., email content), TF-IDF and Bag-of-Words techniques were applied to extract meaningful patterns and convert the raw text into numerical representations.

### 3.2.3 SMOTE (Synthetic Minority Over-sampling Technique):

SMOTE was applied to balance the dataset by generating synthetic samples for the underrepresented class (phishing emails). This technique helps mitigate class imbalance, which is common in cybersecurity datasets, and improves the overall model performance.

### 3.2.4 Label Encoding:

Categorical features were converted into numerical labels using label encoding, ensuring compatibility with machine learning algorithms.

## 3.3 Model Selection

For phishing email detection, several machine learning models were selected based on various factors such as model diversity, ability to handle high-dimensional data, interpretability, and computational efficiency. The following models were chosen:

### 3.3.1 Random Forest (RF)

Random Forest is an ensemble method that builds multiple decision trees and merges their results to improve accuracy and prevent overfitting. Each tree is trained on a random subset of the features and data points, making it a robust and scalable model for classification tasks. Random Forest was selected due to its proven effectiveness in handling high-dimensional data and its ability to balance precision and recall, which is crucial for phishing detection.

### 3.3.2 Gradient Boosting (GB)

Gradient Boosting is a boosting technique that builds a series of models sequentially, where each model tries to correct the errors of the previous one. This method focuses on reducing bias and variance, which often leads to strong predictive performance. Gradient Boosting was included because of its ability to handle complex datasets and improve weak learners iteratively. This makes it a good candidate for detecting phishing emails, which often exhibit subtle and evolving patterns.

### 3.3.3 Logistic Regression (LR)

Logistic Regression is a simple, yet effective linear model used for binary classification tasks. It estimates the probability that a given instance belongs to a particular class (e.g., phishing, or non-phishing) based on the input features. Logistic Regression was selected for its interpretability and computational efficiency. It provides a clear decision boundary and works well with preprocessed features like those used in this study.

### 3.3.4 K-Nearest Neighbors (KNN)

K-Nearest Neighbors classifies new instances based on the majority class of their nearest neighbors in the feature space. The model uses a distance metric to identify similar instances. Although KNN can struggle with high-dimensional datasets, its simplicity and interpretability made it a useful model to include in this study for comparison purposes. It allowed for the exploration of proximity-based learning in phishing detection.

### 3.3.5 Naive Bayes (NB)

Naive Bayes is a probabilistic classifier based on Bayes' Theorem, which assumes that the features are independent. Despite this simplifying assumption, it can perform well in many classification tasks, including text classification. Naive Bayes was selected for its simplicity and computational efficiency. It is particularly well-suited for text classification tasks and was expected to work well with the email content features in the dataset.

### 3.3.6 Decision Tree (DT)

A Decision Tree splits the data into subsets based on feature values, recursively creating decision rules. It is easy to interpret and visualize, making it a popular choice for classification tasks. Decision Trees are interpretable, simple to implement, and provide clear decision boundaries. While it may not always be as accurate as ensemble models like Random Forest, it serves as a useful benchmark for comparison.

### 3.3.7 AdaBoost

AdaBoost works by iteratively adjusting the weights of incorrectly classified samples, allowing subsequent weak learners to focus on hard-to-classify instances. It combines multiple weak classifiers to form a strong classifier. AdaBoost was included for its ability to improve the performance of weak classifiers through boosting. It is computationally efficient and effective at handling noisy and imbalanced datasets.

### 3.3.8 XGBoost

XGBoost is an optimized version of Gradient Boosting that focuses on computational efficiency and scalability. It includes regularization techniques to prevent overfitting and improve model performance. XGBoost was selected due to its proven track record in winning machine learning competitions and its ability to handle imbalanced datasets and high-dimensional features effectively.

## 3.4 Evaluation Metrics

The performance of the machine learning models was evaluated using several standard metrics, which are defined in terms of the following values: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These metrics are as follows:

### 3.4.1 Accuracy:

Accuracy measures the overall correctness of the model. It is the ratio of correctly classified instances (both true positives and true negatives) to the total number of instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \qquad (1)$$

### 3.4.2 Recall:

Recall measures the ability of the model to correctly identify positive instances (phishing emails). It is the ratio of true positives to the total actual positives (true positives + false negatives).

$$\text{Recall} = \frac{TP}{TP + FN} \qquad (2)$$

### 3.4.3 Precision:

Precision measures the accuracy of positive predictions. It is the ratio of true positives to the total predicted positives (true positives + false positives).

$$\text{Precision} = \frac{TP}{TP + FP} \qquad (3)$$

### 3.4.4 F1-Score:

The F1-score is the harmonic mean of precision and recall, providing a balance between the two. It is particularly useful when the class distribution is imbalanced.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (4)$$

These evaluation metrics allow for a comprehensive assessment of each model's performance, considering both the ability to correctly identify phishing emails and minimize false positives and false negatives.

# 4. Results

## 4.1 Model Performance Evaluation

The performance of six machine learning models was evaluated on a balanced dataset of phishing emails. The accuracy, precision, recall, and F1-scores of each model were computed and analyzed to assess their effectiveness.

### 4.1.1 Random Forest

Random Forest model performs exceptionally well with 99.00% accuracy, meaning it correctly classifies almost all instances. Precision (0.99) indicates that 99% of predicted positives are correct, while Recall (0.99) shows that 99% of actual positives are identified. The F1-score (0.99) confirms a strong balance between precision and recall, minimizing both false positives and false negatives.

### 4.1.2 Gradient Boosting

Gradient Boosting model is 97.64% accurate, meaning it gets most predictions right. It's slightly better at detecting class 1 (99% recall) but makes a few more mistakes predicting class 0 (96% recall). With a high F1-score (0.99), it balances precision and recall well, though it's a bit less accurate than Random Forest. Gradient Boosting displayed strong results, with slightly lower precision for phishing emails (class 1) compared to Random Forest.

### 4.1.3 Logistic Regression

Logistic Regression model is 98.46% accurate, meaning it correctly classifies most instances. It maintains high precision (0.98 for both classes), meaning few false positives, while recall is slightly higher for class 1 (0.99) than class 0 (0.98). The F1-score (0.98 for class 0, 0.99 for class 1) shows a strong balance between precision and recall, making it a reliable model. Logistic Regression, a simpler linear model, provided competitive results.

### K-Nearest Neighbors (KNN)

K-Nearest Neighbors (k-NN) model has an accuracy of 91.50%, meaning it makes correct predictions most of the time but is less accurate than other models. Precision (0.91 for class 0, 0.92 for class 1) shows it correctly identifies positives with some false positives. Recall (0.89 for class 0, 0.93 for class 1) indicates it's slightly better at catching class 1 cases. The F1-score (0.90 for class 0, 0.92 for class 1) suggests a decent balance but shows the model could be improved. KNN exhibited lower accuracy and F1-scores compared to ensemble methods. While its simplicity makes it easy to implement, its reliance on distance metrics may limit its performance on high-dimensional datasets.

### 4.1.5 Naive Bayes

Naïve Bayes model is 93.45% accurate, meaning it gets most predictions right. It's slightly better at identifying class 1 (96% precision) but sometimes misses a few actual class 1 cases (92% recall). With F1-scores of 0.93 (class 0) and 0.94 (class 1), it maintains a solid balance, though it's not as strong as more complex models. Naive Bayes performed relatively well given its assumption of feature independence. Its higher precision for phishing emails (class 1) indicates effective identification, but a slightly lower recall shows room for improvement in minimizing false negatives.

### 4.1.6 Decision Tree

Decision Tree model is 97.76% accurate, meaning it makes very few mistakes. It has high precision (0.98 for both classes), so most positive predictions are correct. With recall of 0.97 (class 0) and 0.98 (class 1), it slightly favors class 1. The F1-scores (0.97 for class 0, 0.98 for class 1) show a great balance, making it a strong and reliable model. Decision Tree achieved high accuracy and balanced precision-recall values. Its interpretability and ease of implementation make it suitable for phishing detection, though it slightly underperformed compared to Random Forest and XGBoost.

### 4.1.7 AdaBoost

AdaBoost model has an accuracy of 97.34%, meaning it makes correct predictions most of the time. Precision (0.97 for class 0, 0.98 for class 1) shows it slightly favors class 1, while recall (0.97 for both classes) indicates it catches most actual positives. With F1-scores of 0.97 for both classes, it maintains a great balance between precision and recall, making it a strong but slightly less accurate model than Decision Trees. AdaBoost demonstrated robust performance with consistent metrics. Its emphasis on improving weak classifiers in each iteration contributed to its reliability, although it slightly lagged behind XGBoost.

### 4.1.8 XGBoost

XGBoost model is 99.00% accurate, meaning it almost always makes correct predictions. With precision and recall both at 0.99 for both classes, it rarely misclassifies positives or misses actual cases. The F1-score of 0.99 confirms an excellent balance, making it one of the most reliable models in your comparison. XGBoost emerged as the top-performing model, achieving the highest accuracy and F1-scores. Its scalability and ability to handle imbalanced datasets make it ideal for phishing detection tasks. Comparisons are shown in table 2 and figure 1,2 and 3.

**Table 2: Comparison of Models Performance**

| Model | Existing Accuracy | Proposed Accuracy |
|---|---|---|
| **Random Forest** | 95.04% | 99.00% |
| **Gradient Boosting** | N/A | 97.64% |
| **Logistic Regression** | N/A | 98.46% |
| **K-Nearest Neighbors** | N/A | 91.50% |
| **Naive Bayes** | N/A | 93.45% |
| **Decision Tree** | 95.03% | 97.76% |
| **AdaBoost** | N/A | 97.34% |
| **XGBoost** | 96.61% | 99.03% |

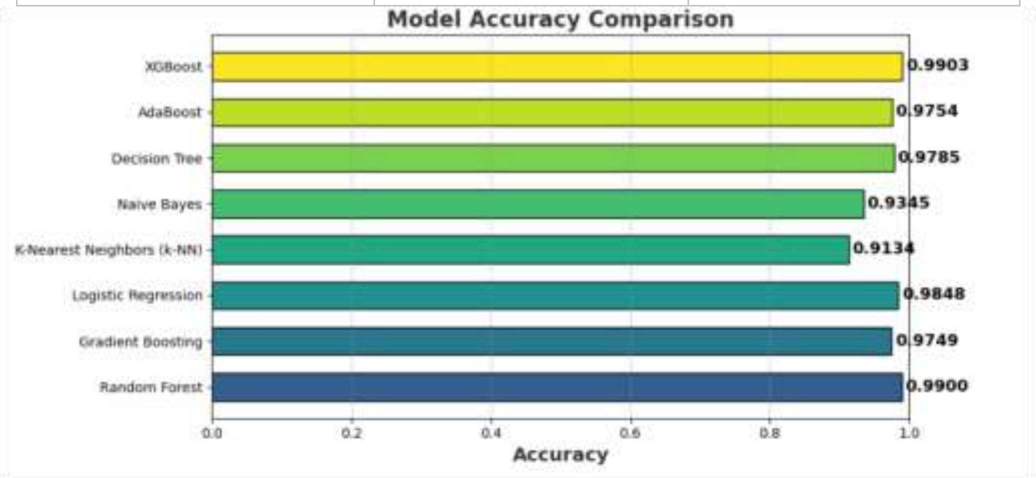

**Figure 1: Modal Accuracy Comparison**

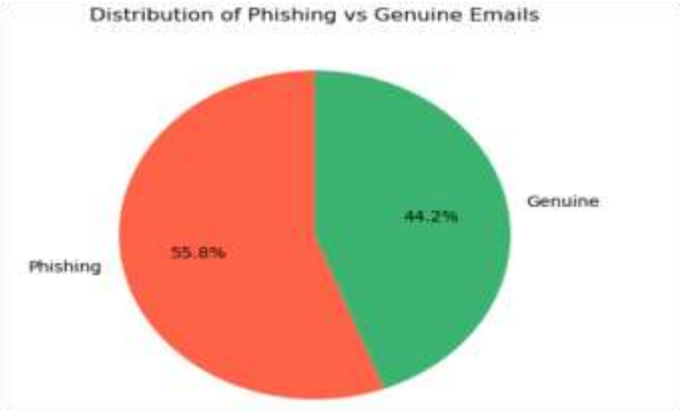

**Figure 2: Pie Chart Label Distribution**

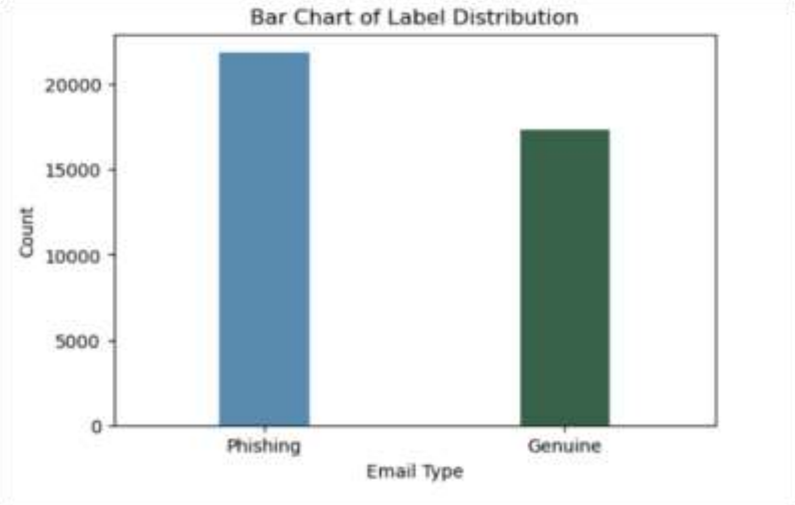

**Figure 3: Bar Chart Label Distribution**

XGBoost outperformed all models in this study, achieving the highest accuracy of 99.004%, consistent with its strong performance in phishing detection studies. Random Forest and Decision Tree demonstrated strong performance, making them reliable options for phishing detection and fraud analysis. Simpler models, such as Logistic Regression and Naive Bayes, achieved lower accuracies but remain useful for interpretable and computationally efficient solutions in cybersecurity tasks.

## Conclusion

This study confirms the effectiveness of ensemble models, particularly XGBoost and Random Forest, in achieving superior classification accuracy. By incorporating preprocessing techniques like SMOTE and feature extraction, the study achieved results that align with and extend findings from prior research in phishing detection and other fields. These results underscore the generalizability of preprocessing and ensemble methods across domains. Evaluate additional metrics, such as F1-score and recall, to provide a holistic understanding of model performance in phishing detection. Validate findings across diverse datasets to ensure robustness and applicability to other domains, such as fraud detection and anomaly analysis. Investigate the integration of deep learning techniques for tasks requiring high-dimensional feature extraction, such as phishing email detection. Explore explainable AI tools to enhance model transparency in cybersecurity applications.

## References

Fares, H., Kilani, J., Fagroud, F., Toumi, H., Lakrami, F., Baddi, Y. and Aknin, N., 2024. Machine learning approach for email phishing detection. Procedia Computer Science, 251, pp.746-751.

Karim, S. and Affandi, D., 2025. Robust Analysis of Hypothyroidism Detection Using Ensemble Modeling Techniques. Spectrum of engineering sciences, 3(2).

Bagui, S., Nandi, D., Bagui, S. and White, R.J., 2021. Machine learning and deep learning for phishing email classification using one-hot encoding. J. Comput. Sci, 17(7), pp.610-623.

Latif, S. and Mussarrat, N., 2024. Impact of Digital Technologies on Consumer E-Purchasing Decisions Using Ensemble Techniques. International Journal of Information Systems and Computer Technologies, 4(1), pp.10-19.

Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A. and Elsoud, E.A., 2022. An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), pp.3819-3828.

Latif, S., Fang, X.W., Arshid, K., Almuhaimeed, A., Imran, A. and Alghamdi, M., 2023. Analysis of birth data using ensemble modeling techniques. Applied Artificial Intelligence, 37(1), p.2158273.

Dewis, M. and Viana, T., 2022. Phish responder: A hybrid machine learning approach to detect phishing and spam emails. Applied System Innovation, 5(4), p.73.

Qi, Q., Wang, Z., Xu, Y., Fang, Y. and Wang, C., 2023. Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. Applied Sciences, 13(15), p.8756.

Alhogail, A. and Alsabih, A., 2021. Applying machine learning and natural language processing to detect phishing email. Computers & Security, 110, p.102414.

Ghaleb, S.A., Mohamad, M., Fadzli, S.A. and Ghanem, W.A.H., 2021. Training neural networks by enhance grasshopper optimization algorithm for spam detection system. IEEE Access, 9, pp.116768-116813.

Shaukat, M.W., Amin, R., Muslam, M.M.A., Alshehri, A.H. and Xie, J., 2023. A hybrid approach for alluring ads phishing attack detection using machine learning. Sensors, 23(19), p.8070.

Fares, H., Kilani, J., Fagroud, F., Toumi, H., Lakrami, F., Baddi, Y. and Aknin, N., 2024. Machine learning approach for email phishing detection. Procedia Computer Science, 251, pp.746-751.

Latif, S., 2024. Robust Decision Support System for Stress Prediction Using Ensemble Techniques. Journal of Innovative Computing and Emerging Technologies, 4(2).

Ghaleb, S.A., Mohamad, M., Fadzli, S.A. and Ghanem, W.A.H., 2021. Training neural networks by enhance grasshopper optimization algorithm for spam detection system. IEEE Access, 9, pp.116768-116813.

Salloum, S., Gaber, T., Vadera, S. and Shaalan, K., 2022. A systematic literature review on phishing email detection using natural language processing techniques. IEEE Access, 10, pp.65703-65727.

Chataut, R., Gyawali, P.K. and Usman, Y., 2024, January. Can ai keep you safe? a study of large language models for phishing detection. In 2024 IEEE 14th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0548-0554). IEEE.

Bagui, S., Nandi, D., Bagui, S. and White, R.J., 2019, June. Classifying phishing email using machine learning and deep learning. In 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security) (pp. 1-2). IEEE.

Latif, S., XianWen, F. and Wang, L.L., 2021. Intelligent decision support system approach for predicting the performance of students based on three-level machine learning technique. Journal of Intelligent Systems, 30(1), pp.739-749.

Hota, H.S., Shrivas, A.K. and Hota, R., 2018. An ensemble model for detecting phishing attack with proposed remove-replace feature selection technique. Procedia computer science, 132, pp.900-907.

Yadav, N. and Panda, S.P., 2022. Feature selection for email phishing detection using machine learning. In International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021, Volume 2 (pp. 365-378). Springer Singapore.

Sharma, S., Sharma, R. and Sharma, M., 2024. Phishing Emails Detection in Cyber Security.