Abstract

Phishing attacks pose a significant threat to online security, targeting individuals and organizations with deceptive emails and websites designed to steal sensitive information. Traditional methods of detecting phishing websites have become inadequate against the evolving tactics employed by cybercriminals. This study explores the application of machine learning techniques in phishing website detection, aiming to enhance accuracy and real-time response capabilities. The results highlight the significance of this research in strengthening online security. By leveraging machine learning techniques, the proposed system provides a proactive defense against phishing attacks, safeguarding users, businesses, and organizations from financial losses, identity theft, and reputational damage. Furthermore, the study underscores the importance of continuous research and collaboration in the ever-changing landscape of cybersecurity, ensuring a safer digital environment for all.

Keywords: Feature Extraction, SVM, Classification, Model-training

Acknowledgments

*It gives us great pleasure in presenting the preliminary project report on **'URL BASED PHISHING DETECTION'**.*

*I would like to take this opportunity to thank my internal guide **Prof. Pradnya Kasture** for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.*

*I am also grateful to **Dr. Vina Lomte**, Head of Computer Engineering Department, RMD sinhgad for his indispensable support, suggestions.*

*In the end our special thanks to **Prof. Pradnya Kasture** for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.*

.

Sanket Shendge

Gautam Sharma

Yadnyesh Chaudhari

(B.E. Computer Engg.)

INDEX

List of Figures

CHAPTER 1

SYNOPSIS

1.1 PROJECT TITLE

PHISHING WEBSITE DETECTION USING URL

1.2 PROJECT OPTION

Machine Learning

1.3 INTERNAL GUIDE

Prof. Pradnya Kasture

1.4 SPONSORSHIP AND EXTERNAL GUIDE

Softtech Solution, Prof. Pradnya Kasture, Mr. Pushkar K

1.5 TECHNICAL KEYWORDS (AS PER ACM KEYWORDS)

- SVM Model Training
- Classification
- Machine Learning

1.6 PROBLEM STATEMENT

To individuals, organizations, and online platforms. Cyber criminals employ sophisticated techniques to create deceptive websites that mimic legitimate ones, aiming to steal sensitive information such as usernames, passwords, and financial details from unsuspecting users. Traditional methods of detecting phishing websites are often insufficient to keep up with the rapidly evolving tactics used by attackers. Therefore, there is a critical need for an advanced and adaptive phishing website detection system based on machine learning techniques.

1.7 ABSTRACT

Phishing attacks pose a significant threat to online security, targeting individuals and organizations with deceptive emails and websites designed to steal sensitive information. Traditional methods of detecting phishing websites have become inadequate against the evolving tactics employed by cybercriminals. This study explores the application of machine learning techniques in phishing website detection, aiming to enhance accuracy and real- time response capabilities. The results highlight the significance of this research in strengthening online security. By leveraging machine learning techniques, the proposed system provides a proactive defense against phishing attacks, safeguarding users, businesses, and organizations from financial losses, identity theft, and reputational damage. Further more, the study underscores the importance of continuous research and collaboration in the everchanging landscape of cybersecurity, ensuring a safer digital environment for all. Keywords: Feature Extraction, SVM, Classification, Model-training

1.8 GOALS AND OBJECTIVES

- Develop machine learning models that can accurately identify phishing websites..
- Implement real-time detection mechanisms to identify phishing websites as soon as they are active.
- Create models that can adapt to new phishing techniques and trends.
- Utilize a comprehensive set of features, including website content, domain information, and user behavior, for analysis.

1.9 RELEVANT MATHEMATICS ASSOCIATED WITH THE PROJECT

Let S be the Whole system S= I,P,O

PHISHING WEBSITE DETECTION

I-input

P-procedure

O-output

Input(I)

I= Dataset as a Text

Where,

Text

Procedure (P),

P=I, System Using Perform the PHISHING WEBSITE DETECTION using svstem Output(O)-

O=Detection PHISHING WEBSITE

1.10 NAMES OF CONFERENCES / JOURNALS WHERE PAPERS CAN BE PUBLISHED

1. A.Y. Ahmad, M. Selvakumar, A. Mohammed, and A.-S. Samer, "TrustQR: A new technique for the detection of phishing attacks on QR code," Adv. Sci. Lett., vol. 22, no. 10, pp. 2905-2909, Oct.2021.

2. C. C. Inez and F. Baruch, "Setting priorities in behavioral interventions: An application.

3. To reducing phishing risk," Risk Anal., vol. 38, no. 4, pp. 826-838, Apr. 2021.

4. Aburrous, Maher Hossain, Mohammed Dahal, Keshav Thabtah, Fadi. (2020). Intelligent phishing detection system for ebanking using fuzzy data mining. Expert Systems with Applications. 37. 7913-7921. 10.1016/j.eswa.-2020.04.044.

5. Rosiello, Angelo Kirda, Engin Kruegel, Ferrandi, Fabrizio. (2007). A layout similarity- based approach for detecting phishing pages. Proceedings of the 3rd International Conference on Security and Privacy in Communication Networks, Secure Comm. 454 - 463. 10.1109/SECCOM..4550367.2021.

6. Chawathe, Sudarshan. Improving Email Security with Fuzzy Rules. 1864- 1869. 10.1109/TrustCom/BigDataSE.2018.00282. 2021.

1.11 REVIEW OF CONFERENCE/JOURNAL PAPERS SUPPORTING PROJECT IDEA

1. **Paper Name** : A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites

   Author: Bhagwat M. D., Dr. Patil P. H, Dr. Patil P. H., Dr. T. S. Vishawanath

   Abstract:

   Detecting and finding some phishing websites in real-time for a day now is really a dynamic and nuanced topic involving several variables and requirements. Fuzzy logic strategies may be an important method in detecting and testing phishing websites due to the ambiguities involved in the detection. Instead of exact principles, Fuzzy logic provides a more intuitive way o f dealing with quality variables. An approach to fuzziness resolution and an open and intelligent phishing website detection model will be proposed in the Phishing website assessment. This approach is based on smooth logic and machine learning algorithms that define various factors on the phishing website. A total of 30 characteristics or features and phishing website attributes can be used for phishing detection with high accuracy. A real-time phishing dataset is used which is downloaded from the UCI machine learning repository. The use of fuzzy logic in phishing website detection acknowledges the inherent uncertainties and ambiguities present in such tasks. Phishing websites often employ tactics that blur the lines between legitimate and malicious content, making traditional binary classification methods less effective. Fuzzy logic, with its ability to handle vague and imprecise information, offers a more nuanced approach to modeling these complex situations.The proposed phishing website assessment model integrates fuzzy logic strategies with machine learning algorithms. Fuzzy logic allows for the representation of quality variables in a more intuitive manner, enabling the model to capture subtle indicators of phishing behavior that may not be easily quantifiable using strict binary rules.

2 **Paper Name:** A Methodical Overview on Phishing Detection along with an Organized Way to Construct an Anti-Phishing Framework

Author: Srushti Patil, Sudhir Dhag

Abstract :

Phishing is a security attack to acquire personal information like passwords, credit card details or other account details of a user by means of websites or emails. Phishing websites look similar to the legitimate ones which make it difficult for a layman to differentiate between them. As per the reports of Anti Phishing Working Group (APWG) published in December 2018, phishing against banking services and payment processor was high. Almost all the phishy URLs use HTTPS and use redirects to avoid getting detected. This paper presents a focused literature survey of methods available to detect phishing websites. A comparative study of the in-use anti-phishing tools was accomplished and their limitations were acknowledged. We analyzed the URL-based features used in the past to improve their definitions as per the current scenario which is our major contribution. Also, a step wise procedure of designing an anti-phishing model is discussed to construct an efficient framework which adds to our contribution. Observations made out of this study are stated along with recommendations on existing systems. Phishing attacks continue to be a significant threat, targeting users to acquire sensitive information such as passwords, credit card details, and other account information through deceptive websites or emails. The similarity between phishing websites and legitimate ones poses a challenge for users, making it crucial to develop robust detection mechanisms. The research paper provides a focused literature survey on existing methods for detecting phishing websites and conducts a comparative study of anti-phishing tools to identify their strengths and limitations. One key finding highlighted in the paper is the prevalence of phishing attacks against banking services and payment processors, underscoring the importance of effective detection measures in these sectors.One notable observation from the study is the widespread use of HTTPS by phishing websites, along with tactics such as redirects to evade detection. This highlights the need for advanced detection techniques that go beyond traditional indicators like URL protocols.

3 **Paper Name:** WC-PAD: Web Crawling based Phishing Attack Detection

Author: Nathezhtha.T1 , Sangeetha.D2 ,Vaidehi.V3

Abstract :

Phishing is a criminal offense which involves theft of user's sensitive data. The phishing websites target individuals, organizations, the cloud storage hosting sites and government websites. Currently, hardware based approaches for anti- phishing is widely used but due to the cost and operational factors software based approaches are preferred. The existing phishing detection approaches fails to provide solution to problem like zero-day phishing website attacks. To overcome these issues and precisely detect phishing occurrence a three phase attack detection named as Web Crawler based Phishing Attack Detector(WC- PAD) has been proposed. It takes the web traffics, web content and Uniform Resource Locator(URL) as input features, based on these features classification of phishing and non phishing websites are done. The experimental analysis of the proposed WC-PAD is done with datasets collected from real phishing cases. From the experimental results, it is found that the proposed WC-PAD gives 98.9phishing and zero-day phishing attack detection. The focus on combating phishing attacks, particularly zero-day phishing attacks, through a software-based approach like the Web Crawler based Phishing Attack Detector (WC-PAD) is commendable.In summary, WC-PAD represents a promising advancement in phishing detection by leveraging software-based methodologies and comprehensive feature analysis. Its ability to address zero-day attacks is particularly noteworthy, offering organizations a proactive defense against evolving cyber threats in the phishing landscape. Continued research and refinement of such approaches hold promise for bolstering cybersecurity resilience across diverse digital environments.

4 **Paper Name:** On Effectiveness of Source Code and SSL Based Features for Phishing Website Detection

Author: Roopak.S, Athira P Vijayaraghavan, Tony Thomas

Abstract :

Phishing is a social engineering method to steal user credentials through data entry forms from malicious websites. Currently available anti-malware soft- wares can only detect black listed phishing websites. Similarity based detection methods such as visual similarity can be easily evaded by making some changes in the textual and visual contents of a phishing site. The phishing behavior of a web page can be identified from its URL, domain and source code based features. However, URL and domain based features can be easily defeated by using black hat SEO techniques. In this paper, we extract the relevant rules based on webpage source code and Secure Socket Layering (SSL) based features from a training dataset using Repeated Incremental Pruning to Produce Error Reduction (RIPPER) algorithm. Further, we check for the presence of these rules in a test dataset. Our implementation results show that the webpage source code based rules can identify phishing websites with an accuracy of 0.92. The approach outlined in the paper addresses the limitations of current anti-malware software and similarity-based detection methods by focusing on webpage source code and SSL-based features for phishing detection. By focusing on URL, domain, webpage source code, and SSL-based features, the study addresses the limitations of traditional feature sets. While URL and domain features can be manipulated using black hat SEO techniques, webpage source code and SSL features provide deeper insights into the behavior and security posture of a website. In conclusion, the study presents a promising approach to phishing detection by leveraging webpage source code and SSL-based features, complemented by a robust rule extraction algorithm. This methodology aligns with the need for dynamic and proactive cybersecurity measures in combating sophisticated phishing threats.

5 **Paper Name :** AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites

Author: Yazan A. Al-Sariera1 , Victor Elijah Adeyemo2 , Abdullateef O. Balogun3,4 and Ammar K. Alazzawi3

Abstract:

Phishing is a type of social web-engineering attack in cyberspace where criminals steal valuable data or information from insensitive or uninformed users of the internet. Existing countermeasures in the form of anti-phishing soft- ware and computational methods for detecting phishing activities have proven to be effective. However, new methods are deployed by hackers to thwart these countermeasures. Due to the evolving nature of phishing attacks, the need for novel and efficient countermeasures becomes crucial as the effect of phishing attacks are often fatal and disastrous. Artificial Intelligence (AI) schemes have been the cornerstone of modern countermeasures used for mitigating phishing attacks. AI- based phishing countermeasures or methods possess their shortcomings particularly the high false alarm rate and the inability to interpret how most phishing methods perform their function. This study pro- posed four (4) 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. The pro- posed AI-based meta- learners were fitted on phishing website datasets (currently with the newest features) and their performances were evaluated. The models achieved a detection accuracy not lower than 97 with a drastically low false-positive rate of not more

0.028. In addition, the proposed models out- perform existing ML based models in phishing attack detection. Hence, we recommend the adoption of meta-learners when building phishing attack detection models.

6 **paper Name:** Phishing Site Detection Using Similarity of Website Structure

Author: Shoma Tanaka, Takashi Matsunaka, Akira Yamada, Ayumu Kubota

Abstract:

The number of phishing sites is increasing and becoming a problem. General phishing sites often have very short lives. Phishers are thought to construct phishing sites using tools such as phishing kits. Phishing sites constructed us- ing the same tools have similar website structures. We propose a new method based on the similarity of website structure information defined by the types and sizes of web resources that make up these websites. Our method can detect phishing sites that is not registered with blocklists or do not have similar URL strings with targeting legitimate sites. In addition, our method can identify phishing sites that differed in appearance but have similar website structures. Our method is particularly effective for detecting phishing sites constructed by the same phishers or using the same tools, as our method identifies structural similarity between websites. We conducted an evaluation to confirm the correctness of our assumption using phishing sites constructed using phishing kits and the PhishTank dataset. We found a large number of phishing sites that were structurally similar to phishing sites constructed using phishing kits. We applied our method to web access logs provided by ordinary Japanese citizens, and detected some unknown phishing sites. We have also examined the possibility of improving our method based on the importance of web resources, determined using the number of occurrences in web access logs. The proposed method for detecting phishing sites based on website structure information represents a novel and promising approach to address the increasing prevalence of phishing attacks.In conclusion, the proposed method offers a proactive and dynamic approach to phishing detection, leveraging structural similarities in website composition to identify malicious activity. Its effectiveness in detecting unknown phishing sites and potential for refinement based on real-world data underscore its value in combating the growing threat of phishing attacks.

7 **Paper Name:** Research on Website Phishing Detection Based on LSTM RNN

Author: SU Yang

Abstract:

In order to effectively detect phishing attacks, this paper designed a new detection system for phishing websites using LSTM Recurrent Neural Networks (RNN). LSTM has the advantage of capturing data timing and long-term dependencies. LSTM has strong learning ability, can automatically learn data characterization without manual extraction of complex features, and has strong potential in the face of complex high-dimensional massive data. Experimental results show that this model approach the accuracy of 99.1neural network algorithm Using LSTM Recurrent Neural Networks (RNN) for phishing website detection represents a significant advancement in leveraging machine learning for cybersecurity. Phishing attacks often exhibit temporal patterns and dependencies in data, making them suitable candidates for modeling with RNN architectures like LSTM. LSTM's ability to capture long- term dependencies and remember relevant information over extended sequences aligns well with the dynamic nature of phishing attacks. One of LSTM's strengths lies in its capability to automatically learn data representations and features from raw input sequences. This eliminates the need for manual feature extraction, especially in scenarios with complex and high-dimensional data such as web traffic and user behavior logs associated with phishing detection. In summary, leveraging LSTM RNNs for phishing website detection showcases the potential of deep learning techniques in bolstering cybersecurity defenses. The high accuracy rates achieved underscore LSTM's effectiveness in learning complex data patterns and its applicability in real-world cybersecurity scenarios.

8 **Paper Name:** Analysis of Phishing Website Detection Using CNN and Bidirectional LSTM

Author: A S S V Lakshmi Pooja1, Sridhar.M2

Abstract:

Phishing is a critical internet hazard and phishing losses progressively and it is caused by electronic means to deprive the users of sensitive information. Feature engineering is remaining essential for website-detection phishing solutions, although the quality of detection depends ultimately on previous knowledge of its features. Moreover, while the functionalities derived from different measurements are more precise, these characteristics take a lot of time to re- move. This suggest a multidimensional approach to the detection of phishings focused on a quick detection mechanism through deep learning to overcome these limitations. The first step is to extract and use the character sequence features of the given URL for rapid classification through in-depth learning; this step does not include support from third parties or previous experience in phishing. It combine statistical URLs, web page code functions, website text features and easily categorise Profound learning in the second level on multidimensional functions. By the approach, the detection time of the threshold is shortened. The experimental results show that a rational adjustment of the threshold allows for the efficiency of the detection. The focus on developing a rapid and efficient phishing detection mechanism using deep learning techniques is crucial given the increasing threat posed by phishing attacks. Phishing attacks evolve continuously, necessitating effective feature engineering for accurate detection. However, traditional feature engineering approaches can be time-consuming and may not capture emerging phishing tactics effectively. Deep learning techniques offer a solution by automating feature extraction from raw data, reducing the reliance on predefined features. In conclusion, the multidimensional approach to phishing detection using deep learning represents a significant advancement in combating phishing threats effectively and efficiently. The emphasis on rapid detection, threshold optimization, and hierarchical learning reflects a proactive stance in addressing evolving cybersecurity challenges.

8.1 PLAN OF PROJECT EXECUTION

Sr. No.	Name/Title	Start Date	End Date
1	Preliminary Survey		
2	Introduction and Problem Statement		
3	Literature Survey		
4	Project Statement		
5	Software Requirement And Specification		
6	System Design		
7	Partial Report Submission		
8	Architecture Design		
9	Implementation		
10	Deployement		
11	Testing		
12	Paper Publish		
13	Report Submission		

CHAPTER 2

TECHNICAL KEYWORDS

2.1 AREA OF PROJECT

Its main Area is Detection PHISHING WEBSITE DETECTION Using URL is ML.

2.2 TECHNICAL KEYWORDS

1. SVM Model Training
2. classification
3. Machine Learning
4. Random Forest
5. Feature Extraction
6. Naive Bayes
7. Decision Tree

CHAPTER 3

INTRODUCTION

7.1 PROJECT IDEA

- Phishing can be defined as impersonating a valid site to trick users by stealing their personal data comprising usernames, passwords, accounts numbers, national insurance numbers, etc. The rapid expansion of the internet and the increasing reliance on online platforms for various activities have undeniably transformed the way we live, work, and communicate. However, this digital revolution has also given rise to new challenges, most notably in the realm of cybersecurity. Phishing attacks, a prevalent and insidious form of cybercrime, continue to exploit the vulnerabilities of users, businesses, and organizations worldwide.
- Phishing attacks involve the creation of deceptive websites or emails that mimic trustworthy sources to trick users into divulging sensitive information, such as login

  credentials and financial details. As these attacks grow in sophistication, traditional methods of detection often prove inadequate. This study holds significant implications for cybersecurity practices and online safety. By advancing the state-of-the-art in phishing website detection, this research contributes to the development of more secure online environments for users, businesses, and organizations. The findings of this study are expected to enhance the effectiveness of cybersecurity measures, reduce the risk of data breaches, and protect individuals and businesses from falling victim to phishing attacks.
- In the subsequent sections, this research will delve into the methodologies employed, the challenges faced, the innovative solutions devised, and the outcomes achieved in the pursuit of building a robust and adaptive phishing website detection system using machine learning techniques.

7.2 MOTIVATION OF THE PROJECT

- The motivation behind using machine learning techniques for phishing website detection lies in their ability to provide scalable, real-time data-driven, and adaptive solutions, ultimately enhancing user protection and bolstering cybersecurity efforts in the face of evolving phishing threats.

7.3 LITERATURE SURVEY

1 **Paper Name** : A Methodical Overview on Detection , Identification and Proactive Prevention of Phishing Websites

Author: Bhagwat M. D., Dr. Patil P. H, Dr. Patil P. H., Dr. T. S. Vishawanath

Abstract:

Detecting and finding some phishing websites in real-time for a day now is really a dynamic and nuanced topic involving several variables and requirements. Fuzzy logic strategies may be an important method in detecting and testing phishing websites due to the ambiguities involved in the detection. Instead of exact principles, Fuzzy logic provides a more intuitive way o f dealing with quality variables. An approach to fuzziness resolution and an open and intelligent phishing website detection model will be proposed in the Phishing website assessment. This approach is based on smooth logic and machine learning algorithms that define various factors on the phishing website. A total of 30 characteristics or features and phishing website attributes can be used for phishing detection with high accuracy. A real-time phishing dataset is used which is downloaded from the UCI machine learning repository. The use of fuzzy logic in phishing website detection acknowledges the inherent uncertainties and ambiguities present in such tasks. Phishing websites often employ tactics that blur the lines between legitimate and malicious content, making traditional binary classification methods less effective. Fuzzy logic, with its ability to handle vague and imprecise information, offers a more nuanced approach to modeling these complex situations. The proposed phishing website assessment model integrates fuzzy logic strategies with machine learning algorithms. Fuzzy logic allows for the representation of quality variables in a more intuitive manner, enabling the model to capture subtle indicators of phishing behavior that may not be easily quantifiable using strict binary rules.

2 **Paper Name:** A Methodical Overview on Phishing Detection along with an Organized Way to Construct an Anti-Phishing Framework

Author: Srushti Patil, Sudhir Dhag

Abstract :

Phishing is a security attack to acquire personal information like passwords, credit card details or other account details of a user by means of websites or emails. Phishing websites look similar to the legitimate ones which make it difficult for a layman to differentiate between them. As per the reports of Anti Phishing Working Group (APWG) published in December 2018, phishing against banking services and payment processor was high. Almost all the phishy URLs use HTTPS and use redirects to avoid getting detected. This paper presents a focused literature survey of methods available to detect phishing websites. A comparative study of the in-use anti-phishing tools was accomplished and their limitations were acknowledged. We analyzed the URL-based features used in the past to improve their definitions as per the current scenario which is our major contribution. Also, a step wise procedure of designing an anti-phishing model is discussed to construct an efficient framework which adds to our contribution. Observations made out of this study are stated along with recommendations on existing systems. Phishing attacks continue to be a significant threat, targeting users to acquire sensitive information such as passwords, credit card details, and other account information through deceptive websites or emails. The similarity between phishing websites and legitimate ones poses a challenge for users, making it crucial to develop robust detection mechanisms. The research paper provides a focused literature survey on existing methods for detecting phishing websites and conducts a comparative study of anti-phishing tools to identify their strengths and limitations. One key finding highlighted in the paper is the prevalence of phishing attacks against banking services and payment processors, underscoring the importance of effective detection measures in these sectors.One notable observation from the study is the widespread use of HTTPS by phishing websites, along with tactics such as redirects to evade detection. This highlights the need for advanced detection techniques that go beyond traditional indicators like URL protocols.

3 **Paper Name:** WC-PAD: Web Crawling based Phishing Attack Detection

Author: Nathezhtha.T1 , Sangeetha.D2 ,Vaidehi.V3

Abstract :

Phishing is a criminal offense which involves theft of user's sensitive data. The phishing websites target individuals, organizations, the cloud storage hosting sites and government websites. Currently, hardware based approaches for anti-phishing is widely used but due to the cost and operational factors software based approaches are preferred. The existing phishing detection approaches fails to provide solution to problem like zero-day phishing website attacks. To overcome these issues and precisely detect phishing occurrence a three phase attack detection named as Web Crawler based Phishing Attack Detector(WC- PAD) has been proposed. It takes the web traffics, web content and Uniform Resource Locator(URL) as input features, based on these features classification of phishing and non phishing websites are done. The experimental analysis of the proposed WC-PAD is done with datasets collected from real phishing cases. From the experimental results, it is found that the proposed WC-PAD gives 98.9phishing and zero-day phishing attack detection. The focus on combating phishing attacks, particularly zero-day phishing attacks, through a software-based approach like the Web Crawler based Phishing Attack Detector (WC-PAD) is commendable. In summary, WC-PAD represents a promising advancement in phishing detection by leveraging software-based methodologies and comprehensive feature analysis. Its ability to address zero-day attacks is particularly noteworthy, offering organizations a proactive defense against evolving cyber threats in the phishing landscape. Continued research and refinement of such approaches hold promise for bolstering cybersecurity resilience across diverse digital environments.

4 **Paper Name:** On Effectiveness of Source Code and SSL Based Features for Phishing Website Detection

Author: Roopak.S, Athira P Vijayaraghavan, Tony Thomas

Abstract :

Phishing is a social engineering method to steal user credentials through data entry forms from malicious websites. Currently available anti-malware soft- wares can only detect black listed phishing websites. Similarity based detection methods such as visual similarity can be easily evaded by making some changes in the textual and visual contents of a phishing site. The phishing behavior of a web page can be identified from its URL, domain and source code based features. However, URL and domain based features can be easily defeated by using black hat SEO techniques. In this paper, we extract the relevant rules based on webpage source code and Secure Socket Layering (SSL) based features from a training dataset using Repeated Incremental Pruning to Produce Error Reduction (RIPPER) algorithm. Further, we check for the presence of these rules in a test dataset. Our implementation results show that the webpage source code based rules can identify phishing websites with an accuracy of 0.92. The approach outlined in the paper addresses the limitations of current anti-malware software and similarity- based detection methods by focusing on webpage source code and SSL-based features for phishing detection. By focusing on URL, domain, webpage source code, and SSL-based features, the study addresses the limitations of traditional feature sets. While URL and domain features can be manipulated using black hat SEO techniques, webpage source code and SSL features provide deeper insights into the behavior and security posture of a website. In conclusion, the study presents a promising approach to phishing detection by leveraging webpage source code and SSL-based features, complemented by a robust rule extraction algorithm. This methodology aligns with the need for dynamic and proactive cybersecurity measures in combating sophisticated phishing threats.

5 **Paper Name :** AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites

Author: Yazan A. Al-Sariera1 , Victor Elijah Adeyemo2 , Abdullateef O. Balogun3,4 and Ammar K. Alazzawi3

Abstract:

Phishing is a type of social web-engineering attack in cyberspace where criminals steal valuable data or information from insensitive or uninformed users of the internet. Existing countermeasures in the form of anti-phishing soft- ware and computational methods for detecting phishing activities have proven to be effective. However, new methods are deployed by hackers to thwart these countermeasures. Due to the evolving nature of phishing attacks, the need for novel and efficient countermeasures becomes crucial as the effect of phishing attacks are often fatal and disastrous. Artificial Intelligence (AI) schemes have been the cornerstone of modern countermeasures used for mitigating phishing attacks. AI- based phishing countermeasures or methods possess their shortcomings particularly the high false alarm rate and the inability to interpret how most phishing methods perform their function. This study pro- posed four (4) 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. The pro- posed AI-based meta- learners were fitted on phishing website datasets (currently with the newest features) and their performances were evaluated. The models achieved a detection accuracy not lower than 97 with a drastically low false-positive rate of not more

0.028. In addition, the proposed models out- perform existing ML based models in phishing attack detection. Hence, we recommend the adoption of meta-learners when building phishing attack detection models.

6 **paper Name:** Phishing Site Detection Using Similarity of Website Structure

Author: Shoma Tanaka, Takashi Matsunaka, Akira Yamada, Ayumu Kubota

Abstract:

The number of phishing sites is increasing and becoming a problem. General phishing sites often have very short lives. Phishers are thought to construct phishing sites using tools such as phishing kits. Phishing sites constructed us- ing the same tools have similar website structures. We propose a new method based on the similarity of website structure information defined by the types and sizes of web resources that make up these websites. Our method can detect phishing sites that is not registered with blocklists or do not have similar URL strings with targeting legitimate sites. In addition, our method can identify phishing sites that differed in appearance but have similar website structures. Our method is particularly effective for detecting phishing sites constructed by the same phishers or using the same tools, as our method identifies structural similarity between websites. We conducted an evaluation to confirm the correctness of our assumption using phishing sites constructed using phishing kits and the PhishTank dataset. We found a large number of phishing sites that were structurally similar to phishing sites constructed using phishing kits. We applied our method to web access logs provided by ordinary Japanese citizens, and detected some unknown phishing sites. We have also examined the possibility of improving our method based on the importance of web resources, determined using the number of occurrences in web access logs. The proposed method for detecting phishing sites based on website structure information represents a novel and promising approach to address the increasing prevalence of phishing attacks.In conclusion, the proposed method offers a proactive and dynamic approach to phishing detection, leveraging structural similarities in website composition to identify malicious activity. Its effectiveness in detecting unknown phishing sites and potential for refinement based on real-world data underscore its value in combating the growing threat of phishing attacks.

7 **Paper Name:** Research on Website Phishing Detection Based on LSTM RNN

Author: SU Yang

Abstract:

In order to effectively detect phishing attacks, this paper designed a new detection system for phishing websites using LSTM Recurrent Neural Networks (RNN). LSTM has the advantage of capturing data timing and long-term dependencies. LSTM has strong learning ability, can automatically learn data characterization without manual extraction of complex features, and has strong potential in the face of complex high-dimensional massive data. Experimental results show that this model approach the accuracy of 99.1neural network algorithms Using LSTM Recurrent Neural Networks (RNN) for phishing website detection represents a significant advancement in leveraging machine learning for cybersecurity. Phishing attacks often exhibit temporal patterns and dependencies in data, making them suitable candidates for modeling with RNN architectures like LSTM. LSTM's ability to capture long-term dependencies and remember relevant information over extended sequences aligns well with the dynamic nature of phishing attacks. One of LSTM's strengths lies in its capability to automatically learn data representations and features from raw input sequences. This eliminates the need for manual feature extraction, especially in scenarios with complex and high-dimensional data such as web traffic and user behavior logs associated with phishing detection. In summary, leveraging LSTM RNNs for phishing website detection showcases the potential of deep learning techniques in bolstering cybersecurity defenses. The high accuracy rates achieved underscore LSTM's effectiveness in learning complex data patterns and its applicability in real-world cybersecurity scenarios.

8 **Paper Name:** Analysis of Phishing Website Detection Using CNN and Bidirectional LSTM

Author: A S S V Lakshmi Pooja1, Sridhar.M2

Abstract:

Phishing is a critical internet hazard and phishing losses progressively and it is caused by electronic means to deprive the users of sensitive information. Feature engineering is remaining essential for website-detection phishing solutions, although the quality of detection depends ultimately on previous knowledge of its features. Moreover, while the functionalities derived from different measurements are more precise, these characteristics take a lot of time to re- move. This suggest a multidimensional approach to the detection of phishings focused on a quick detection mechanism through deep learning to overcome these limitations. The first step is to extract and use the character sequence features of the given URL for rapid classification through in-depth learning; this step does not include support from third parties or previous experience in phishing. It combine statistical URLs, web page code functions, website text features and easily categorise Profound learning in the second level on multidimensional functions. By the approach, the detection time of the threshold is shortened. The experimental results show that a rational adjustment of the threshold allows for the efficiency of the detection. The focus on developing a rapid and efficient phishing detection mechanism using deep learning techniques is crucial given the increasing threat posed by phishing attacks. Phishing attacks evolve continuously, necessitating effective feature engineering for accurate detection. However, traditional feature engineering approaches can be time-consuming and may not capture emerging phishing tactics effectively. Deep learning techniques offer a solution by automating feature extraction from raw data, reducing the reliance on predefined features. In conclusion, the multidimensional approach to phishing detection using deep learning represents a significant advancement in combating phishing threats effectively and efficiently. The emphasis on rapid detection, threshold optimization, and hierarchical learning reflects a proactive stance in addressing evolving cybersecurity challenges.

CHAPTER 4

PROBLEM DEFINITION AND SCOPE

4.1 PROBLEM STATEMENT

- To individuals, organizations, and online platforms. Cyber criminals employ sophisticated techniques to create deceptive websites that mimic legitimate ones, aiming to steal sensitive information such as usernames, passwords, and financial details from unsuspecting users. Traditional methods of detecting phishing websites are often insufficient to keep up with the rapidly evolving tactics used by attackers. Therefore, there is a critical need for an advanced and adaptive phishing website detection system based on machine learning techniques. Detecting phishing websites using Support Vector Machines

4.1.1 Goals and objectives

- Develop machine learning models that can accurately identify phishing websites.
- Implement real-time detection mechanisms to identify phishing websites as soon as they are active.
- Create models that can adapt to new phishing techniques and trends.
- Utilize a comprehensive set of features, including website content, domain information, and user behavior, for analysis.

4.1.2 Statement of scope

- The scope of the "Phishing Website Detection by Machine Learning Techniques" project encompasses a comprehensive exploration of machine learning algorithms and methodologies to develop a robust and adaptive system for identifying phishing websites. (SVM)/NB is a common approach in machine learning-based cybersecurity. Use the testing dataset to evaluate the SVM's/NB performance. Common evaluation metrics for binary classification tasks like phishing detection include accuracy, precision, recall, F1-score, and the ROC curve. Divide your dataset into a training set and a testing set.

4.2 MAJOR CONSTRAINTS

- The project involves selecting and optimizing SVM/NB model parameters, kernel functions, and hyperparameters for phishing classification. The project will evaluate the performance of the SVM/NB phishing classification model using metrics such as accuracy, precision, recall, F1 score, and confusion matrices. Model assessment will cover its effectiveness in classifying different phishing detection.

4.3 METHODOLOGIES OF PROBLEM SOLVING AND EFFICIENCY IS- SUES

SVM/NB Algorithm –

- SVMs are primarily known for classification tasks, where they classify data points into different classes or categories. They can also be used for regression, where the goal is to predict a continuous numerical value. In a two-dimensional space, the hyperplane is a straight line that separates two classes. In higher dimensions, it becomes a hyperplane.

- SVM can handle linear and nonlinear classification by mapping data into a higher-dimensional space.
  SVMs work well for high-dimensional data. They are effective when the number of features exceeds the number of samples.

- SVMs are versatile and can be used for both linear and nonlinear problems. Support Vector Machines are a valuable tool in machine learning and have found applications in a wide range of fields due to their versatility and ability to handle complex classification tasks.

- Naive Bayes Model Selection:

- Gather a dataset of websites, including both legitimate and phishing websites. Preprocess the data by extracting relevant features, such as URL features, page content analysis, and header information.

- Convert the website data into a numerical format suitable for the Naive Bayes algorithm. Common features include the presence of suspicious keywords, URL length, domain age, and various text-based features.

- Divide your dataset into a training set and a testing set. The training set will be used to train the Naive Bayes model, and the testing set will be used to evaluate its performance.

- Choose the appropriate variant of Naive Bayes for your task. In text classification, the Multinomial Naive Bayes classifier is often use

4.4 OUTCOME

- Detecting phishing websites using URLs involves analyzing various features such as domain name, SSL certificate, URL length, presence of subdomains, etc., to determine the likelihood of the website being legitimate or malicious. The outcome of such a detection process typically falls into one of the following categories:

1. Legitimate: The website is determined to be genuine and safe for users to interact with. This outcome is based on positive indicators such as a valid SSL certificate, a reputable domain name, and no suspicious elements in the URL structure.
2. Phishing: The website is flagged as malicious or a phishing attempt. This outcome is based on negative indicators such as a suspicious domain name (e.g., misspelled or misleading), absence of SSL certificate or an invalid one, presence of subdomains mimicking legitimate sites, unusually long or complex URLs, and known phishing patterns in the URL or webpage content.
3. Suspicious: In some cases, the detection system may not be able to definitively classify the website as legitimate or phishing. It may exhibit mixed indicators or fall into a gray area where further analysis is needed. This outcome may trigger additional security checks or user confirmation before allowing access.
4. Error: Sometimes, the URL might be malformed or the detection system encounters an error during analysis. In such cases, the outcome could be an error message indicating that the URL cannot be processed or classified accurately.

- The outcome of phishing website detection using URLs is crucial for protecting users from potential security threats and ensuring a safe browsing experience. Automated systems, machine learning models, and human oversight often work together to improve the accuracy of such detections and reduce false positives or false-negatives.

4.5 APPLICATIONS

- Email Spam Filtering
- Network Security Tools
- Machine Learning Models in Security Suites

4.6 HARDWARE RESOURCES REQUIRED

- RAM : 8 GB
- Hard Disk : 40 GB
- Processor : Intel i5 Processor

4.7 SOFTWARE RESOURCES REQUIRED

Platform :

- IDE : Spyder
- Coding Language : Python Version 3.7,3.8
- Operating System : Windows 10(64 bit)

CHAPTER 5

PROJECT PLAN

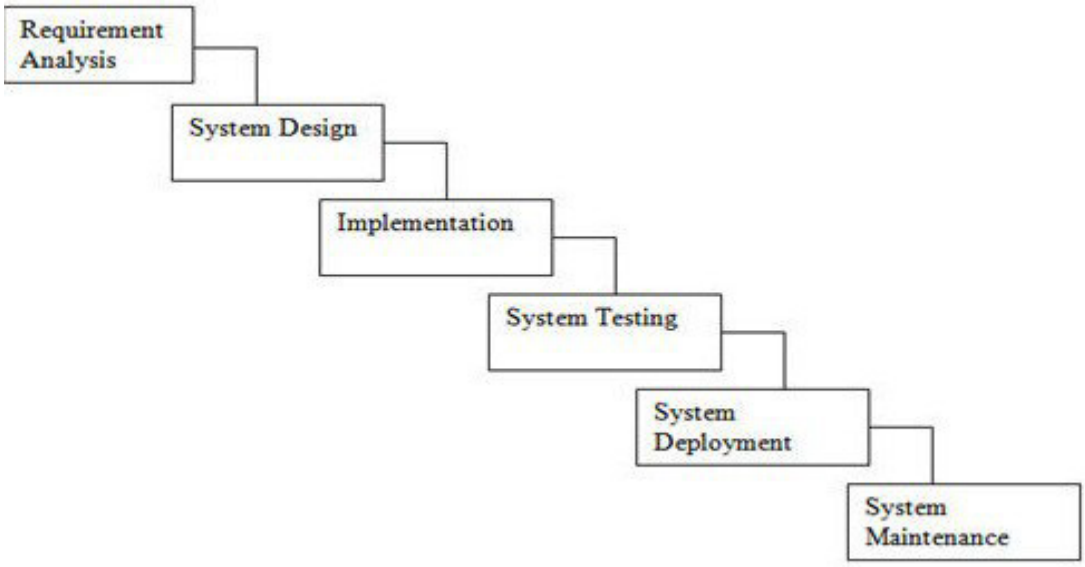
5.1 PROJECT ESTIMATES

We are using waterfall model for our project estimation.

- **Requirement gathering and analysis:** In this step of waterfall we identify what are various requirements are need for our project such are software and hardware required, database, and interfaces.
- **System Design:** In this system design phase we design the system which is easily understood for end user i.e. user friendly. We design some UML diagrams and data flow diagram to understand the system flow and system module and sequence of execution.
- **Implementation:** In implementation phase of our project we have implemented various module required of successfully getting expected outcome at the different module levels. With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.
- **Testing:** The different test cases are performed to test whether the project module are giving expected outcome in assumed time. All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of System:** Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.
- **Maintenance:** There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards like a waterfall through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name

”Waterfall Model”. In this model phases do not overlap.



5.1.1 Reconciled Estimates

Project reconciliation management is a component of Project management which arranges every one of the parts of a project. Project reconciliation guarantees smooth execution of all procedures.

5.1.2 Project Resources

Well configured Laptop, eclipse IDE, 2 GHZ CPU speed, 8 GB RAM, Internet Connection

5.2 RISK MANAGEMENT W.R.T. NP HARD ANALYSIS

1. In appropriate dataset -To overcome this risk we are trying to use well organized and complete dataset.
2. Security- To overcome and improving security we use multilevel security like access permissions of user.

5.2.1 Risk Identification

- Inaccurate shot detection: The system may encounter challenges in accurately identifying and classifying different types of shots, leading to incorrect scoring and analysis. This can occur due to variations in player techniques, occlusions, poor lighting conditions, or noise in the video feed.

- False positives or false negatives: The shot detection system may generate false positives, identifying shots where there was none, or false negatives, missing actual shots played by the batsmen. This can occur due to algorithmic limitations, technical issues, or inadequate training data

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Description 1	Low	Low	High	High
2	Description 2	Low	Low	High	High

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	> 75%
Medium	Probability of occurrence is	26 − 75%
Low	Probability of occurrence is	< 25%

Table 5.2: Risk Probability definitions

Impact	Value	Description
Very high	> 10%	Schedule impact or Unacceptable quality
High	5 − 10%	Schedule impact or Some parts of the project have low quality
Medium	< 5%	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact definitions

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Description 1
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Strategy
Risk Status	Occurred

Risk ID	2
Risk Description	Description 2
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Better testing will resolve this issue.
Risk Status	Identified

Risk ID	3
Risk Description	Description 3
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Very High
Response	Accept
Strategy	Example Running Service Registry behind proxy balancer
Risk Status	Identified

5.3 PROJECT SCHEDULE

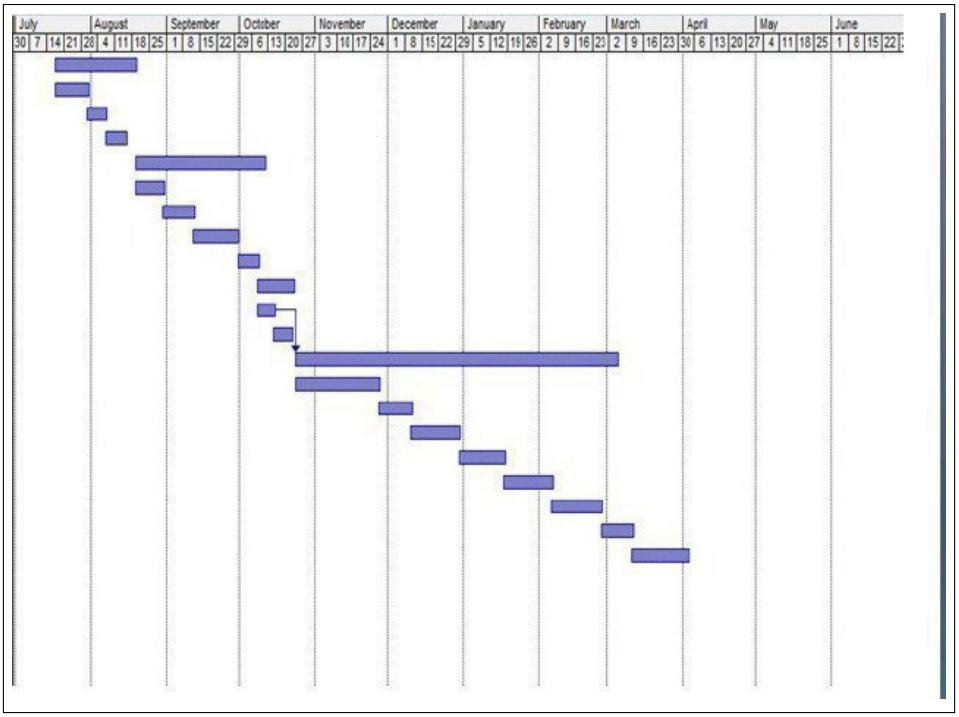


Figure 5.4: Project task set

5.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1: Correctness
- Task 2: Availability
- Task 3: Integrity

5.3.2 Task network

Figure 5.5: Task Network

5.3.3 Timeline Chart

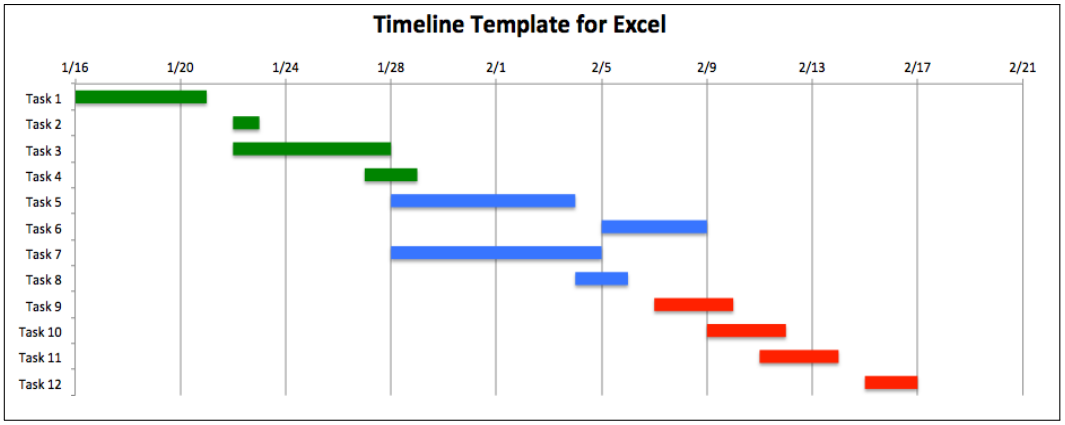


Figure 5.6: Timeline Chart

5.4 TEAM ORGANIZATION

Team consists of 4 members and proper planning mechanism are used and roles of each member are defined.

5.4.1 Team structure

The team structure for the project is identified. There are total 4 members in our team and roles are defined. All members are contributing in all the phases of project.

5.4.2 Management reporting and communication

Well planning mechanisms are used for progress reporting and inter/intra team communication are identified as per requirements of the project

CHAPTER 6

SOFTWARE REQUIREMENT SPECIFICATION

6.1 INTRODUCTION

Phishing website detection plays a crucial role in identifying and mitigating phishing threats. Phishing websites are malicious web pages designed to mimic legitimate sites, with the intent of luring unsuspecting users to enter their sensitive data. Detecting these fraudulent websites is a complex and challenging task, given the constantly evolving tactics used by cybercriminals. Use of machine learning algorithms in phishing website detection, including SVM and NB, is an indispensable part of the cyber-security landscape. The ongoing development and application of these techniques are vital for staying ahead of cyber threats and providing a safe online environment for users and organizations**.**

6.1.1 Purpose and Scope of Document

We will be building a model which will detection of Phishing website.

1. To study Python and developed a GUI in Python.
2. To design master GUI system in Python using Spyder IDE.
3. To learn SVM/NB Algorithm.

6.1.2 Overview of responsibilities of Developer

Researching, designing, implementing, and managing software programs. Testing and evaluating new programs. Identifying areas for modification in existing programs and subsequently developing these modifications. Writing and implementing efficient code.

6.2 USAGE SCENARIO

A case scenario is a made-up situation or problem using real-life constraints and affects in order to discuss and predict how a certain situation could turn out in the real world. By testing the potential outcomes of a problem, those problems are sometimes easier to avoid and solve

6.2.1 User profiles

In the Project there are User, first user will Done their Login and Registration. Then User Upload Text And then by using SVM/NB Algorithm Phishing website Classification

6.2.2 Use Case View

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

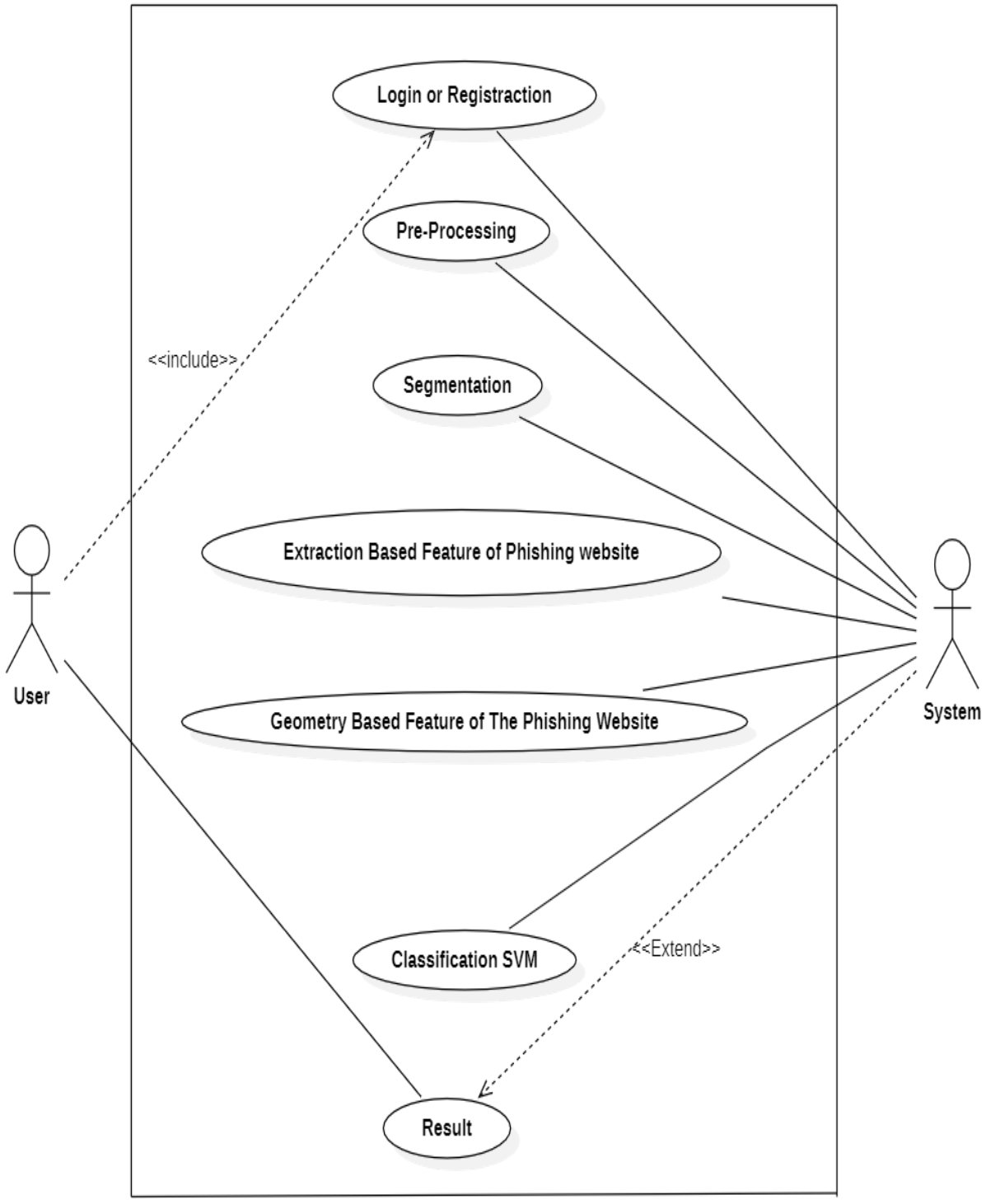


Figure 6.1: Use case Diagram

6.3 DATA MODEL AND DESCRIPTION

6.3.1 Data Description

SQL is, fundamentally, a programming language designed for accessing, modifying and extracting information from relational databases. As a programming language, SQL has commands and a syntax for issuing those commands.

6.4 FUNCTIONAL MODEL AND DESCRIPTION

- The performance of the functions and every module must be well.
- The overall performance of the software will enable the users to work efficiently.
- Performance of response should be fast. Performance of the providing virtual Environment should be fast

6.4.1 Data Flow Diagram

- In Data Flow Diagram, we Show that flow of data in our system in DFD0 we show that base DFD in which rectangle present input as well as output and circle show our system, In DFD1 we show actual input and actual output of system input of our system is text or image and output is rumor detected like wise in DFD 2 we present operation of user as well as admin.

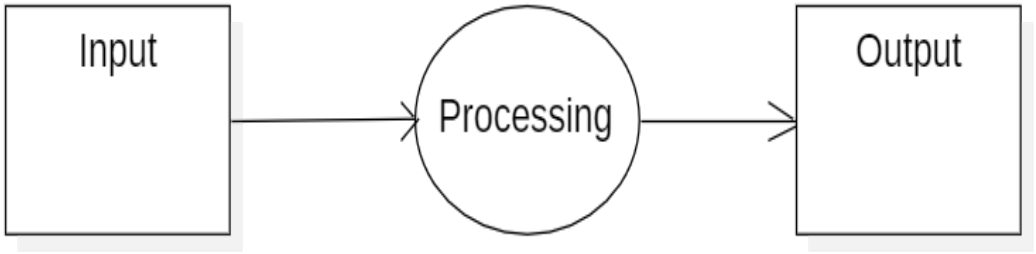


Figure 6.2: Data Flow diagram

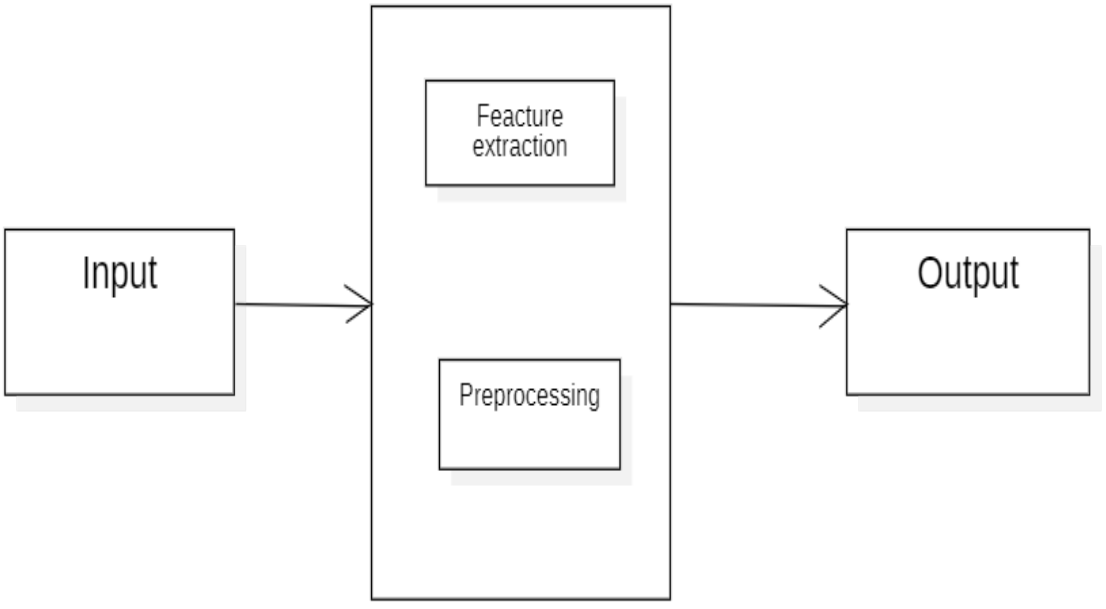


Figure 6.3: Data Flow diagram

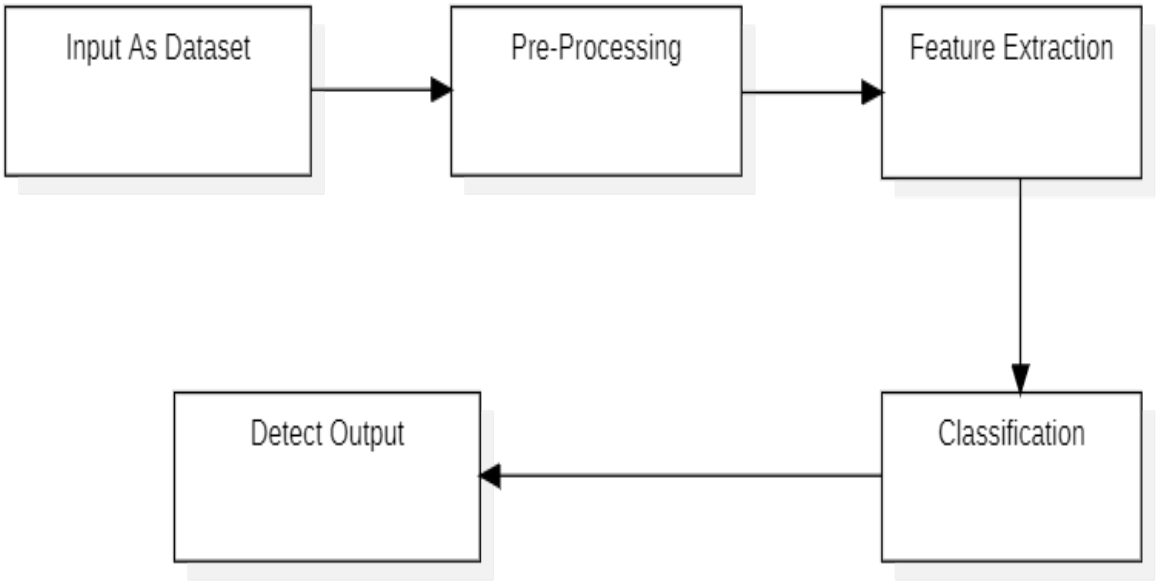


Figure 6.4: Data Flow diagram

6.4.2 Activity Diagram:

Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control they can also include elements showing the flow of data between activities through one or more data stores.

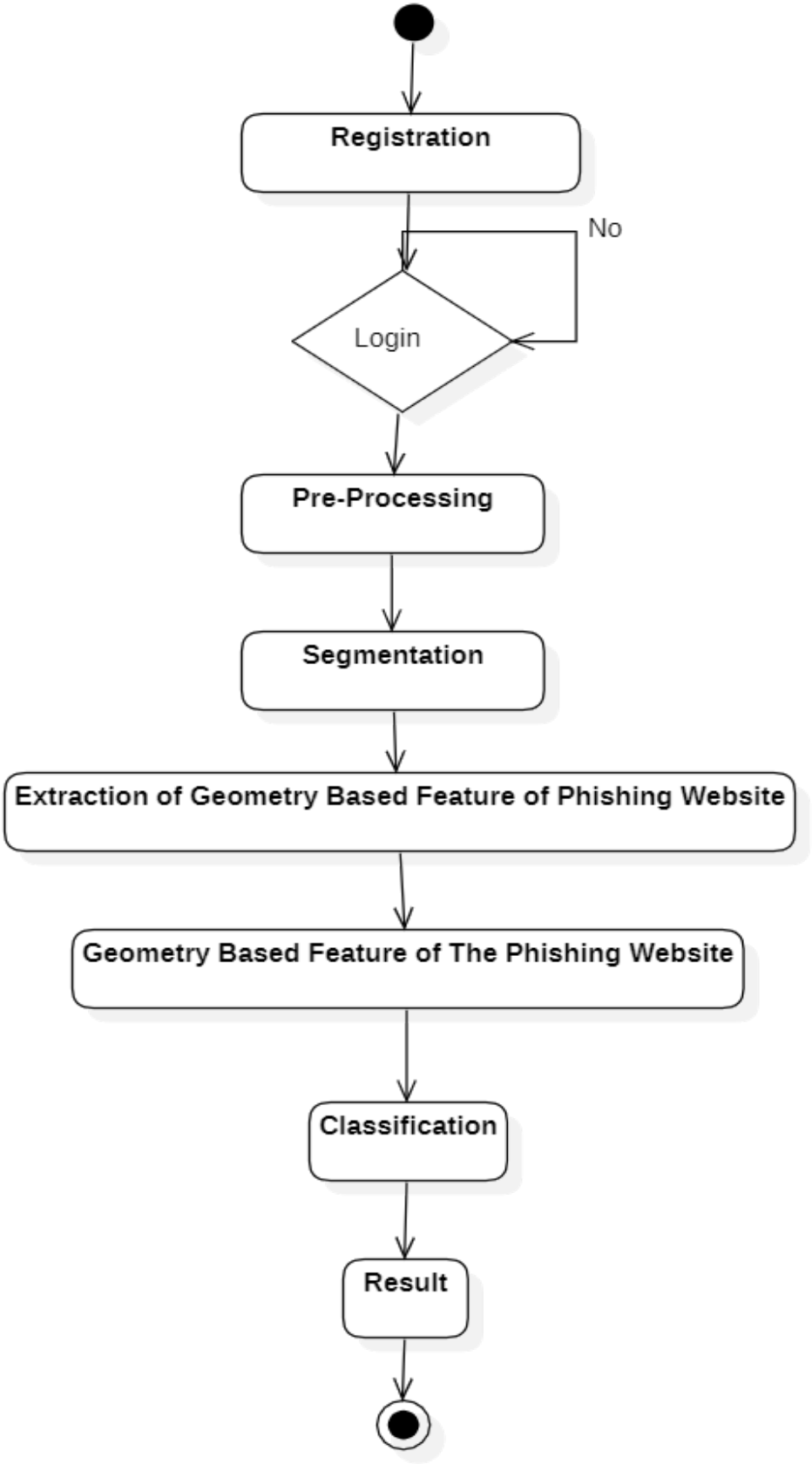


Figure 6.5: Activity Diagram

6.4.3 Non Functional Requirements:

- The performance of the functions and every module must be well.
- The overall performance of the software will enable the users to work efficiently.
- Performance of response should be fast.
- Performance of the providing virtual environment should be fast.

6.4.4 State Diagram:

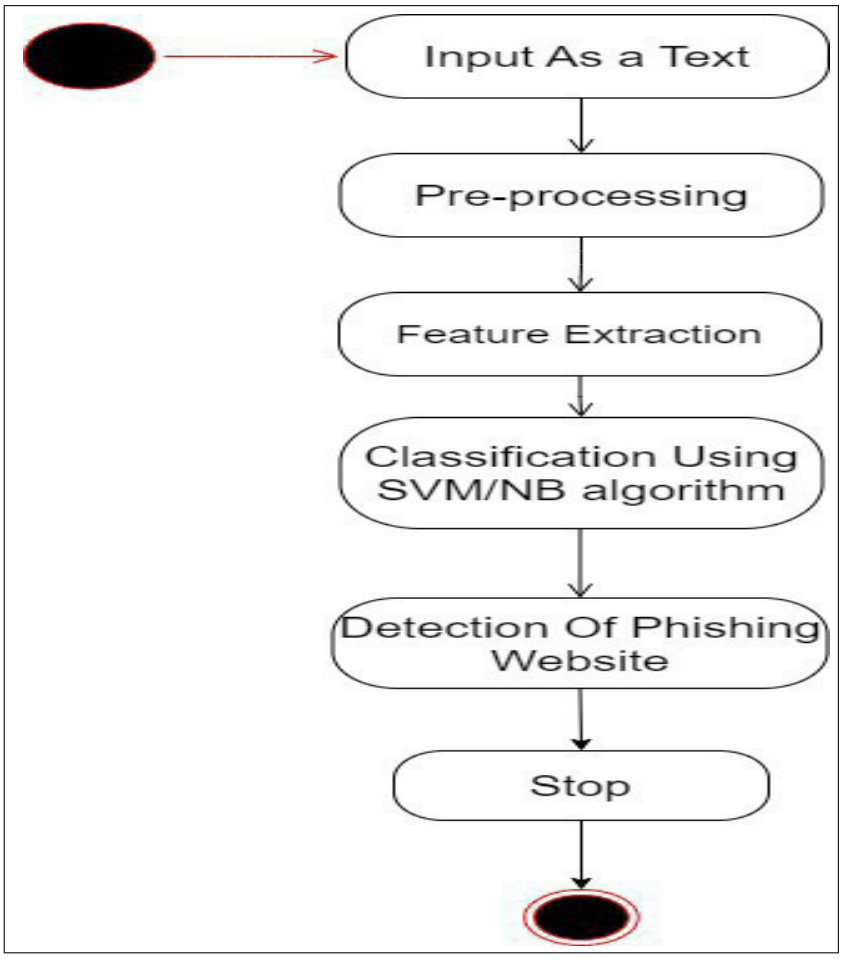


Figure 6.6: State Diagram

6.4.5 Software Interface Description

- IDE : Spyder
- Coding Language : Python Version 3.7,3.8
- Operating System : Windows 10(64 bit)

6.5 ANALYSIS MODEL: SDLC MODEL TO BE APPLIED

- The software development cycle is a combination of different phases such as designing, implementing and deploying the project. These different phases of the software development model are described in this section. The SDLC model for the project development can be understood using the following figure The chosen SDLC model is the waterfall model which is easy to follow and fits bests for he implementation of this project.
- Requirements Analysis: At this stage, the business requirements, definitions of use cases are studied and respective documentations are generated. Design: In this stage, the designs of the data models will be defined and different data preparation and analysis will be carried out.
- Implementation: The actual development of the model will be carried out in this stage. Based on the data model designs and requirements from previous stages, appropriate algorithms, mathematical models and design patterns will be used to develop the agent's back-end and front-end components.
- Testing: The developed model based on the previous stages will be tested in this stage. Various validation tests will be carried out over the trained model.
  Deployment: After the model is validated for its accuracy scores its ready to be deployed or used in simulated scenarios.

CHAPTER 7

DETAILED DESIGN DOCUMENT USING APPENDIX A AND B

7.1 INTRODUCTION

- Machine learning algorithms can analyze vast datasets and identify subtle pat- terns that are often indiscernible to human analysts.
- Machine learning models can process data in real-time, enabling the system to identify phishing websites as soon as they are activated.
- Machine learning systems can learn from new data and adapt to emerging phishing techniques.

7.2 ARCHITECTURAL DESIGN

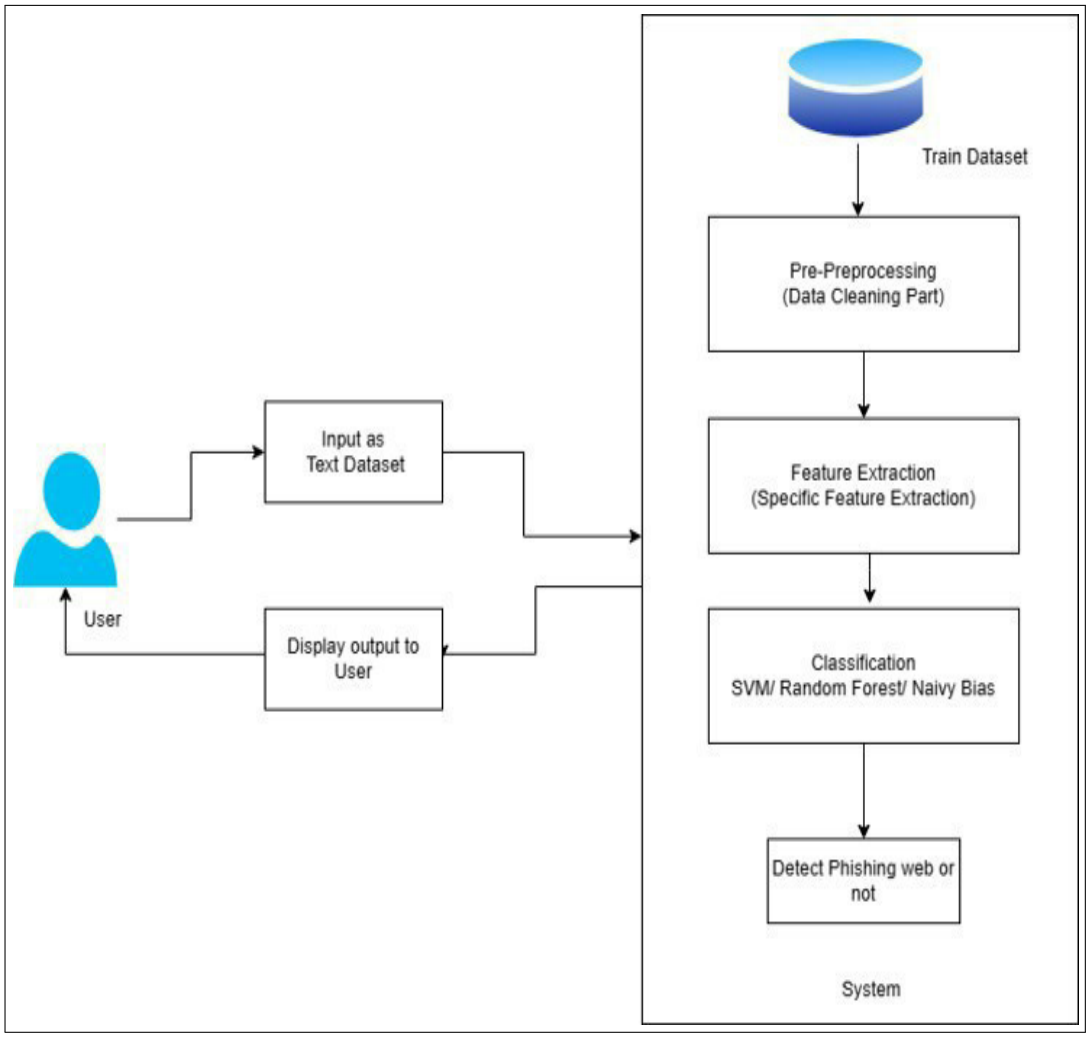


Figure 7.1: System Architecture

7.3 DATA DESIGN (USING APPENDICES A AND B)

7.3.1 Internal software data structure

When building a URL-based phishing detection system using machine learning algorithms like Support Vector Machines (SVM), Naive Bayes (NB), and Random Forest, the internal data structure revolves around features extracted from the URLs and the corresponding labels indicating whether a URL is benign or malicious (phishing).

7.3.2 Global data structure

- In software development, a "global data structure" could refer to a data structure that is declared and defined in a way that it is accessible from any part of a program, regardless of its scope. In many programming languages, global variables and data structures can be used to store information that needs to be shared across different functions or modules

7.3.3 Temporary data structure

- These data structures are used for intermediate storage of data while performing operations like sorting, filtering, or mapping. Simple variables can be used as temporary storage for intermediate values during calculations or transformations. When parsing data, temporary data structures like parse trees or intermediate representations are used to make the parsing process more efficientand organized.

7.3.4 Database description

DB Browser for SQLite (DB4S) is a high quality, visual, open source tool to create, design, and edit database files compatible with SQLite.

DB4S is for users and developers who want to create, search, and edit databases.

DB4S uses a familiar spreadsheet-like interface, and complicated SQL commands do not have to be learned.

Controls and wizards are available for users to:

Create and compact database files Create, define, modify and delete tables Create, define, and delete indexes Browse, edit, add, and delete records Search records Import and export records as text Import and export tables from/to CSV files Import and export databases from/to SQL dump files Issue SQL queries and inspect the results Examine a log of all SQL commands issued by the application Plot simple graphs based on table or query data.

7.4 COMPOENT DESIGN

7.4.1 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

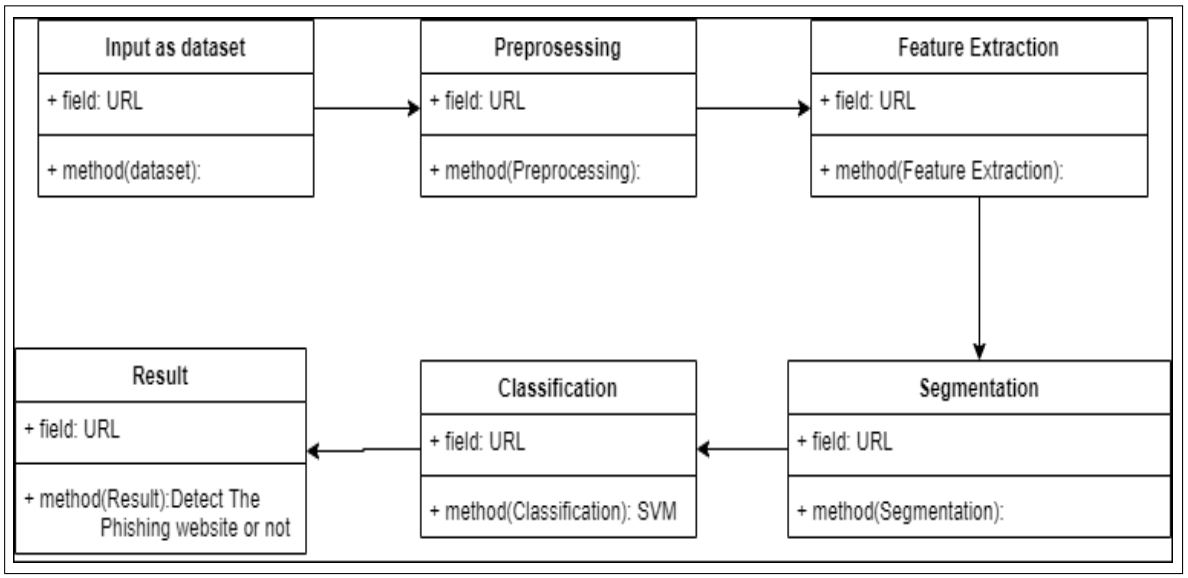


Figure 7.2: Class Diagram

7.5 Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages ex- changed between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

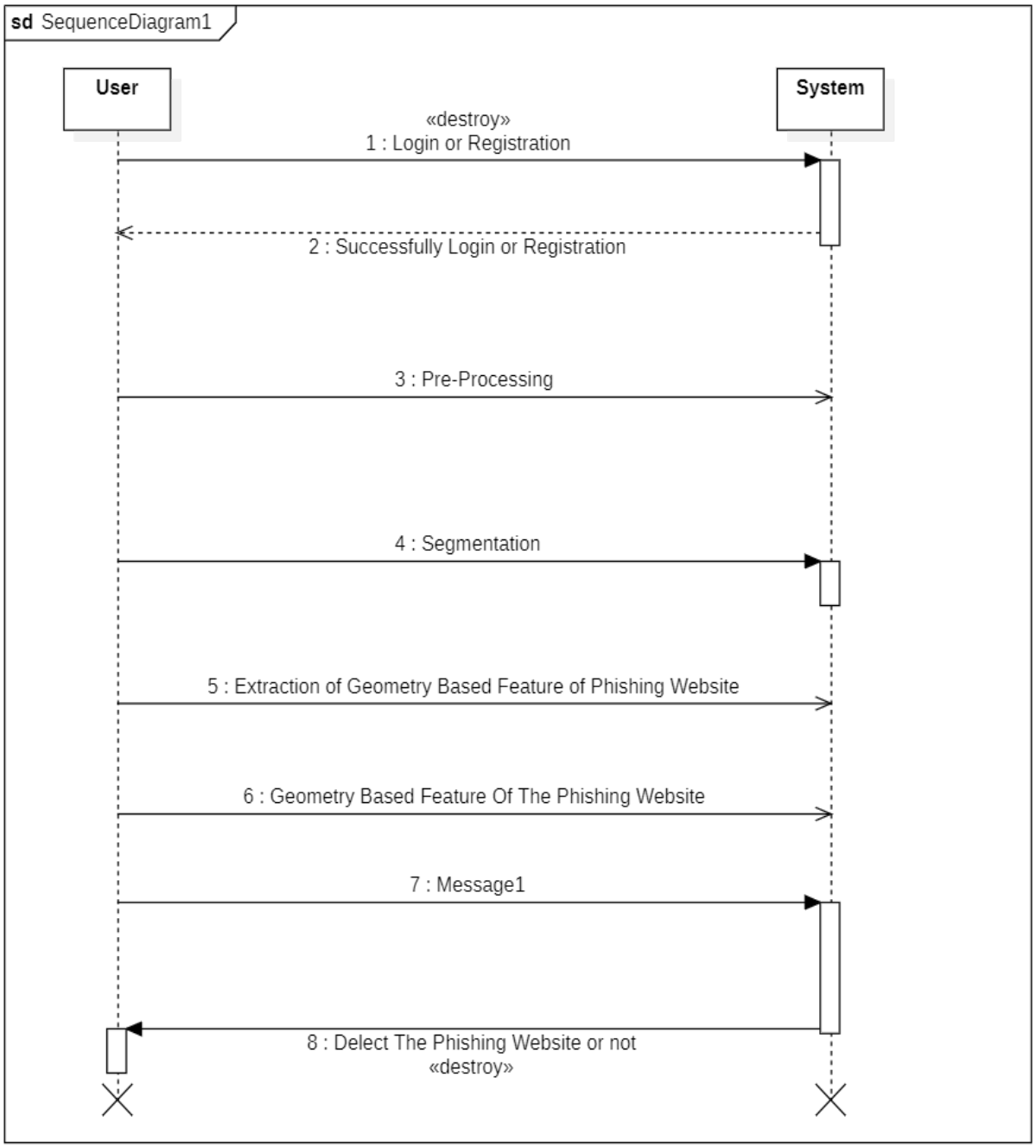


Figure 7.3: Sequence Diagram

7.5.1 Entity Relationship Diagram

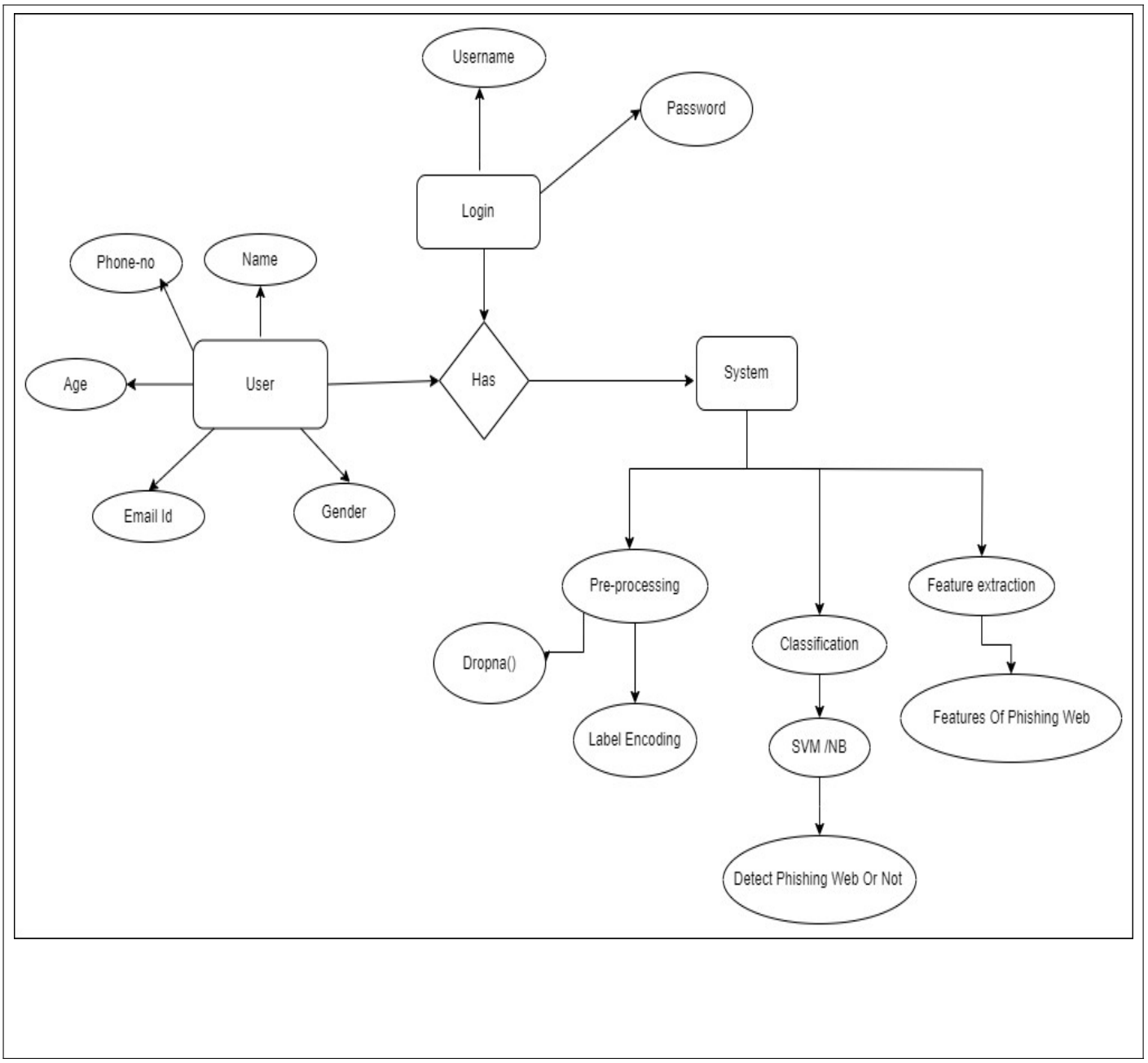


Figure 7.4: ER Diagram

CHAPTER 8

PROJECT IMPLEMENTATION

8.1 INTRODUCTION

In this chapter we are going to have an overview about how much time does it took to complete each task like- Preliminary Survey Introduction and Problem Statement, Literature Survey, Project Statement, Software Requirement and Specification, System Design, Partial Report Submission, Architecture De- sign, Implementation, Deployment, Testing, Paper Publish, Report Submission. This chapter also gives focus on stakeholder list which gives information about project type, customer of the proposed system, user and project member who developed the system.

Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant white space. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python was created in the late 1980s as a successor to the ABC language. Python 3.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

Python interpreters are available for many operating systems. A global community of programmers develops and maintains CPython, a free and open- source reference implementation. A non-profit organization, the Python Soft- ware Foundation, manages and directs resources for Python and CPython development.

8.2 TOOLS AND TECHNOLOGIES USED

Anaconda: Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda,

Inc., which was founded by Peter Wang and Travis Oliphant in 2012. As an Anaconda, Inc. product, it is also known as Anaconda Distribution or Anaconda Individual Edition, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free.

Package versions in Anaconda are managed by the package management system conda. This package manager was spun out as a separate open-source package as it ended up being useful on its own and for other things than Python. There is also a small, bootstrap version of Anaconda called Mini-conda, which includes only conda, Python, the packages they depend on, and a small number of other packages. The big difference between conda and the pip package manager is in how package dependencies are managed, which is a significant challenge for Python data science and the reason conda exists.

When pip installs a package, it automatically installs any dependent Python packages without checking if these conflict with previously installed pack- ages [citation needed]. It will install a package and any of its dependencies regardless of the state of the existing installation [citation needed]. Because of this, a user with a working installation of, for example, Google Tensorflow, can find that it stops working having used pip to install a different package that requires a different version of the dependent numpy library than the one used by Tensorflow. In some cases, the package may appear to work but produce different results in detail.

Spyder

Spyder is a powerful scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It offers a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package.

Beyond its many built-in features, its abilities can be extended even further via its plugin system and API. Furthermore, Spyder can also be used as a PyQt5 extension library, allowing you to build upon its functionality and embed its components, such as the interactive console, in your own software.

Features

- Editor

  Work efficiently in a multi-language editor with a function/class browser, real-time code analysis tools (pyflakes, pylint, and pycodestyle), automatic code completion (jedi and rope), horizontal/vertical splitting, and go-to-definition.

- Interactive console

  Harness the power of as many IPython consoles as you like with full workspace and debugging support, all within the flexibility of a full GUI interface. Instantly run your code by line, cell, or file, and render plots right in line with the output or in interactive windows.

- Documentation viewer

  Render documentation in real-time with Sphinx for any class or function, whether external or user-created, from either the Editor or a Console.

8.3 METHODOLOGIES/ALGORITHM DETAILS

1. Support Vector Machines (SVM):

Algorithm Details:

- Feature Extraction: Extract relevant features from URLs such as domain length, presence of suspicious keywords, use of HTTPS, etc.
- Vectorization: Represent extracted features numerically.
- Training: Train the SVM model using labeled data (phishing or legitimate URLs).
- Kernel Selection: Choose an appropriate kernel function (linear, polynomial, RBF) based on the dataset characteristics.
- Margin Optimization: SVM aims to maximize the margin between different classes, enhancing generalization.
- Prediction: Given a new URL, predict its class (phishing or legitimate) based on the trained model.

2. Naive Bayes (NB):

Algorithm Details:

- Feature Representation: Similar to SVM, extract features from URLs.
- Probability Calculation: Calculate the likelihood of features given each class (phishing or legitimate) using training data.
- Assumption: NB assumes feature independence given the class, simplifying probability calculations.
- Training: Learn the probability distributions for different features and classes.
- Prediction: Given a new URL, calculate the probability of it belonging to each class using Bayes' theorem and select the class with the highest probability.

3. Random Forest:

Algorithm Details:

- Feature Representation: As with SVM and NB, extract relevant features from URLs.
- Ensemble Learning: Train multiple decision trees on random subsets of data and features (bootstrap aggregating).
- Feature Randomization: Randomly select features at each split to promote diversity among trees.
- Voting/Averaging: Combine predictions from multiple trees (voting for classification) to make the final prediction.
- Training: Build an ensemble of decision trees using labeled data.
- Prediction: Given a new URL, aggregate predictions from individual trees to classify it as phishing or legitimate.

8.4 VERIFICATION AND VALIDATION FOR ACCEPTANCE

Verification Testing:

Verification testing for URL-based phishing detection ensures that the implemented machine learning models behave as expected and meet specified requirements in processing URL data and making accurate predictions.

Validation Testing:

Validation testing for URL-based phishing detection focuses on assessing the models' performance and generalization ability on unseen URL data, simulating real-world-scenarios

CHAPTER 9

SOFTWARE TESTING

9.1 INTRODUCTION

- Software testing, depending on the testing method employed, can be implemented at any time in the development process. However, most of the test effort occurs after the requirements have been defined and the coding process has been completed. As such, the methodology of the test is governed by the software development methodology adopted. Different software development models will focus the test effort at different points in the development process. Newer development models, such as Agile, often employ test driven development and place an increased portion of the testing in the hands of the developer, before it reaches a formal team of testers. In a more traditional model, most of the test execution occurs after the requirements have been defined and the coding process has been completed.

9.2 TYPE OF TESTING USED

9.2.1 Unit Testing

- It is the testing of individual software units of the application .It is done after the complexion of an individual unit before integration. Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. This is a structural testing, that relies on knowledge of its construction and is invasive.

- Unit tests perform basic tests at component level and test a specific business process,application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

9.2.2 Integration Testing

- Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

9.2.3 System Test

- To test this application we are going with proper sequencing of testing like unit, integration, validation, GUI, Low level and High level test cases, major scenarios likewise. We will go with the GUI testing first and then integration testing. After integration testing performs the high level test cases and major scenarios which can affect the working on the application. We will perform the testing on the data transmitted using the various inputs and outputs and validate the results.

9.3 WHITE-BOX TESTING

- Software testing methods are traditionally divided into white- and black-box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.
- 1. White-box testing

- In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases.

9.4 BLACK-BOX TESTING

- 2. Black-box testing

- Black-box testing treats the software as a quot;black boxquot;, examining functionality without any knowledge of internal implementation. The testers are only aware of what the software is supposed to do, not how it does it.

9.5 TEST CASES AND TEST RESULTS

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Store Xml File	Xml file	Xml file store	Error Should come	P
002	Parse the xml file for conversion	parsing	File get parse	Accept	P
003	Attribute identification	Check individual Attribute	Identify Attributes	Accepted	P
004	Weight Analysis	Check Weight	Analyze Weight of individual Attribute	Accepted	P
005	Tree formation	Form them-Tree	Formation	Accepted	P
006	Cluster Evaluation	Check Evaluation	Should check Cluster	Accepted	P
007	Algorithm Performance	Check Evaluation	Should work Algorithm Properly	Accepted	P
008	Query Formation	Check Query Correction	Should check Query	Accepted	P

Figure 9.1: GUI TESTING

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Enter the number in username, middle name, last name field	Number	Error Comes	Error Should Comes	P
001	Enter the character in username, middle name, last name field	Character	Accept	Accept	p
002	Enter the invalid email id format in email id field	Kkgmail,com	Error comes	Error Should Comes	P
002	Enter the valid email id format in email id field	kk@gmail.com	Accept	Accept	P
003	Enter the invalid digit no in phone no field	99999	Error comes	Error Should Comes	P
003	Enter the 10 digit no in phone no field	9999999999	Accept	Accept	P

Figure 9.2: Registration test case

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Enter The Wrong username or password click on submit button	Username or password	Error comes	Error Should come	P
002	Enter the correct username and password click on submit button	Username and password	Accept	Accept	P

Figure 9.3: Login test case

CHAPTER 10

RESULTS

10.1 SCREEN SHOTS

10.2 OUTPUTS

Figure 10.1: Login or Sign up Page

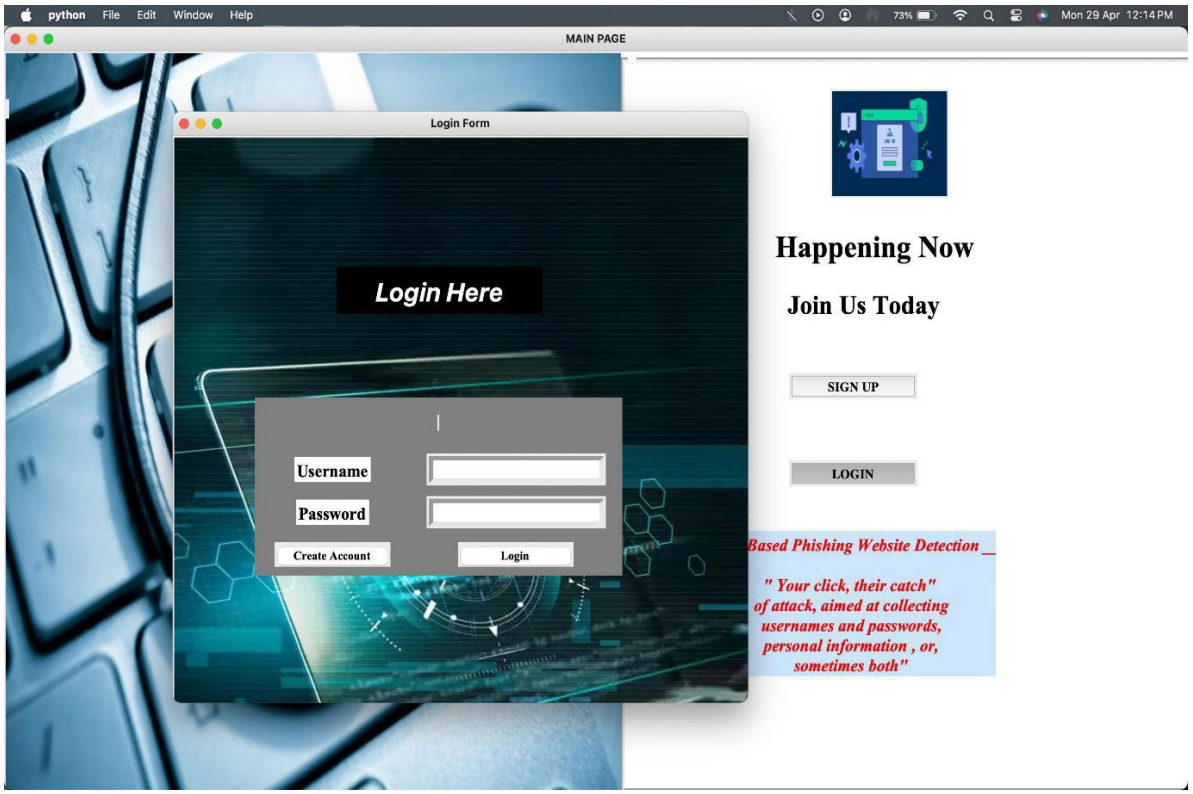


Figure 10.2: Login Page

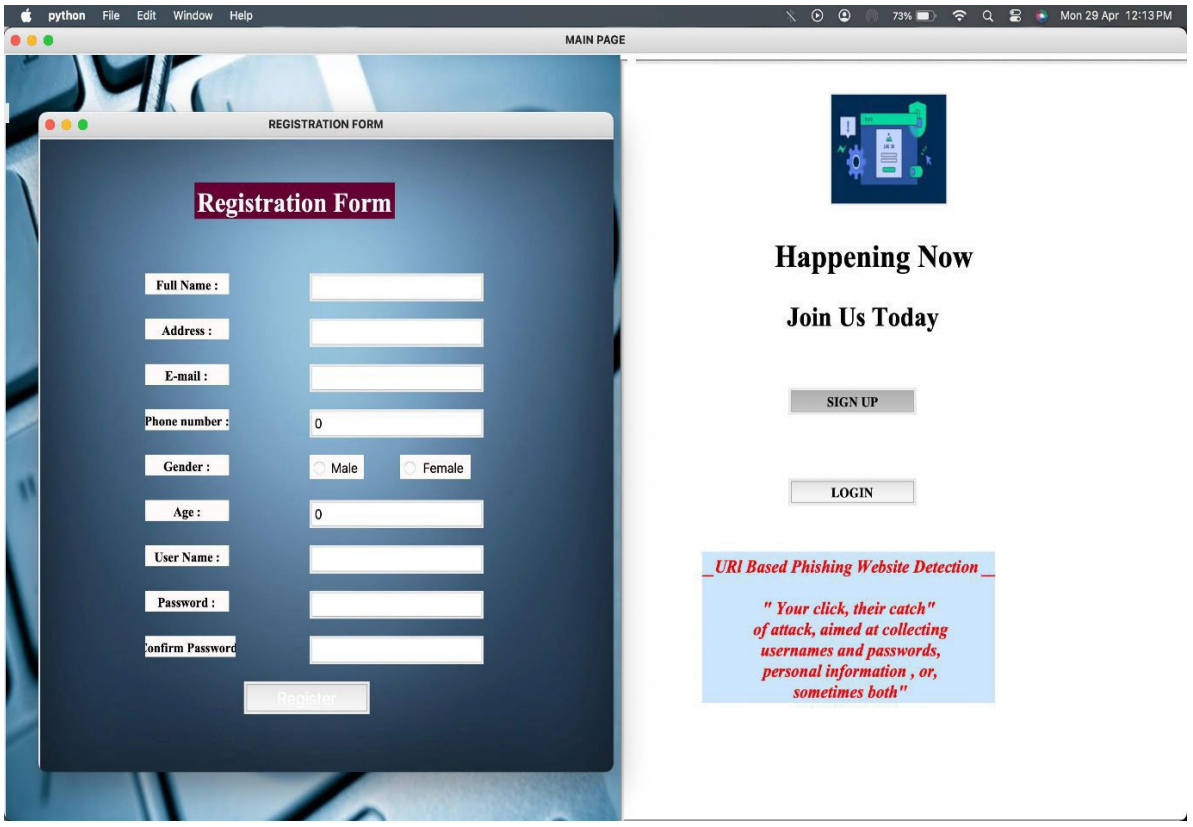


Figure 10.3: Sign up Page

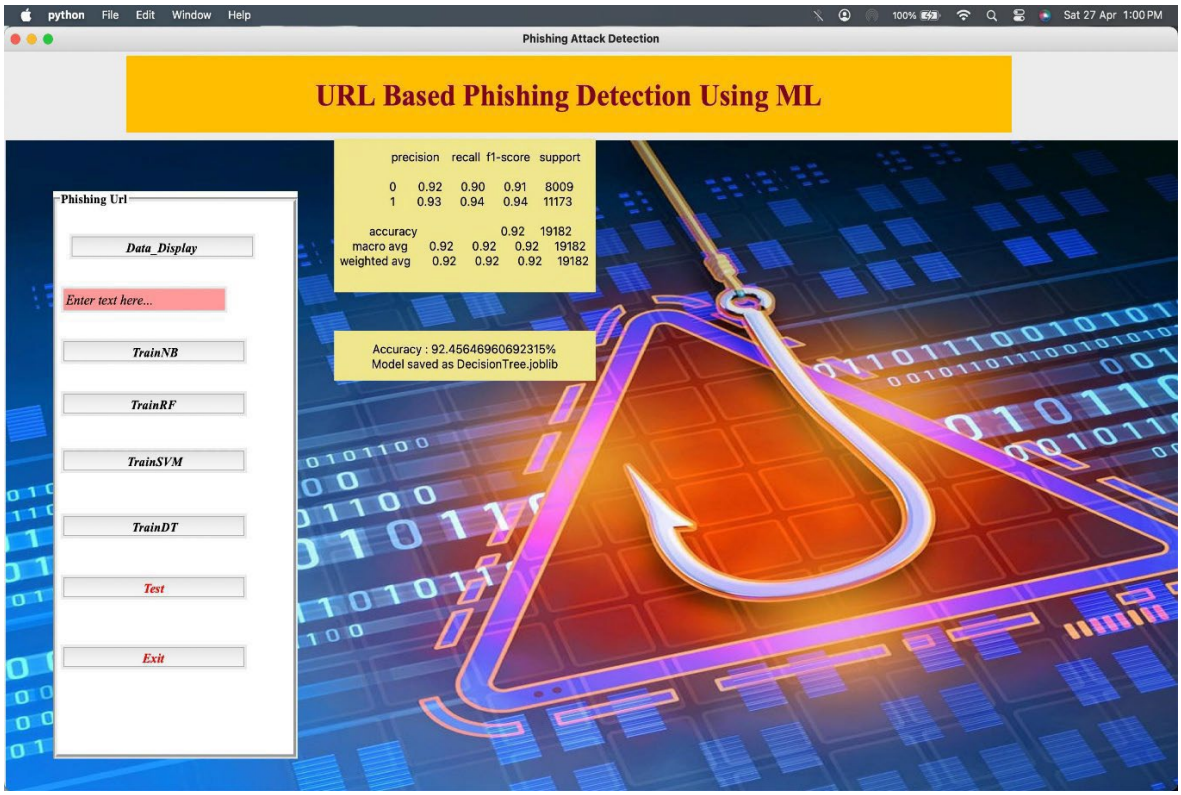


Figure 10.4: Accuracy using DecisionTree

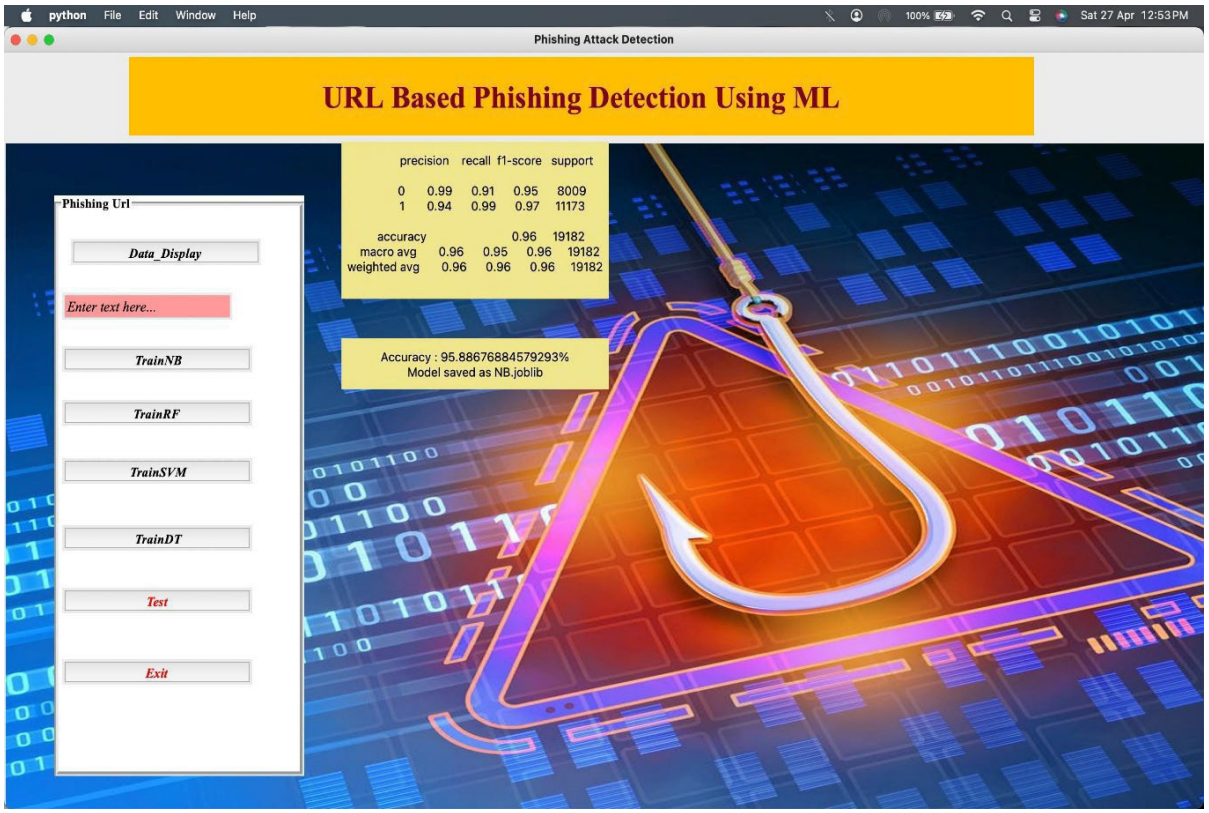


Figure 10.5: Accuracy using NB

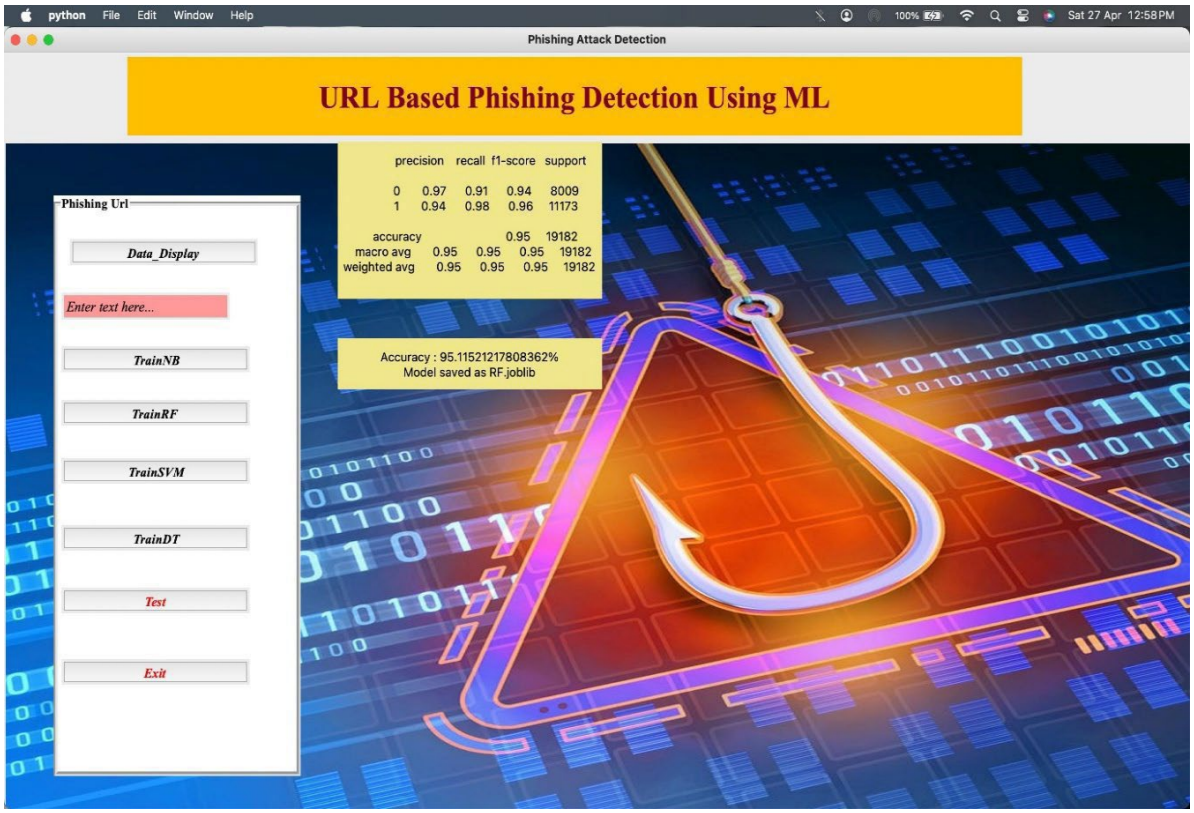


Figure 10.6: Accuracy using RF

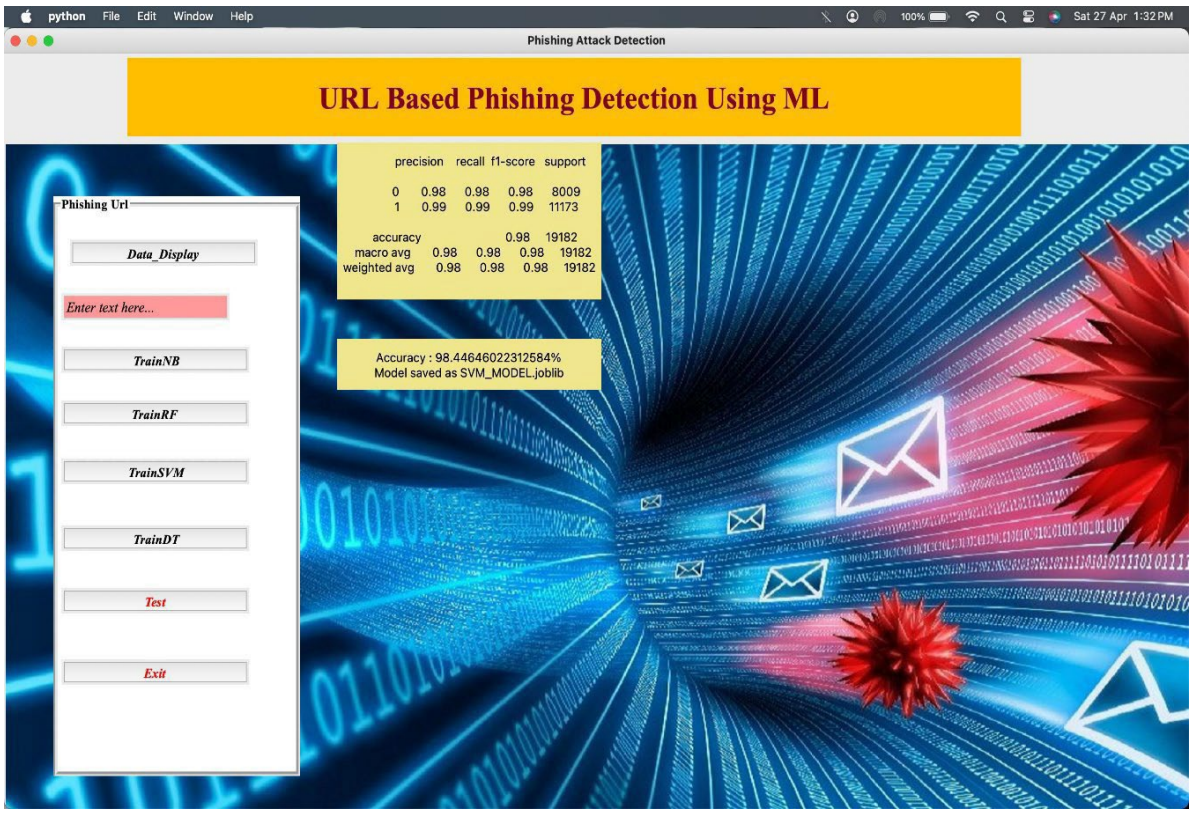


Figure 10.7: Accuracy using SVM

CHAPTER 11

DEPLOYMENT AND MAINTENANCE

11.1 INSTALLATION AND UN-INSTALLATION

Annaconda:

What is Anaconda Navigator? Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® distribution that allows you to launch applications and easily manage conda packages, environments, and channels without using command-line commands. Navigator can search for packages on Anaconda.org or in a local Anaconda Repository. It is available for Windows, macOS, and Linux. To get Navigator, get the Navigator Cheat Sheet and install Anaconda.

The Getting started with Navigator section shows how to start Navigator fromthe shortcuts or from a terminal window.

11.2 USER HELP

1] pip install tenorflow
2] pip install matplot lib
3] pip install keras
4] pip install pillow
5] pip install numpy
6] pip install opencv-python
7] pip install matplot lib
8] pip install scikit-learn
9] pip install tkvideo
10] pip install mediapipe
11] pip install gtts
12] pip install pandas
13] pip install flask
14] pip install mixtend

What applications can I access using Navigator?

The following applications are available by default in Navigator:

1. JupyterLab
2. Jupyter Notebook
3. Spyder
4. PyCharm
5. VSCode
6. Glueviz
7. Orange 3 App
8. RStudio

CHAPTER 12

CONCLUSION AND FUTURE SCOPE

- In summary, the use of machine learning algorithms, such as Support Vector Machines (SVM) and Naive Bayes (NB), for phishing website detection is a critical component of modern cybersecurity. Detecting phishing websites is essential in safeguarding users and organizations from online threats. Support Vector Machines (SVM) and Naive Bayes (NB) are two commonly used machine learning algorithms for identifying phishing websites. The collected dataset is divided into training and testing sets. The model is trained using the training data, and its performance is evaluated using the testing data. Evaluation metrics such as accuracy, precision, recall, and F1-score are used to measure model performance.
- Phishing website detection using machine learning techniques represents a crucial advancement in the field of cybersecurity. In the face of rapidly evolving and increasingly sophisticated phishing attacks, traditional methods have proven insufficient. This study has explored the implementation of intelligent algorithms and real-time analysis to counter the menace of phishing websites effectively. As technology advances, ongoing research and collaboration between the cybersecurity community and machine learning experts will be paramount in staying one step ahead of cyber- criminals, ensuring a safer digital landscape for all.

CHAPTER 13

REFERENCES

- 1. A.Y. Ahmad, M. Selvakumar, A. Mohammed, and A.-S. Samer, “TrustQR: A new technique for the detection of phishing attacks on QR code,” Adv. Sci. Lett., vol. 22, no. 10,
  pp. 2905-2909, Oct.2021.

- 2.C. C. Inez and F. Baruch, “Setting priorities in behavioral interventions: An application to reducing phishing risk,” Risk Anal., vol. 38, no. 4, pp. 826-838, Apr. 2021.

- 3.Aburrous, Maher Hossain, Mohammed Dahal, Keshav Thabtah, Fadi. (2020). Intelligent phishing detection system for ebanking using fuzzy data mining. Expert Systems with Applications. 37. 7913-7921. 10.1016/j.eswa.2020.04.044.

- 4.Rosiello, Angelo Kirda, Engin Kruegel, Ferrandi, Fabrizio. (2007). A layout-similarity-based approach for detecting phishing pages. Proceedings of the 3rd International Conference on Security and Privacy in Communication Networks, Secure Comm. 454 -
  463. 10.1109/SECCOM..4550367.2021.

- 5.Chawathe, Sudarshan. Improving Email Security with Fuzzy Rules. 1864- 1869. 10.1109/TrustCom/BigDataSE.2018.00282. 2021.

- 6. A. Aggarwal, A. Rajadesingan and P. Kumaraguru, “PhishAri: Automatic realtime phishing detection on twitter,” eCrime Researchers Summit, Las Croabas, 2012, pp. 1-12, doi: 10.1109/eCrime.6489521 2022.

- 7.P. Singh, Y. P. S. Maravi and S. Sharma, “Phishing Websites Detection through Supervised Learning Networks”, 2020 International Conference on Computing and Communications Technologies (ICCCT), Chennai, 2020, pp. 61-65.

- 8.K. Thomas, C. Grier, J. Ma, V. Paxson and D. Song, “Design and Evaluation of a Real- Time URL Spam Filtering Service”, IEEE Symposium on Security and Privacy, Berkeley, CA, pp. 447-462. 2021

- 9.C.V. Arulkumar et al., “Secure Communication in Unstructured P2P Networks based on Reputation Management and Self certification”, International Journal of Computer Applications, vol. 15, pp. 1-3,

- 10. H. Shahriar and M. Zulkernine, “Information Source-based Classification of Automatic Phishing Website Detectors”, 2022 IEEE/IPSJ International Symposium on Applications and the Internet, Munich, Bavaria, pp. 190-195. 2022

ANNEXURE A

PROJECT PLANNER

Schedule		Date	Project Activity
July	1st Week	01/07/2023	Project Topic Searching
	2nd Week	08/07/2023	Project Topic Selection
	3rd Week	15/07/2023	Synopsis Submission
August	1st Week	05/08/2023	Presentation On Project Ideas
	2nd Week	12/08/2023	Submission Of Literature Survey
	3rd Week	19/08/2023	Feasibility Assessment
September	1st Week	02/09/2023	Documentation for paper publishing.
	3rd Week	16/09/2023	Design Of Mathematical Model
	4th Week	24/09/2023	Paper is publish.
October	1st Week	09/10/2023	Report Preparation And Submission
December	3rd Week	19/12/2023	1st module presentation
	4th Week	26/12/2023	Discussion and implementation of 2nd module
January	1st Week	02/01/2024	Preparation for conference
	2nd Week	09/01/2024	Study of algorithm.
	3rd Week	16/01/2024	Discussion about modification.
	4th Week	24/01/2024	1st and 2nd module presentation
	5th Week	30/01/2024	Discussion on flow of project and designing new module
February	1st Week	06/02/2024	Modification of modules.
	2nd Week	13/02/2024	Designed test cases for our module.
	3rd Week	20/02/2024	Worked on user interface.
March	1st Week	06/03/2024	Integration of all modules.
April	1st Week	8/04/2024	Final Report.
May	1st Week	10/05/2024	Final Presentation.

ANNEXURE B

REVIEWERS COMMENTS OF PAPER SUBMITTED

Paper 1

Research paper

1. **Paper Title**: URL BASED PHISHING DETECTION
2. **Name of the Conference/Journal where paper submitted** : IJRAR.ORG
3. **Paper accepted/rejected** : Accepted
4. **Review comments by reviewer** : None
5. **Corrective actions if any** : None

Paper 2
Survey paper

1. **Paper Title**: URL BASED PHISHING DETECTION
2. **Name of the Conference/Journal where paper submitted** : IJRAR
3. **Paper accepted/rejected :** Accepted
4. **Review comments by reviewer** : None
5. **Corrective actions if** any : None

ANNEXURE B
PUBLISHED PAPER AND CERTIFICATES

IJRAR.ORG E-ISSN: 2348-1269, P-ISSN: 2349-5138

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

URL-Based Phishing Detection

[1]Yadnyesh Chaudhari, [2]Pradnya Kasture, [3]Sanket Shendge, [4]Gautam Sharma

[1]B.E 4th year, [2]Professor, [3]B.E 4th year, [4]B.E 4th year,
Department of Computer Engineering,
RMDSSOE, Pune-411058, India

Abstract *:* Phishing attacks pose a significant threat to online security, targeting individuals and organizations with deceptive emails and websites designed to steal sensitive information. Traditional methods of detecting phishing websites have become inadequate against the evolving tactics employed by cybercriminals. This study explores the application of machine learning techniques in phishing website detection, aiming to enhance accuracy and real-time response capabilities. The results highlight the significance of this research in strengthening online security. By leveraging machine learning techniques, the proposed system provides a proactive defense against phishing attacks, safeguarding users, businesses, and organizations from financial losses, identity theft, and reputational damage. Furthermore, the study underscores the importance of continuous research and collaboration in the ever-changing landscape of cybersecurity, ensuring a safer digital environment for all.

Index Terms - Feature Extraction, SVM, Classification, Model-training, RF, NB.

I. INTRODUCTION

Phishing websites pose a significant threat to online users, aiming to deceive them into revealing sensitive information such as usernames, passwords, and financial details. Detecting these malicious websites is crucial for ensuring online security. In this context, the use of advanced techniques such as Feature Extraction, Support Vector Machines (SVM), Random Forest (RF), and Naive Bayes (NB) can significantly enhance the effectiveness of phishing website detection.

Phishing is a cyber-attack method where attackers create deceptive websites that mimic legitimate ones, tricking users into divulging confidential information. Phishing attacks often rely on social engineering techniques to exploit human psychology. Detecting phishing websites is challenging due to their dynamic and evolving nature. Therefore, employing sophisticated techniques becomes imperative to stay ahead of cyber threats.

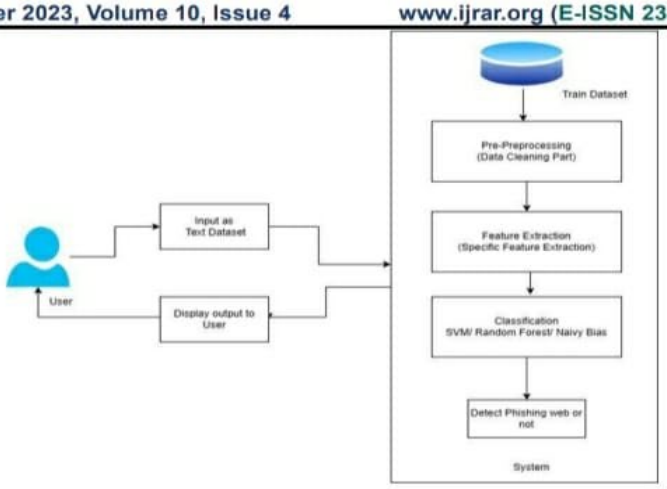
Phishing Website Detection Using Feature Extraction:

Feature extraction involves selecting and transforming relevant information from the raw data to create a feature set that can be used for analysis. In the context of phishing website detection, extracting relevant features from URLs, HTML, JavaScript, and content is crucial.

Support Vector Machines (SVM): SVM is a powerful classification algorithm that works well in high-dimensional spaces. It separates data into different classes by finding the hyperplane that maximizes the margin between classes. SVM can effectively classify phishing and legitimate websites based on extracted features.

Random Forest (RF): RF is an ensemble learning algorithm that builds multiple decision trees and merges their predictions. It is robust, handles high-dimensional data well, and is less prone to overfitting. Random Forest can provide a reliable classification model for phishing website detection.

Naive Bayes (NB): NB is a probabilistic algorithm based on Bayes' theorem. Despite its simplicity, NB can perform well in certain scenarios, especially with a large number of features. It is computationally efficient and can handle real-time classification tasks.



II. Related Work

In this section, we review prior research relevant to our proposed system. We categorize related work into several key areas that have a direct bearing on our research.

1."Trust QR: A New Technique for the Detection of Phishing Attacks on QR Code"

Graphic black and white squares, known as Quick Response (QR) code is a matrix barcode, that allows easy interaction between mobile and websites or printed material by doing away with the necessity of manually typing a URL or contact information. From the pages of magazines to the sides of buses and billboards, QR code technology is being used increasingly in smartphones. Unfortunately, Phishers have started using QR codes for phishing attacks by using some features of QR codes. This paper introduces a new approach called "Trust QR" which detects URL phishing on QR code. It uses QR code-specific features and URL features to detect if the QR code content has a phishing URL. Some of the QR code-specific features use QR code content and its characteristics like length, type, and level of error correction to generate the cryptography key. This technique uses the machine learning classification technique.

2."Setting priorities in behavioral interventions: An application to reducing phishing risk,"

Phishing risk is a growing area of concern for corporations, governments, and individuals. Given the evidence that users vary widely in their vulnerability to phishing attacks, we demonstrate an approach for assessing the benefits and costs of interventions that target the most vulnerable users. Our approach uses Monte Carlo simulation to (1) identify which users were most vulnerable, in signal detection theory terms; (2) assess the proportion of system-level risk attributable to the most vulnerable users; (3) estimate the monetary benefit and cost of behavioral interventions targeting different vulnerability levels; and (4) evaluate the sensitivity of these results to whether the attacks involve random or spear phishing. Using parameter estimates from previous research, we find that the most vulnerable users were less cautious and less able to distinguish between phishing and legitimate emails (positive response bias and low sensitivity, in signal detection theory terms). They also accounted for a large share of phishing risk for both random and spear phishing attacks. Under these conditions, our analysis estimates much greater net benefit for behavioral interventions that target these vulnerable users. Within the range of the model's assumptions, there was generally net benefit even for the least vulnerable users. However, the differences in the return on investment for interventions with users with different degrees of vulnerability indicate the importance of measuring that performance, and letting it guide interventions. This study suggests that interventions to reduce response bias, rather than to increase sensitivity, have greater net benefit.

3."Intelligent phishing detection system for e-banking using fuzzy data mining"

Detecting and identifying any phishing websites in real-time, particularly for e-banking, is really a complex and dynamic problem involving many factors and criteria. Because of the subjective considerations and the ambiguities involved in the detection, fuzzy data mining techniques can be an effective tool in assessing and identifying phishing websites for e-banking since it offer a more natural way of dealing with quality factors rather than exact values. In this paper, we present a novel approach to overcome the 'fuzziness' in the e-banking phishing website assessment and propose an intelligent resilient and effective model for detecting e-banking phishing websites. The proposed model is based on fuzzy logic combined with data mining algorithms to characterize the e-banking phishing website factors and to investigate its techniques by classifying the phishing types and defining six e-banking phishing website attack criteria with a layer structure. Our experimental results showed the significance and importance of the e-banking phishing website criteria (URL & Domain Identity) represented by layer one and the various influence of the phishing characteristic on the final e-banking phishing website rate.

4."A layout-similarity-based approach for detecting phishing pages"

Phishing is a current social engineering attack that results in online identity theft. In a phishing attack, the attacker persuades the victim to reveal confidential information by using web site spoofing techniques. Typically, the captured information is then used to make an illegal economic profit by purchasing goods or undertaking online banking transactions. Although simple in nature, because of their effectiveness, phishing attacks still remain a great source of concern for organizations with online customer services. In previous work, we have developed AntiPhish, a phishing protection system that prevents sensitive user information from being entered on phishing sites. The drawback is that this system requires cooperation from the user and occasionally raises false alarms. In this paper, we present an extension of our system (called DOMAntiPhish) that mitigates the shortcomings of our previous system. In particular, our novel approach leverages layout similarity information to distinguish between malicious and benign web pages. This makes it possible to reduce the involvement of the user and significantly reduces the false alarm rate. Our experimental evaluation demonstrates that our solution is feasible in practice.

5."Improving Email Security with Fuzzy Rules"

Phishing and other malicious email messages are increasingly serious security threats. An important tool for countering such email threats is the automated or semiautomated detection of malicious email. This paper reports work on using fuzzy rules to classify email for such purposes. The effectiveness of a fuzzy rule-based classifier is studied experimentally on a real dataset and compared with results for other classifiers, including those based on crisp rules and decision trees. The human readability and editability of the classifiers produced by these methods is also studied.

6. "Phishing : Automatic real-time phishing detection on Twitter,"

With the advent of online social media, phishers have started using social networks like Twitter, Facebook, and Foursquare to spread phishing scams. Twitter is an immensely popular micro-blogging network where people post short messages of 140 characters called tweets. It has over 100 million active users who post about 200 million tweets every day. Phishers have started using Twitter as a medium to spread phishing because of this vast information dissemination. Further, it is difficult to detect phishing on Twitter unlike emails because of the quick spread of phishing links in the network, the short size of the content, and use of URL obfuscation to shorten the URL. Our technique, Phishing Ari, detects phishing on Twitter in real-time. We use Twitter-specific features along with URL features to detect whether a tweet posted with a URL is phishing or not. Some of the Twitter-specific features we use are tweet content and its characteristics like length, hashtags, and mentions. Other Twitter features used are the characteristics of the Twitter user posting the tweet such as the age of the account, number of tweets, and the follower-followed ratio. These Twitter-specific features coupled with URL-based features prove to be a strong mechanism for detecting phishing tweets. We use machine learning classification techniques and detect phishing tweets with an accuracy of 92.52%. We have deployed our system for end-users by providing an easy-to-use Chrome browser extension that works in real-time and classifies a tweet as phishing or safe. We show that we are able to detect phishing tweets at zero hour with high accuracy which is much faster than public blacklists and as well as Twitter's own defense mechanism to detect malicious content. To the best of our knowledge, this is the first real-time, comprehensive and usable system to detect phishing on Twitter.

7. "Phishing websites detection through supervised learning networks"

Phishing is an unlawful activity of making gullible people to reveal their insightful information into fake websites. The Aim of these phishing websites is to acquire confidential information such as usernames, passwords, banking credentials and some other personal information. Phishing website looks similar to legitimate website therefore people cannot make difference among them. Today users are heavily relying on the internet for online purchasing, ticket booking, bill payments, etc. As technology advances, the phishing approaches being used are also getting progressed and hence it stimulates anti-phishing methods to be upgraded. In this paper, we have implemented two algorithms named Adaline and Backprop ion along with the support vector machine to enhance the detection rate and classification.

8. "Design and Evaluation of a Real-Time URL Spam Filtering Service"

On the heels of the widespread adoption of web services such as social networks and URL shorteners, scams, phishing, and malware have become regular threats. Despite extensive research, email-based spam filtering techniques generally fall short for protecting other web services. To better address this need, we present Monarch, a real-time system that crawls URLs as they are submitted to web services and determines whether the URLs direct to spam. We evaluate the viability of Monarch and the fundamental challenges that arise due to the diversity of web service spam. We show that Monarch can provide accurate, real-time protection, but that the underlying characteristics of spam do not generalize across web services. In particular, we find that spam targeting email qualitatively differs in significant ways from spam campaigns targeting Twitter. We explore the distinctions between email and Twitter spam, including the abuse of public web hosting and redirector services. Finally, we demonstrate Monarch's scalability, showing our system could protect a service such as Twitter -- which needs to process 15 million URLs/day -- for a bit under $800/day.

9. "Trust based communication in unstructured P2P networks using reputation management and self certification mechanism"

In unstructured P2P networks there is a possibility of malicious codes and false transactions. It generates the false identities in order to perform false transactions with other identities. The proposed method uses the concept of DHT and reputation management which provides efficient file searching. The self certification (RSA and MD5) is used for ensuring secure and timely availability of the reputation data of a peer to other peers. The reputations of the peers are used to determine whether a peer is a malicious peer or a good peer. Once the malicious peer is detected the transaction is aborted. The reputation of a given peer is attached to its identity. The identity certificates are generated using self- certification and all peers maintain their own (and hence trusted) certificate authority which issues the identity certificate(s) and digital signature to the peer.

10. "Information Source-Based Classification of Automatic Phishing Website Detectors"

Phishing attacks allure users to submit their personal information to fake websites that mimic legitimate websites. Many anti-phishing techniques have emerged in recent years. However, the number of phishing attacks is still increasing. Two reasons can be blamed for this situation. First, users have too much trust and confidence in existing anti-phishing tools in general. Second, most users believe that they are foolproof against phishing attacks when anti-phishing tools are deployed. We believe that understanding anti-phishing tools based on their common features can be the beginning step to address these issues. However, there is no extensive analysis of existing anti-phishing techniques. This paper attempts to classify existing works based on information sources. The classification would not only provide useful information to develop new anti-phishing techniques or improve existing techniques but also enable our understanding of the limitations of the existing techniques.

III. CONCLUSION

Combining advanced feature extraction techniques with machine learning algorithms like SVM, RF, and NB provides a potent solution for detecting phishing websites. This multi-faceted approach allows for a comprehensive analysis of various aspects of phishing attacks, enabling more accurate and robust detection in the ever-evolving landscape of cyber threats. As cybercriminals continue to refine their tactics, ongoing research and adaptation of detection techniques are essential to stay one step ahead in the ongoing battle for online security.

IV. REFERENCES

[1]. A.Y. Ahmad, M. Selvakumar, A. Mohammed, and A.-S. Samer, "TrustQR: A new technique for the detection of phishing attacks on QR code," Adv. Sci. Lett., vol. 22, no. 10, pp. 2905-2909, Oct.2021.

[2].C.C.Inez and F. Baruch,"Setting priorities in behavioral interventions: An application to reducing phishing risk," Risk Anal., vol. 38, no. 4, pp. 826-838, Apr. 2021.

[3]. Aburrous, Maher Hossain, Mohammed Dahal, Keshav Thabtah, Fadi. (2020). Intelligent phishing detection system for ebanking using fuzzy datamining. Expert Systems with Applications. 37. 7913-7921. 10.1016/ j.eswa.2020.04.044.

[4].Rosiello, Angelo Kirda, Engin Kruegel, Ferrandi, Fabrizio. (2007). A layout-similarity-based approach for detecting phishing pages. Proceedings of the 3rd International Conference on Security and Privacy in Communication

[5].Chawathe, Sudarshan. Improving Email Security with Fuzzy Rules. 1864-1869. 10.1109/TrustCom/ BigDataSE.2018.00282. 2021.

[6]. A. Aggarwal, A. Rajadesingan and P. Kumaraguru, "PhishAri: Automatic realtime phishing detection on twitter," eCrime Researchers Summit, Las Croabas, 2012, pp. 1-12, doi: 10.1109/eCrime.6489521 2022.

[7].P. Singh, Y. P. S. Maravi and S. Sharma, "Phishing Websites Detection through Supervised Learning Networks", 2020 International Conference on Computing and Communications Technologies (ICCCT), Chennai, 2020, pp. 61-65.

[8].K. Thomas, C. Grier, J. Ma, V. Paxson and D. Song, "Design and Evaluation of a Real-Time URL Spam Filtering Service", IEEE Symposium on Security and Privacy, Berkeley, CA, , pp. 447-462. 2021

[9].C.V. Arulkumar et al., "Secure Communication in Unstructured P2P Networks based on Reputation Management and Self certification", International Journal of Computer Applications, vol. 15, pp. 1-3,

[10]. H. Shahriar and M. Zulkernine, "Information Source- based Classification of Automatic PhishingWebsite Detectors", 2022 IEEE/IPSJ International Symposium on Applications and the Internet, Munich, Bavaria, pp. 190-195. 2022.


Certificate of Publication
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG
An International Open Access, Peer-reviewed, Refereed Journal
E-ISSN: 2348-1269, P-ISSN: 2349-5138
The Board of
International Journal of Research and Analytical Reviews (IJRAR)
Is hereby awarding this certificate to
Yadnyesh Chaudhari
In recognition of the publication of the paper entitled
URL Based Phishing Detection
Published In IJRAR (www.ijrar.org) UGC Approved - Journal No : 43602 & 7.17 Impact Factor
Volume 10 Issue 4 December 2023, Date of Publication: 08-December-2023
PAPER ID : IJRAR23D2815
Registration ID : 279157
R.B.Joshi
EDITOR IN CHIEF
UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014
Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org
IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138


Certificate of Publication
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG
An International Open Access, Peer-reviewed, Refereed Journal
E-ISSN: 2348-1269, P-ISSN: 2349-5138
The Board of
International Journal of Research and Analytical Reviews (IJRAR)
Is hereby awarding this certificate to
Gautam Sharma
In recognition of the publication of the paper entitled
URL Based Phishing Detection
Published In IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor
Volume 10 Issue 4 December 2023, Date of Publication: 08-December-2023
PAPER ID : IJRAR23D2815
Registration ID : 279157
R.B.Joshi
EDITOR IN CHIEF
UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014
Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org
IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138


Certificate of Publication
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG
An International Open Access, Peer-reviewed, Refereed Journal
E-ISSN: 2348-1269, P-ISSN: 2349-5138
The Board of
International Journal of Research and Analytical Reviews (IJRAR)
Is hereby awarding this certificate to
Sanket Shendge
In recognition of the publication of the paper entitled
URL Based Phishing Detection
Published In IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor
Volume 10 Issue 4 December 2023, Date of Publication: 08-December-2023
PAPER ID : IJRAR23D2815
Registration ID : 279157
R.B.Joshi
EDITOR IN CHIEF
UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014
Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org
IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138


Certificate of Publication
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG
An International Open Access, Peer-reviewed, Refereed Journal
E-ISSN: 2348-1269, P-ISSN: 2349-5138
The Board of
International Journal of Research and Analytical Reviews (IJRAR)
Is hereby awarding this certificate to
Pradnya Kasture
In recognition of the publication of the paper entitled
URL Based Phishing Detection
Published In IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor
Volume 10 Issue 4 December 2023, Date of Publication: 08-December-2023
PAPER ID : IJRAR23D2815
Registration ID : 279157
R.B.Joshi
EDITOR IN CHIEF
UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014
Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org
IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

URL BASED PHISHING DETECTION

1st Mr. Gautam Sharma
Dept. of Computer Science(RMDSSOE)
Savitribai Phule Pune University
Pune, India
sharmagautam09092001@gmail.com

2nd Mrs. Pradnya Kasture
Dept. of Computer Science(RMDSSOE)
Savitribai Phule Pune University
Pune, India
PradnyaKasture.rmdssoe@sinhgad .edu

3rd Mr. Sanket Shendge
Dept. of Computer Science(RMDSSOE)
Savitribai Phule Pune University
Pune, India
sanketshendge.rmdssoe.comp@gmail .com

4th Mr. Yadnyesh Chaudhari
Dept. of Computer Science(RMDSSOE)
Savitribai Phule Pune University
Pune, India
yadnyeshchaudhari875@gmail.com

Abstract –

Phishing attacks pose a significant threat to online security, targeting individuals and organizations with deceptive emails and websites designed to steal sensitive information. Traditional methods of detecting phishing websites have become inadequate against the evolving tactics employed by cybercriminals. This study explores the application of machine learning techniques in phishing website detection, aiming to enhance accuracy and real-time response capabilities. The results highlight the significance of this research in strengthening online security. By leveraging machine learning techniques, the proposed system provides a proactive defense against phishing attacks, safeguarding users, businesses, and organizations from financial losses, identity theft, and reputational damage. Furthermore, the study underscores the importance of continuous research and collaboration in the ever-changing landscape of cybersecurity, ensuring a safer digital environment for all.

Keywords: ***Feature Extraction, SVM, Classification, Model-training, Random Forest, Naive Bayes, Decision Tree***

1.INTRODUCTION

Phishing websites pose a significant threat to online users, aiming to deceive them into revealing sensitive information such as usernames, passwords, and financial details. Detecting these malicious websites is crucial for ensuring online security. In this context, the use of advanced techniques such as Feature Extraction, Support Vector Machines (SVM), Random Forest (RF), and Naive Bayes (NB) can significantly enhance the effectiveness of phishing website detection. Phishing is a cyber-attack method where attackers create deceptive websites that mimic legitimate ones, tricking users into divulging confidential information. Phishing attacks often rely on social engineering techniques to exploit human psychology. Detecting phishing websites is challenging due to their dynamic and evolving nature. Therefore, employing sophisticated techniques becomes imperative to stay ahead of cyber threats.

Phishing Website Detection Using Feature Extraction:

Feature extraction involves selecting and transforming relevant information from the raw data to create a feature set that can be used for analysis. In the context of phishing website detection, extracting relevant features from URLs, HTML, JavaScript, and content is crucial.

Support Vector Machines (SVM):

SVM is a powerful classification algorithm that works well in high-dimensional spaces. It separates data into different classes by finding the hyperplane that maximizes the margin between classes. SVM can effectively classify phishing and legitimate websites based on extracted features.

Random Forest (RF):

RF is an ensemble learning algorithm that builds multiple decision trees and merges their predictions. It is robust, handles high-dimensional data well, and is less prone to overfitting. Random Forest can provide a reliable classification model for phishing website detection.

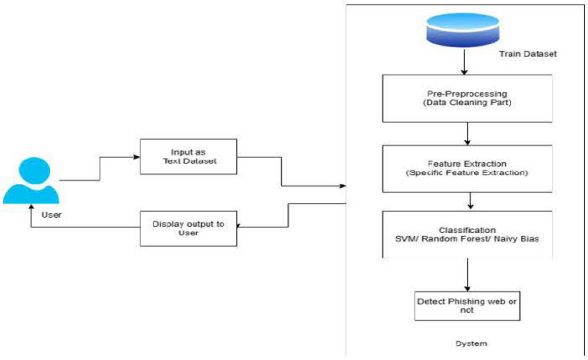
Naive Bayes (NB):

NB is a probabilistic algorithm based on Bayes' theorem. Despite its simplicity, NB can perform well in certain scenarios, especially with a large number of features. It is computationally efficient and can handle real-time classification tasks.

Decision Tree (DT):

A Decision Tree is a versatile machine learning algorithm used for both classification and regression tasks. It models decisions and their possible consequences as a tree-like structure of nodes and branches. Using a Decision Tree for phishing detection based on URLs involves analyzing various features of URLs to distinguish between legitimate and phishing websites.

Here is illustrates diagram that show the process of a phishing detection system that leverages machine learning techniques. Here's a step-by-step explanation of each component and their interactions:



1. **User Interaction**:

Input as Text Dataset: The user provides input, which could be URLs, emails, or text data that needs to be analyzed for phishing.

Display Output to User: After processing, the result is displayed back to the user, indicating whether the input is classified as phishing or not.

2. **System Workflow**:

Train Dataset: This is the initial dataset used to train the machine learning models. It contains labeled examples of phishing and legitimate data.

Pre-Preprocessing (Data Cleaning Part): This step involves cleaning the input data to remove noise and irrelevant information. Common tasks include removing duplicates, handling missing values, and normalizing text.

Feature Extraction (Specific Feature Extraction): Relevant features are extracted from the cleaned data. This might include characteristics such as URL length, presence of certain keywords, domain age, etc., which are indicative of phishing attempts.

Classification: The extracted features are fed into machine learning classifiers. The diagram mentions three possible algorithms:

SVM (Support Vector Machine): A supervised learning model used for classification tasks.

Random Forest: An ensemble learning method that uses multiple decision trees to improve predictive performance.

Naive Bayes: A probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between features.

Decision Tree (DT): A Decision Tree is a versatile machine learning algorithm used for both classification and regression tasks.

Detect Phishing Web or Not: The classifier's output determines whether the input data is classified as phishing or legitimate.

This system combines data cleaning, feature extraction, and machine learning classification to automatically detect and identify phishing attempts, providing real-time feedback to the user.

2. RELATED WORK

In this section, we review prior research relevant to our proposed system. We categorize related work into several key areas that have direct bearing on our research. 1."TrustQR: A New Technique for the Detection of Phishing Attacks on QR Code" Graphic black and white squares, known as Quick Response (QR) code is a matrix barcode, which allows easy interaction between mobile and websites or printed material by doing away with the necessity of manually typing a URL or contact information. From the pages of magazines to the sides of buses and billboards, QR code technology is being used increasingly in smartphones. Unfortunately, Phishers have started using QR code for phishing attacks by using some features of QR code. This paper introduces a new approach called "TrustQR" which detects URL phishing on QR code. It uses QR code specific features and URL features to detect if the QR code content has a phishing URL. Some of the QR code specific features use QR code

content and its characteristics like length, type, and level of error correction to generate the cryptography key. This technique uses the machine learning classification technique.

2.“Setting priorities in behavioral interventions: An application to reducing phishing risk,”

Phishing risk is a growing area of concern for corporations, governments, and individuals. Given the evidence that users vary widely in their vulnerability to phishing attacks, we demonstrate an approach for assessing the benefits and costs of interventions that target the most vulnerable users. Our approach uses Monte Carlo simulation to (1) identify which users were most vulnerable, in signal detection theory terms; (2) assess the proportion of system-level risk attributable to the most vulnerable users; (3) estimate the monetary benefit and cost of behavioral interventions targeting different vulnerability levels; and (4) evaluate the sensitivity of these results to whether the attacks involve random or spear phishing. Using parameter estimates from previous research, we find that the most vulnerable users were less cautious and less able to distinguish between phishing and legitimate emails (positive response bias and low sensitivity, in signal detection theory terms). They also accounted for a large share of phishing risk for both random and spear phishing attacks. Under these conditions, our analysis estimates much greater net benefit for behavioral interventions that target these vulnerable users. Within the range of the model's assumptions, there was generally net benefit even for the least vulnerable users. However, the differences in the return on investment for interventions with users with different degrees of vulnerability indicate the importance of measuring that performance, and letting it guide interventions. This study suggests that interventions to reduce response bias, rather than to increase sensitivity, have greater net benefit.

3."Intelligent phishing detection system for e-banking using fuzzy data mining"

Detecting and identifying any phishing websites in real-time, particularly for e-banking, is really a complex and dynamic problem involving many factors and criteria. Because of the subjective considerations and the ambiguities involved in the detection, fuzzy data mining techniques can be an effective tool in assessing and identifying phishing websites for e-banking since it offers a more natural way of dealing with quality factors rather than exact values. In this paper, we present novel approach to overcome the ‘fuzziness’ in the e-banking phishing website assessment and propose an intelligent resilient and effective model for detecting e-banking phishing websites. The proposed model is based on fuzzy logic combined with data mining algorithms to characterize the e-banking phishing website factors and to investigate its techniques by classifying the phishing types and defining six e-banking phishing website attack criteria’s with a layer structure. Our experimental results showed the significance and importance of the e-banking phishing website criteria (URL & Domain Identity) represented by layer one and the various influence of the phishing characteristic on the final e-banking phishing website rate

4."A layout-similarity-based approach for detecting phishing pages"

Phishing is a current social engineering attack that results in online identity theft. In a phishing attack, the attacker persuades the victim to reveal confidential information by using web site spoofing techniques. Typically, the captured information is then used to make an illegal economic profit by purchasing goods or undertaking online banking transactions. Although simple in nature, because of their effectiveness, phishing attacks still remain a great source of concern for organizations with online customer services. In previous work, we have developed AntiPhish, a phishing protection system that prevents sensitive user information from being entered on phishing sites. The drawback is that this system requires cooperation from the user and occasionally raises false alarms. In this paper, we present an extension of our system (called DOMAntiPhish) that mitigates the shortcomings of our previous system. In particular, our novel approach leverages layout similarity information to distinguish between malicious and benign web pages. This makes it possible to reduce the involvement of the user and significantly reduces the false alarm rate. Our experimental evaluation demonstrates that our solution is feasible in practice.

5."Improving Email Security with Fuzzy Rules"

Phishing and other malicious email messages are increasingly serious security threats. An important tool for countering such email threats is the automated or semiautomated detection of malicious email. This paper reports work on using fuzzy rules to classify email for such purposes. The effectiveness of a fuzzy rule-based classifier is studied experimentally on a real dataset and compared with results for other classifiers, including those based on crisp rules and decision trees. The human-readability and editability of the classifiers produced by these methods is also studied.

6.”PhishAri: Automatic realtime phishing detection on twitter,”

With the advent of online social media, phishers have started using social networks like Twitter, Facebook, and

Foursquare to spread phishing scams. Twitter is an immensely popular micro-blogging network where people post short messages of 140 characters called tweets. It has over 100 million active users who post about 200 million tweets everyday. Phishers have started using Twitter as a medium to spread phishing because of this vast information dissemination. Further, it is difficult to detect phishing on Twitter unlike emails because of the quick spread of phishing links in the network, short size of the content, and use of URL obfuscation to shorten the URL. Our technique, PhishAri, detects phishing on Twitter in realtime. We use Twitter specific features along with URL features to detect whether a tweet posted with a URL is phishing or not. Some of the Twitter specific features we use are tweet content and its characteristics like length, hashtags, and mentions. Other Twitter features used are the characteristics of the Twitter user posting the tweet such as age of the account, number of tweets, and the follower-followee ratio. These Twitter specific features coupled with URL based features prove to be a strong mechanism to detect phishing tweets. We use machine learning classification techniques and detect phishing tweets with an accuracy of 92.52%. We have deployed our system for end-users by providing an easy to use Chrome browser extension which works in realtime and classifies a tweet as phishing or safe. We show that we are able to detect phishing tweets at zero hour with high accuracy which is much faster than public blacklists and as well as Twitter's own defense mechanism to detect malicious content. To the best of our knowledge, this is the first realtime, comprehensive and usable system to detect phishing on Twitter

7."Phishing websites detection through supervised learning networks" Phishing is an unlawful activity of making gullible people to reveal their insightful information into fake websites. The Aim of these phishing websites is to acquire confidential information such as usernames, passwords, banking credentials and some other personal information. Phishing website looks similar to legitimate website therefore people cannot make difference among them. Today users are heavily relying on the internet for online purchasing, ticket booking, bill payments, etc. As technology advances, the phishing approaches being used are also getting progressed and hence it stimulates anti-phishing methods to be upgraded. In this paper, we have implemented two algorithms named Adaline and Backpropion along with the support vector machine to enhance the detection rate and classification.

8.“Design and Evaluation of a Real-Time URL Spam Filtering Service”

On the heels of the widespread adoption of web services such as social networks and URL shorteners, scams, phishing, and malware have become regular threats. Despite extensive research, email-based spam filtering techniques generally fall short for protecting other web services. To better address this need, we present Monarch, a real-time system that crawls URLs as they are submitted to web services and determines whether the URLs direct to spam. We evaluate the viability of Monarch and the fundamental challenges that arise due to the diversity of web service spam. We show that Monarch can provide accurate, real-time protection, but that the underlying characteristics of spam do not generalize across web services. In particular, we find that spam targeting email qualitatively differs in significant ways from spam campaigns targeting Twitter. We explore the distinctions between email and Twitter spam, including the abuse of public web hosting and redirector services. Finally, we demonstrate Monarch's scalability, showing our system could protect a service such as Twitter -- which needs to process 15 million URLs/day -- for a bit under $800/day.

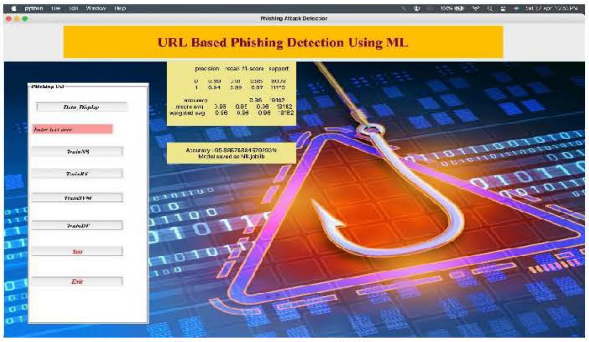
9."Trust based communication in unstructured P2P networks using reputation management and self-certification mechanism"

In unstructured P2P networks there is a possibility of malicious codes and false transactions. It generates the false identities in order to perform false transactions with other identities. The proposed method uses the concept of DHT and reputation management which provides efficient file searching. The self certification (RSA and MD5) is used for ensuring secure and timely availability of the reputation data of a peer to other peers. The reputations of the peers are used to determine whether a peer is a malicious peer or a good peer. Once the malicious peer is detected the transaction is aborted. The reputation of a given peer is attached to its identity. The identity certificates are generated using self-certification and all peers maintain their own (and hence trusted) certificate authority which issues the identity certificate(s) and digital signature to the peer.

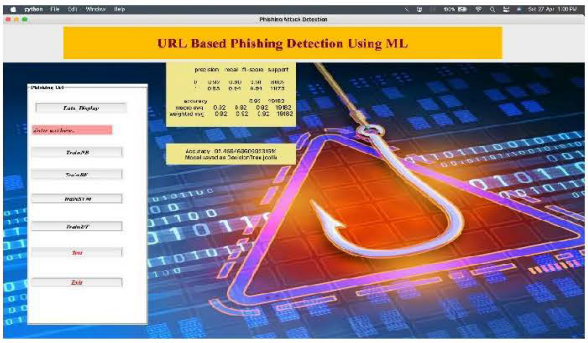
10."Information Source-Based Classification of Automatic Phishing Website Detectors" Phishing attacks allure users to submit their personal information to fake websites that mimic legitimate websites. Many anti-phishing techniques have emerged in recent years. However, the numbers of phishing attacks are still increasing. Two reasons can be blamed for this situation. First, users have too much trust and confidence on existing anti-phishing tools in general. Second, most users believe that they are foolproof against phishing attacks when anti-phishing tools are deployed. We believe that understanding of anti-phishing tools based on their common features can be the beginning step to address these issues. However, there is no extensive analysis of existing anti-phishing techniques. This paper

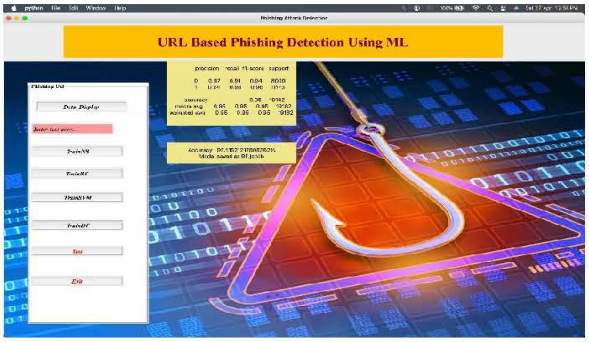
attempts to classify existing works based on information sources. The classification would not only provide useful information to develop new anti-phishing techniques or improve existing techniques, but also enable our understanding on the limitations of the existing techniques

3. RESULTS



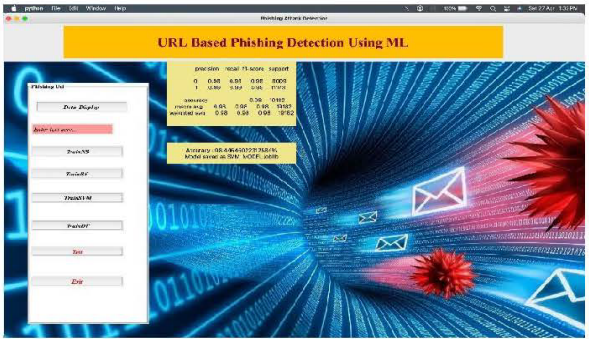
Accuracy using NB





Accuracy using RF

Accuracy using Decision Tree



Accuracy using SVM

4. CONCLUSIONS

Combining advanced feature extraction techniques with machine learning algorithms like SVM, RF, DT and NB provides a potent solution for detecting phishing websites. This multi-faceted approach allows for a comprehensive analysis of various aspects of phishing attacks, enabling more accurate and robust detection in the ever-evolving landscape of cyber threats. As cybercriminals continue to refine their tactics, ongoing research and adaptation of detection techniques are essential to stay one step ahead in the ongoing battle for online security.

ACKNOWLEDGEMENT

We are grateful to our project guides, Mrs. Vina Lomte, HOD, RMDSSOE, and Mrs. Pradnya Kasture, Assistant Professor, RMDSSOE, for their unwavering support, tolerance, and inspiration as well as for their insightful advice and insightful throughout the Research process.

REFERENCES

[1]. A.Y. Ahmad, M. Selvakumar, A. Mohammed, and A.-S. Samer, "TrustQR: A new technique for the detection of phishing attacks on QR code," Adv. Sci. Lett., vol. 22, no. 10, pp. 2905-2909, Oct.2021.

[2].C.C.Inez and F.Baruch,"Setting priorities in behavioral interventions: Anapplication to reducing phishing risk," Risk Anal., vol. 38, no. 4, pp. 826-838, Apr. 2021.

[3]. Aburrous, Maher Hossain, Mohammed Dahal, Keshav Thabtah, Fadi. (2020). Intelligent phishing detection system for ebanking using fuzzy datamining. Expert Systems with Applications. 37. 7913-7921. 10.1016/ j.eswa.2020.04.044.

[4].Rosiello, Angelo Kirda, Engin Kruegel, Ferrandi, Fabrizio. (2007). A layout-similarity-based approach for detecting phishing pages. Proceedings of the 3rd International Conference on Security and Privacy in Communication

[5].Chawathe, Sudarshan. Improving Email Security with Fuzzy Rules. 1864-1869. 10.1109/TrustCom/ BigDataSE.2018.00282. 2021

[6]. A. Aggarwal, A. Rajadesingan and P. Kumaraguru, "PhishAri: Automatic realtime phishing detection on twitter," eCrime Researchers Summit, Las Croabas, 2012, pp. 1-12, doi: 10.1109/eCrime.6489521 2022.

[7].P. Singh, Y. P. S. Maravi and S. Sharma, "Phishing Websites Detection through Supervised Learning Networks", 2020 International Conference on Computing and Communications Technologies (ICCCT), Chennai, 2020, pp. 61-65.

[8].K. Thomas, C. Grier, J. Ma, V. Paxson and D. Song, "Design and Evaluation of a Real-Time URL Spam Filtering Service", IEEE Symposium on Security and Privacy, Berkeley, CA, , pp. 447-462. 2021

[9].C.V. Arulkumar et al., "Secure Communication in Unstructured P2P Networks based on Reputation Management and Self certification", International Journal of Computer Applications, vol. 15, pp. 1-3,

[10]. H. Shahriar and M. Zulkernine, "Information Source-based Classification of Automatic PhishingWebsite Detectors", 2022 IEEE/IPSJ International Symposium on Applications and the Internet, Munich, Bavaria, pp. 190-195. 2022

Published Paper Certificates Screenshots:

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Mr. Yadnyesh Chaudhari

in recognization to the publication of paper titled

URL BASED PHISHING DETECTION

published in IJSREM Journal on Volume 08 Issue 05 May, 2024

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Mrs. Pradnya Kasture

in recognization to the publication of paper titled

URL BASED PHISHING DETECTION

published in IJSREM Journal on Volume 08 Issue 05 May, 2024

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

ANNEXURE C

PLAGIARISM REPORT

Dupli Checker

PLAGIARISM SCAN REPORT

21%
Plagiarised

79%
Unique

Date	2024-05-02
Words	8976
Characters	74047

Plagiarism Report

ANNEXURE D

INFORMATION OF PROJECT GROUP MEMBERS

1. Name : Sanket Vishwas Shendge
2. Date of Birth : 21/05/2002
3. Gender : Male
4. Permanent Address : Holkar Nagar, Near Tapkir S.T.D, Ambegoan B.K., Pune- 46
5. E-Mail : sanketshendge.rmdssoe.comp@gmail.com
6. Mobile/Contact No. : 7709136080
7. Placement Details : NA
8. Paper Published : URL-Based Phishing Detection

1) Name : Gautam Sharma

2) Date of Birth : 17/08/2001

3) Gender : Male

4) Permanent Address : Vishwanarayan Complex ,Narhe ,Pune-411041

5) E-Mail : gautamsharma.rmdssoe.comp@gmail.com

6) Mobile/Contact No. : 7219462235

7) Placement Details : NA

8) Paper Published :URL-Based Phishing Detection

1) Name : Yadnyesh Vinayak Chaudhari

2) Date of Birth : 13/09/2002

3) Gender : Male

4) Permanent Address : Gat No- 89/1, Plot No -14, Behind Gujral Petrol Pump, Jalgaon

5) E-Mail : yadnyeshchaudhari875@gmail.com

6) Mobile/Contact No. : 8766937956

7) Placement Details : NA

8) Paper Published : URL-Based Phishing Detection