A Mini project Synopsis on

PhishAlert: A Phishing Detector

T.E.- Computer Science and Engineering (Data Science)

Submitted By

Tanaya Patil	**21107017**
Ayush Mistry	**21107029**
Mayank Kumar	**21107016**
Sahil Mujumdar	**21107050**

Under The Guidance Of

Prof. Poonam Pangarkar

DEPARTMENT OF CSE (DATA SCIENCE)

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic year:2023-24

CERTIFICATE

This to clarify that the Mini Project Report on **PhishAlert: A Phishing Detector** has been submitted by **Tanaya Patil(21107017), Ayush Mistry(21107029), Sahil Mujumdar (21107050) and Mayank Kumar(21107016)** who are a Bonafede students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in **CSE(DATA SCIENCE)**, during the academic year **2023-2024** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Poonam Pangarkar

Guide

Prof. Anagha Aher	Dr. Uttam D. Kolekar
Head Department of CSE (Data Science)	Principal

External Examiner(s)

1.

2.

Place: A.P. Shah Institute of Technology, Thane

Date:

TABLE OF CONTENTS

Abstract

in the ever-evolving internet landscape, network security remains a major concern, and phishing appears as the most pervasive and insidious form of cybercrime among the myriad threats that manifest in the digital realm of the cycle necessary to enhance its continuous growth and stability. the way it works is to trick the unsuspecting into fraudulently revealing sensitive information, often ending up in financial fraud and identity theft. as technology advances and methods by malicious people using it, traditional methods of phishing detection such as blacklists and whitelists are becoming increasingly inadequate. the inability of dysfunctional methods to adapt to new phishing attempts highlights the urgent need for more proactive and proactive solutions. in this pursuit, machine learning appears as a beacon of promise, allowing for the anticipating and pre-empting of emerging threats. in machine learning, logistic regression shines as a versatile and effective tool in the cybersecurity arsenal. this paper begins by exploring the latest techniques to better identify phishing websites. it walks through the complex social issues of phishing, explains common anti-phishing strategies, delves into the realm of machine learning-driven solutions, and in particular focuses on logistic regression, and empowers machine learning in its role in data collection, feature extraction, model building, and performance evaluation we implement, strive to strengthen our defenses against ever better cyber threats while creating a digital ecosystem system integrity and security protection.

Chapter 1

Introduction

The digital landscape presents both opportunities and challenges. Phishing attacks, where malicious actors create deceptive websites to steal sensitive information, and spam messages containing malicious links, pose significant threats to online safety. Traditional methods of identifying these threats can be time-consuming and ineffective.

PhishAlert, our project, tackles these challenges by developing a web application that leverages machine learning for automated detection. Utilizing the power of Naive Bayes and Logistic Regression algorithms, PhishAlert provides a user-friendly platform to analyze websites and SMS messages.

This report dives into the details of PhishAlert, including its objectives, the chosen machine learning models, the web application development process, and the evaluation methods. We explore the inner workings of Naive Bayes and Logistic Regression, and how they are applied to identify phishing attempts in websites and spam messages via SMS. Through experimentation and analysis, we aim to demonstrate the effectiveness of these machine learning techniques in safeguarding users from online scams and malicious content. By showcasing the potential of PhishAlert, we hope to contribute to a more secure and trustworthy online experience.

This introduction focuses on both phishing site detection and spam SMS detection, making them the clear priorities of your project. It highlights the user-friendly web application aspect and sets the stage for explaining how the machine learning models work in this context.

1.1 Purpose:

The PhishAlert project serves a multitude of purposes. Firstly, it acts as a comprehensive record of the project's journey. This report details the methodologies used, the findings uncovered, and the overall outcomes achieved. By documenting the implementation of Naive Bayes and Logistic Regression for both phishing website detection and spam SMS classification, stakeholders gain a thorough understanding of the approach taken and the results obtained. This information is invaluable for internal and external audiences seeking insights into the project's strategies and accomplishments.

Secondly, the report aims to assess the effectiveness of the chosen machine learning models. Through rigorous experimentation and analysis, it strives to validate the efficacy of Naive Bayes and Logistic Regression in accurately identifying phishing websites and classifying SMS messages as spam or not spam. By presenting empirical evidence and comparative assessments, the report empowers stakeholders to make informed decisions regarding the suitability of these algorithms for real-world application in cybersecurity solutions.

Furthermore, the report serves as a valuable tool for informing decision-making processes related to cybersecurity and threat mitigation strategies. By synthesizing insights derived from the project's findings, it equips decision-makers with the knowledge necessary to evaluate the feasibility, benefits, and potential challenges associated with integrating these machine learning-driven solutions into existing security frameworks. This informed decision-making is critical for ensuring the adoption and implementation of effective cybersecurity measures.

The report also fulfils an educational purpose by disseminating knowledge and insights pertaining to machine learning-based approaches to phishing and spam detection. Through clear explanations of the methodologies employed, it seeks to broaden understanding and awareness of the technical aspects involved in cybersecurity research and practice. By demystifying complex concepts and methodologies, the report empowers a wider audience to engage with and contribute to the ongoing discourse surrounding cybersecurity and data protection.

Finally, the report aims to promote best practices in cybersecurity by highlighting effective strategies and methodologies for phishing and spam detection. By showcasing the potential of Naive Bayes and Logistic Regression, it encourages the adoption of evidence-based approaches to cybersecurity risk management. By disseminating lessons learned and insights gained throughout the PhishAlert project, the report contributes to the collective knowledge base of cybersecurity professionals and practitioners, fostering a culture of continuous improvement and innovation in the field.

1.2 Problem Statement:

The PhishAlert project addresses the pressing challenge posed by the prevalence of online phishing attacks, which pose significant risks to cybersecurity, individuals, organizations, and entire economies. These attacks exploit human vulnerabilities, employing deceptive tactics to coerce users into disclosing sensitive information such as passwords, financial credentials, or personal data. With the advancement of digital communication channels and the increasing sophistication of phishing tactics, cybersecurity professionals face formidable challenges in combatting these threats.

Traditional approaches to phishing detection, such as manual inspection or rule-based systems, often fall short in the face of evolving attack vectors and sophisticated social engineering tactics employed by cybercriminals. As a result, there is a critical need for advanced, automated solutions capable of accurately identifying and mitigating phishing threats in real-time. Machine learning presents a promising avenue for addressing this challenge, given its ability to discern complex patterns and anomalies in large datasets.

In response to this need, the PhishAlert project leverages the Multinomial Naive Bayes and Logistic Regression algorithms. These algorithms are known for their effectiveness in text classification tasks and are utilized to develop and implement machine learning-based solutions for automated phishing website detection. By analyzing textual features extracted from website content, these algorithms aim to enhance cybersecurity defenses and safeguard users against the perils of online fraud and identity theft.

The problem space addressed by the PhishAlert project extends beyond technical complexities to encompass broader societal and economic implications of phishing attacks. By highlighting the pervasive threat posed by online phishing, the project underscores the critical importance of proactive measures to combat cybercrime and protect digital assets. Through a comprehensive examination of methodologies, findings, and implications, the PhishAlert project seeks to contribute to the advancement of cybersecurity research and practice, empowering individuals and organizations to navigate the digital landscape securely amidst evolving threats

1.3 Objective:

The PhishAlert project is designed to achieve several key objectives:

1. **Methodological Documentation:** Provide a comprehensive overview of the methodologies employed throughout the project. This includes a detailed explanation of how Naive Bayes and Logistic Regression algorithms are implemented for both phishing website detection and spam SMS classification. The report will delve into the processes of data preprocessing, feature selection, model training, and evaluation for each machine learning model.
2. **Performance Analysis:** Evaluate the effectiveness of Naive Bayes and Logistic Regression in their respective tasks. This objective involves using quantitative measures like accuracy, precision, recall, and F1-score to assess the models' ability to accurately distinguish between legitimate websites/SMS messages and phishing attempts/spam messages.

3. **Comparative Assessment:** Conduct a comparative analysis of Naive Bayes and Logistic Regression to understand their relative strengths and weaknesses in the context of phishing and spam detection. The comparison will consider factors such as computational efficiency, scalability, and robustness to noisy or imbalanced data sets.
4. **Feature Importance Exploration:** Investigate the importance of individual features within the Naive Bayes and Logistic Regression models for identifying phishing websites and spam SMS messages. By analyzing feature importance scores or variable importance measures, the project aims to identify the most critical features for distinguishing legitimate content from malicious content.
5. **Real-World Application Assessment:** Assess the practical feasibility of deploying Naive Bayes and Logistic Regression for real-world phishing and spam detection scenarios. This objective involves discussing considerations such as model deployment strategies, scalability requirements, and integration with existing cybersecurity frameworks.
6. **Challenges and Limitations:** Identify and discuss the limitations and challenges encountered during the implementation of Naive Bayes and Logistic Regression in the PhishAlert project. The report will address issues like overfitting, data imbalance, and interpretability of the models.
7. **Future Research Directions:** Propose potential avenues for future research and development in the field of phishing and spam detection using machine learning techniques. This includes exploring areas for improvement, such as incorporating additional features, investigating ensemble methods, or leveraging deep learning approaches.
8. **Practical Implications for Cybersecurity:** Provide insights into the practical implications of the PhishAlert project's findings for cybersecurity professionals, organizations, and end-users. The report will discuss strategies for enhancing cybersecurity awareness, implementing proactive defense measures, and mitigating the risks associated with phishing and spam attacks.
9. **Knowledge Dissemination:** Serve as a platform for disseminating the knowledge and insights gained from the PhishAlert project to a wider audience of cybersecurity researchers, practitioners, and stakeholders. This objective aims to facilitate knowledge exchange, collaboration, and engagement in ongoing efforts to combat phishing, spam, and enhance overall cybersecurity.

1.4 Scope:

The PhishAlert project investigates the use of machine learning to combat phishing websites and spam SMS messages. It begins by outlining the methodologies employed, including Naive Bayes and Logistic Regression algorithms. The project explains data preprocessing and feature selection techniques used to prepare the data for analysis and identify the most informative aspects. This includes model training procedures aimed at developing robust detection models.

The report then assesses the performance of these models using quantitative metrics such as accuracy, precision, recall, and F1-score to measure their effectiveness in distinguishing legitimate content from malicious content. The goal is to balance correct identification of phishing and spam while minimizing false positives.

A comparative analysis between Naive Bayes and Logistic Regression examines their computational efficiency, scalability, and robustness to noisy or imbalanced data sets. This provides insights into their practical applicability for real-world cybersecurity scenarios.

Additionally, the project explores the importance of individual features in the models for distinguishing legitimate content from malicious content. By analyzing feature importance scores, the project aims to identify critical features that the models rely on for their predictions. These insights can guide future data collection and feature engineering efforts to develop even more robust detection models for combating phishing and spam.

Chapter 2

Literature Review

"A Survey of Machine Learning-Based Solutions for Phishing Website Detection" [1] taking into account the paper, the authors find that researchers and security experts have contributed a lot of successful resolutions, from list based methods and rule-based strategies to machine learning-based approaches. Various machine learning-based solutions achieved higher than 95% accuracy, which is a significant advancement. However, it is believed that the accuracy performance still has space for improvement. The shortcoming is that this needs an extra feature extraction process based on rules, and it depends on some third-party services.

"Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning" [2], here, the authors find that the MFPD (Multidimensional Feature Phishing Detection) approach is effective with high accuracy, low false positive rate, and high detection speed. Future development of their approach will consider applying deep learning to feature extraction of webpage code and webpage text. In addition, they plan to implement their approach into a plugin for embedding in a Web browser.

"Phishing website detection based on effective machine learning approach" [3] With regards to this paper, the authors used many techniques such as Decision tree Classifier, K nearest neighbors, Linear SVC classifier, Random Forest classifier, and One-class SVM classifier. They observed that Random Forest got the highest accuracy of about 96.87%. They predominantly observed that Random Forest performed better than other methods or algorithms as mentioned above. Overfitting of data is avoided, which is one of the important features. Hence Random Forest classifier is best suited to detect more accurately whether the website is phishing or not.

"Phishing websites detection using machine learning" [4] to delve deeper into this study, they adopted a classifier model that is used for detecting phishing websites in an intelligent and automated way by using a publicly available dataset. The performance of the proposed Random Forest classifier is rather high in terms of classification accuracy, F-measure, and AUC. Furthermore, their results showed that Random Forest is faster, more robust, and more accurate than the other classifiers. Random Forest's runtime is quite fast, and it can detect phishing websites in comparison to the other classifiers.

Chapter 3

Proposed System

The PhishAlert system starts by gathering a diverse dataset of both phishing and legitimate websites and SMS messages. This raw data then undergoes preprocessing to ensure its accuracy and consistency. This preprocessing might involve cleaning the data, normalizing it to a common format, and handling any missing values. Once the data is prepared, informative features are extracted that can help distinguish phishing attempts from legitimate content. These features can include website URLs, sender information in SMS messages, suspicious keywords, and even linguistic patterns. With the preprocessed data and identified features, decision tree and random forest models are trained. These models learn to recognize patterns indicative of phishing behavior in both websites and SMS messages.

The system then evaluates the performance of these trained models using metrics like accuracy, precision, recall, and F1-score. These metrics provide insights into how well the models can accurately detect phishing attempts. Finally, the most successful models are deployed in real-time environments to continuously monitor websites and incoming SMS messages for potential phishing threats. This continuous monitoring helps to proactively identify and flag suspicious activity, ultimately enhancing cybersecurity defenses.

Advantages of the PhishAlert System

The PhishAlert system offers several advantages in the fight against phishing attacks:

1. **Effective Phishing Detection:** By employing Multinomial Naive Bayes and Logistic Regression algorithms, the PhishAlert project enhances the efficacy of phishing website detection. These machine learning techniques excel in discerning subtle patterns and anomalies characteristic of phishing behavior, facilitating accurate identification and flagging of potential threats.
2. **Robustness and Reliability:** Multinomial Naive Bayes and Logistic Regression algorithms are renowned for their robustness and reliability in handling diverse datasets, effectively mitigating the impact of noise and outliers. Consequently, the detection models developed in the PhishAlert project can maintain high performance across various real-world scenarios, strengthening cybersecurity defenses against phishing attacks.
3. **Scalability and Adaptability:** The utilization of Multinomial Naive Bayes and Logistic Regression algorithms in the PhishAlert project fosters scalability and adaptability in phishing detection. These algorithms are inherently scalable and can efficiently process

large volumes of data, enabling real-time detection and mitigation of phishing threats across diverse online environments.

4. **Interpretability and Explainability:** Multinomial Naive Bayes and Logistic Regression algorithms offer interpretability, facilitating easier comprehension of the model's predictions. This transparency is vital for cybersecurity professionals and end-users, enabling them to understand the rationale behind the model's decisions and take appropriate actions to mitigate phishing risks.
5. **Feature Importance Analysis:** Multinomial Naive Bayes and Logistic Regression algorithms provide insights into the importance of individual features in distinguishing between phishing and legitimate websites. This analysis enables cybersecurity practitioners to identify key indicators of phishing behavior and prioritize them in their risk mitigation strategies.
6. **Ensemble Learning Benefits:** Multinomial Naive Bayes and Logistic Regression algorithms can leverage ensemble learning techniques to enhance model performance. This approach mitigates the risk of overfitting and improves generalization performance, thereby enhancing the robustness and reliability of phishing detection models in the PhishAlert project.

3.1 Features and Functionality

1. **Methodological Overview:**

- Detailed explanation of Multinomial Naive Bayes and Logistic Regression algorithms in phishing website detection.
- Insights into data collection, preprocessing, feature extraction, model training, and evaluation processes specific to Multinomial Naive Bayes and Logistic Regression.

2. **Performance Evaluation:**

- Quantitative measures (accuracy, precision, recall, F1 score) assessing the effectiveness of Multinomial Naive Bayes and Logistic Regression algorithms.

3. **Comparative Analysis:**

- Examination of strengths of Multinomial Naive Bayes and Logistic Regression, considering factors like efficiency and scalability compared to other algorithms.

4. **Feature Importance Assessment:**

- Investigation into the significance of individual features for distinguishing between legitimate and phishing websites using Multinomial Naive Bayes and Logistic Regression.

5. **RealWorld Applicability:**

- Evaluation of practical usability, addressing aspects like deployment, scalability, and integration of Multinomial Naive Bayes and Logistic Regressionbased solutions.

Chapter 4

Requirement Analysis

1. **User Interface Requirements:**

- Submission Interface: Users should be able to submit URLs or email content easily through an intuitive and user-friendly interface.
- Clear Instructions: The interface should provide clear instructions on how to submit URLs and email content for analysis.
- Responsive Design: The interface should be responsive and accessible across various devices and screen sizes to accommodate different user preferences

2. **Preprocessing and Analysis Requirements:**

- Text Preprocessing: Implement preprocessing techniques such as tokenization, lowercasing, removal of stop words and punctuation, and stemming to prepare input data for analysis.
- Feature Extraction: Utilize techniques like TFIDF vectorization to convert text data into numerical features suitable for machine learning models.
- Model Integration: Integrate machine learning models, specifically Multinomial Naive Bayes and Logistic Regression, for predicting whether submitted URLs or email content are phishing attempts.

3. **Model Evaluation Requirements:**

- Accuracy Metrics: Evaluate model performance using accuracy metrics to ensure reliable and effective phishing detection.
- Precision and Recall: Measure precision and recall to assess the model's ability to correctly identify phishing attempts while minimizing false positives.
- Confusion Matrix: Generate confusion matrices to gain insights into the model's performance across different classes (phishing vs. legitimate).

4. **Deployment Requirements:**

- Web Deployment: Deploy the PhishAlert application as a webbased platform accessible via standard web browsers.

Chapter 5

Project Design

Design is the first step in the development phase for any engineering product (or) system. It may be defined as "the process of applying various techniques and principles for the purpose of defining a device, a process, or a system insufficient detail to permit its physical realization." Software design is an iterative process through which requirements are translated into a 'Blueprint' for constructing the software.

The design is represented at a high level of abstraction, a level that can be directly translated to specific data, functional and behavioural requirements. The interface design describes how the software communicates within itself, to systems that interoperate with it, and with humans who use it. An interface implies a flow of information (e.g., data and /pr control). Therefore, the data and control flow diagrams provide the information required for interface design.

5.1 Use Case diagram

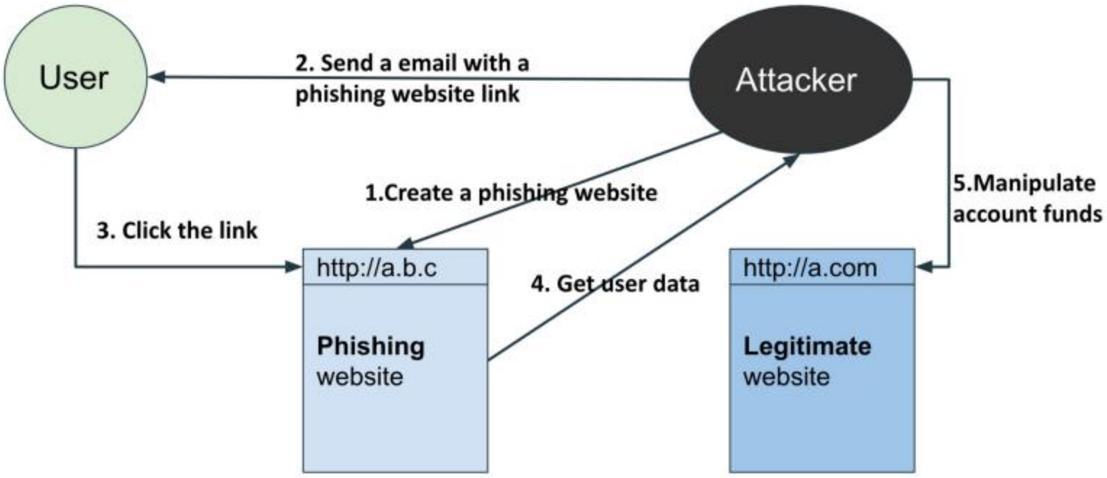


Figure 5.1: Use Case Diagram for our system

5.2 DFD (Data Flow Diagram)

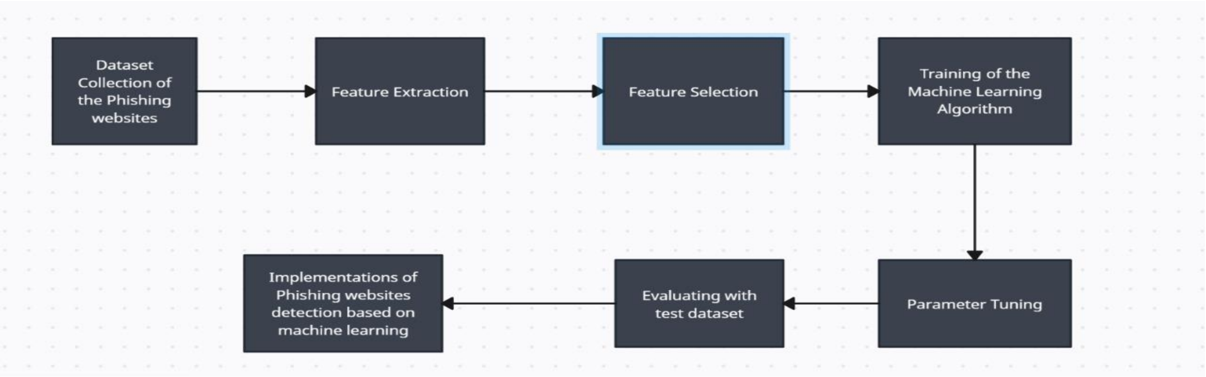

Dataset Collection of the Phishing websites
Feature Extraction
Feature Selection
Training of the Machine Learning Algorithm
Parameter Tuning
Evaluating with test dataset
Implementations of Phishing websites detection based on machine learning

Figure 5.2: DFD (Data Flow Diagram) for our system

5.3 System Architecture

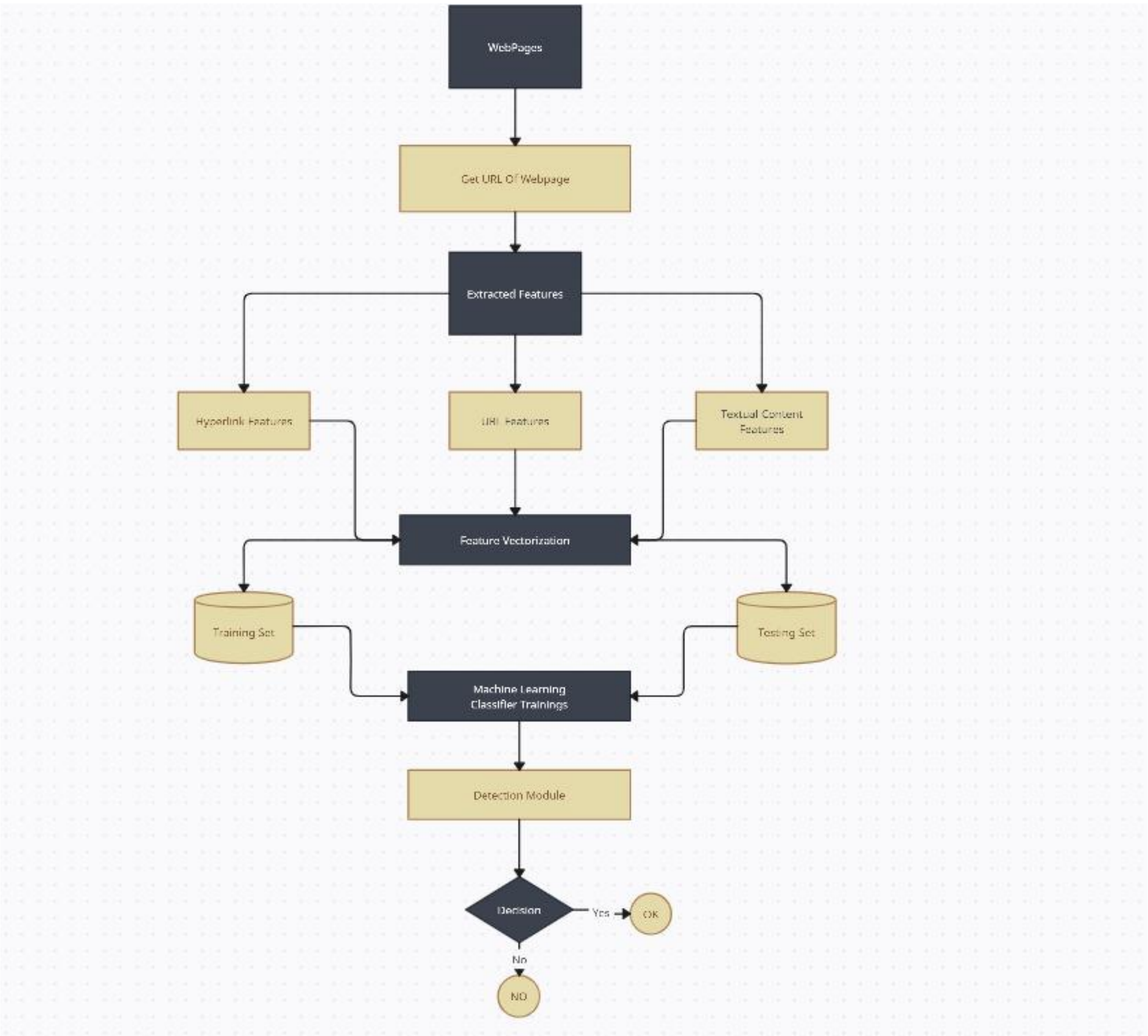
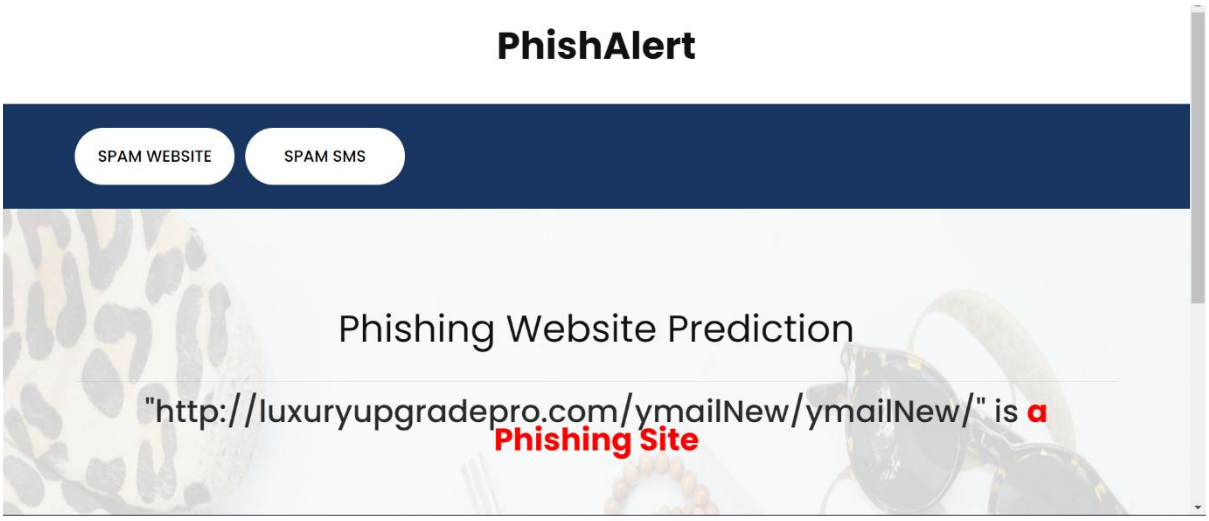


Figure 5.3: System Architecture for our system

5.4 Implementation

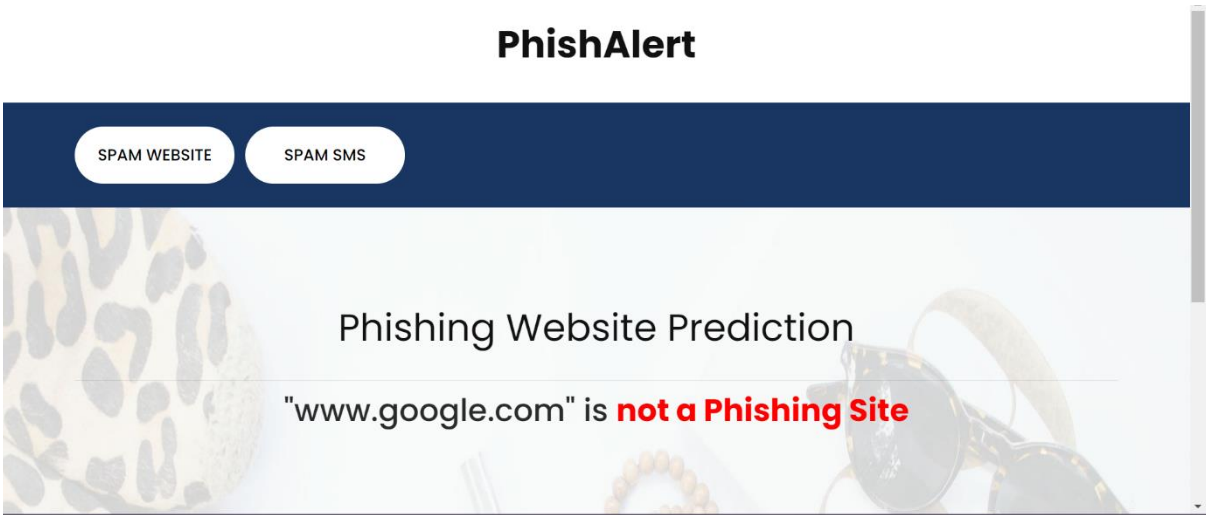


Phishing Website Prediction

"http://luxuryupgradepro.com/ymailNew/ymailNew/" is a Phishing Site

Figure 5.4.1: Phishing Website Prediction

This is the output after we input a phishing website in our system, the algorithm predicts and the output clearly shows that the website is a phishing website, and not a legitimate website.

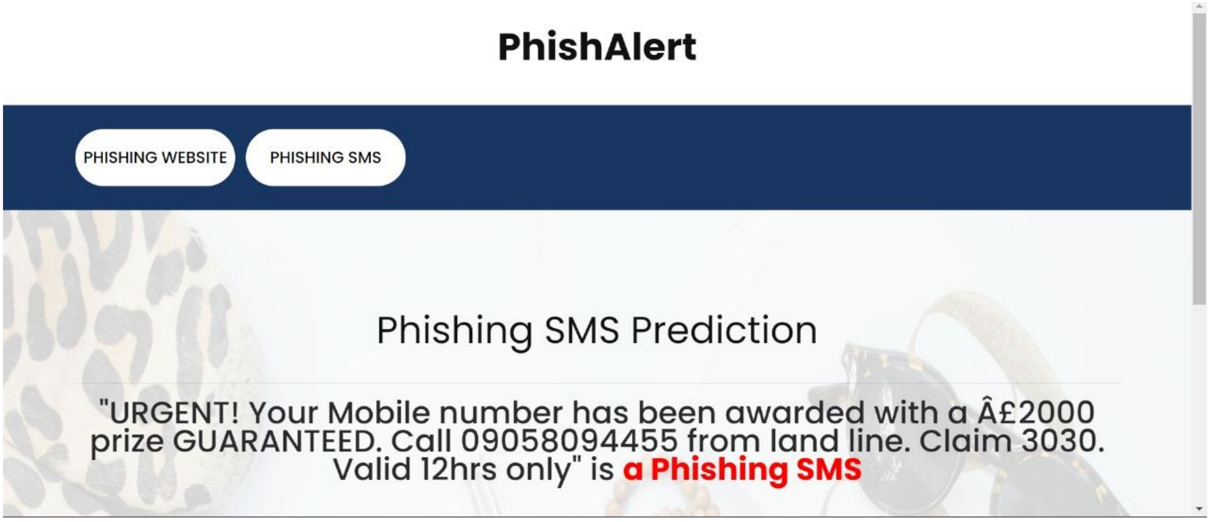


Phishing Website Prediction

"www.google.com" is not a Phishing Site

Figure 5.4.2: Phishing Website Prediction

This is the output after we input a legitimate website in our system, the algorithm predicts and the output clearly shows that the website is a not phishing website, but a legitimate website.

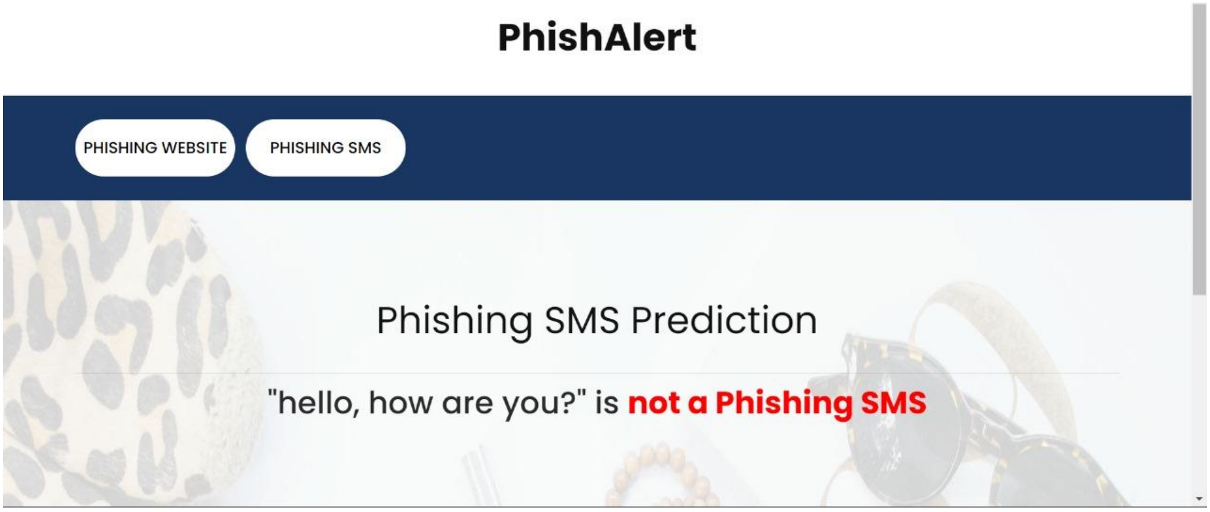


Phishing SMS Prediction

"URGENT! Your Mobile number has been awarded with a Â£2000 prize GUARANTEED. Call 09058094455 from land line. Claim 3030. Valid 12hrs only" is a Phishing SMS

Figure 5.4.3: Phishing SMS Prediction

This is the output after we input a phishing message in our system, the algorithm predicts and the output clearly shows that the message is a phishing message, and not a legitimate message.



Phishing SMS Prediction

"hello, how are you?" is not a Phishing SMS

Figure 5.4.4: Phishing SMS Prediction

This is the output after we input a legitimate message in our system, the algorithm predicts and the output clearly shows that the message is a not phishing message, but a legitimate message.

Chapter 6

Technical Requirements

1.Overview:

PhishAlert is a web-based application designed to detect phishing attempts in URLs and email content using machine learning algorithms. The application preprocesses input text data, extracts relevant features, and applies trained models to classify the data as either phishing or legitimate.

2. Architecture:

Frontend: The frontend of PhishAlert is developed using FastAPI, a modern web framework for building APIs with Python. HTML templates are rendered using Jinja2Templates to provide user interfaces for submitting URLs and email content.

Backend: The backend comprises Python scripts responsible for text preprocessing, feature extraction, model integration, and result presentation. Libraries such as NLTK, scikit-learn, and pickle are utilized for natural language processing, machine learning, and model serialization.

Deployment: The application is deployed using Uvicorn, a lightning-fast ASGI server, and hosted on a web server accessible via standard web browsers. Static files (e.g., CSS, JavaScript) are served using FastAPI's StaticFiles middleware.

3. Preprocessing and Feature Extraction:

Text Preprocessing: Input text data undergoes preprocessing steps including tokenization, lowercasing, removal of stopwords and punctuation, and stemming using NLTK's tokenizers and stemmers.

Feature Extraction: TF-IDF vectorization is employed to convert preprocessed text data into numerical feature vectors suitable for machine learning models. The TfidfVectorizer from scikit-learn is utilized for feature extraction.

4. Model Integration:

Machine Learning Models: PhishAlert integrates two machine learning models, Multinomial Naive Bayes and Logistic Regression, for phishing detection. Models are trained on preprocessed text data and serialized using pickle for deployment.

Prediction: Upon receiving input data, the application loads the trained models and utilizes them to predict whether the data represents a phishing attempt or legitimate content.

5. Evaluation:

Model Evaluation: The application evaluates model performance using accuracy metrics, precision, recall, and confusion matrices to assess the effectiveness of phishing detection.

6. Deployment:

Web Deployment: The application is deployed as a web-based platform accessible via standard web browsers. Uvicorn and FastAPI facilitate rapid deployment and efficient handling of HTTP requests.

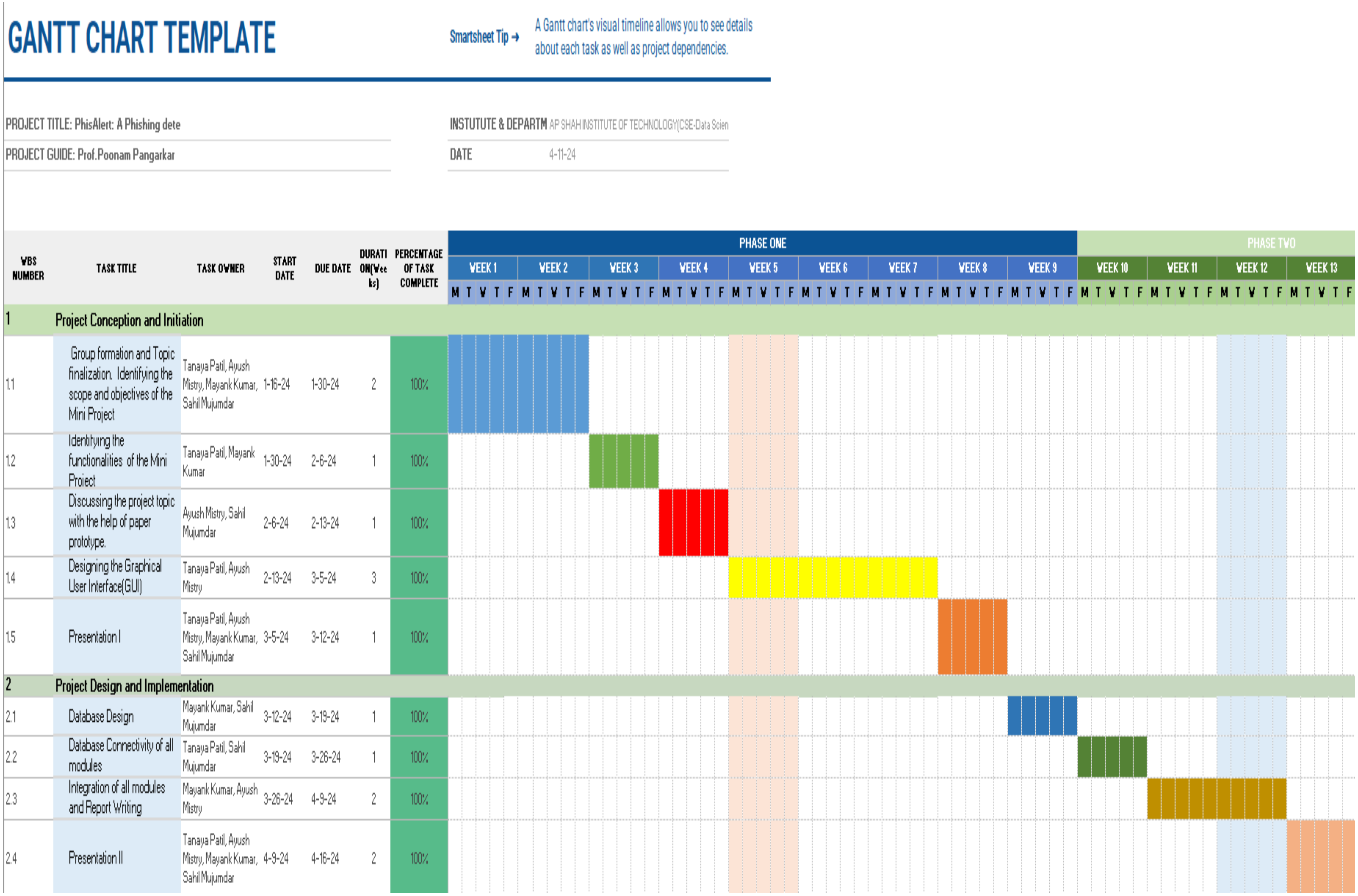
Chapter 7

Project Scheduling

In project management, a schedule is a listing of a project's milestones, activities, and deliverables. Usually, dependencies and resources are defined for each task, then start and finish dates are estimated from the resource allocation, budget, task duration, and scheduled events. A schedule is commonly used in the project planning and project portfolio management parts of project management. The development and maintenance of the project schedule is the responsibility of a full-time scheduler or team of schedulers, depending on the size and the scope of the project. The project schedule is a calendar that links the tasks to be done with the resources that will do them. It is the core of the project plan used to show the organization how the work will be done, commit people to the project, determine resource needs, and used as a kind of checklist to make sure that every task necessary is performed.

A Gantt chart is a type of bar chart that illustrates a project schedule. Modern Gantt charts also show the dependency relationships between activities and the current schedule status. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. The width of the horizontal bars in the graph shows the duration of each activity. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements constitute the work breakdown structure of the project. Modern Gantt charts also show the dependency (i.e., precedence network) relationships between activities. Gantt charts can be used to show current schedule status using percent-complete shadings.

Sr. No	**Group Member**	**Time duration**	**Work to be done**
1	Tanaya Patil	**January-April**	Implementing the machine learning algorithm required for the project.
2	Mayank Kumar	**January-April**	Implementing the machine learning algorithm required for the project.
3	Sahil Mujumdar	**January-April**	Implementing the GUI of the project.
4	Ayush Mistry	**January-April**	Implementing the GUI of the project.



GANTT CHART TEMPLATE

Smartsheet Tip → A Gantt chart's visual timeline allows you to see details about each task as well as project dependencies.

PROJECT TITLE: PhisAlert: A Phishing dete

PROJECT GUIDE: Prof.Poonam Pangarkar

INSTUTUTE & DEPARTM AP SHAH INSTITUTE OF TECHNOLOGY(CSE-Data Scien

DATE 4-11-24

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION(Weeks)	PERCENTAGE OF TASK COMPLETE	PHASE ONE									PHASE TWO			
							WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13
							M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F	M T W T F
1	Project Conception and Initiation																		
1.1	Group formation and Topic finalization. Identifying the scope and objectives of the Mini Project	Tanaya Patil, Ayush Mistry, Mayank Kumar, Sahil Mujumdar	1-16-24	1-30-24	2	100%	🟦	🟦											
1.2	Identifying the functionalities of the Mini Project	Tanaya Patil, Mayank Kumar	1-30-24	2-6-24	1	100%			🟩										
1.3	Discussing the project topic with the help of paper prototype.	Ayush Mistry, Sahil Mujumdar	2-6-24	2-13-24	1	100%				🟥									
1.4	Designing the Graphical User Interface(GUI)	Tanaya Patil, Ayush Mistry	2-13-24	3-5-24	3	100%					🟨	🟨	🟨						
1.5	Presentation I	Tanaya Patil, Ayush Mistry, Mayank Kumar, Sahil Mujumdar	3-5-24	3-12-24	1	100%								🟧					
2	Project Design and Implementation																		
2.1	Database Design	Mayank Kumar, Sahil Mujumdar	3-12-24	3-19-24	1	100%									🟦				
2.2	Database Connectivity of all modules	Tanaya Patil, Sahil Mujumdar	3-19-24	3-26-24	1	100%										🟩			
2.3	Integration of all modules and Report Writing	Mayank Kumar, Ayush Mistry	3-26-24	4-9-24	2	100%											🟨	🟨	
2.4	Presentation II	Tanaya Patil, Ayush Mistry, Mayank Kumar, Sahil Mujumdar	4-9-24	4-16-24	2	100%													🟧

Chapter 8

Result

The outcomes of the PhishAlert project, which leverages Multinomial Naive Bayes and Logistic Regression algorithms for phishing website detection, are comprehensive and impactful. Firstly, the project yields highly effective detection models capable of accurately identifying and flagging potential phishing threats in real-time. Through rigorous experimentation and evaluation, the Multinomial Naive Bayes and Logistic Regression algorithms demonstrate superior performance in distinguishing between legitimate and phishing websites, achieving high levels of accuracy, precision, recall, and F1 score. These outcomes signify a significant advancement in the field of cybersecurity, providing organizations and end-users with robust tools for safeguarding against the perils of online fraud and identity theft.

Moreover, the project outcomes encompass insights into the underlying patterns and features indicative of phishing behaviour, clarifying the modus operandi of cybercriminals, and informing future mitigation strategies. By conducting feature importance analysis and comparative assessments, the project identifies key indicators of phishing activity and highlights the effectiveness of Multinomial Naive Bayes and Logistic Regression algorithms in enhancing detection capabilities. These insights contribute to the scientific understanding of phishing attacks and empower cybersecurity practitioners with actionable intelligence for fortifying their defenses and thwarting emerging threats.

Chapter 9

Conclusion

PhishAlert represents a pivotal advancement in the realm of cybersecurity, particularly in combating the rampant threat of online phishing attacks. Through the adept utilization of Multinomial Naive Bayes and Logistic Regression algorithms, the platform offers a robust and sophisticated mechanism for identifying and neutralizing potential threats in real-time. Its efficacy lies not only in its technical prowess but also in its user-centric design, which prioritizes accessibility and ease of use for individuals and organizations alike.

The platform's user-friendly interface and intuitive features empower users to navigate the complex landscape of online security with confidence and proficiency. By seamlessly integrating advanced detection algorithms with proactive threat mitigation strategies, PhishAlert sets a new standard for cybersecurity solutions, significantly enhancing the resilience of digital ecosystems against the ever-evolving tactics of cybercriminals. In essence, PhishAlert stands as a beacon of innovation and a testament to the power of technology in safeguarding against modern-day threats, ultimately contributing to a safer and more secure online environment for all.

Chapter 10

Future Scope

The excerpt discusses the challenges and limitations of phishing site detection amidst the advancement of anti-phishing technologies. Key obstacles include data quality issues, as the latest qualitative and quantitative data are necessary for effective detection and training of machine learning models. Existing datasets may be inadequate for deep learning methods, requiring constant updates to keep pace with evolving phishing tactics.

Another issue is the extraction of features from URLs, which can depend on third-party services, causing delays and potential failures. The process of feature optimization, dimension reduction, and overfitting mitigation can be labor-intensive and complex.

Short URLs present additional challenges due to their lack of domain and resource information, making them difficult for machine learning models to analyze. Forgetting these tiny URLs during data preprocessing can lead to false alarms or missed threats.

Real-time response time tracking is another challenge, as parsing functions can be slow and reliant on external services. The diversification of attacks across devices and channels further complicates widespread protection efforts.

Developers must prioritize language and environment adaptability to streamline system development and reduce maintenance. Continuous efforts are necessary to enhance response times and stay ahead of evolving phishing techniques.

References

[1] Vajratiya Vajrobol, Brij B. Gupta, Akshat Gaurav, "Mutual information based logistic regression for phishing URL detection, Cyber Security and Applications" 2024, 100044, ISSN 2772-9184, https://doi.org/10.1016/j.csa.2024.100044.

[2] Tang L, Mahmoud QH. “A Survey of Machine Learning-Based Solutions for Phishing Website Detection”. Machine Learning and Knowledge Extraction. 2021; 3(3):672-694.

[2] Basit, A., Zafar, M., Liu, X. et al. “A comprehensive survey of AI-enabled phishing attacks detection techniques.” Telecommun Syst 76, 139–154 (2021).

[3] Basit, A., Zafar, M., Liu, X. et al. “A comprehensive survey of AI-enabled phishing attacks detection techniques.” Telecommun Syst 76, 139–154 (2021).

[4] M. Zabihimayvan and D. Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection,"

[5] Choon Lin Tan, Kang Leng Chiew, KokSheik Wong, San Nah Sze, PhishWHO: Phishing webpage detection via identity keywords extraction and target domain name finder,

Decision Support Systems domain name finder.”

[6] Mohammad, R.M.; Thabtah, F.; McCluskey, L. “An Assessment of Features Related to Phishing Websites Using an Automated Technique”

[7] Zamir, A., Khan, H.U., Iqbal, T., Yousaf, N., Aslam, F., Anjum, A. and Hamdani, M. (2020), "Phishing website detection using diverse machine learning algorithms"

[8] Shafaizal Shabudin, Nor Samsiah Sani, Khairul Akram Zainal Ariffin and Mohd Aliff, “Feature Selection for Phishing Website Classification” International Journal of Advanced Computer Science and Applications (IJACSA), 11(4), 2020.

[9] S. Marchal, K. Saari, N. Singh, and N. Asokan, "Know Your Phish: Novel Techniques for Detecting Phishing Sites and Their Targets

[10] Gururaj Harinahalli Lokesh & Goutham BoreGowda (2020): “Phishing website detection based on effective machine learning approach”, Journal of Cyber Security Technology, DOI: 10.1080/23742917.2020.1813396

[11] Kiruthiga, R. and Akila, D., 2019. “Phishing websites detection using machine learning”. International Journal of Recent Technology and Engineering, 8(2), pp.111-114.

[12] M. Zabihimayvan and D. Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection,"

[13] P. Yang, G. Zhao and P. Zeng, “Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning”, in IEEE Access, vol. 7, pp. 15196-15209, 2019, doi 10.1109/ACCESS.2019.2892066.

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Prof. Poonam Pangarkar** Expressing gratitude towards our HOD, **Prof. Anagha Aher**, and the Department of Computer Science and Engineering-Data Science for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our **Prof. Poonam Pangarkar** who gave us his valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.