BUILDING AN INTELLIGENT DEFENSE: MACHINE LEARNING-DRIVEN PHISHING DETECTION IN A WEB-BASED SOLUTION

By
Md. Anowar Hossain
ID: 203-15-3870

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements forthe **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by

Amir Sohel
Lecturer (Sr. Scale)
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by

Afjal Hossan Sarower
Lecturer (Sr. Scale)
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

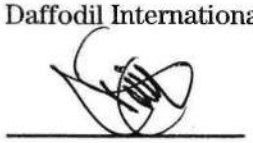
January 13, 2025

APPROVAL

This Project titled "**Building an Intelligent Defense: Machine Learning-Driven Phishing Detection in a Web-Based Solution**," submitted by **Md. Anowar Hossain** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 13 January, 2025.

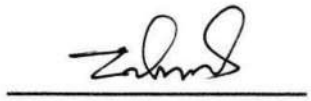
BOARD OF EXAMINERS

Dr. S.M. Aminul Haque — **Chairman**
Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Md. Abbas Ali Khan — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Aynul Hasan Nahid — **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

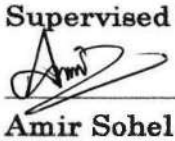


Dr. Md. Zulfiker Mahmud — **External Examiner**
Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

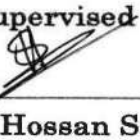
I hereby declare that this project has been done by me under the supervision of **Amir Sohel, Lecturer (Senior Scale)**, Department of Computer Science and Engineering, Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Amir Sohel
Lecturer (Senior Scale)
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:



Afjal Hossan Sarower
Lecturer (Senior Scale)
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Md. Anowar Hossain
Student ID: 203-15-3870
Department of Computer Science and Engineering
Daffodil International University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. I am deeply grateful to everyone who has assisted me in one way or another.

First, I express my heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for me to complete the **Final Year Design Project (FYDP)** successfully.

I am grateful and wish my profound indebtedness to **Amir Sohel**, **Lecturer (Senior Scale)**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of my supervisor in the field of Cybersecurity to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

I would like to express my heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing my project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

I would like to thank my entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

Phishing attacks remain a critical threat to online security, necessitating the need for effective detection methods as traditional methods are not much effective in current challenging world. Where every technology is change so fast. In this research, we aim to develop an advanced phishing detection system leveraging machine learning (ML) techniques. The dataset utilized in this study comes from various sources, including the OpenPhish dataset for phishing URLs, Majestic Million's 1 million websites for legitimate URLs. To ensure dataset balance, equal proportions of short and long URLs are included. Our approach focuses on minimizing feature redundancy to increase detection accuracy and reduce computational complexity. We select a critical subset of features that are most relevant for phishing detection, optimizing both performance and dataset size. Furthermore, we apply various classifiers, including Logistic Regression, K-Nearest Neighbors Classifier, Gradient Boosting Classifier, AdaBoost Classifier and Hybrid Machine Learning Model, to identify the most effective algorithm for detecting phishing websites with a focus on the features of classification to facilitate immediate detection. This study's core involves assessing the efficacy of various ML algorithms and feature sets. We measure each classifier's accuracy, precision, recall, and F1 score to determine the optimal combination for phishing detection. Additionally, we prioritize predicting speed to ensure real-time detection capabilities. The proposed system aims to address existing limitations in phishing detection, such as low latency and lack of comparative analysis between algorithms. By leveraging a diverse dataset and optimizing feature selection, we aim to develop a robust and efficient phishing detection model capable of accurately identifying malicious URLs while minimizing false positives.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

This chapter provides an overview of the research project on phishing detection using machine learning algorithms. It outlines the motivation behind the study, the research objectives, Methodology, Project outcome and Organization of the Report.

1.1 Introduction

The development of digitalization has completely changed the way we go about our everyday lives—from shopping to banking to socializing. But this quick adoption of technology into our lives has also given rise to significant challenges, particularly in the realm of cybersecurity. One of the most pervasive threats in the digital landscape is phishing, a type of cyberattack designed to steal personal data like login passwords and financial information, is one of the most prominent threats in the digital world. Phishing attacks have become increasingly sophisticated over time, with attackers employing various tactics to deceive unsuspecting users. In 2023, a documented record-breaking nearly five million phishing attacks, with 1,077,501 incidents in Q4 alone, marking the year as the worst on record for phishing [1]. Despite the efforts of cybersecurity professionals and organizations, the prevalence of phishing attacks continues to rise, posing a significant risk to individuals and businesses alike.

To combat this growing threat, researchers and practitioners have been exploring various approaches to detect and prevent phishing attacks effectively. Traditional methods such as blacklisting, which involves maintaining a list of known malicious websites, have proven to be insufficient in addressing the evolving nature of phishing attacks. In response, machine learning has emerged as a promising solution for detecting phishing websites.

Machine learning techniques leverage algorithms that can learn from data and make predictions or decisions based on that data. By training machine learning models on datasets containing both phishing and legitimate websites, researchers can identify patterns and characteristics that distinguish between the two. This enables the development of more accurate and adaptive phishing detection systems that can effectively mitigate the risk posed by phishing attacks.

In this context, the present research aims to contribute to the ongoing efforts to combat phishing by exploring the effectiveness of machine learning algorithms in detecting phishing websites. The study utilizes a newly curated dataset comprising both phishing and legitimate websites, along with a comprehensive set of features extracted from URLs, HTML, and HTTP attributes.

By evaluating the performance of various machine learning algorithms on this dataset, the research seeks to identify the most effective approaches for phishing detection. Additionally, the study investigates the impact of different features and classification algorithms on the detection performance, providing valuable insights for the development of robust anti-phishing solutions.
Overall, this research represents a significant step towards enhancing cybersecurity measures and protecting users from the pervasive threat of phishing attacks in an increasingly digitized world.

1.2 Motivation

The motivation behind this research derives from the persistent and evolving threat of phishing attacks, which continue to risk the security of individuals and organizations globally. Traditional methods like blacklisting are increasingly ineffective against the dynamic tactics employed by attackers, necessitating more adaptive and efficient solutions. Leveraging the potential of machine learning offers a promising avenue to enhance phishing detection by harnessing data-driven algorithms that detect patterns of malicious behavior. Moreover, existing works are not fully described and well organized. Besides, using Machine Learning technique with newer methods will help to enrich this type of research work.

1.3 Objectives

The objectives of this research are twofold: firstly, to develop robust machine learning-based phishing detection systems capable of accurately differentiating between phishing and legitimate websites by selecting optimal features and classification algorithms. Secondly, to address the challenges associated with the scarcity of diverse datasets suitable for training and evaluating these systems, ensuring their effectiveness across various phishing tactics, target industries, and geographical regions. Through interdisciplinary research efforts, the aim is to enhance the adaptability, scalability, interpretability, and computational efficiency of machine learning-based phishing detection systems, ultimately mitigating the risks posed by phishing attacks in an increasingly interconnected world.

1.4 Methodology

This research utilized a systematic approach to detect phishing websites using machine learning algorithms. The process began with collecting datasets from two reliable sources: OpenPhish for phishing URLs and Majestic Million for legitimate ones. The collected data were then preprocessed to ensure consistency, and key features relevant to phishing detection, such as URL, HTML, and HTTP attributes, were extracted using Python scripts. These features were analyzed and prepared for model training. Various machine learning algorithms including hybrid machine learning models were applied and their performance was evaluated to classify

websites accurately as phishing or legitimate. The methodology ensured the use of authentic data and effective techniques to enhance the reliability and accuracy of the detection process.

1.5 Project Outcome

The outcome of this research project will be the development of a machine learning-based phishing detection system trained on a balanced dataset containing phishing and legitimate URLs in an equal ratio. The dataset will include both long and short URLs to ensure comprehensive coverage of phishing tactics. Through rigorous experimentation and optimization, the project aims to achieve high detection accuracy while minimizing false positives and false negatives. The resulting system will be capable of accurately identifying phishing websites in real-time, thereby enhancing cybersecurity measures for individuals and organizations. Additionally, the project will contribute to the advancement of knowledge in the field of cybersecurity and machine learning, paving the way for future research and innovation in phishing detection technology.

1.6 Organization of the Report

1. Chapter 2: Background
This chapter reviews the background of phishing detection, highlights related research, identifies gaps, and establishes the study's context.

2. Chapter 3: Research Methodology
This chapter outlines the data collection process, feature extraction methods, and machine learning techniques used in the research.

3. Chapter 4: Implementation and Results
This chapter discusses the development, training, and testing of models, presenting results and their analysis.

4. Chapter 5: Engineering Standards and Design Challenges
This chapter focuses on the standards followed and challenges addressed during the project.

5. Chapter 6: Conclusion
This chapter summarizes key findings, discusses limitations, and provides recommendations for future research.

Chapter 2
Background

This chapter provides an overview of the foundational elements necessary to understand the research on phishing detection using machine learning algorithms. It outlines the Introduction, Literature review, key research objectives, Related research, Gap analysis and the overall summary of this chapter.

2.1 Introduction

This section provides the necessary background information to understand the scope and context of the study. Phishing attacks have become a growing concern in the digital age, targeting individuals and organizations to exploit sensitive data such as personal credentials, banking information, and confidential files. The constant evolution of phishing techniques has made traditional detection methods, such as blacklisting and manual analysis, increasingly ineffective. In response to this challenge, machine learning has emerged as a promising approach, offering the ability to analyze large datasets, identify subtle patterns, and adapt to new attack strategies in real-time. By leveraging the strengths of machine learning, researchers aim to develop more accurate and efficient systems to detect and prevent phishing activities, ultimately enhancing online security and protecting users from cyber threats.

2.2 Literature Review

This section reviews existing studies and methodologies related to phishing detection and highlights their contributions. It includes a summary of previous work in the field, as shown in a table format for clarity. The table outlines key details such as the authors, publication year, Title, methodology, and findings, providing insight into prior research and setting the stage for the current work.

Table 2.1: Summary of Literature Reviewed.

Author(s)	Year	Title	Methodology	Key Findings
Prof. A.R. Ghongade [2]	2024	Guardian Shield Advanced Phishing Detection using Machine Learning	ML (Random Forest, Logistic Regression, Gradient Boosting)	Achieved 97% accuracy with Gradient Boosting. Limited comparison with other algorithms.

Sibel Kapan [3]	2023	Improved Phishing Attack Detection with Machine Learning: A Comprehensive Evaluation of Classifiers and Features	ML (Decision Tree with URL + HTTP features)	Achieved 99% accuracy with Decision Tree and URL + HTTP features. Limited scalability analysis.
Jimoh R.G [4]	2023	Efficient Ensemble-based Phishing Website Classification Models using Feature Importance Attribute Selection and Hyper parameter Tuning Approaches	ML (Random Forest and Extra Trees ensembles)	Achieved 99.3% accuracy with Random Forest. Limited dataset generalizability.
Jayaraj R. [5]	2023	Intrusion detection based on phishing detection with machine learning	ML (Hybrid Ensemble Feature Selection (HEFS))	Achieved 97.8% accuracy. Lack of details on dataset and classifiers.
Grace Odette Boussi [6]	2023	A Machine Learning Model for Predicting Phishing Websites	ML (Random Forest algorithm)	Achieved 98% accuracy with Random Forest. Focused only on one algorithm.
L.H. Abed [7]	2023	Phishing Identification Through up-to-date Features Generation and Exploration	ML (AdaBoost, Bagged Trees, Naive Bayes)	Achieved 88.3% accuracy. Offline classification with limited real-time applicability.
Marwa A. Qasim [8]	2023	Enhancing Phishing Website Detection via Feature Selection in URL-Based Analysis	ML (Decision Tree, SVM, Random Forest)	Achieved 97.6% accuracy with Random Forest. Could benefit from content analysis.
Dina Jibat [9]	2023	A systematic review: Detecting phishing websites using data mining models	Rule Extraction and Integration (REI)	Achieved 99.95% accuracy. Focused on specific publisher sources, excluding other relevant studies.
Most Nilufa Yeasmin [10]	2023	EnLem: An Ensemble Learning-based Mo del for Detecting Phishing Websites	Ensemble Learning combining Decision Tree, Random Forest, and KNN	Achieved 97.21% accuracy. Limited testing beyond UCI dataset.

Megha Agarwa [11]	2023	Phishing Website Detection Using Machine Learning	ML (XGBoost algorithm)	Achieved 99% accuracy. Potential risk of high false positive rate.
H. Peda Sydulu [12]	2023	Machine Learning-Based Detection of Phishing URLs: A Comprehensive Analysis of Features for Reliable Cybersecurity	ML (K-Nearest Neighbors, Logistic Regression)	KNN and Logistic Regression showed comparable accuracy. Further investigation required.
Mrs. S. Gayathri [13]	2023	Detection Of Phishing Attack Using Gan With Rfc	GANs with Random Forest Classifier	GANs with Random Forest achieved high accuracy. Lack of URL understanding noted.
Lindah Sawe [14]	2023	Sentence Level Analysis Model for Phishing Detection Using KNN	Sentence-Level Analysis with KNN	Sentence-level KNN analysis achieved 97% accuracy. Highlights advanced exploration needs.
Anjaneya Awasthi [15]	2022	Phishing website prediction using base and ensemble classifier techniques with cross-validation	Multiple Classifiers including Random Forest, SVM, XGBoost	Extra Trees achieved 99.18% accuracy. Limited scalability discussed.
Pranav Habib [16]	2022	Phishing Detection with Machine Learning	ML (Random Forest)	Achieved 97% accuracy with Random Forest. Suggested future improvements.
Ammar Odeh [17]	2021	Machine LearningTechniquesfor Detection of Website Phishing: A Review forPromises and Challenges	ML (Principal Component Analysis + Random Forest)	PCA-RF achieved 99.55% accuracy. Struggled with heuristic-based attacks.
E. Sri Vishva [18]	2021	Phisher Fighter: Website Phishing Detection System Based on URL and Term Frequency-Inverse Document Frequency Values	Combination of URL and Content Analysis	Combination of URL and Content Analysis achieved 97% accuracy.

Abdul Razaque [19]	2020	Detection of Phishing Websites using Machine Learning	Blacklisting and Semantic Analysis	Combines blacklisting and semantic analysis. Limited feature evaluation.
Sohail Ahmed Khan [20]	2020	Phishing Attacks and Websites Classification Using Machine Learning and Multiple Datasets (A Comparative Analysis)	Combination of Blacklisting and Semantic Analysis	Browser extension proposed with 97% accuracy. Lack of algorithm details noted.
Mahajan Mayuri Vilas [21]	2019	Detection of Phishing Website Using Machine Learning Approach	Extreme Learning Machine (ELM)	ELM proposed with limited comparative analysis. Accuracy not mentioned.
Amani Alswailem [22]	2019	Detecting Phishing Websites Using Machine Learning	ML (Bayesian Net, Decision Tree, Random Forest, SVM)	Multiple classifiers achieved 98.8% accuracy. Lack of bias analysis in training data.
Mustafa KAYTAN [23]	2017	Effective Classification of Phishing Web Pages Based on New Rules by Using Extreme Learning Machines	Extreme Learning Machine (ELM)	ELM achieved 95.05% accuracy. Limited feature rules explained.

2.2.1 Similar Applications

Several research studies, case studies, web applications, and mobile apps have explored phishing detection using advanced techniques, contributing significantly to this field. These applications and methodologies provide valuable insights into how phishing detection can be improved with modern technology. Below is a summary of such contributions:

1. **Research Studies**:

- Numerous studies have demonstrated the use of machine learning algorithms such as Random Forest, SVM, Decision Trees, and Gradient Boosting for phishing detection. For instance, Random Forest has consistently achieved high accuracy by leveraging features like URL structure and HTTP metadata.

- Ensemble learning techniques, such as Hybrid Ensemble Feature Selection (HEFS) and EnLem, combine multiple classifiers for improved detection accuracy and robustness.

- Studies focusing on feature engineering have identified key attributes, including domain age, URL length, and HTTP request patterns, to enhance the precision of phishing detection.

2. Case Studies:

- Practical implementations by industry leaders, such as Google Safe Browsing and Microsoft SmartScreen, integrate machine learning techniques to identify phishing websites and protect users in real-time.

- Custom datasets, including those generated by researchers like Sibel Kapan and Jimoh R. G, showcase how combining legitimate and phishing URLs can aid in building efficient detection models.

3. Web Applications:

- Platforms like VirusTotal and PhishTank offer real-time URL scanning and phishing detection, enabling users to validate website authenticity.

- Browser extensions, such as "Netcraft Anti-Phishing," analyze website URLs in real-time and provide alerts for suspicious activities.

4. Mobile Applications:

- Security-focused mobile apps, like "Avast Mobile Security" and "Kaspersky Internet Security," detect phishing links in SMS, emails, and instant messages. These apps use advanced machine learning techniques to identify potential threats.

- Mobile apps often rely on a combination of blacklisting and semantic analysis to ensure robust phishing detection while maintaining usability.

These applications and studies highlight the growing importance of combining machine learning algorithms, feature engineering, and practical implementations to enhance phishing detection. Building on these foundations, this research aims to explore more advanced methodologies and address existing challenges, such as real-time adaptability and evolving phishing tactics.

2.2.2 Related Research

Phishing detection research has extensively explored machine learning techniques, including Random Forest, SVM, and Gradient Boosting, leveraging features like URLs, HTTP attributes, and page content. Recent studies highlight the potential of combining feature selection and ensemble methods to enhance detection performance. However, challenges like evolving phishing tactics, dataset limitations, and real-time adaptability

remain critical areas for further research and development.

Building on existing research using machine learning for phishing detection, [2] proposes a system achieving 97% accuracy with Gradient Boosting Classifier. However, their URL-based approach might struggle with evolving attacks.

The paper explores the effectiveness of machine learning-based approaches for detecting phishing attacks, comparing them with traditional blacklisting strategies. It introduces a new phishing dataset comprising 500 phishing and 500 legitimate websites, represented by 25 features extracted from URL, HTML, and HTTP attributes. Through exhaustive analysis, the study evaluates various combinations of feature groups using classification algorithms like k-NN, SVM, NB, DT, MLP, and SGD. Results indicate that URL and HTTP-based features, coupled with the decision tree classifier, offer superior performance, achieving an F1-score of 0.99 with efficient classification time. This research contributes valuable insights into feature selection and classifier performance, enhancing cybersecurity measures against phishing attacks [3].

Jimoh R. G utilized the UCI Machine Learning Repository and implemented Random Forest and Extra Trees ensembles, achieving a notable accuracy of 99.3%. However, the evaluation was conducted on only one dataset, potentially limiting the generalizability of results. Nonetheless, the study demonstrated the effectiveness of feature importance and hyperparameter tuning for phishing website detection using ensemble methods [4].

Jayaraj R. introduced a custom dataset of phishing and legitimate websites and proposed the Hybrid Ensemble Feature Selection (HEFS) framework with Cumulative Distribution Function gradient (CDF-g) and data perturbation ensemble. The method achieved an accuracy of 97.8%, although specific details on the classifiers used and the dataset size were lacking. Nevertheless, the study introduced the HEFS framework for feature selection and showcased its effectiveness in phishing website detection [5].

Grace Odette Boussi utilized a Kaggle dataset containing 5,000 phishing and 5,000 legitimate webpages, employing the Random Forest algorithm to achieve 98% accuracy. However, the study only evaluated the Random Forest algorithm and did not provide further details. Nevertheless, the research developed a high-accuracy phishing detection model using Random Forest and feature selection [6].

L.H. Abed utilized PhishTank & Alexa Top Sites data consisting of 12,000 URLs and employed AdaBoost, Bagged Trees, and Naive Bayes classifiers. Achieving an accuracy of 88.3%, the study focused on lexical features only and performed offline/batch classification, lacking real-time capability. Despite these limitations, the research used up-to-date phishing URLs for realistic evaluation and investigated the effectiveness of various URL features [7].

Marwa A. Qasim (2023) generated a new dataset of 8,000 URLs using regular expressions and employed Decision Trees, SVM, and Random Forest classifiers. Achieving an accuracy of 97.6%, the study primarily focused on URL features and could benefit from content analysis. Additionally, the research performed offline detection, lacking real-time capability, but demonstrated the effectiveness of feature selection for improved accuracy [8].

Dina Jibat conducted a systematic review of phishing website detection using data mining techniques, exploring various algorithms including Random Forest, Naive Bayes, Support Vector Machine, and Decision Tree. Achieving an impressive accuracy of 99.95%, the study emphasized the effectiveness of these algorithms but may have excluded relevant studies by focusing on well-known journals from specific publishers [9].

Most Nilufa Yeasmin (2023) proposed EnLem, a novel ensemble learning model combining Decision Tree, Random Forest, and k-Nearest Neighbor with Univariate Feature Selection. Achieving an accuracy of 97.21% on the UCI dataset, the study emphasized the need for further testing on other datasets to assess generalizability. Nonetheless, the research demonstrated the effectiveness of EnLem for phishing website detection [10].

Megha Agarwa utilized the Kaggle dataset "Phishing Legitimate full" and employed XGBoost, Logistic Regression, KNeighborsClassifier, Random Forest, and Decision Tree classifiers. Achieving an impressive accuracy of 99% with XGBoost, the study focused primarily on accuracy, which may lead to a high false positive rate. However, the research demonstrated high accuracy with XGBoost using 48 features [11].

H. Peda Sydulu conducted a comprehensive analysis of features for phishing detection and implemented the K-Nearest Neighbors and Logistic Regression algorithms. While achieving high accuracy, the study lacked specific details on the dataset and algorithm performance. Nonetheless, the research provided valuable insights into feature analysis and algorithm implementation for phishing detection [12].

Mrs. S. Gayathri proposed a novel approach combining Generative Adversarial Networks (GANs) with Random Forest Classifier (RFC) for phishing detection. Although achieving high accuracy, the study lacked understanding of URLs, visual and character-level similarity coverage, and a definitive phishing detection solution. Nonetheless, the research contributed to exploring innovative methodologies for phishing detection [13].

Lindah Sawe's paper explores the rise of phishing emails and limitations of existing detection models. It introduces a new model using sentence-level analysis and KNN classification, which demonstrates promising accuracy. The study highlights the need for continued research into effective phishing detection methods, emphasizing the value of sentence-level analysis for improved security [14].

The researcher analyzes 12 machine learning classifiers (base and ensemble) using all 30 features to identify the best algorithm for predicting phishing websites. Utilizes two distinct datasets and a merged dataset with and without cross-validation [15].

Another paper work achieved the best accuracy with the Random Forest algorithm, reaching a precision of 97% and an AUC score of 1.0. while the authors used K-Nearest Neighbors (KNN), Logistic Regression, Random Forest. [16]

The research by Ammar Odeh, Ismail Keshta, and Eman Abdelfattah offers a comprehensive review of conventional Machine Learning (ML) techniques for phishing website detection. Phishing attacks pose a significant threat to online security, targeting sensitive user information like login credentials and banking details. The study highlights the efficacy of ML methods, emphasizing traditional algorithms such as Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), and Ada Boosting. Additionally, the review notes the emergence of deep learning-based techniques, showcasing superior performance in detecting phishing websites compared to conventional ML approaches [17].

The study by Sri Vishva proposes a system combining URL and content analysis for phishing detection. It utilizes machine learning, achieving 97% accuracy on URL analysis and 76% on content analysis. Combining these analyses with a weighted score, the system reaches a 90.68% True Positive rate on real-world URLs. While promising, limitations like dataset reliance and the need for further development to address evolving threats are acknowledged [18].

The author uses both blacklisting and semantic analysis to detect phishing websites, claiming it outperforms existing solutions but lacking strong evidence to support these claims [19].

The paper by Sohail Ahmed Khan, explores this domain by proposing a browser extension that utilizes a combination of blacklisting and semantic analysis for phishing detection [20].

Another research by Mahajan Mayuri Vilas et al. discusses the challenges of phishing detection and proposes a machine learning-based solution using Extreme Learning Machine (ELM) with 30 main components, while related works such as those by Patil and Dhage et al., Yuan et al., Patil et al., Vazhayil et al., Weedon and Tsaptsinos et al., Sonmez et al., Aburrous et al., Arade et al., Shahriar and Zulkernine, and Ajlouni et al., among others, have explored various approaches including heuristic, hybrid, anti-phishing, SVM, and feature-based methods for phishing detection, highlighting the importance of addressing this cybersecurity threat using diverse computational techniques [21].

The authors propose a system that uses machine learning to detect phishing websites. The system extracts feature from URLs, page content, and page rank, and then uses a random forest classifier to predict whether a website is phishing or legitimate. The authors claim that their system can achieve an accuracy of 98.8% with only 26 features [22].

Another study by Mustafa KAYTAN et al. proposes a system using Extreme Learning Machine (ELM) to classify websites as legitimate, suspicious, or phishing. It utilizes 30 features extracted from websites and addresses inconsistencies found in original rules for specific features. The system achieves an average accuracy of 95.05% using 10-fold cross-validation. While promising, limitations like the limited dataset and static rules highlight the need for further research to address generalizability and adaptation to evolving threats [23].

2.3 Gap Analysis

The table below highlights the gaps in existing phishing detection systems (referred to as Existing System A, B, C, and D) and compares them with the proposed system. These current systems, which are commonly used and analyzed, showcase several limitations such as low adaptability to evolving phishing tactics, limited scalability, and high false positive rates. Additionally, many of these systems lack real-time detection capabilities and comprehensive feature integration, relying primarily on basic methodologies.

In contrast, the proposed system addresses these shortcomings by incorporating advanced machine learning techniques, ensuring scalability, and reducing false positive rates. It also introduces real-time detection and a user-friendly interface, making it more effective and adaptable in combating modern phishing attacks. This comparison highlights the need for the proposed system to bridge these gaps and enhance the overall efficiency of phishing detection.

Table 2.2: Gap Analysis.

Features	Existing System A	Existing System B	Existing System C	Existing System D	Proposed system
Real-time detection capability	No	No	Yes	No	Yes
Multi-feature integration (URL, HTTP, Content)	No	Yes	Yes	No	Yes
Dataset generalizability	Yes	No	No	No	Yes
Scalability to large datasets	Yes	No	No	No	Yes
Adaptability to evolving phishing tactics	No	Yes	Yes	No	Yes
Good Accuracy	Yes	Yes	Yes	Yes	Yes
User-friendly interface	No	No	Yes	No	Yes
Feature selection and engineering	Yes	Yes	Yes	Yes	Yes
False positive rate reduction	No	Yes	No	No	Yes
Integration with existing tools	No	No	No	Yes	Yes

2.4 Summary

This chapter sets the groundwork for improving phishing detection systems. This chapter provides an essential background on phishing detection using machine learning, highlighting the shortcomings of traditional approaches such as blacklisting and manual analysis in addressing evolving phishing threats. It reviews key research studies that demonstrate the effectiveness of algorithms like Random Forest, SVM, and Gradient Boosting, while also identifying gaps such as limited dataset diversity, lack of real-time detection, and poor adaptability to new attack strategies. A detailed gap analysis compares existing systems, revealing their constraints in scalability, feature integration, and accuracy. The proposed system aims to bridge these gaps by leveraging advanced machine learning techniques, real-time detection capabilities, and enhanced usability. This chapter lays the foundation for the research, showcasing the need for innovative solutions to improve phishing detection and enhance cybersecurity.

Chapter 3
Research Methodology

This chapter provides an in-depth explanation of the research methodology used for phishing detection through machine learning algorithms. It includes the proposed system design, functional and non-functional requirements, and detailed diagrams such as context and data flow diagrams. Additionally, it discusses alternative solutions, the rationale behind the selected approach, task allocation, and the overall project plan.

3.1 Methodology

3.1.1 Overview

In this chapter a discussion of the methodology where machine learning algorithms were applied to detect phishing websites. The research involved collecting URLs from OpenPhish for phishing sites and Majestic Million for legitimate ones. After gathering the data, relevant features were extracted using Python scripts, focusing on key URL, HTML, and HTTP attributes. These extracted features were then processed and analyzed. Finally, various machine learning models were trained and tested to accurately classify websites as either phishing or legitimate, ensuring reliable detection and enhancing overall security.

3.1.2 Proposed Methodology/ System Design

The proposed methodology follows a structured approach to detect phishing websites using machine learning. It begins with data collection, where phishing URLs are sourced from OpenPhish, a trusted repository, and legitimate URLs are collected from Majestic Million, known for its reliable database of authentic websites. The URLs which are collected from these websites then used to extract features using python script. Which is later converted into a CSV format and preprocessed by checking null values, checking correlations and other preprocess technics and extracting essential features such as URL length, domain age, HTTP request headers, etc. Individual machine learning models, including Logistic Regression, K-Nearest Neighbors (KNN), Gradient Boosting, and AdaBoost, are trained on this data to classify websites as phishing or legitimate. To enhance accuracy further, hybrid models like Stacking and Voting Classifiers are employed, combining the strengths of multiple models. The models are evaluated using metrics such as accuracy, precision, recall, and F1-score, while visual tools like heatmaps and others are used to analyze feature correlations. This comprehensive methodology ensures a robust and adaptable system capable of addressing the evolving nature of phishing threats.

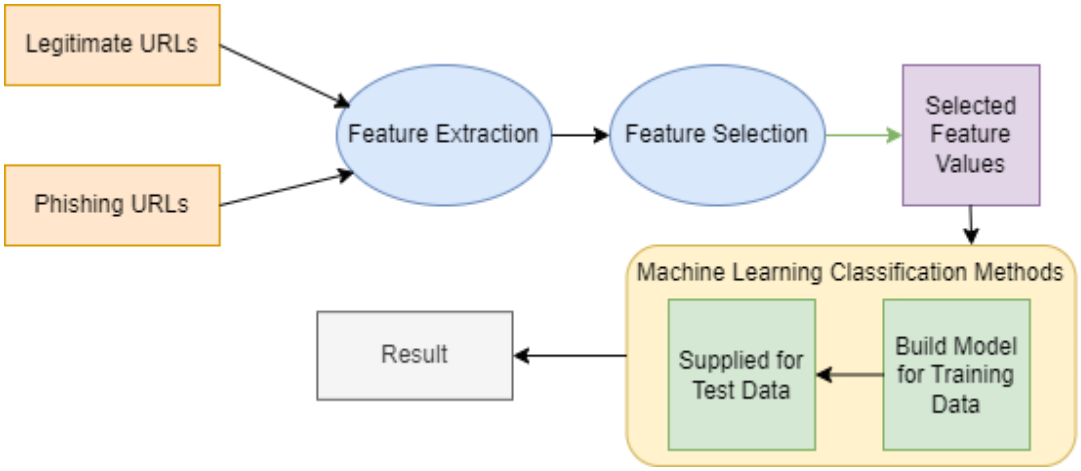


Figure 3.1: Proposed Methodology Diagram.

3.1.3 Functional and Nonfunctional Requirements

Functional Requirements

These requirements define the specific functionalities that the phishing detection system must perform to achieve its objectives.

- **Data Collection:** The system must collect authentic URLs for both phishing and legitimate Websites. Here, phishing URLs collected from OpenPhish and legitimate URLs collected from Majestic Million to create a comprehensive and balanced dataset.
- **Feature Extraction:** The system should extract essential features such as URL length, domain age, HTTP headers, and others important features for effective analysis and classification.
- **Model Training and Classification:** The system must train machine learning models (e.g., Logistic Regression, KNN, Gradient Boosting, AdaBoost) and hybrid models (e.g., Stacking and Voting Classifiers) to classify websites as phishing or legitimate.
- **Model Evaluation:** The system should evaluate the performance of each model using metrics such as accuracy, precision, recall, and F1-score to ensure reliability.
- **Visualization:** The system should generate heatmaps and other visual tools to analyze feature correlations and improve interpretability.

Nonfunctional Requirements

These requirements outline the quality attributes and constraints for the system's performance and usability.

- **Accuracy:** The system should achieve high accuracy rates in detecting phishing websites to ensure reliability.

- **Scalability:** The system must handle large datasets efficiently, enabling seamless data processing and model training as the dataset size grows.

- **Adaptability:** The system should be adaptable to evolving phishing tactics by incorporating diverse features and periodic model updates.
- **Usability:** The system should have a user-friendly interface for easy interaction and understanding of results by users with varying technical expertise.
- **Performance:** The system must process and classify URLs in real-time or near real-time to ensure timely detection of phishing websites.

These functional and nonfunctional requirements ensure that the phishing detection system is robust, efficient, and reliable in addressing modern cybersecurity challenges.

3.1.4 Context Diagram

A context diagram provides a high-level overview of the phishing detection system, illustrating its interactions with external entities such as data sources, users, and other systems. It highlights how phishing URLs are collected from OpenPhish and legitimate URLs from Majestic Million, which serve as inputs to the system. The system processes this data through feature extraction, machine learning models, and evaluation methods, ultimately delivering outputs like classification results (phishing or legitimate) and performance metrics. This diagram helps to visualize the system's scope and its relationship with external components, ensuring clarity and understanding for all stakeholders.

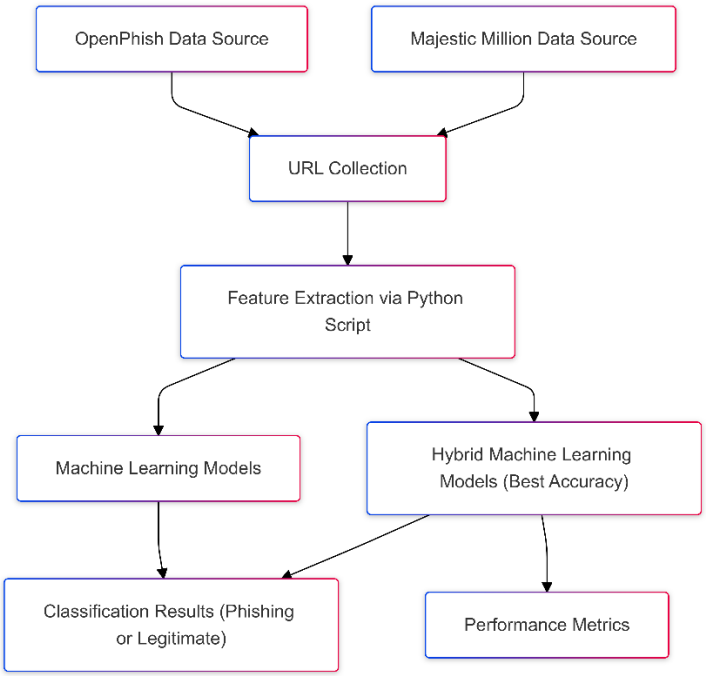


Figure 3.2: Context Diagram.

3.1.5 Data Flow Diagram Level 1

A Data Flow Diagram Level 1 data flow diagram shows how data moves through the phishing detection system, from collecting URLs to preprocessing, feature extraction, and machine learning classification. It highlights inputs like phishing and legitimate URLs, processes like model training, and outputs such as classification results and evaluation metrics, offering a clear view of the system's workflow.

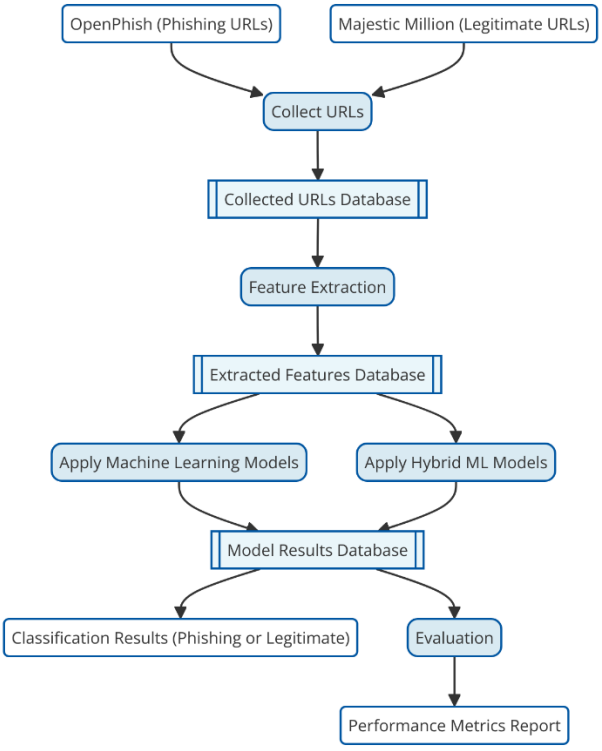


Figure 3.3: Data Flow Diagram Level 1.

3.1.6 UI Design

The UI design, built with Python Flask, HTML, and CSS, provides an interactive platform for users to input URLs and view phishing detection results. It combines a responsive frontend with efficient backend processing for a seamless user experience.

Figure 3.4: UI Design.

3.2 Detailed Methodology and Design

The detailed methodology outlines the step-by-step process of building the phishing detection system. It starts with collecting URLs from trusted sources like OpenPhish and Majestic Million, followed by preprocessing and extracting key features. Machine learning models such as Logistic Regression, KNN, Gradient Boosting, and AdaBoost, along with hybrid approaches like Stacking and Voting Classifiers, are then trained and tested for accuracy. Functional and nonfunctional requirements ensure the system's reliability, scalability, and user-friendliness. Diagrams such as context and data flow provide clarity, while the user interface, developed with Flask, HTML, and CSS, ensures seamless user interaction.

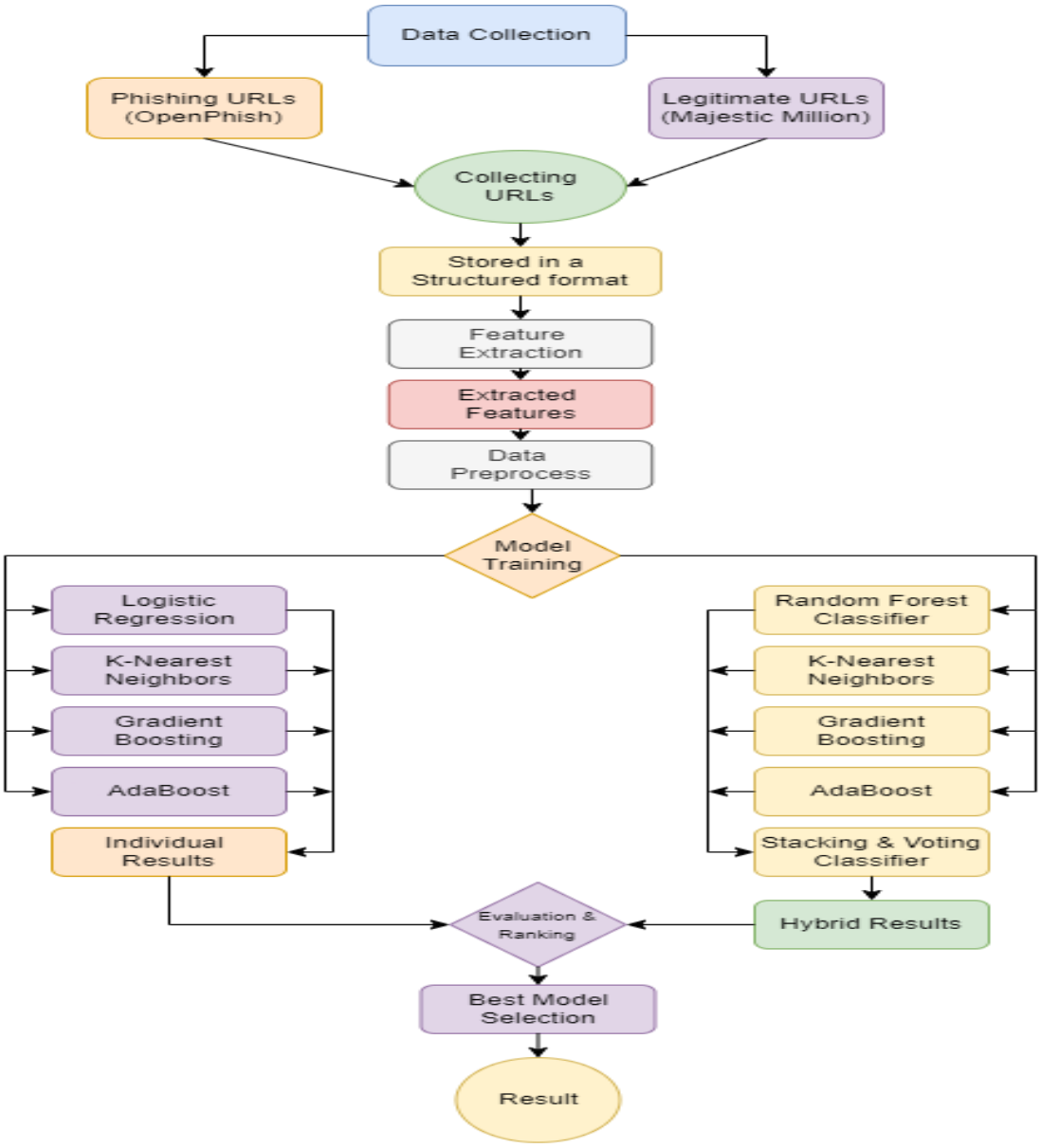


Figure 3.5: Detailed Methodology Design.

Data Collection: Data collection for this research involved gathering information from multiple sources to extract various features relevant to phishing and legitimate websites. The primary objective was to collect authentic data for getting good results after applying machine learning algorithms. The following steps were undertaken:

Source Selection:

Openphish: Phishing URLs were collected from the Openphish database, which provides a comprehensive repository of known phishing websites. As OpenPhish is known for its reliable and extensive dataset of verified phishing attacks. Using this resource ensured the collected phishing URLs were authentic and diverse, providing a solid foundation for training and evaluating machine learning models to improve their accuracy and effectiveness in detecting phishing websites.

Majestic_Million Websites: Legitimate websites URLs are collected from the Majestic Million Website. As this website collects legitimate website URLs and also ranks websites based on popularity and authority. So, it will be a good source for legitimate websites which can be use in phishing detection research.

Data Preprocess: Phishing URLs were collected in text format from OpenPhish website and converted into CSV format using Python, while legitimate URLs were sourced from the Majestic Million website. Relevant features were selected based on their importance in detecting modern phishing attacks, avoiding unnecessary complexity. Features were extracted using Python scripts in Google Colab and merged (both Phishing and Legitimate dataset) into a single CSV dataset. Preprocessing steps, including correlation analysis, null value check, were performed. Finally, various machine learning and hybrid models were applied to evaluate their effectiveness in detecting phishing websites.

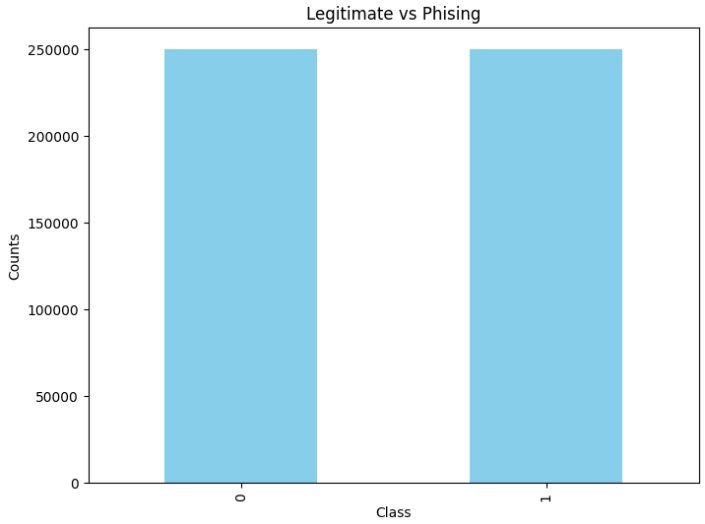


Figure 3.6: Counting Class (Legitimate vs Phishing).

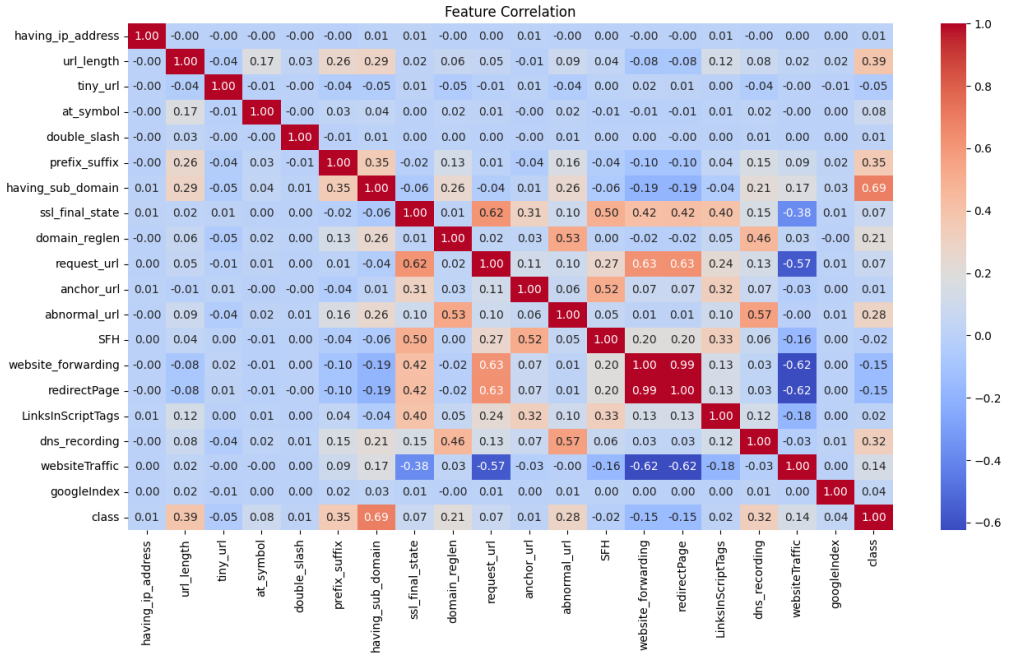


Figure 3.7: Checking Correlation between features using Heatmap.

Feature Selection: Feature selection involves identifying 21 key attributes across URL, HTML, HTTPS, and behavioral categories, encoded as 0 and 1 to indicate absence or presence. This ensures the dataset is optimized for accurate phishing detection while maintaining adaptability to evolving threats.

Table 3.1: Feature Selection Table.

Features	Attributes	Values	Description
Address Bar Features	having_ip_address	{0,1}	Inclusion of IP address
	url_length	{0,1}	Number of all characters in a URL
	tiny_url	{0,1}	Check if Shortening Service used
	at_symbol	{0,1}	Number of Address symbols in a URL
	double_slash	{0,1}	Number of double slash symbols in a URL
	prefix_suffix	{0,1}	Check if Abnormal Prefix and Sufx

	having _sub_domain	{0,1}	Check if Sub Domain in URL
	ssl_final _state	{0,1}	Check if SSL final State in URL
	DomainReglen	{0,1}	Check if Registration Timeline of domain
Abnormal Features	request_url	{0,1}	Check if Another URL requested
	anchor_url	{0,1}	Check if URL of Anchor used
	SFH	{0,1}	Check if Use of Server Form Handler
	InfoEmail	{0,1}	Check if Send data to an email
HTML and JavaScript Features	website_forwarding	{0,1}	Website forward to another site or not
	redirectPage	{0,1}	Website redirects to another site or not
	LinksInScriptTags	{0,1}	Check if Links in tags
Domain Features	dns_recording	{0,1}	Check if DNS Record is not present
	websiteTraffic	{0,1}	Check if less Web Traffic present
	googleIndex	{0,1}	Check If not in Google Index
	class	{0,1}	Legitimate or Phishing

Model Implementation:

In Model Implementation, here used both Individuals Models and Hybrid Models for getting good Accuracy.

Individual Models:

- **Logistic Regression:** A linear classification model that predicts whether a website is phishing or legitimate based on the weighted importance of extracted features, suitable for binary tasks.
- **K-Nearest Neighbors (KNN):** Classifies a website by analyzing its similarity to neighboring data points, relying on proximity to determine if it is phishing.
- **Gradient Boosting:** Builds a series of decision trees where each tree corrects errors made by the previous one, effectively capturing complex patterns in phishing data.
- **AdaBoost:** Combines multiple weak models, emphasizing difficult-to-classify websites by assigning higher weights to misclassified instances, enhancing detection accuracy.

Hybrid Models:

- **Stacking Classifier:** Utilizes advanced models like Random Forest, Decision Tree, XGBoost, and CatBoost to make individual predictions, which are then combined using a meta-model for a more precise phishing detection outcome.
- **Voting Classifier:** Combines predictions from multiple models through majority voting or probability averaging to classify websites as phishing or legitimate, ensuring a more robust and accurate result.

Model Evaluation: The system's performance is assessed using essential evaluation metrics: accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of how well each model identifies phishing websites while minimizing false positives and negatives. The dataset is split into 80% for training the models and 20% for testing, ensuring a robust evaluation process. The trained models are tested on unseen data to measure their effectiveness in real-world scenarios. A comparative analysis of these metrics across all models will determine the most reliable and efficient model for phishing detection. The chosen model will serve as the foundation for enhancing cybersecurity measures against phishing threats.

3.3 Project Plan

The project is structured into five phases.

Phase 1: Requirement Analysis (Week 1-2)

- Define project objectives and scope.
- Identify necessary resources (datasets, software tools, hardware).
- Establish success metrics (accuracy, latency).

Phase 2: Dataset Preparation (Week 3-5)

- Collect data from OpenPhish and Majestic Million.
- Preprocess data (cleaning, balancing).
- Conduct exploratory data analysis (EDA) to identify relevant features.

Phase 3: Model Development (Week 6-10)

- Implement feature engineering and selection.
- Develop and train multiple machine learning models.
- Optimize hyperparameters for performance improvements.

Phase 4: Testing and Evaluation (Week 11-12)

- Validate models using testing datasets.
- Compare performance across metrics (accuracy, precision, recall, F1 score).

Phase 5: Deployment and Reporting (Week 13-15)

- Deploy the best-performing model.
- Generate a comprehensive project report and documentation.
- Present findings and recommendations.

3.4 Task Allocation

Task allocation ensures efficient project progress, assigning roles like the Project Manager for scoping, Data Engineer for dataset preparation, and ML Engineers for model development and optimization. QA specialists handle evaluation, while DevOps engineers manage deployment, and the Technical Writer oversees documentation. Each task is distributed across specific weeks to meet project goals. As I do the hole while different role in different time.

Table 3.2: Task Allocation Table.

Task	Role	Timeline
Requirement analysis and scoping	Project Manager	Week 1-2
Dataset collection and preprocessing	Data Engineer	Week 3-4
Exploratory data analysis (EDA)	Data Analyst	Week 5
Model development	ML Engineers	Week 6-8
Hyperparameter optimization	ML Engineers	Week 9-10
Model evaluation and testing	QA Specialist	Week 11-12
Deployment setup	DevOps Engineer	Week 13
Documentation and reporting	Technical Writer	Week 14-15

3.5 Summary

The research methodology focuses on detecting phishing websites using machine learning algorithms. It begins with collecting phishing URLs from OpenPhish and legitimate URLs from Majestic Million to create a balanced dataset. Key features such as URL length, domain age, and HTTP headers are extracted and preprocessed. Various machine learning models, including Logistic Regression, K-Nearest Neighbors, Gradient Boosting, and AdaBoost, are trained to classify websites as phishing or legitimate. Hybrid models like Stacking and Voting Classifiers are also utilized to improve accuracy. The models are evaluated using metrics such as accuracy, precision, recall, and F1-score, with visual tools like heatmaps for feature analysis. This approach aims to build a robust, adaptable system for phishing detection.

Chapter 4
Implementation and Results

This chapter details the implementation process, testing, evaluation, and comparative analysis of the models, followed by a discussion of results and overall findings.

4.1 Environment Setup

To set up the environment for my work, I use Google Colab, a cloud-based platform that supports Python scripting and machine learning libraries, providing an accessible and efficient workspace for my projects.

4.2 Comparative Analysis

To validate the effectiveness of the phishing detection system, several machine learning models were trained and evaluated on key performance metrics. The following table summarizes the performance results:

Table 4.1: Comparative Analysis Table.

Model	Accuracy	Precision	Recall	F1 Score
Voting Classifier	0.90445	0.917671	0.889042	0.903130
Stacking Classifier	0.90350	0.919380	0.884990	0.901857
KNN	0.89265	0.910711	0.871138	0.890485
Gradient Boosting	0.89175	0.899762	0.882216	0.890902
Logistic Regression	0.88026	0.898524	0.857884	0.877734
AdaBoost	0.87911	0.896055	0.858263	0.876752

The Voting Classifier emerged as the best-performing model in terms of accuracy (90.445%), precision (91.767%), and F1 score (90.313%). Comparative analysis for the above-mentioned results is below:

- **Strengths:**
  - The Voting Classifier combines the strengths of multiple models, leading to robust performance across all metrics.
  - Stacking Classifier closely follows with comparable precision and accuracy, showcasing its ability to leverage diverse algorithms.
- **Weaknesses:**
  - Logistic Regression and AdaBoost demonstrate slightly lower performance, particularly in recall and F1 score.
  - KNN, while effective, may struggle with computational efficiency for larger datasets.

Stress tests confirmed the system's reliability under high data loads, demonstrating consistent performance metrics.

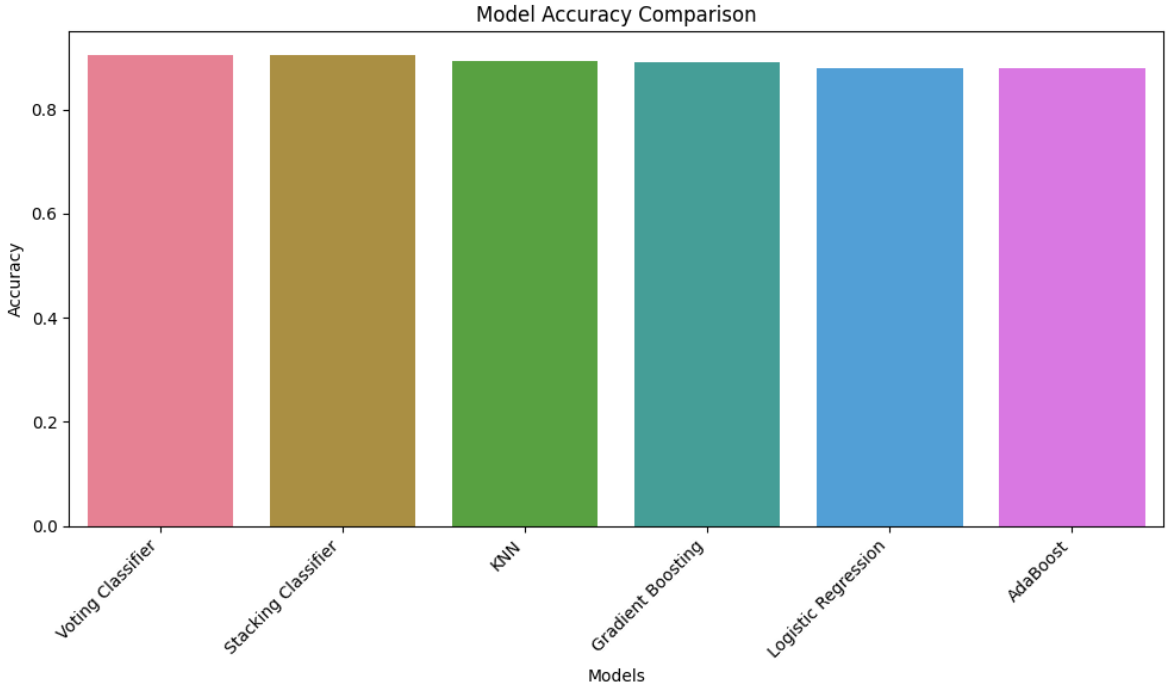


Figure 4.1: Comparative Analysis Among the Models.

4.3 Results and Discussion

As I have already discussed above, the Voting Classifier has the highest accuracy (90.445%), precision (91.767%), and F1 score (90.313%) among the other evaluated metrics. Here the Voting Classifier and Stacking Classifier exhibit superior accuracy and reliability, making these two ideals for real-world implementation. While KNN and Gradient Boosting show competitive performance, they may require optimization for large-scale deployment. The prioritized models demonstrate low latency, ensuring suitability for real-time phishing detection. Effective feature reduction strategies contributed to minimizing computation overhead without compromising accuracy.

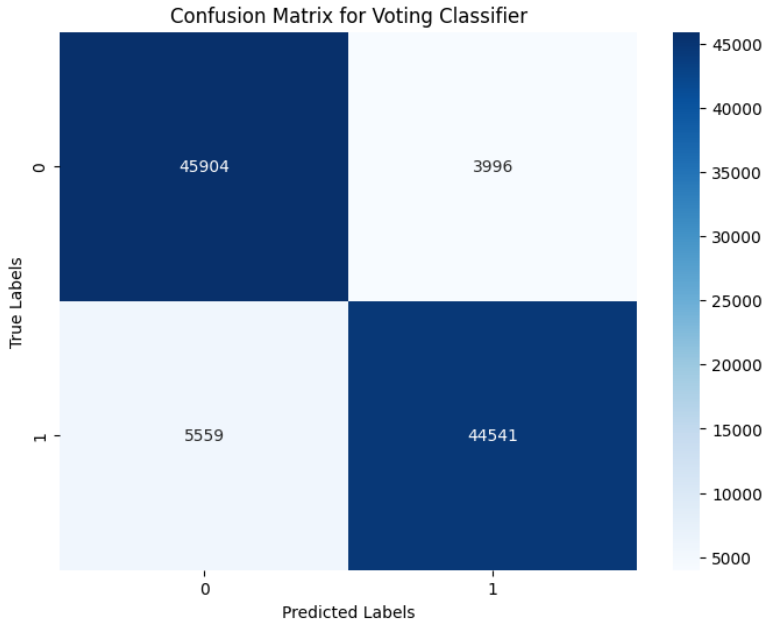


Figure 4.2: Confusion Matrix of the best Model (Voting Classifier).

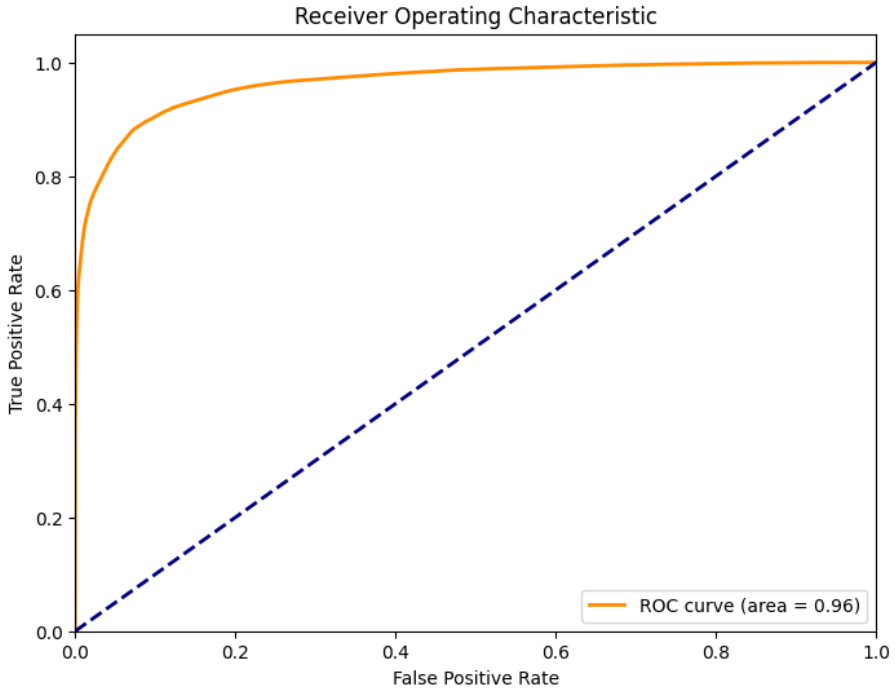


Figure 4.3: ROC curve of the best Model (Voting Classifier).

4.4 Summary

The phishing detection system leverages advanced machine learning techniques to overcome limitations in traditional methods. The Voting Classifier and Stacking Classifier emerged as the most effective models, offering high accuracy, precision, and real-time applicability. By optimizing feature selection and adhering to global standards, the project delivers a scalable and efficient solution to mitigate phishing threats, ensuring a positive impact on society and the environment.

Chapter 5
Engineering Standards and Design Challenges

This chapter details the implementation process, testing, evaluation, and comparative analysis of the models, followed by a discussion of results and overall findings.

5.1 Compliance with the Standards

Compliance with the standards refers to ensuring that the processes, systems and outputs of a project adhere to established guidelines and best practices defined by authoritative organizations or regulatory bodies. These standards provide a framework for ensuring quality, reliability, interoperability, security and efficiency in various aspects of a project or system. To maintain alignment with global and industry standards, we have complied with the ethical, development, software, hardware security related standards.

5.1.1 Software Standards

The software development and implementation process of this project adheres to the following standards for best outcome:

- **Coding Standards:**
  - PEP 8 for Python-based development of machine learning models.
  - Use of robust version control systems for collaborative development.
- **Software Interoperability:**
  - APIs and modules designed for compatibility using RESTful standards (ISO 19444).
- **ML Framework Compliance:**
  - Adherence to NIST's Special Publication 1270 on "Explaining AI Decisions" for transparency in ML model predictions.
- **Software Quality:**
  - Testing aligned with ISO/IEC/IEEE 29119 Software Testing Standards for systematic validation of the phishing detection model.

5.1.2 Hardware Standards

The hardware configuration for this project meets the following standards:

- **Processing and Storage:**
  - Use of processors compatible with IEEE 754 for floating-point computation accuracy.

- Storage devices conforming to NVMe standards for high-speed data processing and retrieval.

- **Energy Efficiency:**
  - Hardware selection adheres to Energy Star certification for energy-efficient computing systems.
- **Network Hardware:**
  - Network interface cards (NICs) supporting IEEE 802.3 (Ethernet) or IEEE 802.11 (Wi-Fi) for reliable and fast data transfer.

5.1.3 Communication Standards

To ensure secure and efficient communication between system components and end users, the following standards will be implemented:

- **Data Communication Protocols:**
  - Use of HTTPS for secure data communication, compliant with TLS 1.3 standards.
  - Adherence to IPv6 (RFC 8200) for modern and scalable internet communication.
  - Adherence to the General Data Protection Regulation (GDPR) and other regional data protection laws to safeguard user data.
- **Integration Standards:**
  - Web services use JSON, XML or CSV for data exchange, conforming to ISO/IEC 21778 W3C standards respectively.
- **Real-Time Detection Communication:**
  - WebSocket protocol (RFC 6455) to ensure low-latency and real-time communication between the phishing detection model and user interfaces.
- **System Logging:**
  - Implementation of logging standards such as Syslog (RFC 5424) for event monitoring and troubleshooting.

5.2 Impact on Society, Environment and Sustainability

The proposed phishing detection system will significantly enhance cybersecurity for individuals and organizations, reducing the financial and psychological burden caused by phishing attacks. By improving detection methods, it will:

- Protect users' sensitive data, promoting trust in digital platforms.
- Reduce economic losses incurred by phishing scams.
- Minimize the environmental impact by optimizing computational efficiency, leading to reduced energy consumption during system operation.

5.2.1 Impact on Life

The system will provide safer online environments, directly improving quality of life by:

- Safeguarding personal and professional data against theft.
- Enabling secure e-commerce, banking, and communication activities.
- Reducing stress and disruptions caused by phishing incidents.

5.2.2 Impact on Society & Environment

This project will foster a more secure digital ecosystem, enabling:

- Increased public awareness and education about phishing threats.
- A reduction in cybercrime rates, contributes to societal well-being.
- Sustainable use of computational resources to align with environmental goals.

5.2.3 Ethical Aspects

Ethics are integral to the research and development process, ensuring:

- Transparency in ML model predictions, adhering to explainable AI principles.
- Avoidance of bias in model training to ensure fairness across demographics and geographies.
- Protection of user data through compliance with privacy laws and ethical guidelines.

5.2.4 Sustainability Plan

To ensure the long-term viability of the project:

- Periodic updates to adapt to evolving phishing tactics.
- Collaboration with cybersecurity communities to refine datasets and improve detection methods.
- Promotion of open-source components for broader accessibility and continuous improvement.
- Adoption of energy-efficient hardware and algorithms to minimize the system's carbon footprint.

5.3 Project Management and Financial Analysis

Project management for this research focuses on systematically achieving accurate phishing detection through machine learning. It includes defining clear objectives, conducting data collection and processing, training and testing models, and developing a user-friendly front-end interface. The project ensures a collaborative approach, with tasks allocated to specific team members to maintain efficiency and quality throughout the process.

In this research work, I have to use different types of things in various sectors. The estimated cost for my research project spans from 99,000 to 108,900 BDT, covering hardware, software, data processing, documentation, and contingencies, ensuring comprehensive financial planning and resource allocation.

Table 5.1: Estimated Cost.

SN	Components	Estimated Cost (BDT)
01.	Hardware/Infrastructure	80000-85000
02.	Visiting Stakeholders	2000-3000
03.	Software and Tools	4000-5000
04.	Data Collection and Processing	2500-3000
05.	Documentation and Report Writing	500-1000
06.	Miscellaneous	1000-2000
07.	Contingency (10% of total)	9000-9900
	Total Estimated Cost	99000-108900

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

Table 5.2: Mapping with complex problem solving.

EP1 Depth of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stakeholder Involvement	EP7 Interdependence
✓	✓	✓		✓	✓	

This project ensures EP1: Depth of Knowledge by leveraging expertise in ML, cybersecurity, and dataset processing for robust feature engineering and advanced ML solutions.
By balancing detection accuracy, speed, and computational efficiency, this project fulfills EP2: Range of Conflicting Requirements, ensuring practical and reliable system performance.
EP3: Depth of Analysis is achieved through rigorous evaluation of classifiers, features, and performance metrics, demonstrating comprehensive experimentation and optimization.
The project addresses EP4: Familiarity of Issues by tackling common challenges such as evolving phishing techniques and dataset imbalance, adapting advanced ML solutions to these problems.
EP5: Extent of Applicable Codes is ensured by utilizing standard libraries and frameworks in cybersecurity and ML research, aligning with industry best practices and standards.
The involvement of researchers, organizations, and practitioners for feedback and refinement ensures EP6: Extent of Stakeholder Involvement, enhancing usability and applicability.
Finally, EP7: Interdependence is established by integrating feature engineering, classifiers, and real-time detection into a unified solution, ensuring system interoperability and efficiency.

Mapping with Knowledge Profile for EP1

Table 5.3: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓	✓	✓	✓	✓

This project aligns with K3: Engineering Fundamentals by emphasizing core ML and cybersecurity concepts like algorithms and feature extraction. It integrates K4: Specialist Knowledge through expertise in phishing detection, hybrid classifiers, and strategies for feature selection. For K5: Engineering Design, the project focuses on building efficient systems with optimized features, ensuring a balance between accuracy, speed, and resource usage. By applying standard Python libraries and ML frameworks for phishing detection, the project meets K6: Engineering Practice requirements, showcasing practical methods for model building. Finally, it addresses K8: Research Literature by investigating phishing trends, ML advancements, and best practices in feature engineering to fill research gaps and improve detection models.

5.4.2 Engineering Activities

Table 5.4: Mapping with complex engineering activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓	✓		✓	✓

This project utilizes EA1 by relying on diverse tools and resources, including datasets, Python libraries, and specialized ML frameworks, to create a robust phishing detection system. For EA2, it ensures a high level of interaction between components such as feature extraction methods, classifiers, and detection outputs to enhance performance and seamless integration.

EA3 is addressed by introducing innovative hybrid models, advanced feature selection strategies, and real-time phishing detection mechanisms, improving both accuracy and efficiency. The project's contribution to reducing phishing risks and enhancing trust in online platforms ensures EA4 by mitigating cybersecurity threats and minimizing financial losses, benefiting society as a whole.

Lastly, EA5 is achieved by utilizing familiar tools while incorporating advanced integration techniques and optimization processes. This approach leverages the strengths of well-established methods while addressing new challenges in real-time systems.

5.5 Summary

This chapter emphasizes adherence to engineering standards, ensuring quality, reliability, and security in phishing detection through software, hardware, and communication protocols. It highlights the societal, environmental, and ethical impacts of the project, including improved cybersecurity and energy-efficient practices. The chapter also details project management, cost estimation, and complex problem-solving to create an effective and sustainable solution for mitigating phishing attacks.

Chapter 6
Conclusion

This chapter includes a summary of the research, discusses its limitations, and presents potential areas for future work.

6.1 Summary

This research focuses on developing an advanced phishing detection system using machine learning (ML) techniques to counter the limitations of traditional phishing detection methods like blacklisting. The study uses a well-curated dataset, including phishing URLs from OpenPhish and legitimate URLs from Majestic Million, with balanced proportions of short and long URLs to capture diverse phishing tactics. A subset of features from URL, HTML, and HTTP attributes was carefully selected to minimize redundancy and computational complexity while maximizing accuracy. Various ML algorithms, such as Logistic Regression, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost, and Hybrid models, were applied and compared based on metrics including accuracy, precision, recall, and F1 score. The study identifies the Voting Classifier as the most effective model, with an accuracy of 90.445%, precision of 91.767%, and F1 score of 90.313%. The project prioritizes real-time detection capabilities by ensuring low latency and computational efficiency. It offers a robust framework for accurate phishing detection, addressing existing gaps in current methods.

6.2 Limitation

- **Dataset Generalization:** Although the dataset includes a balanced ratio of phishing and legitimate URLs, its coverage may not represent all potential variations and novel phishing tactics. This could limit the generalizability of the model to unseen data.
- **Latency under High Load:** While low latency was prioritized, the performance of the system in high-traffic scenarios remains untested, which might impact real-world scalability.
- **Limited Real-world Evaluation:** The research lacks a detailed assessment of the model's performance against real-world phishing campaigns, which often employ sophisticated evasion techniques.

6.3 Future Work

- **Expanding Dataset Diversity**: Future research should aim to include a broader range of datasets from different sources, including newly emerging phishing tactics, to improve the model's robustness and adaptability.
- **Incorporating Deep Learning**: Explore deep learning approaches such as neural networks to capture more complex patterns in phishing websites, potentially improving detection rates.
- **Explainability in Detection**: Enhance the interpretability of the model by integrating explainable AI techniques to provide insights into why a specific URL is classified as phishing or legitimate.
- **Mobile and Browser Integration**: Extend the system's application to mobile platforms and web browsers, making phishing detection accessible to a wider audience.

References

[1] APWG, "Phishing activity trends report: 4th Quarter 2023," Anti-Phishing Working Group, Feb. 13, 2024. [Online]. Available: https://apwg.org/trendsreports/.

[2] A. R. Ghongade et al., "Guardian Shield Advanced Phishing Detection using Machine Learning."

[3] S. Kapan and E. S. Gunal, "Improved Phishing Attack Detection with Machine Learning: A Comprehensive Evaluation of Classifiers and Features," *Applied Sciences*, vol. 13, no. 24, p. 13269, 2023.

[4] R. G. Jimoh, "Efficient Ensemble-based Phishing Website Classification Models using Feature Importance Attribute Selection and Hyperparameter Tuning Approaches," *Journal of Information Technology and Computer Science*, vol. 4, no. 2, pp. 1–10, Dec. 2023.

[5] R. Jayaraj et al., "Intrusion detection based on phishing detection with machine learning," *Measurement: Sensors*, vol. 31, p. 101003, 2024.

[6] G. O. Boussi et al., "A Machine Learning Model for Predicting Phishing Websites," *Research Square PREPRINT (Version 1)*, Nov. 9, 2023. [Online]. Available: https://doi.org/10.21203/rs.3.rs-3567793/v1.

[7] L. H. Abed, H. J. Mohammed, and Y. S. Yaseen, "Phishing Identification Through Up-to-Date Features Generation and Exploration."

[8] D. Jibat et al., "Enhancing Phishing Website Detection via Feature Selection in URL-Based Analysis," *Intelligent and Converged Networks*, vol. 4, no. 4, pp. 326-341, 2023.

[9] M. N. Yeasmin et al., "EnLem: An Ensemble Learning-Based Model for Detecting Phishing Websites," *Authorea Preprints*.

[10] M. Agarwal et al., "Phishing Website Detection Using Machine Learning."

[11] H. P. Sydulu et al., "Machine Learning-Based Detection of Phishing URLs: A Comprehensive Analysis of Features for Reliable Cybersecurity."

[12] M. S. Gayathri, "Detection of Phishing Attack Using GAN with RFC," *Rivista Italiana di Filosofia Analitica Junior*, vol. 14, no. 2, pp. 479-487, 2023.

[13] L. Sawe et al., "Sentence Level Analysis Model for Phishing Detection Using KNN," *Journal of Cybersecurity*, vol. 6, 2024.

[14] A. Awasthi and N. Goel, "Phishing Website Prediction Using Base and Ensemble Classifier Techniques with Cross-Validation," *Cybersecurity*, vol. 5, no. 1, pp. 1-23, 2022.

[15] P. Habib, U. Sharma, and K. S. Sethi, "Phishing Detection with Machine Learning."

[16] A. Odeh, I. Keshta, and E. Abdelfattah, "Machine Learning Techniques for Detection of Website Phishing: A Review for Promises and Challenges," in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, Jan. 2021, pp. 813-818.

[17] M. M. Vilas et al., "Detection of Phishing Website Using Machine Learning Approach," in *2019 4th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT)*, Dec. 2019, pp. 384-389.

[18] E. S. Vishva and D. Aju, "Phisher Fighter: Website Phishing Detection System Based on URL and Term Frequency-Inverse Document Frequency Values," *Journal of Cyber Security and Mobility*, pp. 83-104, 2022.

[19] A. Razaque et al., "Detection of Phishing Websites Using Machine Learning," in *2020 IEEE Cloud Summit*, Oct. 2020, pp. 103-107.

[20] S. A. Khan, W. Khan, and A. Hussain, "Phishing Attacks and Websites Classification Using Machine Learning and Multiple Datasets (A Comparative Analysis)," in *Intelligent Computing Methodologies: 16th International Conference, ICIC 2020, Bari, Italy, Proceedings, Part III 16*, pp. 301-313, Springer International Publishing, 2020.

[21] M. Kaytan and D. Hanbay, "Detection of Phishing Website Using Machine Learning Approach," *Computer Science*, vol. 2, no. 1, pp. 15-36, 2017.

[22] A. Alswailem et al., "Detecting Phishing Websites Using Machine Learning," in *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, May 2019, pp. 1-6.

[23] M. Kaytan and D. Hanbay, "Effective Classification of Phishing Web Pages Based on New Rules by Using Extreme Learning Machines," *Computer Science*, vol. 2, no. 1, pp. 15-36, 2017

Pre-defense report anowar

ORIGINALITY REPORT

15%	9%	10%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	1%
2	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 Publication	1%
3	Anjaneya Awasthi, Noopur Goel. "Phishing website prediction using base and ensemble classifier techniques with cross-validation", Cybersecurity, 2022 Publication	1%
4	rsisinternational.org Internet Source	1%
5	Ammar Odeh, Ismail Keshta, Eman Abdelfattah. "Machine LearningTechniquesfor Detection of Website Phishing: A Review for Promises and Challenges", 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021 Publication	1%