|  |
| --- |
| **VISVESVARAYA TECHNOLOGICAL UNIVERSITY**  **BELAGAVI, KARNATAKA, INDIA**    **PROJECT REPORT PHASE-I**  **ON**  “PHISHING DETECTION OF MALICIOUS URLs USING MACHINE LEARNING”  ***Submitted in partial fulfillment of the requirement for the award of the degree of***  ***Bachelor of Engineering***  ***in***  ***Information Science and Engineering***  ***Submitted by***  **ANIMESH DWIVEDI [1SG20IS008]**  **CHIRAG TILWANI [1SG20IS023]**  **HARDIK SINGH [1SG20IS035]**  **LAVKESH PRASAD [1SG20IS048]**  ***Under the guidance of***  **Prof. Priyanka MR**  **Assistant Professor**    **DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING SAPTHAGIRI COLLEGE OF ENGINEERING**  (Affiliated to Visvesvaraya Technological University, Belagavi& Approved by AICTE, New Delhi)  #14/5 , Chikkasandra, Hesaraghatta Main Road, Bengaluru – 560057  a (Accredited by NBA and NAAC with ‘A’ GRADE)  (An ISO 9001:2015 & ISO 14001:2015 Certified Institution)  **2023-24** |

# SAPTHAGIRI COLLEGE OF ENGINEERING

(Affiliated to Visvesvaraya Technological University, Belagavi& Approved by AICTE, New Delhi)

#14/5, Chikkasandra, Hesaraghatta Main Road, Bengaluru – 560057

(Accredited by NBA and NAAC with ‘A’ GRADE)

(An ISO 9001:2015 & ISO 14001:2015 Certified Institution)

## Department of Information Science & Engineering



# CERTIFICATE



Certified that the project work entitled **“PHISHING DETECTION OF MALICIOUS URLs USING**

**MACHINE** phase-I carried out by **ANIMESH DWIVEDI [1SG20IS008], CHIRAG TILWANI**

**[1SG20IS023], HARDIK SINGH [1SG20IS035], LAVKESH PRASAD [1SG20IS048]** bonafide students of 7th semester, department of **Information Science & Engineering** carried out at our college **Sapthagiri College of Engineering**, Bengaluru in partial fulfillment of the award of by **Bachelor of Engineering in Information Science & Engineering of the Visvesvaraya Technological University,** Belagavi during the year 2023-24. It is certified that all corrections/suggestions indicated for final Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project prescribed for the said Degree work.

**Signature of the Guide Signature of the HOD Signature of Principal Prof. Priyanka MR Dr. H.R. Ranganatha Dr. H Ramakrishna Assistant Professor Professor & Head Principal**

## 

**ABSTRACT**

**Cybercrime through malicious phishing URLs is prevalent among the numerous types facilitated by the internet, leading this study to focus on phishing attacks. Despite its existence since 1996, phishing has emerged as the most dangerous internet crime, utilizing email trickery and mock websites to steal data. While research on prevention, identification, and awareness exists, a definitive solution to thwart it remains elusive. Therefore, this study focuses on combating phishing attacks amidst the rampant cybercrime landscape.**

i

# 

# ACKNOWLEDGEMENT

Any achievement doesn’t depend solely on the individual efforts but on the guidance, encouragement and co-operation of intellectuals, elders and friends. A number of personalities have helped us. We would like to take this opportunity to thank them all.

We would like to express our heart-felt gratitude to **Dr. H Ramkrishna**, Principal, Sapthagiri College of Engineering, Bengaluru, for his help and inspiration during the tenure of the course.

It is great privilege to extend our deep sense of gratitude to **Dr. H R Ranganatha**, Head of the Department, Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru, who patronized throughout our career, for his constant support and encouragement and for the facilities provided to carry out this work successfully.

We wish to express our sincere thanks to our guide **Prof. Priyanka M R**, Assistant Professor, Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru for helping us throughout and guiding us from time to time.

We also extend our sense of gratitude and sincere thanks to all faculty members and non-teaching staff members of Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru for their views and encouraging ideas.

Finally, we also thank our family and friends for their co-operation and motivation.

**ANIMESH DWIVEDI [1SG20IS008]**

**CHIRAG TILWANI [1SG20IS023]**

**HARDIK SINGH [1SG20IS035]**

**LAVKESH PRASAD [1SG20IS048]**

ii

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **TABLE OF CONTENTS** | |  | |
| **Chapter No.** | **Chapter Name** | | **Page No.** | |
| **1.** | **INTRODUCTION** | | **1** | |
| 1.1 | Overview | | 1 | |
| 1.2 | Challenges | | 1 | |
| 1.3 | Motivation | | 2 | |
| 1.4 | Benefits | | 2 | |
| 1.5 | Applications | | 2 | |
| 1.6 | Problem Statement | | 2 | |
| 1.7 | Objectives | | 2 | |
| **2.** | **LITERATURE SURVEY** | | **3** | |
| 2.1 | Overview | | 3 | |
|  |  | |  | |
| **3.** | **ANALYSIS** | | **10** | |
| 3.1 | Existing System | | 10 | |
| 3.2 | Problem Statement | | 11 | |
| 3.3 | Proposed System | | 11 | |
| 3.4 | Functional Requirements | | 11 | |
| 3.5 | Non-Functional Requirements | | 12 | |
| 3.6 | Software Requirements | | 12 | |
| 3.7 | Hardware Requirements | | 13 | |
| **4.** | **DESIGN** | | **14** | |
| 4.1 | Architectural Diagram | | 14 | |
| 4.2 | Data flow Diagram | | 15 | |
| 4.3 | Use-case Diagram | | 16 | |
| 4.4  4.5  4.6 | Activity Diagram  Class Diagram  Sequence Diagram | | 17  18  19 | |
| **5.** | **IMPLEMENTATION** | | **20** | |
| 5.1 | Modules Description | | 20 | |
| 5.2 | Data Preprocessing Module | | 20 | |
| 5.3 | Model Training Module | | 21 | |
| 5.4 | Code Snippets | | 23 | |
| **6.** | **CONCLUSION** | | **32** | |
|  | | |  | |
|  | | | **BIBILIOGRAPHY** | |
|  | | |  | |

iii

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure no** | **Name of the figure** | **Page no** |
| 4.1 | Architectural Diagram | 14 |
| 4.2 | Data flow diagram | 15 |
| 4.3 | Use-case diagram | 16 |
| 4.4  4.5  4.6 | Activity Diagram  Class Diagram  Sequence Diagram | 17  18  19 |

iv

**CHAPTER 1**

# INTRODUCTION

**1.1 Overview**

Phishing is a cyber-attack that lures the victim, using a technological bait to collect personal or confidential information. It started in 1995 with the American Online (AOL) attack. Afterwards, the phishers-individuals or teams steering the phishing attacks, moved to more profitable targets such as online banking and e-commerce services. Financial gains are the primary motivating factors for phishers; however, fame and notoriety are also interesting psychological aspects of phishing.

Phishing is an Internet threat and has a top rank in the cyber threat landscape. It has become a leading cyber threat to the financial sectors and has spread into many sectors. The recent statistics show the number of phishing attacks doubled in 2020 compared to the past, and nearly 84% of phishing sites have been recorded in the latter part of 2020 which used SSL protection. It indicates that HTTPS is not a vital feature when detecting phishing attacks at present. In fact, half of the phishing attacks’ lifetime ended in less than a day, and new phishing trends have also emerged rapidly due to the dynamic nature of phishing attacks.

PhishDet is proposed as a representation learning-based phishing detection solution that uses the URL and the HTML content of a website. It combines two deep networks – i.e. the Long-term Recurrent Convolutional Network (LRCN) and the Graph Convolutional Network (GCN). It performed well during the experiments and achieved 96.42% detection accuracy with a real-world public dataset. PhishDet has been further tested with a benchmark dataset, and it has outperformed similar solutions by recording 99.57% detection accuracy. The key contributions of this project are,

1). PhishDet - A phishing detection solution that automatically selects features from a website’s raw URL and HTML content;

2). The first GNN based phishing detection approach that uses raw HTML content.

**1.2 Challenges**

Phishing is the most significant issue in the field of networks and the Internet. Many researchers have attempted to provide facilities to protect users from cyber-attacks by preventing the phishing of URLs using machine learning, Two groups of phishing detection systems have been proposed and implemented in previous studies.

**1.3 Motivation**

The Internet provides a great opportunity for attackers to engage in criminal activities such as online fraud, malicious software, computer viruses, ransomware, worms, intellectual property rights, denial of service attacks, money laundering, vandalism, electronic terrorism, and extortion. Hacking is a major destroyer of the Internet through which any person can hack computer information and use it in different ways to harm others. Immorality, which harms moral values, is a major issue for the younger generations. Detecting these websites rather than websites that appear simple and secure, will help people. Therefore, an awareness of these websites is necessary. Viruses can damage an entire computer network and confidential information by spreading to multiple computers. It is not suitable to use unauthorized websites on the internet

**1.4 Benefits**

By implementing this project we can get following benefits.

1. Automatic detection of phishing website.
2. Comparing to current technology this project saves time of phishing website detection.
3. Comparing to current mechanism this project gives high accuracy in phishing website detection.

**1.5 Applications**

* Society
* Education
* Network Security

**1.6 Problem Statement**

Phishing attacks remain threat to online security, with attackers continually evolving their techniques to deceive users into revealing sensitive information. This project aims to develop a robust machine learning model capable of effectively detecting phishing URLs, thus contributing to a safer online environment.

**1.7 Objectives**

* To improve the accuracy in phishing website detection using machine learning.
* To implement the real time phishing website detection.
* To build an automated phishing website detection.
* To combine URL statistical features, webpage code features, webpage text features

to improve the accuracy in phishing website detection.

Top of Form

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 OVERVIEW**

A literature survey or a literature review in a project report shows the various analyses and research made in the field of interest and the results already published, taking into account the various parameters of the project and the extent of the project. Literature survey is mainly carried out in order to analyze the background of the current project which helps to find out flaws in the existing system & guides on which unsolved problems we can work out. So, the following topics not only illustrate the background of the project but also uncover the problems and flaws which motivated to propose solutions and work on this project.

A literature survey is a text of a scholarly paper, which includes the current knowledge including substantive findings, as well as theoretical and methodological contributions to a particular topic. Literature reviews use secondary sources, and do not report new or original experimental work. Most often associated with academic-oriented literature, such as a thesis, dissertation or a peer-reviewed journal article, a literature review usually precedes the methodology and results sectional though this is not always the case. Literature reviews are also common in are search proposal or prospectus (the document that is approved before a student formally begins a dissertation or thesis). Its main goals are to situate the current study within the body of literature and to provide context for the particular reader. Literature reviews are a basis for researching nearly every academic field. demic field. A literature survey includes the following:

* Existing theories about the topic which are accepted universally.
* Books written on the topic, both generic and specific.
* Research done in the field usually in the order of oldest to latest.
* Challenges being faced and on-going work, if available.

Literature survey describes about the existing work on the given project. It deals with the problem associated with the existing system and also gives user a clear knowledge on how to deal with the existing problems and how to provide solution to the existing problems.

**Objectives of Literature Survey**

* Learning the definitions of the concepts.
* Access to latest approaches, methods and theories.
* Discovering research topics based on the existing research
* Concentrate on your own field of expertise– Even if another field uses the same words, they usually mean completely.
* It improves the quality of the literature survey to exclude sidetracks– Remember to explicate what is excluded.

Before building our application, the following system is taken into consideration:

**2.2 Related Works:**

**[1] Title: Using Classical Machine Learning For Phishing Websites Detection Form URLs Author: Iman Akour Noha Alnazzawi, Ahmad Aburayya, Raghad Alfaisal, Sultan Idris Said Salloum, 2022** The paper addresses the urgent need to research phishing attacks, highlighting methods like blacklisting and website characteristics, emphasizing the importance of detection for cybersecurity. vUtilizing machine learning classifiers, the study identifies Support Vector Machine (SVM) as the most effective, achieving a 96.30% accuracy in detecting phishing URLs. The study proposes a phishing detection method using classical machine learning, employing models like SVM, KNN, LR, and NB on datasets with phishing and legitimate URLs. SVM achieved the highest accuracy (96.30%) using host-based and lexical features. The research underscores user education, combining ML and URL analysis for effective phishing detection, and suggests future work in dataset enhancement, hybrid algorithms, and NLP-based features.

**[2] Title: Lexical Feature Based Feature Selection And Phishing URL Classification Using Machine Learning techniques Author:** [**Bireswar Banik**](https://www.researchgate.net/profile/Bireswar-Banik?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19)**,** [**Abhijit Sarma**](https://www.researchgate.net/profile/Abhijit-Sarma?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), **2021** The paper addresses phishing detection challenges by focusing on lexical features of URLs, employing feature selection to enhance accuracy. It evaluates various feature combinations using machine learning techniques, aiming to identify the most effective approach for phishing and legitimate URL classification**.** The study focuses on phishing detection through lexical features in URLs. Employing feature selection. It evaluates various feature combinations and assesses performance using machine learning techniques for accurate phishing and legitimate URL classification, emphasizing network security.Lexical features of URLs from different combination and datasets. Effective use of lexical features, machine learning, and feature selection; optimized accuracy with Random Forest across diverse datasets, promising for combating evolving phishing tactics.Limited insight into specific lexical features; absence of detailed methodology; unclear generalization to evolving phishing tactics; potential dataset bias.

**[3] Title: Phishing Detection System Through Hybrid Machine Learning Based on URL**

**Author: Abdul karim samir brahim belhaouari, mobeenshahroz, khabib mustofa, ands. Ramanakumarjoga, 2023** The study emphasizes phishing attacks as a severe cybercrime and proposes a machine learning-based defense using a hybrid LSD model.Utilizing various machine learning algorithms and evaluation metrics, the proposed approach outperforms others in preventing phishing attacks with high accuracy and efficiency.The proposed system employs machine learning models including decision tree, random forest, support vector classifier, and a hybrid LSD model for efficient phishing URL detection.Using canopy feature selection and Grid Search Hyperparameter Optimization, the proposed approach outperforms other models, demonstrating superior precision, accuracy, recall, F1-score, and specificity.Effective null value handling. Integration of diverse ML modelsLack of details on dataset size and characteristics. Limited explanation on the choice of models.

**[4] Title: Phishing Websites detection Based on Hybrid Model of Deep Belief Network and Support Vector Machine, 2020** The paper proposes a hybrid model combining Deep Belief Network and Support Vector Machines to address financial crimes employing technical methods.The model achieves impressive results with 99.96% accuracy, 99.94% precision, and outperforms other comparison models in phishing URL detection.The paper proposes a hybrid model for detecting phishing websites by combining Deep Belief Network (DBN) and Support Vector Machine (SVM). It employs URL filtering, extracts features, and achieves high accuracy (99.96%) using a large dataset, outperforming other models. DBN, SVM and Feature Extraction This suggests the effectiveness of the hybrid approach in extracting deeper features and improving phishing detection accuracy, precision and false negative rate. Leverages hybrid approach for robust phishing URL detection, combining feature-rich DBN for extraction and SVM for accurate classification.

**[5] Title: PhiDMA- A Phishing detection model with multi-filter approach Author: G**unikhan Sonowal, K.S. Kuppusamy, **2020** The abstract introduces a multilayer phishing detection model, PhiDMA, addressing the inadequacy of single filter methods in countering phishing attempts.PhiDMA incorporates five layers, including auto-upgrade whitelist, URL features, lexical signature, string matching, and accessibility score, achieving 92.72% detection accuracy. PhiDMA, a multilayer model, combats phishing with components like Auto-upgrade whitelist, URL features, Lexical signature, String matching, and Accessibility Score.The approach attains a robust 92.72% accuracy in identifying phishing sites, showcasing its effectiveness in cyber threat detection**.** DBN, SVM and Feature Extraction**.** This suggests the effectiveness of the hybrid approach in extracting deeper features and improving phishing detection accuracy, precision and false negative rate.Leverages hybrid approach for robust phishing URL detection, combining feature-rich DBN for extraction and SVM for accurate classification.

**[6] Title: A Multivocal Literature Review on Growing Social Engineering Based Cyber-Attacks/Threats During the COVID-19 Pandemic: Challenges and Prospective Solutions, 2020** The study explores social engineering cyber-attacks during the COVID-19 pandemic, highlighting techniques like phishing, scamming, and their impact on critical infrastructure.Findings include the prevalence of fake emails, websites, and mobile apps, along with ransomware and trojans, targeting healthcare and posing economic challenges. from the researcher and practitioner communities by using the latest technology, such as artificial intelligence, blockchain, and big data analytics.The paper proposes a hybrid model for detecting phishing websites by combining Deep Belief Network (DBN) and Support Vector Machine (SVM). It employs URL filtering, extracts features, and achieves high accuracy (99.96%) using a large dataset, outperforming other models.This suggests the effectiveness of the hybrid approach in extracting deeper features and improving phishing detection accuracy, precision and false negative rate.Leverages hybrid approach for robust phishing URL detection, combining feature-rich DBN for extraction and SVM for accurate classification.

**[7] Title: A survey of phishing attacks: Their types, vectors and technical approaches Author: K. L. Chiew, K. S. C. Yong, and C. L. Tan, 2018** The paper provides a comprehensive review of past and current phishing approaches, detailing their mediums, vectors, and combinations used by phishers. The review aims to enhance understanding, raise awareness, and guide the development of effective anti-phishing systems, benefiting developers and policymakers. This comprehensive review examines the evolution of phishing approaches, detailing the mediums and vectors used by attackers. It explores combinations of techniques and aims to enhance understanding for developing a robust anti-phishing system.It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website.

**[8] Title: Machine Learning Techniques for detection of website phishing: A review for promises and challenge Author:** **A. Odeh, I. Keshta, and E. Abdelfattah, 2021** The paper reviews phishing website detection methods, emphasizing Machine Learning (ML) techniques like Random Forest, SVM, and deep learning for improved performance.Identified challenges include overfitting and the need for sufficient training data, emphasizing the importance of user awareness and proposing automated anti-phishing solutions**.** The proposed solution addresses phishing attacks on websites, employing Machine Learning (ML) techniques such as Random Forest, Support Vector Machine (SVM), Naïve Bayes, and Ada Boosting. While identifying challenges in ML, it emphasizes user awareness and proposes an automated phishing detection solution.Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), and Ada Boosting these are some methodologies used here.It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website.

**[9] Title: A survey of machine learning-based solutions for phishing website detection Author:** **L. Tang and Q. H. Mahmoud, 2021** The paper highlights the significance of network security in the Internet era, focusing on the evolving threat of phishing and the need for advanced detection methods.It conducts a comprehensive survey on phishing website detection, emphasizing the role of machine learning in predicting and improving accuracy against emerging phishing links. This paper offers a state-of-the-art survey on methods for phishing website detection. It starts with the life cycle of phishing, introduces common anti-phishing methods, mainly focuses on the method of identifying phishing links, and has an in-depth understanding of machine learning-based solutions, including data collection, feature extraction, modeling, and evaluation performance.This paper provides a detailed comparison of various solutions for phishing website detection. It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[10] Title: A Survey of URL-Based Phishing Detection Author:** **E. S. Aung, C. T. Zan, and H. Yamana, 2022** The paper addresses the growing threat of cyber phishing, with a focus on URL-based detection techniques due to their significance and potential for faster processing.Through a comprehensive survey, the paper analyzes diverse URL-based features, detection techniques, and their performance on various datasets to enhance phishing prevention systems. The proposed solution focuses on combating cyber phishing by surveying URL-based phishing detection techniques. Emphasizing URL's significance, it examines diverse detection mechanisms, analyzes performance across datasets, and advocates improved URL-based phishing detection systems.Naïve Bayes, Support Vector Machine, Random Forest, Convolutional Neural Network are some methodologies used for making the models accurate as much as possible.It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[11] Title: A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites Author: M. D. Bhagwat; P. H. Patil; T. S. Vishawanath, 2021** The paper explores the dynamic and nuanced challenge of real-time phishing website detection, proposing a model based on fuzzy logic and machine learning algorithms.The model utilizes 30 features to assess phishing websites with high accuracy, addressing the complexities through smooth logic and an open, intelligent approach. The proposed solution uses fuzzy logic and machine learning to build a dynamic, real-time phishing detection model analyzing 30 website features from a live dataset for high accuracy. It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[12] Title: Detecting Phishing Websites Through Domain and Content Analysis Author:** **Cristian Pascariu; Ioan C. Bacivarov, 2021** The paper addresses security incidents by proposing a novel solution for detecting phishing websites through domain and content analysis.Detection involves identifying similarities to legitimate services in the domain and analyzing content for elements indicative of credential theft. The proposed solution combines domain similarity checks and content analysis to identify phishing websites. Domains mimicking legitimate sites and those containing suspicious credential forms.It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[13] Title: SenseInput: An Image-Based Sensitive Input Detection Scheme for Phishing Website Detection Author: Shih-Chun Lin; Pang-Cheng Wl; Hong-Yen Chen; Tomohiro Morikawa; Takeshi Takahashi, 2022** The paper addresses evolving phishing threats by proposing SenseInput, a hybrid deep learning model, achieving high accuracy (96.94%) in detecting sensitive inputs. SenseInput's approach, incorporating statistical and sensitive input features, outperforms previous methods with a 98.48% f1-score on validation datasets. The proposed solution, SenseInput, employs hybrid deep learning models to enhance the detection of sensitive inputs and information on phishing websites.To also Achieving high f1-scores and outperforming previous approaches in phishing detection.Accuracy is 98.48 and 95.87**.** In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[14] Title: A Lightweight Phishing Website Detection Algorithm by Machine Learning Author: Chenyu Gu, 2021** The paper addresses the challenges of increased data and reduced detection speed in phishing website detection, proposing a lightweight framework. The framework utilizes Minhash signatures for fast URL matching, employing both similarity detection and machine learning-based intention detection for effective phishing identification. The proposed solution introduces a lightweight framework for phishing website detection, utilizing Minhash signature for faster URL matching. It combines similarity detection for suspicious sites and machine learning-based intention detection to identify phishing websites efficiently. It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**[15] Title:** **AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites Author:** **Yazan Ahmad Alsariera; Victor Elijah Adeyemo; Abdullateef Oluwagbemiga Balogun, 2021** The study addresses the evolving nature of phishing attacks, proposing four meta-learner models based on extra-tree classifiers for efficient and accurate detection.The models achieve high accuracy (not lower than 97%) and low false-positive rates, outperforming existing machine learning-based models, recommending their adoption for phishing detection. The proposed solution addresses phishing attacks using four meta-learner models (ABET, BET, RoFBET, LBET) developed with the extra-tree base classifier. Achieving over 97% accuracy, it outperforms existing models, advocating meta-learners for phishing attack detection. Meta-learner models (AdaBoost-Extra Tree (ABET), Bagging - Extra tree (BET), Rotation Forest - Extra Tree (RoFBET) and LogitBoost-Extra Tree (LBET)) developed using the extra-tree base classifier.It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

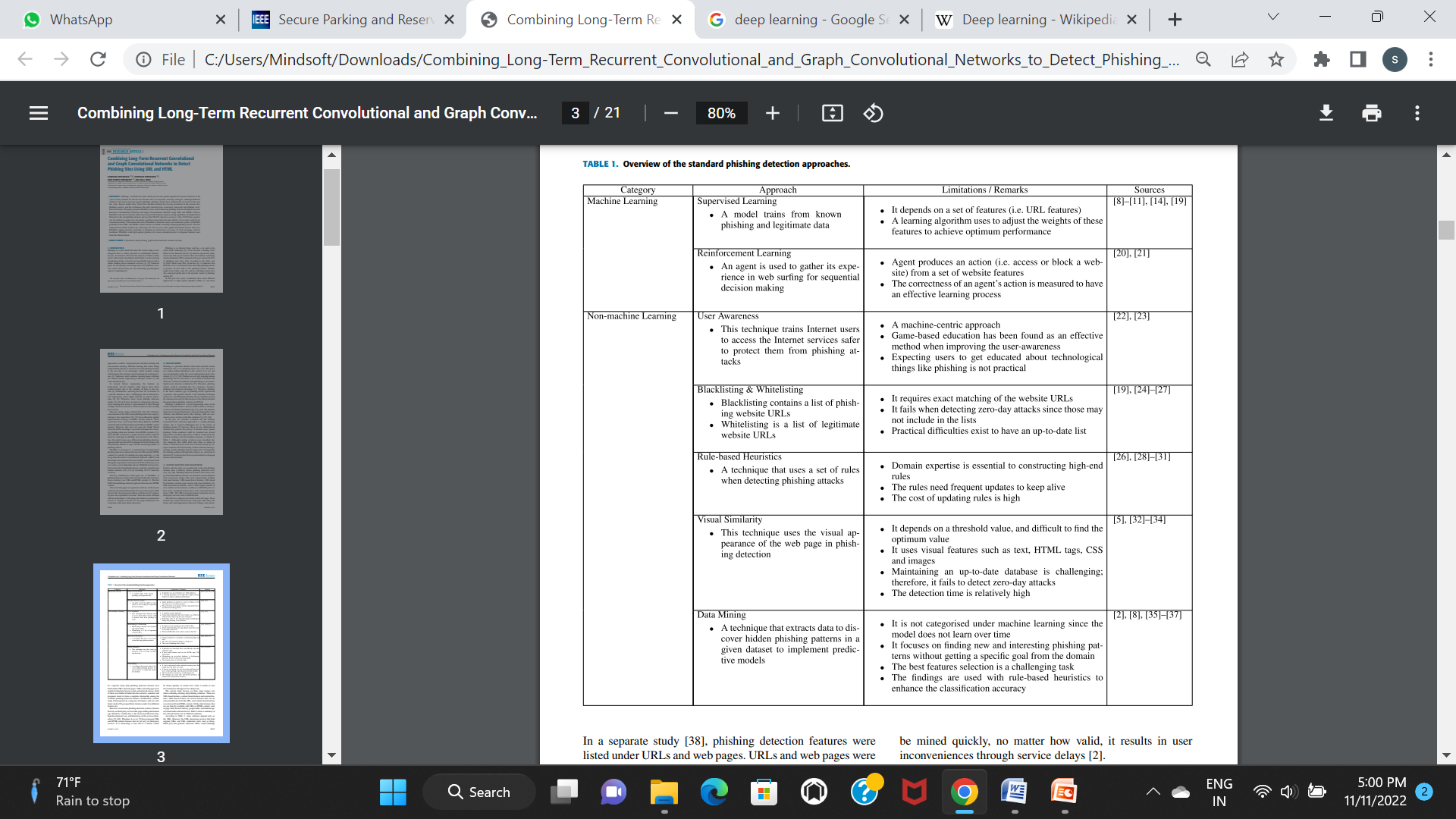
**[16] Title:** **An Ensemble Method for Phishing Websites Detection Based on XGBoost Author: Jiaqi Gu; Hui Xu, 2022** The paper addresses the need for phishing website detection, presenting an ensemble model, XGBoost, combining Random Forest and K-Nearest Neighbors, achieving 99.74% accuracy on training and 96.44% on testing data. The proposed method effectively detects phishing websites, suggesting its potential integration into applications or web extensions for enhanced user privacy protection. The proposed solution introduces an ensemble model for phishing website detection using URL features, specifically the XGBoost model combining Random Forest and K-Nearest Neighbors. With an accuracy of 99.74% on training data and 96.44% on testing data, it outperforms other models, offering effective protection for internet users' privacy. It classifies the phishing websites efficiently. In this paper they are considering only the URL attributes to classify the phishing website. This is not considering web HTML contents.

**CHAPTER 3**

**ANALYSIS**

**3.1 Existing System**

In the past two decades, academia and the industry researched better detection approaches to combat phishing attacks, but it seemed challenging due to the nature of phishing attacks however, there are now differentiated solution that protects the privacy of Internet users against phishing. These solutions could be clustered into several approaches, and these approaches could be categorized into machine learning and non-machine learning, as shown in Table 1.



**Limitations:**

* Most of the machine learning based existing system is implemented only using URL features.
* Most of the non machine learning based existing system is based on blocklist and non blocklist URLs are manually updated by administrator, So It is not accurate and time consuming process.
* Most of existing system is based on specific rule based so it is not suitable for new patterns of phishing attacks.

**3.2 Problem Statement**

Phishing Websites are duplicate Web pages created to mimic real Websites in-order to deceive people to get their personal information. Because of the adaptability of their tactics with little cost Detecting and identifying Phishing Websites is really a complex and dynamic problem. Feature engineering is important in phishing website detection solutions, but the accuracy of detection critically depends on prior knowledge of features. Moreover, although features extracted from different dimensions are more comprehensive, a drawback is that extracting these features requires a large amount of time. To overcome these problems we need an efficient phishing website detection system.

**3.3 Proposed System**

In the proposed study, machine learning algorithms were used with the features of the URL to solve classification problems. Effective features for training purposes were selected based on an effective phishing detection mechanism. The proposed system presents a differentiated phishing detection approach called PhishDet that exercises representation learning techniques simultaneously for URL and HTML content of a web page. It combines two separate models named URLDet and HTMLDet that were modelled using ML techniques to process URLs and HTML contents. PhishDet performs well at present. However, the solution should be retained occasionally to be more effective in future attacks, and the retaining process may not be costly due to the advantages of the representation learning technique.

**Advantages**

* + Testing on a dataset containing millions of phishing URLs and legitimate URLs, the accuracy reaches 98.99%, and the false positive rate is only 0.59%.
  + It will support large amount of data.
  + It will analysis the phishing website based URL and web page contents.
  + It takes less computational time.

**3.4 Functional Requirements**

* **Image Acquisition**: The system should be able to receive digital medical images, such as X-rays, in standard formats (e.g., JPEG, PNG).
* **Preprocessing**: The system should perform preprocessing tasks such as noise reduction, image normalization, and enhancement to optimize the input images for fracture detection.
* **Region of Interest (ROI) Identification**: Implement algorithms to automatically identify and extract the region of interest within the images, focusing on the skeletal structures under examination.
* **Feature Extraction**: Extract relevant features from the identified regions, including but not limited to bone density, shape, and texture features.
* **Machine Learning Model**: Develop and train a machine learning model (e.g., convolutional neural network) to classify images as either normal or containing fractures.
* **Error Handling:** Implement error handling mechanisms to detect and report issues related to image processing, model inference, or system integration.
* **User Interface**: Provide a user-friendly interface for healthcare professionals to interact with the system.

**3.5 Non- Functional Requirements**

* Performance : Process images and provide results in a timely manner.
* Scalability : Handle a large number of concurrent users and image submissions.
* Reliability : Minimize the risk of errors and false positives/negatives.
* Interoperability : Ensure compatibility with various medical imaging devices and file formats.
* Usability : Provide clear and concise instructions and guidance.

**3.6 Software Requirements**

* Microsoft Windows : Microsoft Windows is a series of graphical user interface-based operating systems. It provides a user-friendly interface and a wide range of applications and services to support various computing needs.
* Python : Python is a high-level, interpreted programming language known for its readability, simplicity, and versatility.
* IDE (VS code preferred) : Visual Studio Code (VS Code) is a free, open-source source code editor developed by Microsoft.

**3.7 Hardware Requirements**

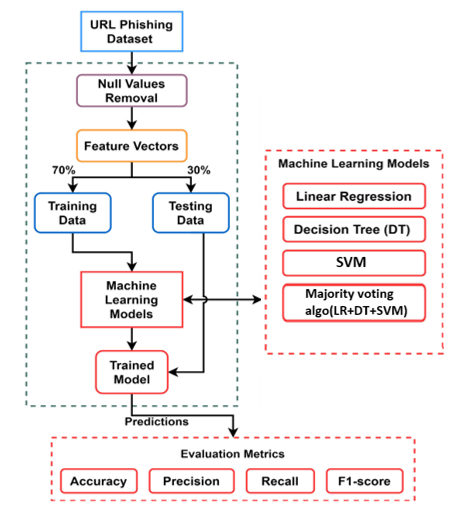
**2 GB of RAM -** It seems like your statement is about a computer or a device having 2 GB of RAM (Random Access Memory). RAM is a crucial component in a computer system, and its size impacts the system's overall performance and ability to handle multiple tasks simultaneously.

* **Dual Core processor -** A dual-core processor is a type of central processing unit (CPU) that has two independent physical cores on a single chip. Each core in a dual-core processor functions as a separate central processing unit, capable of executing its own set of instructions.
* **Hard Disk of 128 GB -** A 128 GB hard disk can store up to 128 gigabytes of data. This storage capacity is relatively modest compared to larger drives available today, such as 1 TB (terabyte) or 2 TB drives.
* **Modern Wi-Fi/Ethernet -** Modern Wi-Fi technology is based on different standards, such as 802.11ac and 802.11ax (Wi-Fi 6). These standards offer increased data transfer rates, improved range, and better performance in crowded environments with multiple connected devices.
* **I/O Devices -** Input/Output (I/O) devices are peripherals connected to a computer or electronic system to provide input or receive output. Examples include keyboards, mice, printers, monitors, and external storage devices.

**CHAPTER 4**

**DESIGN**

* 1. **System Architecture:**



**Fig 4.1 System Architecture**

Phishing detection based on URLs proposed in this study. The classification of phishing URLs was implemented using machine learning algorithms. Cybercrimes are growing with the growth of Internet architecture worldwide, which needs to provide a security mechanism to prevent an attacker from getting confidential content by breaching the network through fake and malicious URLs. A phishing dataset was used to perform the experiments. The dataset is in the form of data vectors that require null-value removal to remove unnecessary empty values. Multiple machine learning algorithms, such as decision tree (DT), linear regression (LR), naive Bayes (NB), random forest (RF), gradient boosting machine (GBM), support vector classifier (SVC), K-neighbors classifier, and the proposed hybrid model (LR+SVC+DT) LSD with soft and hard voting were used based on functional features, as shown in Figure 4.1. To improve the prediction results, a crossvalidation technique with grid search hyper-parameter tuning based on canopy feature selection was designed using the proposed LSD hybrid model. Finally, predictions were made to classify the phishing URLs and evaluate their performance in terms of accuracy, precision, recall, specificity, and F1- score.

* 1. **Data Flow Diagram:**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing.

**Level:0**

Website Dataset

Phishing/Legitimate

**Fig 4.2 Dataflow Diagram**

**Level 0** Describes the overall process of this project. we are passing website dataset as a input the system will classify the website is phishing or legitimate using the ML model.

**4.3 UseCase Diagram**

**Level: 1**

Dataset

Features

encode

Clean

features

**Fig 4.3 UseCase Diagram**

**Level 1** Describes the first stage process of this project. we are passing website dataset as a input the system will perform the preprocess and extract the important features. The process begins with the input of an URL, which is then preprocessed to remove noise and enhance the image quality. The preprocessed image is then encoded into a numerical format using a feature extraction algorithm. The extracted features are then cleaned to remove any irrelevant or redundant information. Finally, the cleaned features are used to train a machine learning model, which can be used for various applications such as object recognition or image classification. The diagram shows the step-by-step process of encoding and extracting features from an image, highlighting the importance of each step in achieving accurate and reliable results.

**4.4 Activity Diagram:**

**Level: 2**

Legitimate

Classification score ?

Features

Trained Data

-ve

Phishing

+ve

**Fig 4.4 Activity Diagram**

**Level 2** Describes the final stage process of this project. we are passing extracted features from level 1and trained data as a input the system will classify the given website is phishing or legitimate using ML Model. The diagram starts with the input of the dataset, which is then processed through several stages. The first stage involves reading the data, followed by training the data using algorithms. The next stage involves reading the trained data to generate a score, which is then used to classify the data. The final stage involves reading the classified data to generate a score. The diagram also includes a question mark, which could represent a decision point or a potential issue that needs to be addressed. Overall, the diagram shows the flow of data through the various stages of processing and classification.

**4.5 Class Diagram:**

**Level: 3**

**User**

dataset: .csv

Load dataset()

View Result()

**System**

dataset:.csv

Read Dataset()

Extract features()

Classify()

**Fig 4.5 Class Diagram**

**Level 3**  A user interacts with a system by loading data, viewing results, and extracting features. The system reads the data, classifies it, and extracts features. At the top, there is a user who interacts with the system. The system consists of a database, which stores and retrieves data. The data is processed by the system's load and read data components, which extract features and classify the data. The extracted features are then passed to the view result component, which presents the data to the user. The system also includes an extract features component that processes the data and an extract features view component that displays the processed data. The data flow between these components is represented by arrows, indicating the movement of data between them. Overall, the diagram illustrates a system that processes and presents data to the user.

**4.6 Sequence Diagram:**

**Level: 4**

Result()

Classify()

Extract Features()

Read()

Load dataset()

User

System

Preprocess()

Train()

**Fig 4.6 Sequence Diagram**

**Level 4** The sequence diagram will determine the users states (active/ inactive) with our applications. The system consists of a load dataset, a process, and a train. The load dataset is responsible for loading data into the system, while the process component is responsible for processing the data. The train component is responsible for training the system using the processed data.The data flow in the system starts with the load dataset, which loads the data into the system. The data is then processed by the process component, which transforms the data into a format suitable for training. The processed data is then passed on to the train component, which uses it to train the system. The trained system can then be used to perform various tasks, such as making predictions or classifications. The diagram provides a clear representation of the components and data flow within the system, highlighting the interconnectedness of the different components and their roles in the overall system.

**CHAPTER 5**

**IMPLEMENATION**

**5.1 Modules Description**

* **Data Collection Module**: This module is responsible for gathering datasets containing both legitimate and malicious URLs. It may involve web scraping, API integration with threat intelligence services, or accessing publicly available datasets.
* **Feature Extraction Module**: This module focuses on extracting relevant features from the URLs that can be used by machine learning algorithms for classification. Features may include URL length, presence of suspicious keywords, domain age, presence of HTTPS, etc.
* **Model Training Module**: This module encompasses the training of machine learning models such as Random Forest, Gradient Boosting, Support Vector Machines (SVM), and Neural Networks. It involves data preprocessing, feature selection, model initialization, hyperparameter tuning, and cross-validation.
* **Model Evaluation Module**: This module involves evaluating the trained models' performance using various metrics such as accuracy, precision, recall, F1-score, ROC-AUC curve, and confusion matrices. It may also include techniques like k-fold cross-validation to ensure robustness of the model.
* **Real-time Detection Module**: This module is responsible for deploying the trained model into a production environment where it can classify URLs in real-time. It involves integrating the model with web servers, APIs, or browser extensions to detect potentially malicious URLs as users access them.

Each module plays a crucial role in the overall phishing detection system, from data collection to real-time detection, ensuring the system's effectiveness and reliability in identifying malicious URLs.

**5.2 Data Preprocessing Module**

The data preprocessing module is a crucial initial step in the phishing detection project, aimed at preparing the dataset of URLs for subsequent machine learning tasks. This module ensures that the dataset is well-structured, balanced, and enriched to facilitate effective model training. The following steps are undertaken in this module:

* **Data Analysis**: The module starts with an in-depth analysis of the dataset to understand the distribution of legitimate and malicious URLs. This analysis provides insights into class imbalance, data quality issues, and distribution characteristics, ensuring a representative dataset for model training.
* **Visualization**: Visual representations of sample URLs from both legitimate and malicious classes are generated to gain a visual understanding of the dataset's composition. This visualization aids in identifying any patterns, anomalies, or challenges present in the data, such as skewed class distributions or common features in malicious URLs.
* **Feature Extraction**: Relevant features are extracted from the URLs to represent them numerically for machine learning algorithms. Features may include URL length, presence of suspicious keywords, domain age, character n-gram frequencies, and others. Feature extraction techniques ensure that important information is captured effectively for model training.
* **Data Augmentation**: Augmentation techniques are applied to enhance the dataset diversity and mitigate overfitting. Techniques such as URL tokenization, character-level perturbations, and synthetic data generation can be utilized to introduce variations in the dataset, improving the model's ability to generalize to unseen URLs.
* **Data Balancing:** Class imbalance is addressed by applying sampling techniques such as oversampling, undersampling, or synthetic minority oversampling technique (SMOTE). Balancing the dataset ensures that the model is trained effectively on both legitimate and malicious URLs, preventing bias towards the majority class.
* **Data Encoding:** Categorical features within the URLs, such as domain extensions or protocol types, are encoded into numerical representations using techniques like one-hot encoding or label encoding. This encoding process prepares the data for input into machine learning algorithms, ensuring compatibility and effectiveness in model training.
* **Data Splitting:** The dataset is split into training, validation, and testing sets to evaluate the model's performance accurately. Common splitting ratios such as 70-15-15 or 80-10-10 are employed to ensure sufficient data for training, validation, and testing while maintaining data integrity.

By meticulously preparing and preprocessing the dataset, this module lays the foundation for training robust machine learning models capable of accurately detecting malicious URLs and distinguishing them from legitimate ones.

**5.3 Model Training Module**

The Model Training Module is a fundamental component in the development of the phishing detection system, focusing on training machine learning models to accurately classify URLs as legitimate or malicious. This module utilizes a range of machine learning algorithms to analyze URL characteristics and make predictions regarding their malicious intent. Through this module, diverse models are trained on a dataset comprising both legitimate and malicious URLs, aiming to optimize their performance for accurate classification.

This module involves the training of several machine learning models, including Random Forest,Decision Tree and Support Vector Machines (SVM) and Majority voting(RF+DT+SVM). Each model offers unique architectural features and learning capabilities, contributing to the overall effectiveness of the phishing detection system.

The training process consists of the following steps:

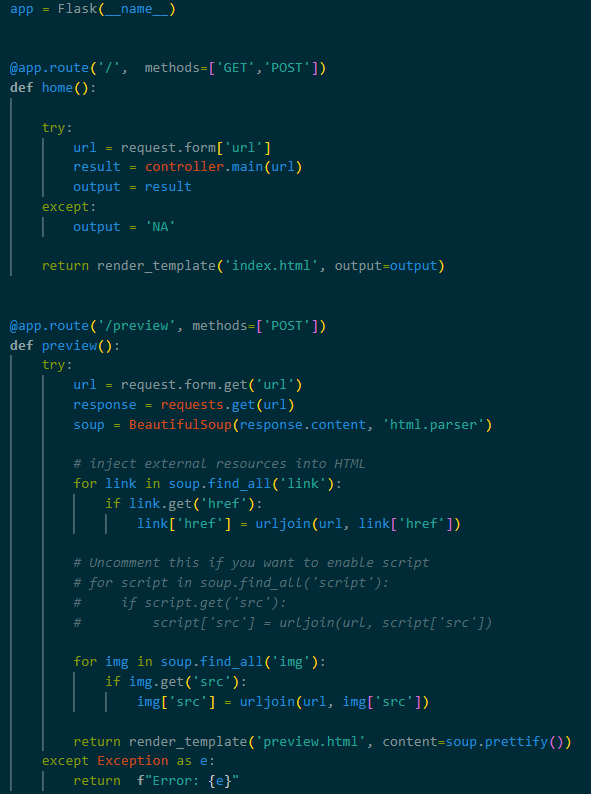
1. **Data Preprocessing:** The dataset of URLs is preprocessed to extract relevant features and prepare it for model training. This may involve feature extraction, data encoding, balancing classes, and splitting the dataset into training, validation, and test sets.
2. **Model Architecture Definition:** The architectures of the selected machine learning models are defined, specifying the number of layers, neurons, activation functions, and other parameters. Each model architecture is tailored to the characteristics of the phishing detection task and the dataset.
3. **Model Compilation:** The models are compiled with appropriate loss functions and optimizers, tailored to the classification task of distinguishing between legitimate and malicious URLs. This step ensures that the models are optimized for learning and making accurate predictions.
4. **Training:** The models are trained on the preprocessed dataset using training algorithms such as stochastic gradient descent (SGD) or Adam. During training, the models learn to classify URLs based on their features, adjusting their parameters to minimize the classification error.
5. **Hyperparameter Tuning:** The hyperparameters of the models are fine-tuned through experimentation to optimize their performance. This may involve adjusting parameters such as learning rate, batch size, and regularization techniques to improve the models' accuracy and generalization ability.
6. **Evaluation:** The trained models are evaluated on the validation set to assess their performance using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC curve. This evaluation helps select the best-performing models for deployment in the phishing detection system.

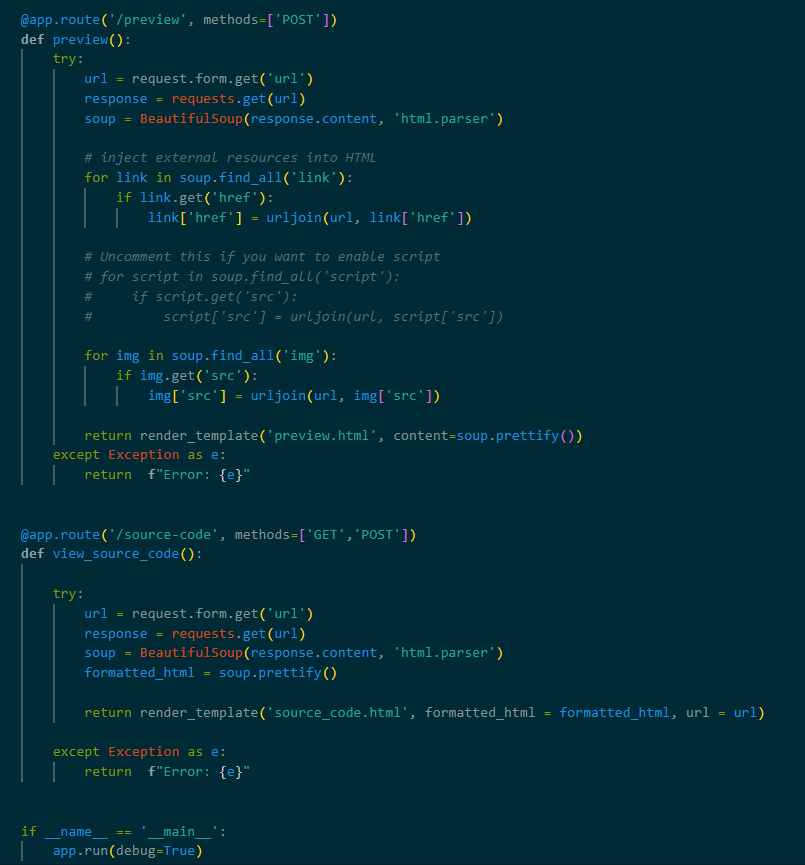
By rigorously training and evaluating multiple machine learning models, this module establishes a foundation for building an effective phishing detection system capable of accurately identifying malicious URLs and protecting users from phishing attacks. Each model contributes unique insights and strengths to the overall system, enhancing its robustness and reliability in detecting malicious online threats.

**5.4 Code Snippets**

**5.4.1 Data preprocessing module**

The presented code snippet delves into the data preprocessing module, a critical stride in developing machine learning models for phishing website detection. It establishes the dataset's structural organization and implements pivotal preprocessing techniques, akin to essential data augmentation practices such as feature extraction and encoding. Each preprocessing function is customized to cater to the nuances of training, validation, and test datasets, ensuring meticulous data preparation. Furthermore, the instantiation of data generators expedites the batch processing workflow during both model training and evaluation phases. This systematic methodology guarantees the dataset's readiness by optimizing its format and enriching its contents, thereby laying a robust groundwork for subsequent model training and evaluation endeavors.

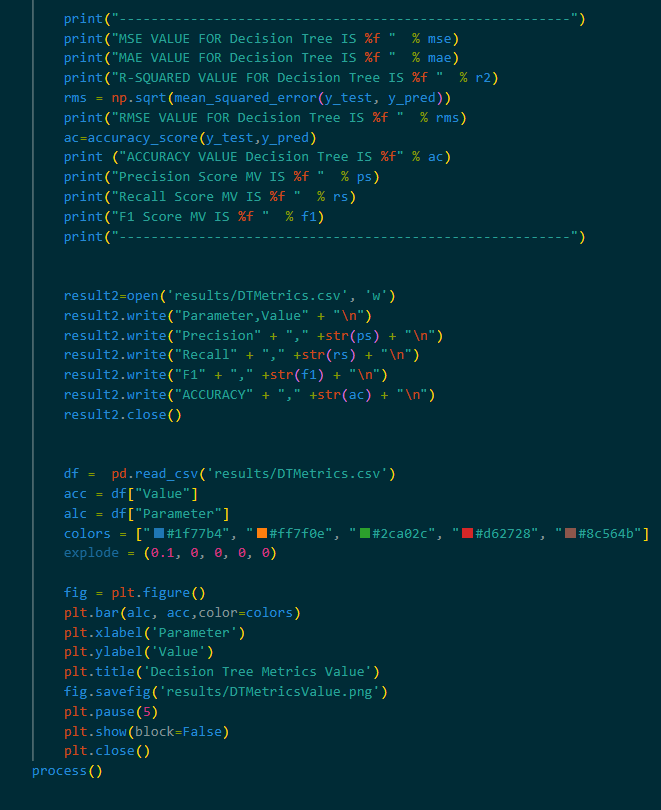
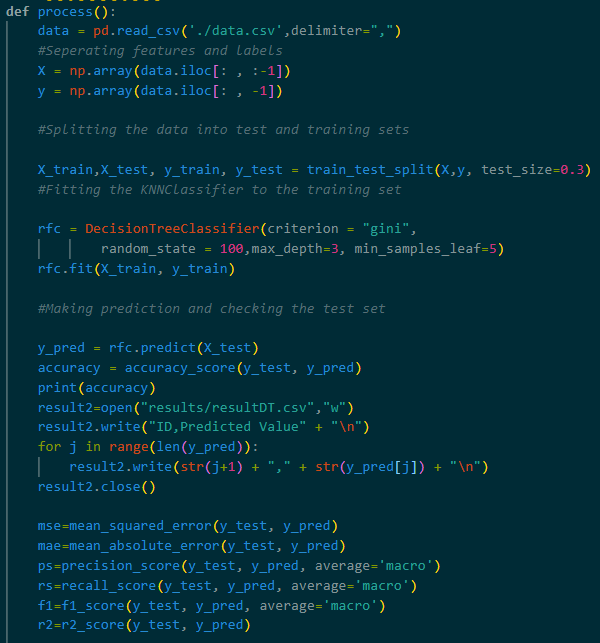




**Fig 5.4.1 Code snippet of Data Processing module**

**5.4.2 Decision Tree model**

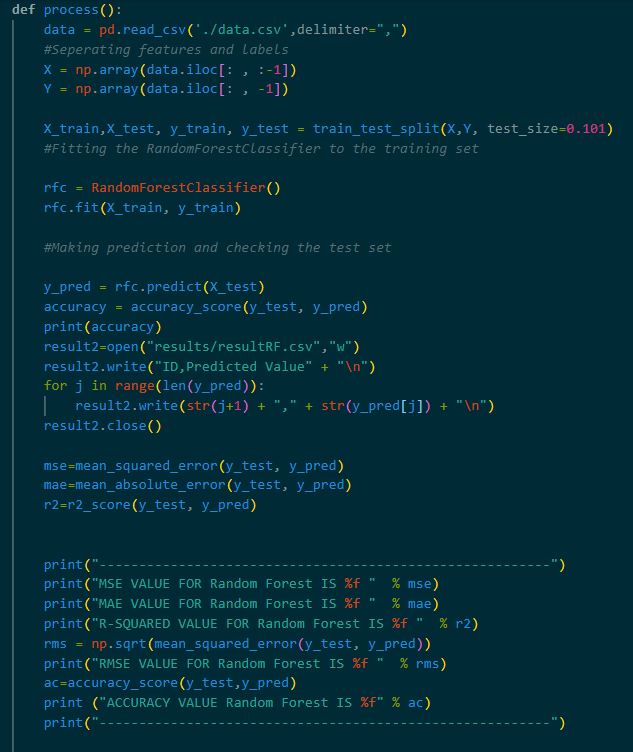
This code snippet initializes a decision tree model for phishing URL detection using machine learning. The decision tree model is chosen due to its simplicity and interpretability, making it suitable for classifying URLs as legitimate or malicious based on various features. The model is trained using a dataset containing features extracted from URLs, such as URL length, presence of suspicious keywords, and domain age. The decision tree model is then compiled with appropriate parameters, such as the criterion for splitting nodes (e.g., Gini impurity or information gain), to optimize its performance during training. Additionally, hyperparameters may be fine-tuned through experimentation to improve the model's accuracy and generalization ability. Overall, this decision tree model serves as an effective tool for detecting phishing URLs and safeguarding users against online threats.



**Fig 5.4.2 Code snippet of Decision Tree Model**

**5.4.3 Random Forest model**

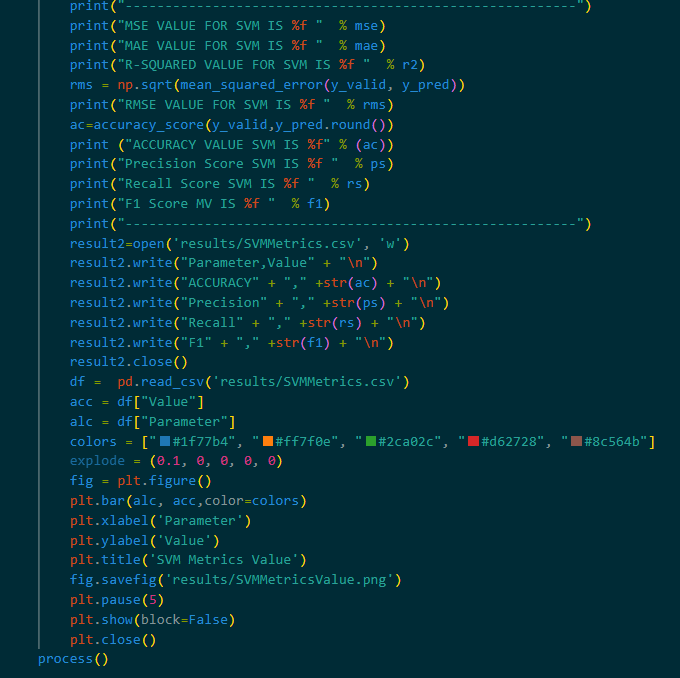
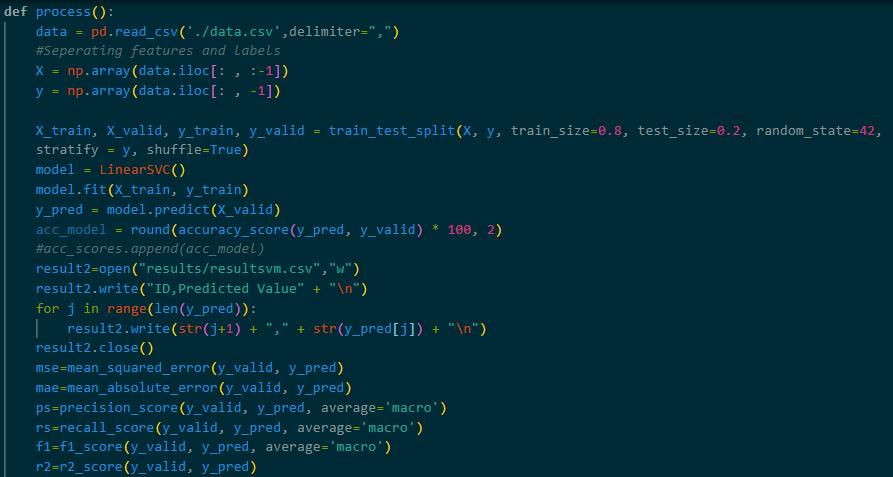
This code snippet initializes a Random Forest model for phishing URL detection using machine learning. The Random Forest model is chosen for its versatility and robustness in handling high-dimensional feature spaces, making it well-suited for classifying URLs as legitimate or malicious based on various features. The model is trained using a dataset containing features extracted from URLs, such as URL length, presence of suspicious keywords, and domain age. Unlike deep learning models that require extensive computational resources, Random Forest can efficiently handle large datasets and complex feature interactions without the need for extensive hyperparameter tuning. The model is then compiled with appropriate parameters, such as the number of decision trees and maximum depth, to optimize its performance during training. Additionally, feature importance analysis can be conducted to identify the most discriminative features for phishing URL detection. Overall, this Random Forest model serves as a powerful tool for detecting phishing URLs and protecting users from online threats.



**Fig 5.4.3 Code Snippet of Random Forest model**

**5.4.3 Support Vector Machine model**

This code snippet initializes a Support Vector Machine (SVM) model for phishing URL detection using machine learning. SVM is chosen for its effectiveness in handling high-dimensional feature spaces and its ability to find the optimal hyperplane that best separates different classes of data. The model is trained using a dataset containing features extracted from URLs, such as URL length, presence of suspicious keywords, and domain age. SVMs are particularly well-suited for binary classification tasks like phishing URL detection, where the goal is to accurately distinguish between legitimate and malicious URLs. The model is then compiled with appropriate parameters, such as the choice of kernel function (e.g., linear, polynomial, or radial basis function), to optimize its performance during training. Additionally, hyperparameter tuning can be performed to fine-tune the model's parameters and improve its accuracy and generalization ability. Overall, this SVM model serves as a robust and reliable tool for detecting phishing URLs and protecting users from online threats.



**Fig 5.4.4 Code Snippet of SVM model**

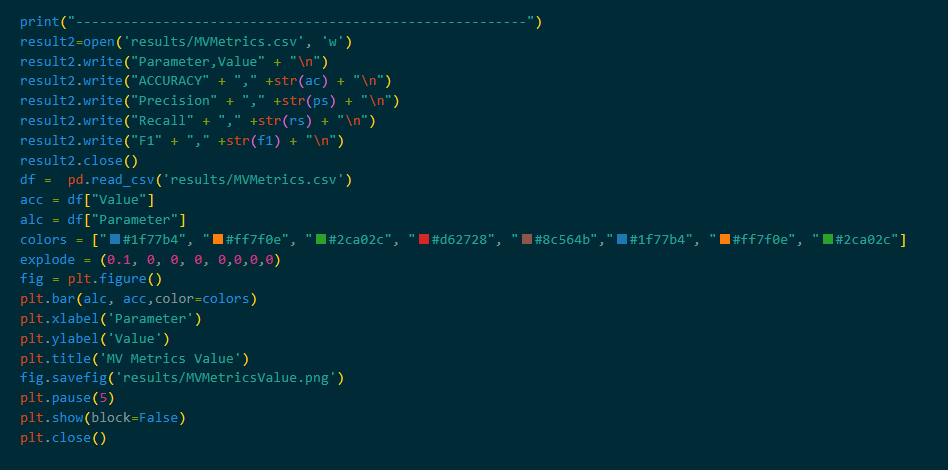
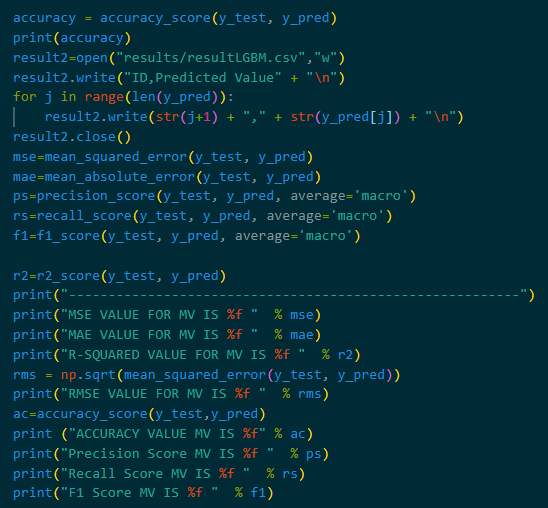
**5.4.5 Majority Voting Model**

This code snippet implements a Majority Voting ensemble method for phishing URL detection using machine learning. The Majority Voting ensemble technique combines the predictions of multiple base classifiers to make a final decision. In this context, several base classifiers, such as Decision Trees, Random Forests, Support Vector Machines (SVM), and Neural Networks, can be trained on features extracted from URLs.

First, each base classifier is trained independently on the training dataset. Then, during the prediction phase, each classifier makes its prediction for whether a URL is legitimate or malicious. The final prediction is determined by a majority vote among the predictions of all the base classifiers. If the majority of classifiers predict a URL as malicious, it is classified as such; otherwise, it is classified as legitimate.

This ensemble approach leverages the diversity of base classifiers to improve overall prediction accuracy and robustness, as it aggregates multiple models' insights. Additionally, it helps mitigate overfitting and reduces the risk of bias associated with individual classifiers. Overall, the Majority Voting ensemble method serves as a powerful tool for detecting phishing URLs and enhancing users' online security





**Fig 5.4.4 Code Snippet of Majority Voting model**

**CONCLUSION**

In conclusion, this study will effectively address the challenge of detecting malicious phishing URLs through a comprehensive machine learning approach. The investigation will employ a diverse array of algorithms, including decision trees, linear regression, random forests, support vector machines, gradient boosting machines, K-nearest neighbors, naive Bayes, and hybrid models with soft and hard voting. To enhance model performance, canopy feature selection, cross-fold validation, and grid search hyperparameter optimization techniques will be judiciously applied within the LSD ensemble framework. The results will conclusively demonstrate that this meticulously designed approach will successfully achieve its aim with remarkable efficiency, offering a promising solution to combat phishing attacks and safeguard online security.

**BIBLIOGRAPHY**

[1] Using Classical Machine Learning For Phishing Websites Detection Form URLs Author: Iman Akour Noha Alnazzawi, Ahmad Aburayya, Raghad Alfaisal, Sultan Idris Said Salloum, 2022

[2] Title: Lexical Feature Based Feature Selection And Phishing URL Classification Using Machine Learning techniques Author: [Bireswar Banik](https://www.researchgate.net/profile/Bireswar-Banik?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Abhijit Sarma](https://www.researchgate.net/profile/Abhijit-Sarma?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), 2021

[3] Phishing Detection System Through Hybrid Machine Learning Based on URL

Author: Abdul karim samir brahim belhaouari, mobeenshahroz, khabib mustofa, ands. Ramanakumarjoga, 2023

[4] Phishing Websites detection Based on Hybrid Model of Deep Belief Network and Support Vector Machine, 2020

[5] PhiDMA- A Phishing detection model with multi-filter approach Author: Gunikhan Sonowal, K.S. Kuppusamy , 2020

[6] A Multivocal Literature Review on Growing Social Engineering Based Cyber-Attacks/Threats During the COVID-19 Pandemic: Challenges and Prospective Solutions, 2020

[7] A survey of phishing attacks: Their types, vectors and technical approaches Author: K. L. Chiew, K. S. C. Yong, and C. L. Tan, 2018

[8] Machine Learning Techniques for detection of website phishing: A review for promises and challenge Author: A. Odeh, I. Keshta, and E. Abdelfattah, 2021

[9] A survey of machine learning-based solutions for phishing website detection Author: L. Tang and Q. H. Mahmoud, 2021

[10] A Survey of URL-Based Phishing Detection Author: E. S. Aung, C. T. Zan, and H. Yamana,

2022

[11] A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites Author: M. D. Bhagwat; P. H. Patil; T. S. Vishawanath, 2021

[12] Detecting Phishing Websites Through Domain and Content Analysis Author: Cristian Pascariu Ioan C. Bacivarov , 2021

[13] SenseInput: An Image-Based Sensitive Input Detection Scheme for Phishing Website Detection Author: Shih-Chun Lin; Pang-Cheng Wl; Hong-Yen Chen; Tomohiro Morikawa; Takeshi Takahashi, 2022

[14] A Lightweight Phishing Website Detection Algorithm by Machine Learning Author: Chenyu Gu, 2021

[15] AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites Author: Yazan Ahmad Alsariera; Victor Elijah Adeyemo; Abdullateef Oluwagbemiga Balogun, 2021

[16] An Ensemble Method for Phishing Websites Detection Based on XGBoost Author: Jiaqi Gu; Hui Xu, 2022.

[17] PhishSim: Aiding Phishing Website Detection With a Feature-Free Tool Authors: Rizka Widyarini Purwanto; Arindam Pal; Alan Blair; Sanjay Jha.Published in: IEEE Transactions on Information Forensics and Security ( Volume: 17).

[18] A Deep Learning-Based Framework for Phishing Website Detection Authors: Lizhen Tang; Qusay H. Mahmoud. Published in: IEEE Access ( Volume: 10) 2021

[19] Multilayer Stacked Ensemble Learning Model to Detect Phishing Websites Authors: Lakshmana Rao Kalabarige; Routhu Srinivasa Rao; Ajith Abraham; Lubna Abdelkarei..Published in: IEEE Access ( Volume: 10), 2022.

[20] PDGAN: Phishing Detection With Generative Adversarial Networks Authors: Saad Al-Ahmadi; Afrah Alotaibi; Omar Alsaleh. Published in: IEEE Access ( Volume: 10), 2022.