BINDURA UNIVERSITY OF SCIENCE EDUCATION

FACULTY OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

DEVELOPMENT AND IMPLEMENTATION OF A REAL TIME MACHINE LEARNING BASED APPROACH FOR DETECTING PHISHING WEBSITES

BY DEMPSSEY MUZANARGO

B190503B

SUPERVISOR: MR O MUZURURA

A RESEARCH PROJECT SUBMITTED TO THE COMPUTER SCIENCE DEPARTMENT AT BINDURA UNIVERSITY OF SCIENCE EDUCATION IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE BACHELOR OF SCIENCE HONOURS DEGREE IN COMPUTER SCIENCE

JUNE 2023

Abstract

This undertaking offers a plugin for detecting phishing for the Chrome browser that detects and notifies users concerning phishing websites in real time using a random forest classifier. According to the author's review of the literature, the random forest classifier outperforms other algorithms in detecting phishing websites.

One frequent method consists of executing the categorization on a server and then enable the plugin to query the server for the results. In contrast to this strategy, our project proposes to carry out the categorization within the browser. The advantages of classifying in the client side browser include increased privacy (the user's data from browsing does not need to leave his computer) and phishing detection is not affected by network latency.

The proposed system will accurately predict phishing websites in the browser and prevent users from sharing sensitive information with potential hackers. The proposed system will use Javascript so that it can run as a browser plugin. Because Javascript does not support many ML libraries, and taking into account the processing capability of the client machines, the strategy must be lightweight.

The model must be trained using Python scikit-learn on the phishing websites dataset, and the model parameters must then be exported into a portable format for usage in Javascript.

Dedication

My parents, who have always been my strongest supports and motivators throughout my academic career, are the ones I dedicate this dissertation to. They have helped me reach this milestone with their constant love, support, and advice. I will always be indebted to them for their efforts and the principles they taught me since they helped me become the person I am today.

I also dedicate this work to my siblings, who have supported me no matter what has happened. They have been a regular source of encouragement and strength for me.

Last but not least, I dedicate my dissertation to all the professors and guides who have contributed significantly to the development of my academic career. Their wisdom, direction, and mentorship have been of immeasurable assistance.

Acknowledgements

I am appreciative of all the folks who have helped me along the way while working on my dissertation. First and foremost, I want to sincerely thank Mr Muzurura, my supervisor, for all of his advice, support, and helpful criticism. This dissertation would not have been feasible without his assistance.

I am appreciative to Bindura Univeristy's staff and professors for giving me access to the facilities and resources I needed to carry out my study.

My sincere gratitude is extended to my family , siblings and friends who have been a continual source of love, encouragement, and support along this journey.

Table of Contents

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 1 : PROBLEM IDENTIFICATION

1.1 Introduction

Machine learning techniques can be used to create models used for phishing activity detection by extracting certain features that are common in already existing phishing websites. These features can then be used in the browser for identifying potential phishing websites. When a user accesses a website, machine learning model will instantly recognize the illegitimate page and send the output to the user. The creation of ML models for automated data analysis relies on the availability of reliable websites with relevant data sets.

According to Sarker (2021), inorder to successfully implement this prediction, we must take three restrictions into account while selecting the machine learning method for our product.

1. The prediction accuracy of the trained model should be good, because a product used by end users in the real world should not generate inaccurate results.

2. The developed method should be able to produce classifications in real-time, which means it should have a very short execution time and require less computer resources.

3. False positives and false negatives must be considered when picking a machine learning algorithm for the problem of phishing detection.

1.2 Background of the study

Phishing is a type sort of cybercrime in which case perpetrators send spam messages with dangerous links in an effort to trick their targets into either downloading malware or visiting spoof websites. Emails were historically used to send these messages, but texts, social media, and phone calls have recently been used as well.

According to AAG-IT (2023), the most frequent type of cybercrime in the world is still phishing. 83% of firms in the UK who experienced a cyber-attack in 2022 said it was due to phishing. Phishing is still the most common sort of cybercrime in the globe. In 2022, 83% of UK businesses who had a cyber-attack stated it was the result of phishing.

Phishing attacks hit 323,972 internet users worldwide in 2021. This suggests that phishing assaults were used to deceive 50% of cybercrime victims. Despite the fact that Google's cyber security processes prevented 99.9% of phishing attempts from reaching users. In 2021, hackers stole $44.2 million through phishing attacks, with a loss of $136 on an average on each incident.

Phishing attacks usually select victims via emails, which route them to malicious websites. There were 16.5 phished emails for every 100 internet users worldwide in the year 2021.Businesses must now take cyber security more seriously more than ever before, especially in highly regulated industries such as law firms and financial services businesses.

According to a 2019 poll, the most popular technique of attack for internet criminals was spear phishing. The primary objective for these assaults was for intelligence collection purposes, with 96% of organizations targeted attacks for this purpose. In 2022, the bulk of URLs in phishing emails (54%) link to websites with the '.com' domain and '.net' is the second most common domain, accounting for fewer than 8.9% of all domains.

1.3 Statement of the problem

As a result of the exponential development in the number of individuals who use the internet today, the internet currently controls a large portion of the world. Because of the anonymity provided by the internet and the fast increase of online transactions, hackers attempt to deceive end users by employing tactics such as phishing, SQL injection, malware, man-in-the-middle attacks, domain name system tunneling, ransomware , web trojans, and others (Alkawaz , 2021). Phishing is regarded to be the most deceitful of all attacks.

Phishing assaults are becoming more complex and difficult to detect, resulting in large financial losses for both individuals and corporations. Manual examination and blacklisting are time-consuming and generally inefficient techniques of detecting phishing websites. As a result, an automated system capable of detecting phishing websites in real time is required.

Machine learning (ML) has showed potential in detecting phishing websites, although research on the usefulness of ML algorithms in detecting new and previously unknown phishing websites is lacking. This project's issue statement is to create a ML-based system that can detect phishing websites in real-time with a high accuracy.

1.4 Research objectives

- To identify and classify phishing websites accurately and efficiently.
- To provide real-time alerts to users and security teams about potential phishing threats.
- To prevent users from sharing sensitive information to potential phishing websites.

1.5 Research questions

1. What are the most effective methods for accurately and efficiently identifying and classifying phishing websites?

2. How can real time alerts be developed and implemented to notify users and security teams about potential phishing threats?

3. What strategies can be employed to prevent users from sharing sensitive information with potential phishing websites ?

1.6 Significance of study

A key effort is detecting phishing using machine learning since it guards against cybercrime and safeguards people and companies from financial loss, identity theft, and other security breaches. The sophistication of phishing attempts is rising, making it challenging for conventional security procedures to identify them. Large-scale data analysis using machine learning algorithms can spot patterns that signify phishing attacks. This technique is more efficient than conventional rule-based systems since it can adjust to new varieties of phishing attempts.

Phishing attacks pose a serious risk to both individuals and corporations. Phishing scams caused approximately $54 million in damages in 2020 alone, according to the FBI's Internet Crime Complaint Center (2021). These losses can be avoided with the aid of machine learning-based phishing detection systems, which can do this by spotting bogus emails, websites, and other kinds of communication before they can do any damage.

Machine learning-based phishing detection helps safeguard sensitive data including login credentials, personal information, and intellectual property in addition to averting financial loss. Organizations can take action to defend their networks and limit additional damage by spotting phishing attempts early on.

Overally, the importance of machine learning-based phishing detection lies in its capacity to offer a proactive approach to cyber security that is more successful than conventional methods.

1.7 Assumptions

- Phishing websites have certain characteristics that distinguish them from legitimate websites, such as misspellings, grammatical errors, and suspicious links
- Machine learning algorithms can learn to recognize patterns in large datasets of known phishing websites and use this knowledge to identify new phishing attempts.
- The accuracy of the system improves over time as it is trained on more data and learns from its mistakes.

1.8 Limitations

- Limited training data: ML models require a large amount of high-quality training data to learn and make accurate predictions. However, phishing attacks are constantly evolving, and it can be challenging to collect enough data to train the model effectively.
- False positives: ML models may sometimes classify legitimate emails or websites as phishing attacks, leading to false positives. This can result in users being denied access to legitimate sites or emails, which can be frustrating and reduce trust in the system.
- Cost and complexity: Implementing an ML-based phishing detection system requires significant resources, including skilled personnel, hardware, and software

tools. This can be a barrier for smaller organizations with limited budgets or technical expertise.

1.9 Definition of terms

Machine learning is a type of artificial intelligence that allows computer systems to learn and improve without having to be explicitly programmed.

A cyber-attack is an attempt by hackers or cybercriminals to gain unauthorized access to a computer system or network in order to steal data, cause damage, or disrupt operations.

Malware refers to any software designed to harm, damage, or disrupt computer systems. It includes viruses, worms, spyware, adware, and other malicious programs.

Ransom-ware is a type of malware that encrypts files on a victim's computer and then demands payment for the decryption key.

Trojans are malicious programs disguised as legitimate software that can give hackers remote access to a victim's computer system.

Datasets refer to collections of data used for machine learning and other analytical purposes. They can include structured data such as spreadsheets and databases as well as unstructured data such as text documents and images.

CHAPTER 2 : LITERATURE REVIEW

2.1 Introduction

The purpose of this literature review is to explore the different machine learning approaches used for phishing detection and their effectiveness. The review will also examine the challenges faced in implementing these techniques and propose possible solutions. The review will draw upon various research articles published on the topic of machine learning for phishing detection.

The literature review will begin with an overview of phishing attacks and their impact on individuals and organizations. It will then delve into the different machine learning algorithms used for phishing detection such as decision trees, linear model, neural networks and random forest.

Finally, this literature review's goal is to offer a thorough grasp of the present cutting-edge technology in phishing detection utilizing machine learning approaches. It is hoped that this review would be a useful resource for the author as he works on the project.

2.2 Phishing

According to Paliath et al (2019) , phishing is a type of social engineering attack that is frequently used by cybercriminals to obtain personal information from internet users , including credit card details, usernames, and passwords (Ramana et al, 2021).In some cases, phishing attacks are also used to distribute malware within a network (Gupta et al, 2021).

There are several types of phishing attacks, such as spoofing, malware-based phishing, DNS-based phishing, data theft, email/spam, web-based delivery, and phone phishing, (Kathrine et al, 2019).These attacks can take various forms and often involve multiple communication channels such as email, instant messages, QR codes (Geng et al, 2018), and social media. Attackers typically impersonate well-known banks or e-commerce websites to deceive users into providing their login credentials.

2.3 The impact of phishing on cyber-security

2.4 Organizational impact

Phishing attacks can have a big impact on businesses, especially in terms of losses in money and harm to their reputation. According to a research by the Ponemon Institute (2020), a business would loss an average of $1.6 million on a successful phishing attack in 2020. This includes the price of lost productivity as well as other expenditures like legal fees and IT cleanup work.

Phishing attacks can cost a business money, but they can also harm its reputation. 60% of consumers would be less likely to do business with a company that had experienced a data breach (CyberSecurityVentures survey, 2021). This emphasizes how crucial strong cyber security measures are to guard against phishing assaults.

2.5 Individual impact

Attacks that involve phishing can have a big effect on people. The attack may cause victims to suffer from emotional discomfort and worry in addition to money losses from stolen personal information or fraudulent transactions. According to a research by Krombholz (2015), phishing attack victims experienced rage, irritation, and embarrassment.

Additionally, individuals could endure long-term repercussions like identity theft or damage to their credit score. According to a report by Javelin Strategy & Research (2020), identity theft cost American consumers $16.9 billion in losses in 2019.

2.6 Global Statistics on cyber security threats

According to Fortra's Pen Testing Report (2020), phishing is the most well-known cyber security danger, and data show that it is at the forefront of most cyber security experts' minds. The purpose remains primarily to steal credentials, and younger users appear to be less prepared for this type of assault.

Terranova Security (2023) presented the following graph showing distribution of global security threats. The graph depicts that phishing is at 80% making it the biggest security concern to internet users. This is because phishing attacks mainly rely on human error and deception.

Cybercriminals utilize social engineering strategies to instill anxiety or urgency in their victims, increasing their likelihood of clicking on a malicious link or providing sensitive information (Magawa et al, 2021).

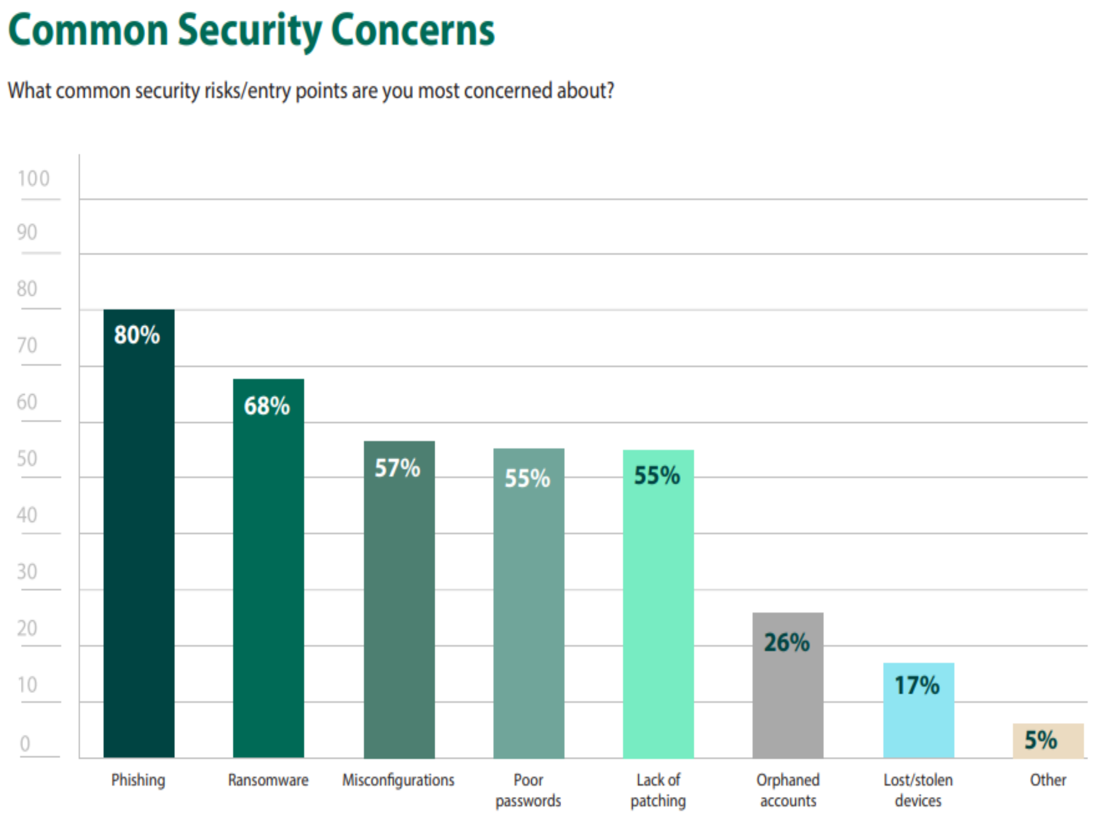


Figure 1 Common security concerns

Retrieved from https://terranovasecurity.com/cyber-security-statistics/

2.7 Existing Machine Learning techniques used for phishing detection

2.8 List Based

To identify phishing websites, browsers like Microsoft Edge, Firefox, and Google Chrome use List Based techniques. There are two different kinds of List Based approaches: whitelisting and blacklisting. A list of legitimate URLs that browsers can access is contained in the "whitelist," and if a URL is on the list, the browser may download the corresponding web page. Additionally, phishing or fraudulent URLs are included in the blacklisting database, preventing browsers from downloading the web pages. The main drawback is that List Based techniques can be defeated with a small URL modification, necessitating frequent list updates in order to stop new phishing URLs (Yang et al, 2021).

2.9 Visual Similarity

Using a variety of visual traits, this approach compares authentic and questionable websites. These tools compare similarity because the phishing web page seems to be very similar to its genuine page. This strategy makes use of CSS, text layout, source code, the website logo, web page screenshots, and other visual components. These methods cannot identify zero-hour phishing attacks because they compare the suspect web page to previously visited or saved web pages (Jain and Gupta , 2018)

2.10 Heuristic

The heuristic method makes use of elements taken directly from phishing websites. This tactic is based on a number of characteristics that set phishing websites apart from legitimate ones. These techniques collect information from a variety of sources, including website traffic, DNS, digital certificates, text content, and URLs. The effectiveness of this method depends on the feature set, training data, and classification algorithms. This method has the benefit of being able to identify Zero-hour phishing attacks (Jain and Gupta, 2018).

2.11 Deep Learning

Recent advancements in deep learning techniques suggest that deep neural networks will be more effective at identifying phishing websites than traditional machine learning methods. Deep Neural Network, Recurrent Neural Network, Feed-Forward Deep Neural Network, Limited Boltzmann Machine, Convolutional Neural Network, Deep Belief Network, and Deep Auto-Encoder are a few popular Deep Learning algorithms used for phishing detection (Basit et al, 2020).

2.12 Limitations of existing phishing detection techniques

i. Lack of Real-Time Detection: Many existing phishing detection techniques rely on static analysis and signature-based approaches, which are not effective in detecting new and evolving phishing attacks in real-time (Sharma et al, 2019).

ii. Inability to Detect Sophisticated Attacks: Phishing attacks are becoming increasingly sophisticated, with attackers using advanced techniques such as social engineering and spear-phishing to bypass traditional detection methods (Kumar et al, 2018). Existing techniques may not be able to detect these types of attacks.

iii. False Positives: Some phishing detection techniques may generate false positives, which can lead to legitimate emails being flagged as phishing attempts (Wang et al, 2017). This can result in a loss of productivity and user frustration.

iv. Limited Coverage: Some existing phishing detection techniques only focus on specific types of attacks or platforms, such as email or web-based attacks (Zhang et al, 2020). This limited coverage can leave other attack vectors vulnerable to phishing attacks.

v. Dependence on User Behavior: Many existing phishing detection techniques rely on user behavior, such as clicking on links or opening attachments, to detect potential threats (Kumar et al, 2018). However, this approach is not foolproof as users may still fall victim to sophisticated social engineering tactics.

2.13 The need for Machine learning in phishing detection

Phishing attacks are one of the most common forms of cybercrime, where attackers trick users into divulging sensitive information such as login credentials, credit card details, and personal information. Phishing attacks can be carried out through various channels, including email, social media, and websites (Kumar et al, 2018).

Detecting phishing websites is a challenging task because attackers use sophisticated techniques to make their websites look legitimate. Machine learning algorithms can be used to detect phishing websites by analyzing various features such as website content, domain name, SSL certificate, and user behavior (Basit et al, 2020).

According to Basit et al (2020), machine learning algorithms can be trained on a large dataset of known phishing websites to learn patterns and characteristics that distinguish them from legitimate websites. These algorithms can then be used to automatically detect new phishing websites in real-time.

Machine learning algorithms can also analyze user behavior on a website to detect potential phishing attacks. For example, if a user enters their login credentials on a website that is not associated with the legitimate service provider, the machine learning algorithm can flag it as a potential phishing attack (Kahksha, S. N, 2021).

In conclusion, machine learning is essential for detecting phishing in websites due to its ability to analyze large amounts of data and identify patterns that are difficult for humans to detect. With the increasing number of phishing attacks every day, machine learning-based solutions are becoming more critical in protecting users from these types of cyber threats.

2.14 Machine Learning

Machine learning is a subset of artificial intelligence that involves the use of algorithms and statistical models to enable computer systems to learn from data, without being explicitly programmed. It is a process by which machines can improve their performance on a specific task by learning from experience (Zhang et al, 2020).

According to Jordan (2020), a leading researcher in the field of machine learning, "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

There are three main types of machine learning: supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, the algorithm is trained on labeled data to predict outcomes for new data. In unsupervised learning, the algorithm identifies patterns and relationships in unlabeled data. In reinforcement learning, the algorithm learns through trial and error by receiving feedback in the form of rewards or punishments (Kumar et al, 2018).

Machine learning has numerous applications across various industries such as healthcare, finance, marketing, and transportation. Some examples include predicting disease outbreaks based on social media activity, detecting fraudulent transactions in banking systems, optimizing supply chain management using predictive analytics, and developing self-driving cars (Magawa et al, 2022).

2.15 Using machine learning for phishing detection

In the present day, phishing websites can be identified using machine learning Sindhu et al, (2020).To represent phishing URLs and related websites, common characteristics like URL details, website architecture, and JavaScript features are gathered. Then, phishing data sets are acquired based on those features. Then, based on those features, Machine Learning classifiers are trained to identify the phishing website. Big Data sets (with high Velocity, Variety, Volume, Value, and Veracity) respond very well to this technique. The most successful approach, according to Alkawaz et al (2021), was machine learning-based classifiers that had more than 99% accuracy.

2.16 Comparison between research papers on machine learning

Author(s)	Aim	Findings	Limitations
Qabajeh et al, 2018	This review compares conventional anti-phishing techniques, which include user education, phishing awareness campaigns, recurring training sessions, and legal analysis. List-based and machine-learning techniques are discussed in the computerized anti-phishing techniques.	Machine learning and rule induction are effective methods for phishing defense because of their high detection rates and, more importantly, their clear results.	In the research, 67 studies were analyzed, but Deep Learning techniques were not covered.
Zuraiq and Alkasassbeh, 2019	This study examines several phishing detection methods, including heuristic, content-based, and fuzzy rule-based methods.	The study indicated no perfect method for identifying phishing websites.	The work analyzed only 18 studies and did not include Machine Learning, List-based, and Deep Learning approaches.
Korkmaz, 2020	Proposed a review work on selecting features that can be used in URL based phishing detection systems.	According to research, URL-based detection strategies are preferred to increase detection speed.	This study's limitation is that the work discussed only five studies in the literature.

Kathrine et al, 2019	This work presents different phishing attacks with the latest prevention approaches. This paper proposed a framework to detect and prevent phishing attacks	According to this study, Machine Learning-based algorithms effectively detect true positive results.	The work discussed only 11 studies, and the research does not include Deep Learning techniques for mitigating phishing websites.

Table 1 : Comparison between research papers

2.17 Machine learning algorithms used for phishing detection

Machine learning algorithms can help improve the accuracy and efficiency of detecting phishing attacks by automatically identifying patterns in large datasets of features extracted from URLs or other sources.

Four machine learning algorithms have been used in this work for detection of phishing websites. These include Decision tree, Random forest, Neural Network and Linear model.

2.18 Decision tree

DT algorithms are supervised machine learning algorithms that can be applied to classification and regression issues. The internal nodes of a decision tree (DT) represent different attributes, and the branches between the nodes represent potential outcomes that these attributes might have in the observed samples, whereas the terminal nodes represent the dependent variables' final results (Zhao, 2012).

2.19 Linear model

The conventional approach for fitting a statistical model to data is a linear regression model. When the target variable is continuous and numeric, it is appropriate. The family of generalized linear models allows for the inclusion of targets with non-normal (non-gaussian) distributions in linear regression. Following the conversion of the target variable into a continuous numeric, linear regression models are iteratively fitted to the data (Zhao , 2012).

2.20 Neural network

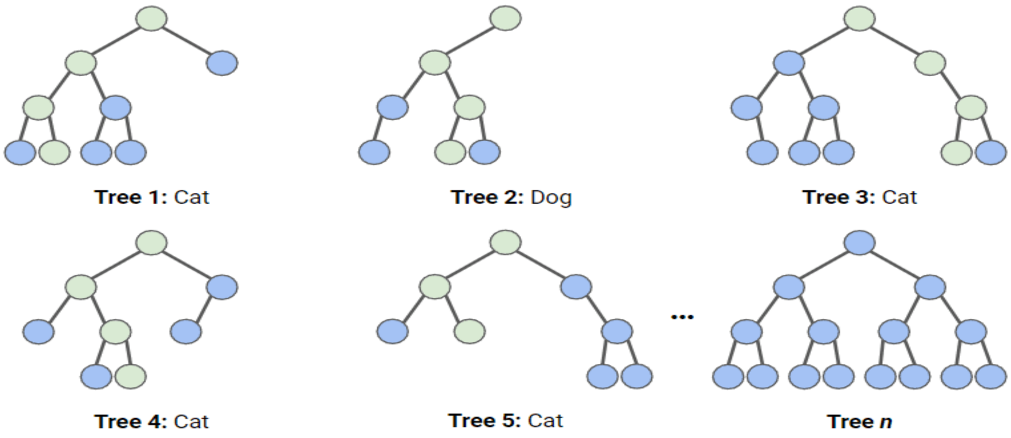
Neural networks are made up of a group of interconnected units called neurons. Like the synapses in the brain, neurons can communicate with one another in order to learn and make decisions in a more human-like way (Williams , 2009).

2.21 Random Forest

Assume you have a complex challenge to tackle and you enlist the help of a group of specialists from various professions. Each expert offers an opinion based on their knowledge and experience. The experts would then vote to make a final conclusion.

Multiple decision trees are built in a random forest classification utilizing various random subsets of the data and characteristics. Each decision tree functions as an expert, advising on how to classify the data. Predictions are created by computing the forecast for each decision tree and then selecting the most popular outcome.

We have a random forest with n decision trees in the picture below, and we've presented the top 5 with their predictions (either "Dog" or "Cat"). Because each tree is exposed to a different number of features and a distinct sample of the original dataset, each tree is unique. Each tree provides a forecast. We can see from the first five trees that 4/5 guessed the sample was a Cat. The green circles represent a possible path the tree followed to make its conclusion. The random forest would tally the number of predictions made by decision trees for Cat and Dog and select the most popular guess (Williams , 2009).



2.22 Performance Metrix

Confusion Matrix has been used to calculate different parameters such as accuracy, sensitivity or true positive rate (TPR), specificity or true negative rate (TNR), false positive (FP) and f-measure (Kahksha and Naaz, 2018).

Accuracy is percentage of correct classification (true positive and negative) from overall numbers of instance.

Accuracy (A) = (TP + TN) / (TP + TN + FP +FN)

Sensitivity is percentage of correct positive classifications (true positive) from instances that are actually positive.

Sensitivity (S)/Recall/ TPR = TP/ (TP+FN)

Specificity is the percentage of positive records classified correctly out of all positive records

Specificity (SS) = TN / (TN + FP)

Precision is the percentage of the correct positive classification (true positive) from instances that predicate as positive.

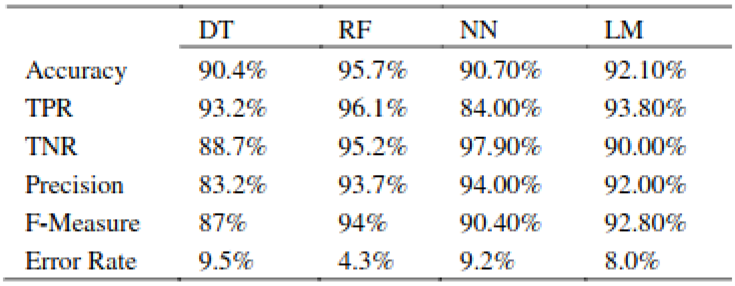
Precision = TP / (TP + FP)

The F measure is defined as the weighted harmonic mean of the precision and recall of the test.

F-Measure = 2 × Precision × Recall/ (Precision + Recall)

2.23 Measuring the accuracy of machine learning-based detection algorithms

According to test run by Kahksha, S. N. (2021), the accuracy of different machine learning algorithms has been calculated using confusion matrix and random forest has been found to have highest accuracy of 95.7%, linear model showed accuracy of 92.10% followed by neural network with 90.7% whereas decision tree has been found to perform worst with accuracy of 90.4% as shown below.



	DT	RF	NN	LM
Accuracy	90.4%	95.7%	90.70%	92.10%
TPR	93.2%	96.1%	84.00%	93.80%
TNR	88.7%	95.2%	97.90%	90.00%
Precision	83.2%	93.7%	94.00%	92.00%
F-Measure	87%	94%	90.40%	92.80%
Error Rate	9.5%	4.3%	9.2%	8.0%

Figure 2 ***Retrieved from Kahksha, S. N. (2021). Detection of Phishing Websites using Machine Learning Approach. Department of Computer Science and Engineering, School of Engineering Sciences and Technology, New Delhi – 110062, India.***

2.24 Measuring the error rate of machine learning based detection algorithms

According to test run by Kahksha, S. N. (2021), the error rate of different machine learning algorithms has been calculated using confusion matrix and random forest has been found to have minimum error rate of 4.3%, linear model showed error rate of 8% followed by neural network with 9.2% whereas decision tree has been found to perform worst with error rate of 9.5% as shown below.

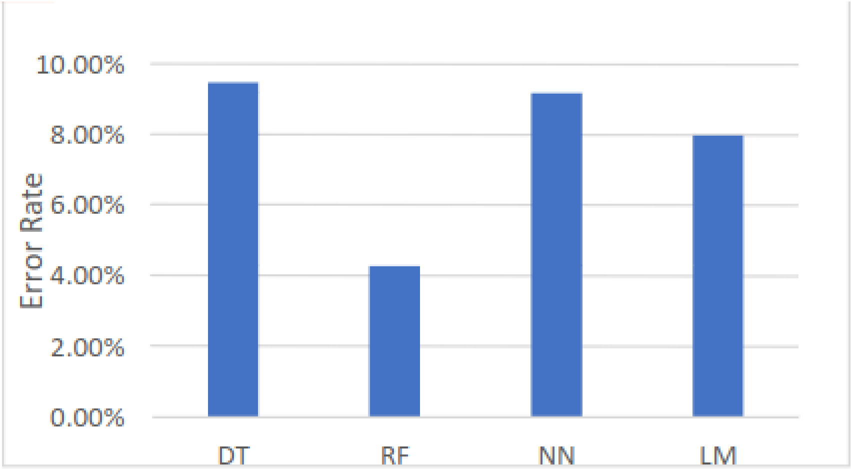
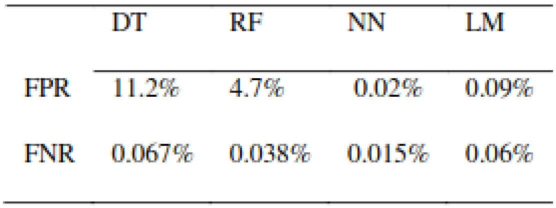


Figure 3 ***Retrieved from Kahksha, S. N. (2021). Detection of Phishing Websites using Machine Learning Approach. Department of Computer Science and Engineering, School of Engineering Sciences and Technology, New Delhi – 110062, India.***

2.25 False Positive rate (FPR), and False Negative rate (FNR)

According to test run by Kahksha, S. N. (2021), the false positive rate and false negative rate of different machine learning algorithms has been calculated using confusion matrix and random forest has been found to have the second FPR and FNR of 4.7% and 0.038% respectively.



	DT	RF	NN	LM
FPR	11.2%	4.7%	0.02%	0.09%
FNR	0.067%	0.038%	0.015%	0.06%

Figure 4 ***Retrieved from Kahksha, S. N. (2021). Detection of Phishing Websites using Machine Learning Approach. Department of Computer Science and Engineering, School of Engineering Sciences and Technology, New Delhi – 110062, India.***

2.26 Proposed approach

From the above matrix, the author has therefore chosen the random forest algorithm to implement in his project due to the following factors as compared to the other mentioned algorithms

i. High Accuracy (95.7%)

ii. Low Error rate (4.3%)

iii. Low FPR (4.7%)

In addition random forest algorithm was chosen because it is an ensemble approach that combines multiple decision trees to improve accuracy and robustness of the model. The random forest algorithm also has the ability to handle high dimensional data with many features, as well as noisy and missing data.

CHAPTER 3 : RESEARCH METHODOLOGY

3.1 Introduction

This chapter explains the development of a phishing detection plugin for chrome browser that can detect and warn the user about phishing web sites in real-time using random forest classifier.

The method used consists of executing the categorization on a server and then enable the plugin to query the server for the results. In contrast to this strategy, our project proposes to carry out the categorization within the browser. The advantages of classifying in the client side browser include increased privacy (the user's data from browsing does not need to leave his computer) and phishing detection is not affected by network latency.

The proposed system will accurately predict phishing websites in the browser and prevent users from sharing sensitive information with potential hackers. The proposed system will use Javascript so that it can run as a browser plugin. Because Javascript does not support many ML libraries, and taking into account the processing capability of the client machines, the strategy must be lightweight.

The model must be trained using Python scikit-learn on the phishing websites dataset, and the model parameters must then be exported into a portable format for usage in Javascript.

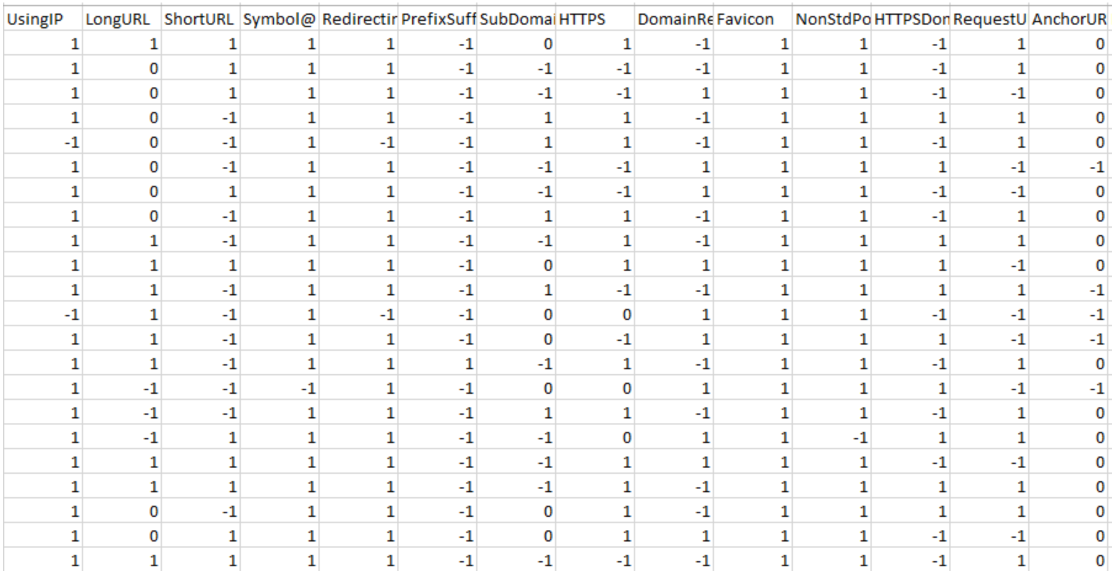
3.2 Research design

The acquisition of data is a crucial step in software development as it can determine the success or failure of the final product, given that it must align with the customer's requirements. This study used a machine learning dataset obtained from an online repository.

3.3 Data collection

The dataset is obtained from the UCI repository and placed in a humpy array. The dataset contains 30 characteristics that must be reduced in order for them to be extracted on the browser.

https://archive.ics.uci.edu/ml/datasets/phishing+websites



UsingIP	LongURL	ShortURL	Symbol@	Redirectir	PrefixSuff	SubDomai	HTTPS	DomainRe	Favicon	NonStdPo	HTTPSDon	RequestU	AnchorUR
1	1	1	1	1	-1	0	1	-1	1	1	-1	1	0
1	0	1	1	1	-1	-1	-1	-1	1	1	-1	1	0
1	0	1	1	1	-1	-1	-1	1	1	1	-1	-1	0
1	0	-1	1	1	-1	1	1	-1	1	1	1	1	0
-1	0	-1	1	-1	-1	1	1	-1	1	1	-1	1	0
1	0	-1	1	1	-1	-1	-1	1	1	1	1	-1	-1
1	0	1	1	1	-1	-1	-1	1	1	1	-1	-1	0
1	0	-1	1	1	-1	1	1	-1	1	1	-1	1	0
1	1	-1	1	1	-1	-1	1	-1	1	1	1	1	0
1	1	1	1	1	-1	0	1	1	1	1	1	-1	0
1	1	-1	1	1	-1	1	-1	-1	1	1	1	1	-1
-1	1	-1	1	-1	-1	0	0	1	1	1	-1	-1	-1
1	1	-1	1	1	-1	0	-1	1	1	1	1	-1	-1
1	1	-1	1	1	1	-1	1	-1	1	1	-1	1	0
1	-1	-1	-1	1	-1	0	0	1	1	1	1	-1	-1
1	-1	-1	1	1	-1	1	1	-1	1	1	-1	1	0
1	-1	1	1	1	-1	-1	0	1	1	-1	1	1	0
1	1	1	1	1	-1	-1	1	1	1	1	-1	-1	0
1	1	1	1	1	-1	-1	1	-1	1	1	1	1	0
1	0	-1	1	1	-1	0	1	-1	1	1	1	1	0
1	0	1	1	1	-1	0	1	1	1	1	-1	-1	0
1	1	1	1	1	-1	-1	-1	-1	1	1	-1	1	0

Figure 5 UCL dataset

3.4 Requirements Analysis

The process of finding, assessing, and documenting a system's needs is known as requirements analysis. Understanding the needs and expectations of stakeholders, defining the scope of the system, and stating the functional and non-functional criteria that must be satisfied are all part of the process.

3.5 Functional Requirements

The functional requirements establish the system's function and specify what the system performs.

In the case of this project , a user visits a phishing website, the plugin alerts them. The plugin needs to meet the following criteria:

- The plugin needs to be quick enough to stop the user from giving the phishing website any important information.

- Any external online service or API that can reveal a user's browsing habits shouldn't be used by the plugin.
- The plugin need to be capable of spotting recently developed phishing websites.
- The plugin ought to have a system in place for upgrading itself to incorporate new phishing tricks.

3.6 Non-functional requirements

Non-functional requirements of a system are the characteristics or qualities that define how well the system performs its intended functions. These requirements are not related to the specific features or functionalities of the system, but rather to its overall performance, usability, reliability, security, and other aspects that affect its effectiveness and efficiency.

1. Accuracy: The system should have a high accuracy rate in detecting phishing attacks to minimize false positives and false negatives.
2. Scalability: The system should be able to handle large volumes of data and traffic without compromising its performance.
3. Reliability: The system should be reliable and available at all times, with minimal downtime or disruptions.

3.7 Hardware Requirements

1. CPU
2. Keyboard
3. Mouse Monitor

3.8 Software Requirements

1. Operating system:
    - Windows
2. Software tools:

- Google Chrome
- Google co-laboratory
- VS Code
- Python

3.9 Development model

The best SDLC to use for phishing detection system using machine learning is the Agile SDLC. The Agile SDLC is a flexible and iterative approach that allows for continuous feedback and improvement throughout the development process. This is particularly important for a project like this because it requires constant updates and improvements to stay ahead of evolving threats.

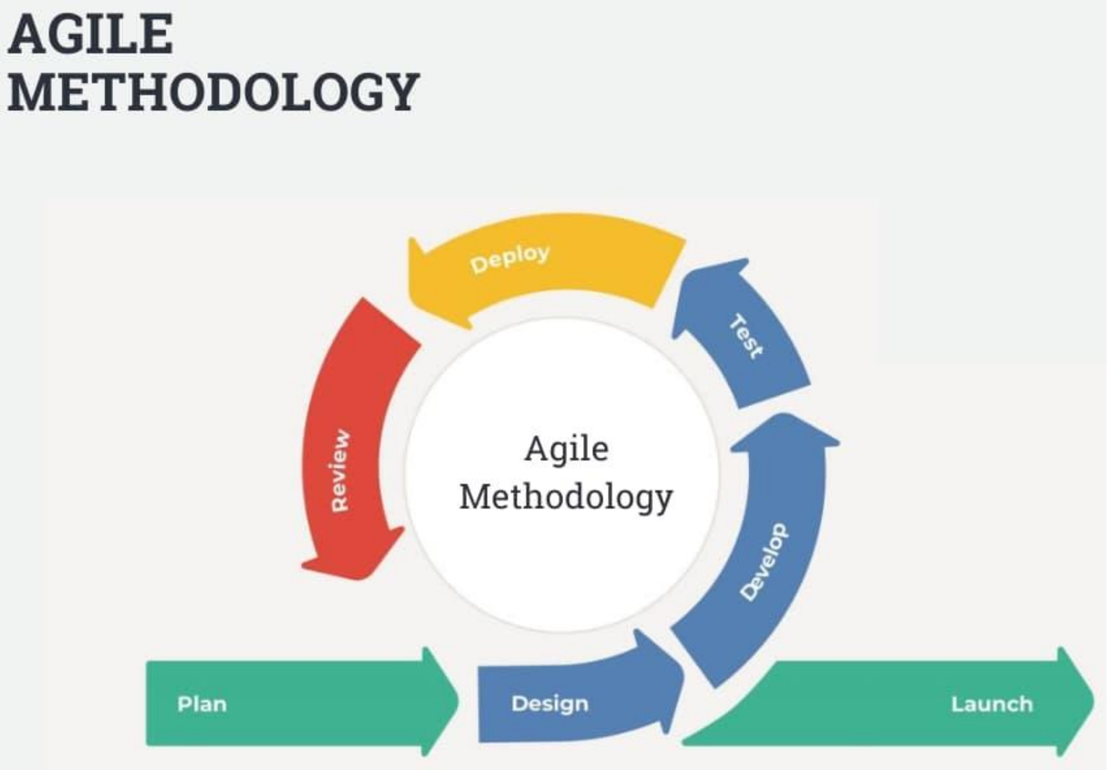


Figure 6 Agile methodology

The steps for using the Agile SDLC for phishing detection system using machine learning are as follows:

1. Planning: In this stage, the author will define the scope of the project, identify stakeholders, and establish goals and objectives. This will involve identifying the types

of phishing attacks that need to be detected, as well as the data sources that will be used to train the machine learning algorithms.

2. Analysis: In this stage, the author will conduct a detailed analysis of the requirements for the system. This will involve identifying any technical or operational constraints that need to be considered in designing and implementing the system.

3. Design: In this stage, the author will design a prototype of the system based on the requirements identified in the analysis phase. This may involve developing wireframes or mockups of user interfaces, as well as designing algorithms for detecting phishing attacks.

4. Development: In this stage, developers will begin coding and building out features of the system based on the design specifications developed in previous stages.

5. Testing: In this stage, testers will conduct functional testing to ensure that all features are working correctly and meeting requirements. They may also conduct performance testing to ensure that the system can handle large volumes of data and traffic.

6. Deployment: In this stage, developers will deploy the system to production environments where it can be used by end-users.

7. Maintenance: In this final stage, developers will continue to monitor and maintain the system over time, making updates and improvements as needed based on feedback from users or changes in threat landscape.

3.10 Class diagram

The following is a class diagram of the whole Machine Translation system. This picture clearly displays the functions of various modules in the system. It also depicts the interplay of the system's parts, offering a clear notion for implementation.

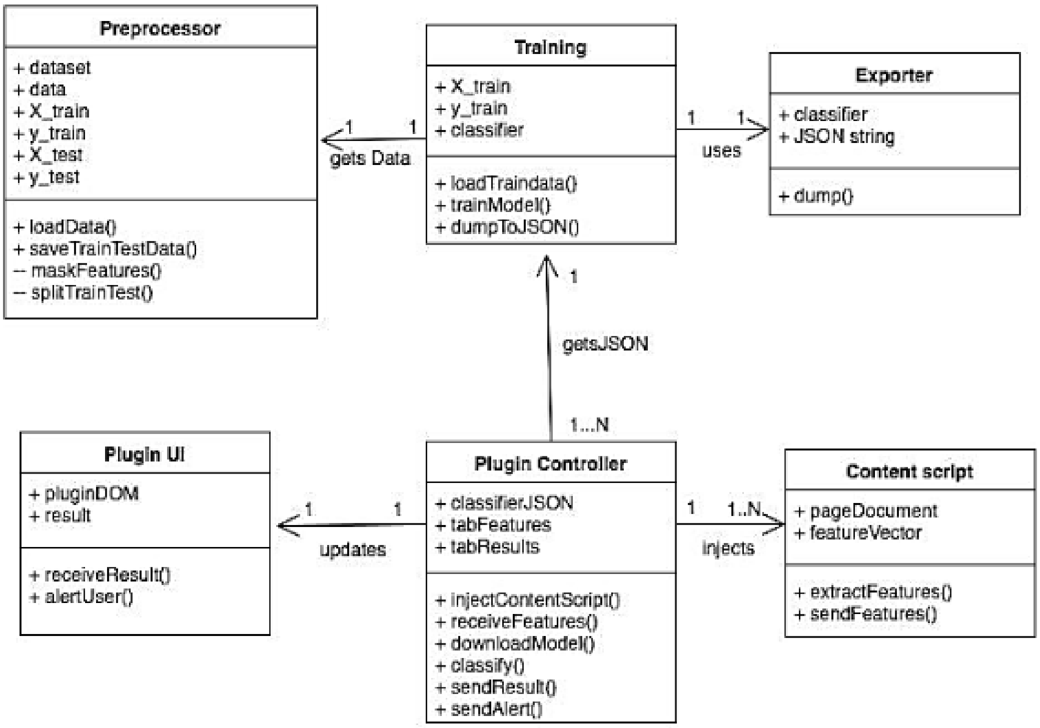


Figure 7 Class diagram

3.11 Flow chart of how the system works

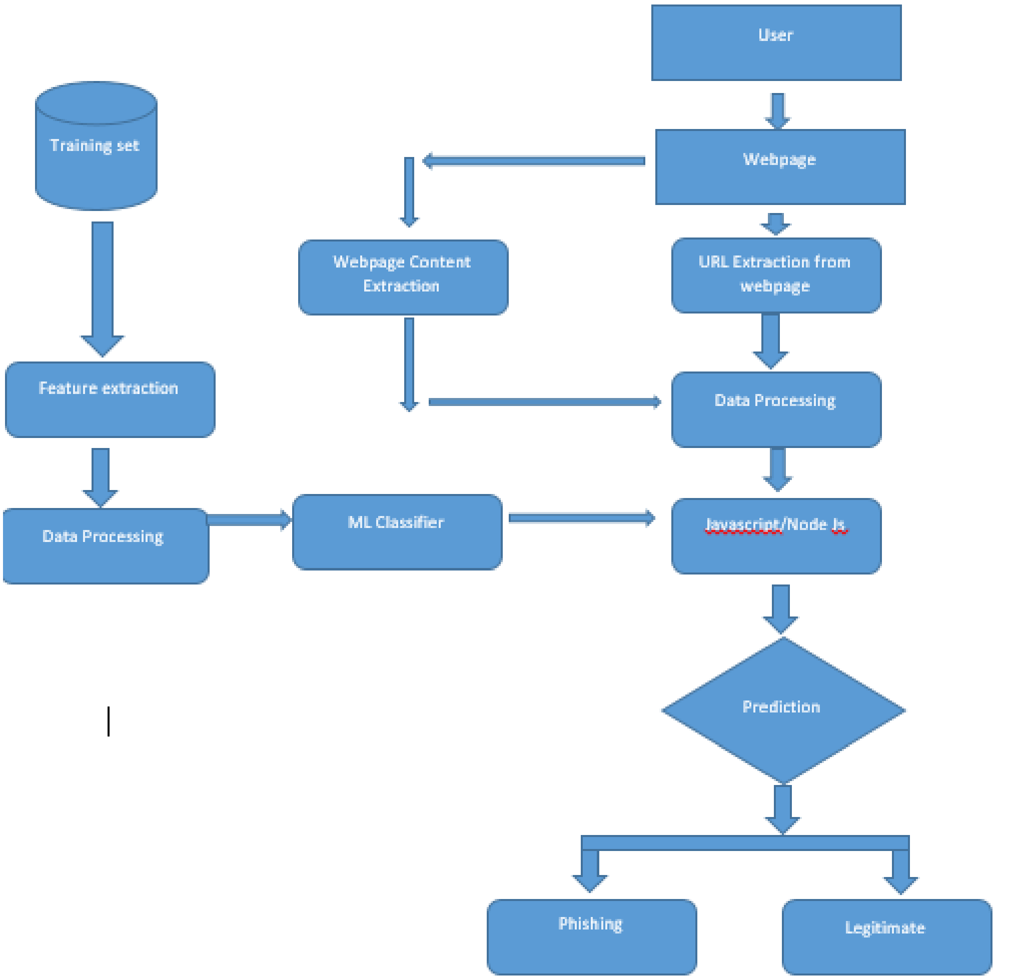


Figure 8 Flow chart of the system

3.12 Use case diagram for the proposed system

A use case diagram is a visual representation of a user's potential interactions with a technology. It also displays various system users and the actions that they may conduct on the system.

Pre-condition: The user goes to a website that has a plugin installed.

Post-condition: The user is warned incase it's a phishing website.

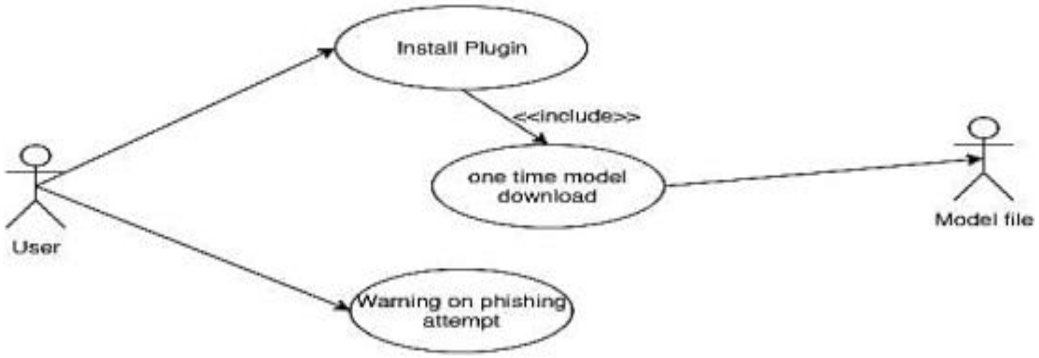


Figure 9 Use case diagram

3.12.1 Prototype across the modules

This section describes the system's input and output to each module.

Preprocessing: This module accepts a downloaded dataset in arff format and generates four new files: training features, training class labels, testing features, and testing class labels.

Training: This module accepts the four pre-processor output files and generates a trained Random Forest object as well as the cross validation score on the training set.

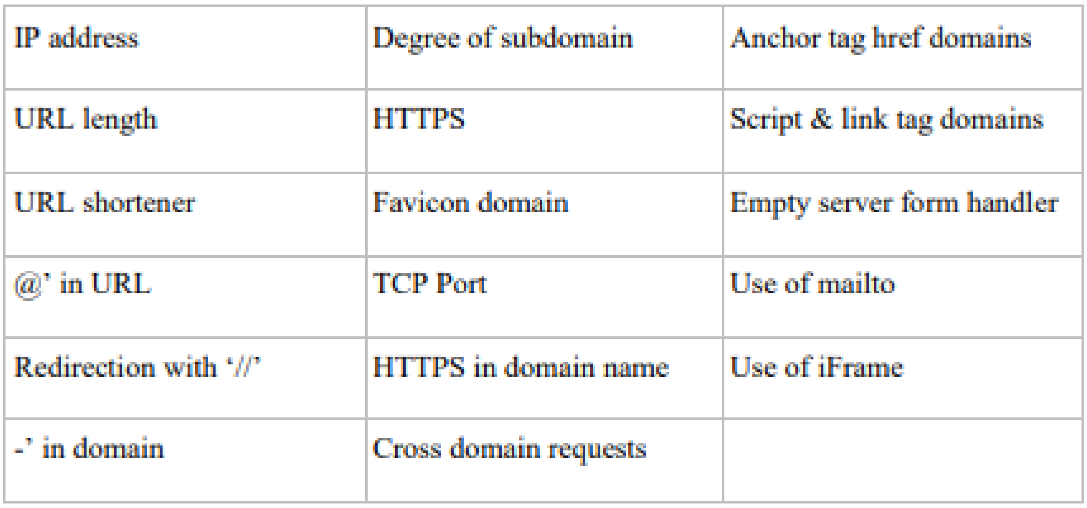
Exporting model: This module accepts the learnt Random Forest classifier object and creates its JSON representation recursively together with the trained ML model, which is saved to disk.

Plugin Feature Extraction: This program takes a webpage as input and creates a feature vector with 17 encoded features.

Classification: Using the feature vector from the feature extraction module and the JSON format from the Exporting model module, this module generates a boolean output indicating if the webpage is authentic or phishing based on the ML model.

3.12.2Preprocessing

The dataset is obtained from the UCI repository and placed in a numpy array. The dataset contains 30 characteristics that must be reduced in order for them to be extracted on the browser. Based on previous research by other authors, 17 characteristics were chosen out of 30 with little loss in test data accuracy.



IP address	Degree of subdomain	Anchor tag href domains
URL length	HTTPS	Script & link tag domains
URL shortener	Favicon domain	Empty server form handler
@’ in URL	TCP Port	Use of mailto
Redirection with ‘//’	HTTPS in domain name	Use of iFrame
-’ in domain	Cross domain requests	

Figure 10 Feature extraction

The dataset is then divided into training and testing sets, with 30% set aside for testing. Training and testing data are both stored to disk.

3.13 Training

The preprocessing module's training data is imported from disk. Using the scikit-learn package, a random forest classifier is trained on the data. Because Random Forest is an ensemble learning approach, an ensemble of ten decision tree estimators is employed. Each decision tree employs the CART algorithm to decrease gini impurity.

$$Gini(E) = 1 - \sum_{j=1} p_j^2$$

On the training data, the cross validation score is also computed. The F1 score is derived from the testing data. The trained model is then exported to JSON using the following module.

3.14 Exporting Model

During the training phase, every machine learning algorithm learns the values of its parameters. Each decision tree in Random Forest is an autonomous learner, and each decision tree learns node threshold values as the leaf nodes learn class probabilities. As a result, a format for representing the Random Forest in JSON must be developed. The entire JSON structure is made up of keys such as the number of estimators, the number of classes, and so on. It also includes an array with each item being a JSON-encoded estimator. Each decision tree is encoded as a JSON tree with nested objects comprising recursively the threshold for that node as well as left and right node objects.

3.15 Plugin Feature Extraction

The above-mentioned 17 properties must be retrieved and encoded in real time for each webpage as it loads. A content script is used to gain access to the webpage's DOM. While each page loads, the content script is automatically inserted. The content script is in charge of gathering the features and sending them to the plugin. The major goal of this work is to not use any external online services, and the features must be independent of network latency and extraction must be quick. All of this is ensured while creating strategies for feature extraction. Once a feature is retrieved, it is encoded into the values -1, 0, 1 using the following formula:

-1 - Legitimate

0 - Suspicious

1 - Phishing

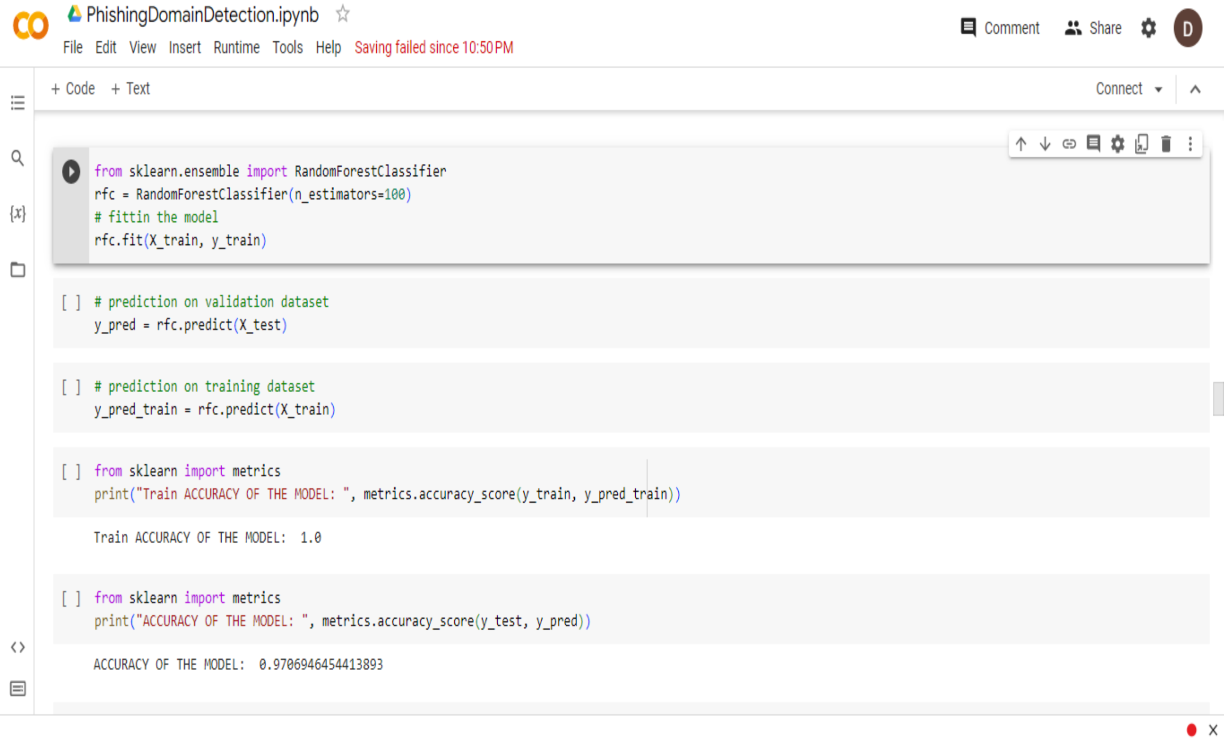
The content script passes the feature vector with 17 encoded values to the plugin.

3.16 Classification

The Random Forest classification algorithm is used to the feature vector acquired from the content script. The JSON file containing the Random Forest parameters and model is downloaded and cached on disk. The script attempts to load the JSON and model from disk, and if the cache is not found, the JSON and model are downloaded again. A javascript framework has been created to simulate the Random Forest behavior using JSON by comparing the feature vector to the node threshold. The binary classification output is based on the leaf node values, and the user is notified if the webpage is classed as phishing.

3.16.1 Implementation

3.16.2 Training module


PhishingDomainDetection.ipynb
File Edit View Insert Runtime Tools Help Saving failed since 10:50 PM
Comment Share
+ Code + Text
Connect

```
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(n_estimators=100)
# fittin the model
rfc.fit(X_train, y_train)
```

```
# prediction on validation dataset
y_pred = rfc.predict(X_test)
```

```
# prediction on training dataset
y_pred_train = rfc.predict(X_train)
```

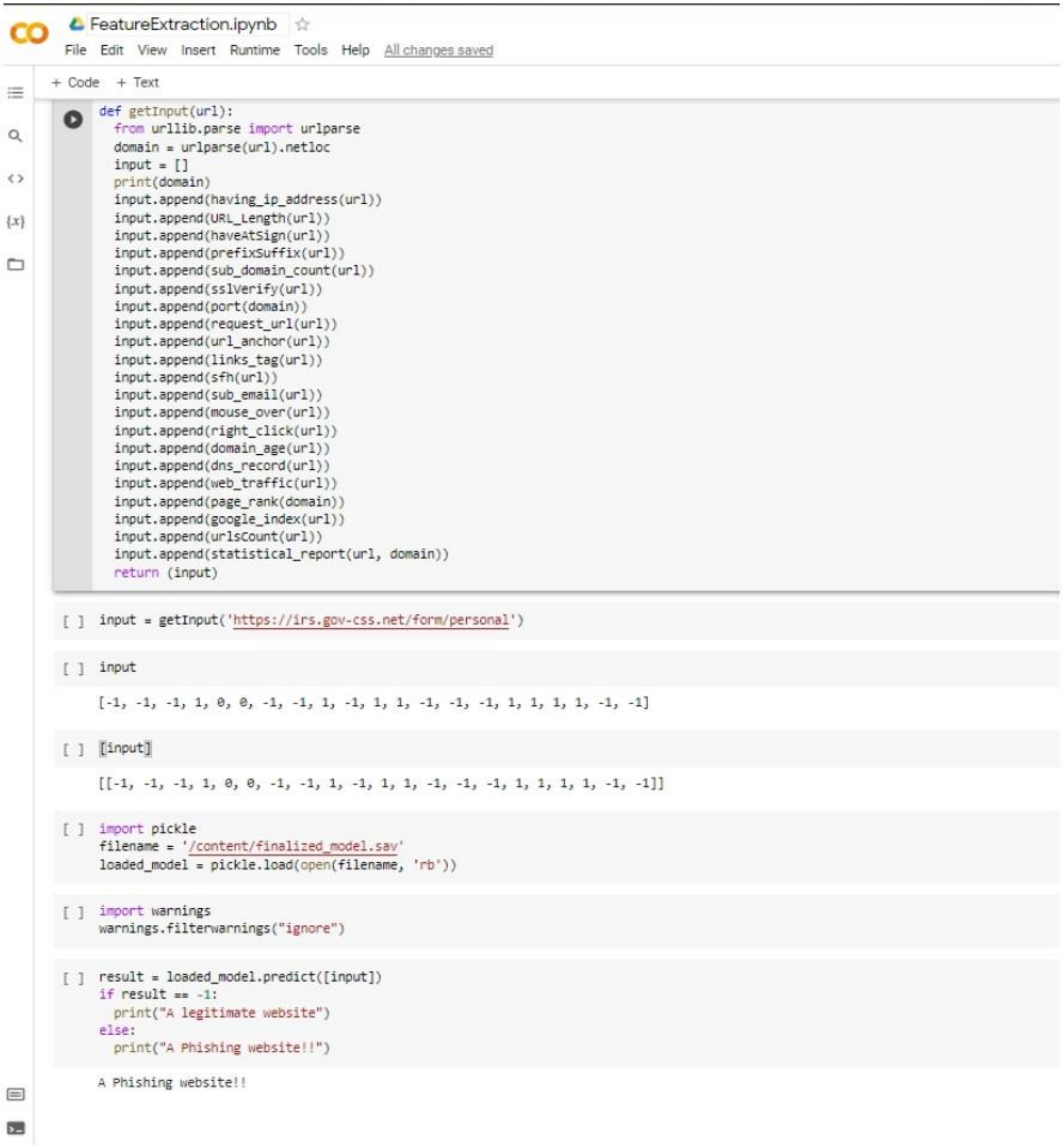
```
from sklearn import metrics
print("Train ACCURACY OF THE MODEL: ", metrics.accuracy_score(y_train, y_pred_train))
```

```
Train ACCURACY OF THE MODEL:  1.0
```

```
from sklearn import metrics
print("ACCURACY OF THE MODEL: ", metrics.accuracy_score(y_test, y_pred))
```

```
ACCURACY OF THE MODEL:  0.9706946454413893
```

3.16.3Feature extraction using the classifier



```
def getInput(url):
  from urllib.parse import urlparse
  domain = urlparse(url).netloc
  input = []
  print(domain)
  input.append(having_ip_address(url))
  input.append(URL_Length(url))
  input.append(haveAtSign(url))
  input.append(prefixSuffix(url))
  input.append(sub_domain_count(url))
  input.append(sslVerify(url))
  input.append(port(domain))
  input.append(request_url(url))
  input.append(url_anchor(url))
  input.append(links_tag(url))
  input.append(sfh(url))
  input.append(sub_email(url))
  input.append(mouse_over(url))
  input.append(right_click(url))
  input.append(domain_age(url))
  input.append(dns_record(url))
  input.append(web_traffic(url))
  input.append(page_rank(domain))
  input.append(google_index(url))
  input.append(urlsCount(url))
  input.append(statistical_report(url, domain))
  return (input)
```

```
[ ] input = getInput('https://irs.gov-css.net/form/personal')
```

```
[ ] input

    [-1, -1, -1, 1, 0, 0, -1, -1, 1, -1, 1, 1, -1, -1, -1, 1, 1, 1, 1, -1, -1]
```

```
[ ] [input]

    [[-1, -1, -1, 1, 0, 0, -1, -1, 1, -1, 1, 1, -1, -1, -1, 1, 1, 1, 1, -1, -1]]
```

```
[ ] import pickle
    filename = '/content/finalized_model.sav'
    loaded_model = pickle.load(open(filename, 'rb'))
```

```
[ ] import warnings
    warnings.filterwarnings("ignore")
```

```
[ ] result = loaded_model.predict([input])
    if result == -1:
      print("A legitimate website")
    else:
      print("A Phishing website!!")

    A Phishing website!!
```

3.17 Deployment details

Python 3 is required for the backend, and the Classifier JSON and model set are delivered over HTTP via Github. The plugin is delivered as a single file and must be executed in the Chrome browser. For distribution, the plugin (frontend) is packaged as a crx file.

CHAPTER 4 : DATA PRESENTATION, ANALYSIS AND INTERPRETATION

4.1 Introduction

The preceding section covered the methodology used for research, while this section is dedicated to the presentation, analysis, and interpretation of data. Once the author had implemented the system, there was a requirement to assess its efficiency. To determine this, accuracy and performance were used as metrics. Additionally, the behavior of the developed solution was observed under various scenarios and presented in tabular form.

4.2 Testing

The development process relies heavily on testing, and this chapter presents the tests that were conducted and their outcomes. These tests are evaluated based on the functional and non-functional requirements outlined in the preceding chapter

4.3 Black box testing

Testing the system through black box testing does not require knowledge of its internal structure. This method assesses the functional and non-functional requirements of the system.

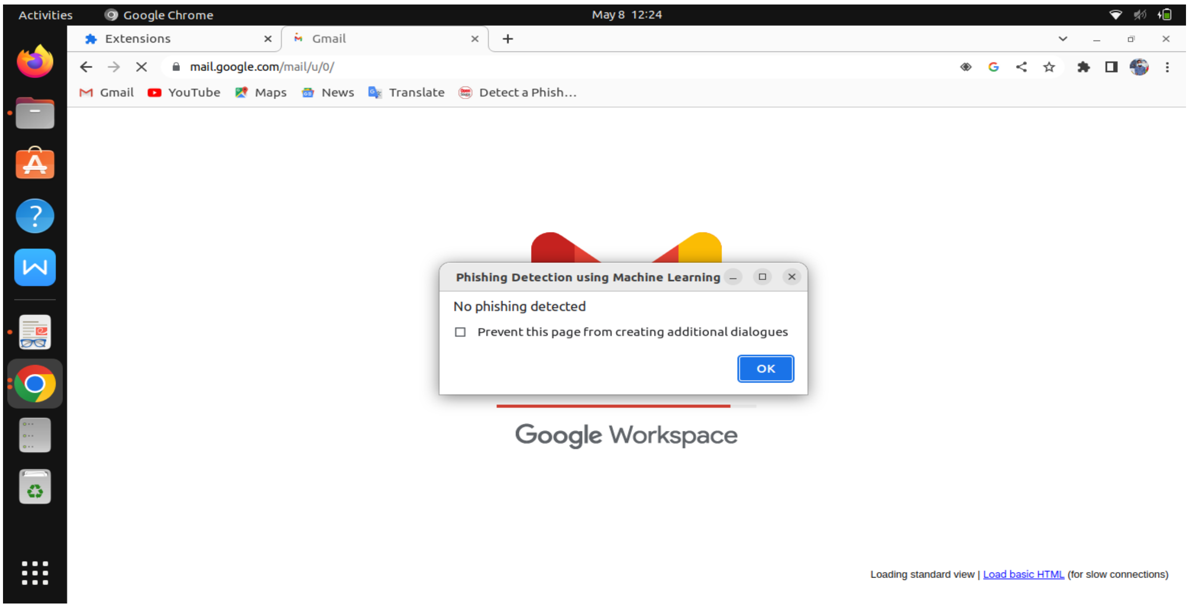
4.4 White box testing

White box testing is a software testing approach that includes inspecting an application's or systems internal structure and workings. The purpose of white box testing is to find bugs in the code, ensuring that all pathways have been checked, and confirm that the product fulfills its functional and non-functional criteria.

4.5 Meeting the objectives

- **Providing real-time alerts to users and security teams about potential phishing threats.**

Showing pop-up alert for a non-phishing website



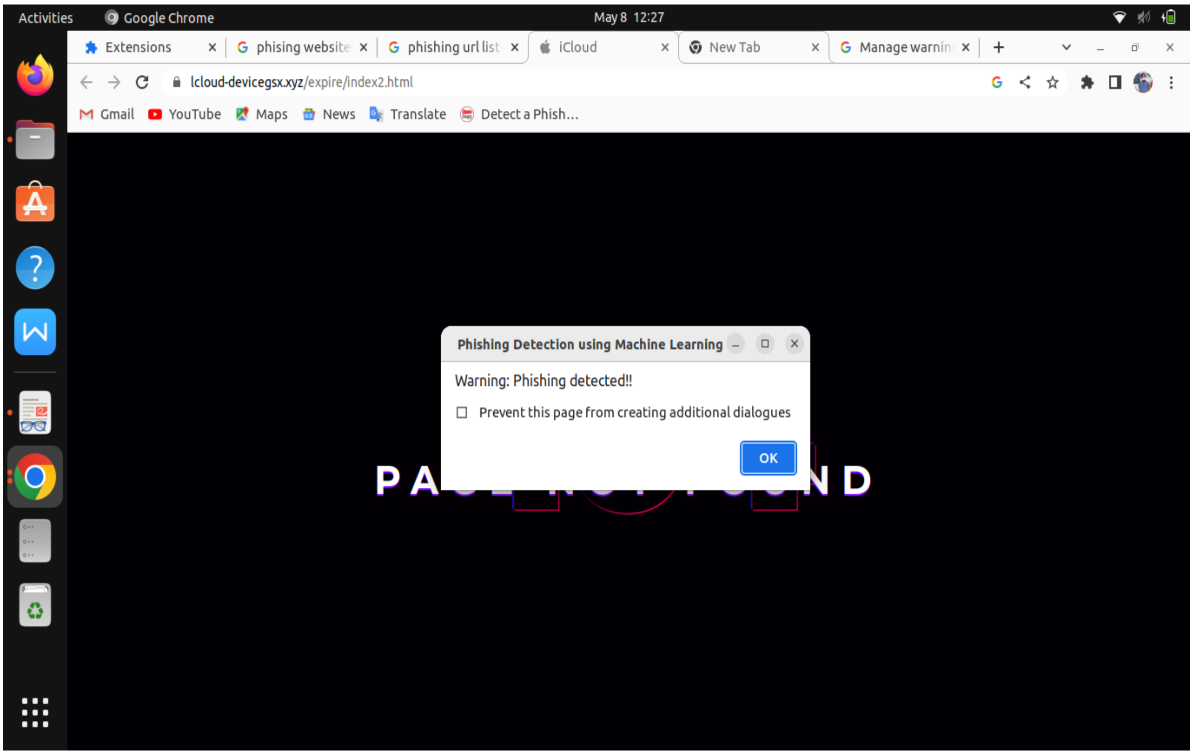
Phishing Detection using Machine Learning

No phishing detected

☐ Prevent this page from creating additional dialogues

OK

Showing pop-up alert for phishing detection



Phishing Detection using Machine Learning

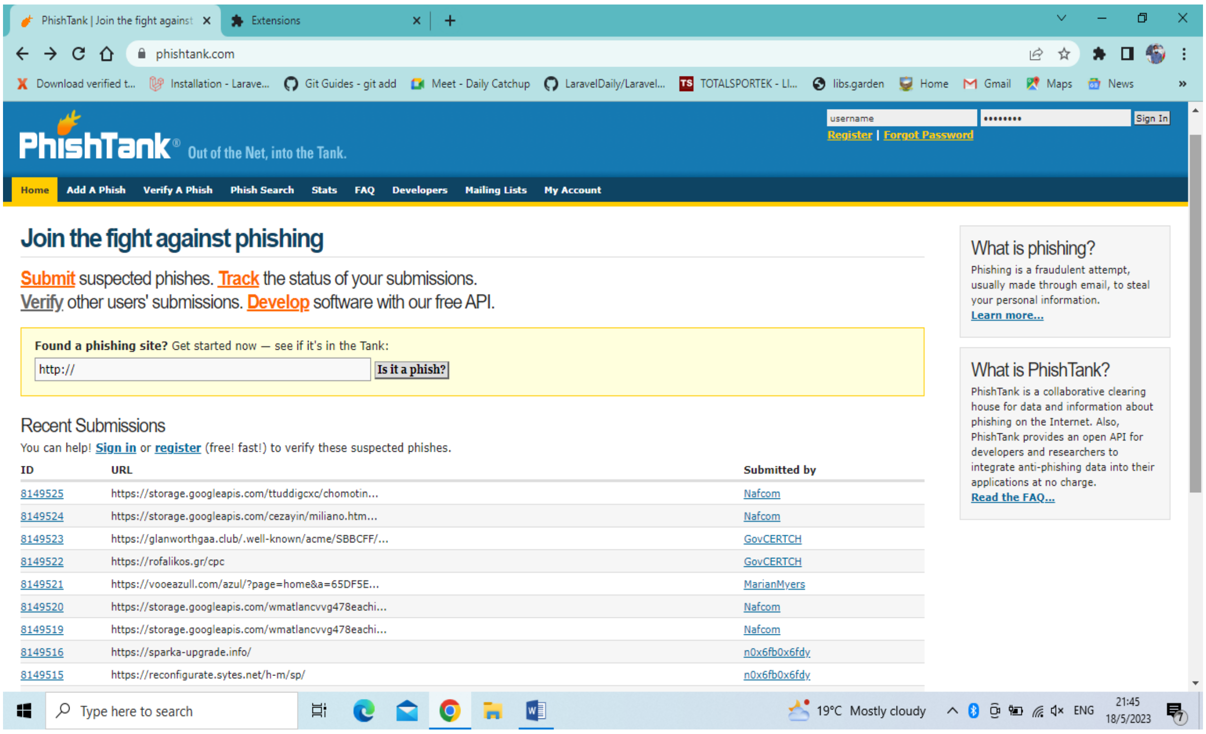
Warning: Phishing detected!!

□ Prevent this page from creating additional dialogues

OK

- **Identifying and classifying phishing websites accurately and efficiently.**

Classification of websites



Join the fight against phishing

Submit suspected phishes. Track the status of your submissions.
Verify other users' submissions. Develop software with our free API.

Found a phishing site? Get started now — see if it's in the Tank:

http://   Is it a phish?

Recent Submissions

You can help! Sign in or register (free! fast!) to verify these suspected phishes.

ID	URL	Submitted by
8149525	https://storage.googleapis.com/ttuddigcxc/chomotin...	Nafcom
8149524	https://storage.googleapis.com/cezayin/miliano.htm...	Nafcom
8149523	https://glanworthgaa.club/.well-known/acme/SBBCFF/...	GovCERTCH
8149522	https://rofalikos.gr/cpc	GovCERTCH
8149521	https://vooeazull.com/azul/?page=home&a=65DF5E...	MarianMyers
8149520	https://storage.googleapis.com/wmatlancvvg478eachi...	Nafcom
8149519	https://storage.googleapis.com/wmatlancvvg478eachi...	Nafcom
8149516	https://sparka-upgrade.info/	n0x6fb0x6fdy
8149515	https://reconfigurate.sytes.net/h-m/sp/	n0x6fb0x6fdy

What is phishing?

Phishing is a fraudulent attempt, usually made through email, to steal your personal information.

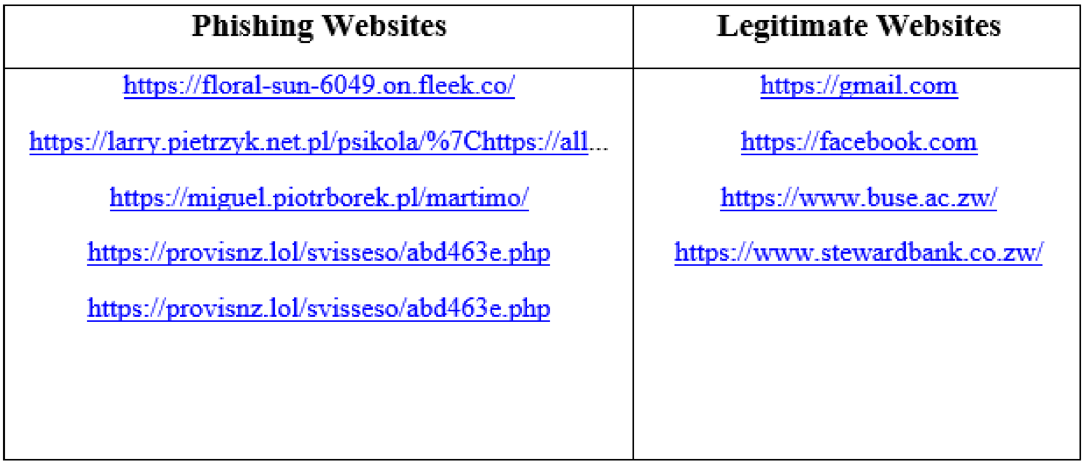
Learn more...

What is PhishTank?

PhishTank is a collaborative clearing house for data and information about phishing on the Internet. Also, PhishTank provides an open API for developers and researchers to integrate anti-phishing data into their applications at no charge.

Read the FAQ...

Table 2 : Classification of websites



Phishing Websites	Legitimate Websites
https://floral-sun-6049.on.fleek.co/	https://gmail.com
https://larry.pietrzyk.net.pl/psikola/%7Chttps://all...	https://facebook.com
https://miguel.piotrborek.pl/martimo/	https://www.buse.ac.zw/
https://provisnz.lol/svisseso/abd463e.php	https://www.stewardbank.co.zw/
https://provisnz.lol/svisseso/abd463e.php	

4.6 Experimental design

The author decided to use the experimental research design as it allows him to observe changes and response of systems and objects as he changed factors (visiting different phishing and non-phishing websites).

Table 3 Testing for phishing

Test cases	Phishing detected	Number of tests	Correct readings	False Readings	Classification
1	Yes	50	48	2	True positive
2	No	50	43	7	True negative

4.6.1 Accuracy

Accuracy is the number of right predictions divided by the total number of forecasts in each category. It is then multiplied by 100 to get the percentage of correctness. It is calculated using the equation below:

Equation 1: Accuracy calculation as adopted from Karl Pearson (1904)

i. Accuracy = (TP+TN)/ (TP+TN+FP+FN)*100

Accuracy in presence of phishing= (48+43)/ (48+43+2+7)

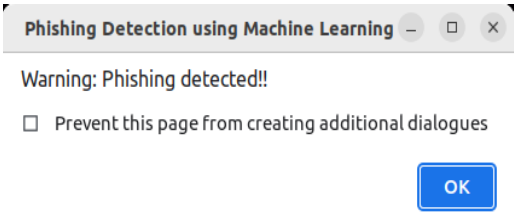
= 91/100

=0.91*100

= 91%

- **Preventing users from sharing sensitive information to potential phishing websites**

Once the user had been informed by the pop-up alert that the website is of phishing nature, he/she can stop navigating the website and prevent himself from sharing sensitive information to potential hackers.



Phishing Detection using Machine Learning

Warning: Phishing detected!!

☐ Prevent this page from creating additional dialogues

CHAPTER 5 : CONCLUSION

5.1 Introduction

A system for detecting phishing websites in real time has been developed. Real time alerts and system accuracy are the main focus of the system. The system sends alerts to the end user every time a phishing website is detected and prevents them from sharing confidential information in the long run.

5.2 Contribution

The proposed system contributes by decreasing the number of computer fraud cases committed by individuals who use phished data to commit malicious crimes. The system showed an accuracy of 91% and produced a real-time alert mechanism as required by the author.

5.3 Future scope

The future scope of the system may include the use of two or more classification algorithms at the same time. An advantage of having two algorithms will be to improve accuracy but this will decrease response time in the long run

5.4 Recommendations

- The author recommends making sure that the chrome extension is on before browsing the internet.
- The author recommends to always have the latest version of Google Chrome in order to avoid compatibility issues with the extension.
- The author recommends avoiding further browsing of websites that have been deemed as phishing websites by the extension to prevent sharing sensitive information with potential hackers.

5.5 Challenges faced

- **Data quality**: To learn and generate precise predictions, machine learning algorithms need a lot of high-quality data. Phishing assaults are uncommon occurrences, though, and gathering enough data to train the algorithm can be difficult. The quality of the data itself might also be problematic since it can include bias or noise in it, which would reduce the system's accuracy.

- **Overfitting** : Overfitting happens when a model performs badly on fresh data after being trained too well on a particular dataset. This may occur if the model is very complicated or if there is insufficient diversity in the training data.

- **Adversarial attacks** : Attackers may evade machine learning-based detection systems by using adversarial strategies like obfuscation or evasion.

5.6 Conclusion

The phishing detection system was effectively developed, and the researcher gained new information and experience along the project's beneficial course. Overally, this study effort was a success.

References

1. AAG-IT. (2023, January 18). The Latest Phishing Statistics. Retrieved April 14, 2023, from https://aag-it.com/the-latest-phishing-statistics/.
2. Alkawaz, M.H., Steven, S.J., Hajamydeen, A.I., Ramli R.(2021). A comprehensive survey on identification and analysis of phishing website based on machine learning methods.In Proceedings of the IEEE 11th Symposium on Computer Applications and Industrial Electronics (ISCAIE), pp. 82-87. https://doi.org/10.1109/ISCAIE51753.2021.
3. Alkhalil, Z., Hewage, C., Nawaf, L., & Khan, I. (2021). Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. Cardiff School of Technologies, Cardiff Metropolitan University, Cardiff, United Kingdom.
4. Basit,A., Zafar,M., Javed,A.R., Jalil,Z.(2020).A Novel Ensemble Machine Learning Method to Detect Phishing Attack.In Proceedings of the IEEE International Multi-Topic Conference INMIC 2020, pp. 1-6. https://doi.org/10.1109/INMIC50486.2020.9318210.
5. Dutta, A.K. (2021) Detecting phishing websites using machine learning technique. PLoS ONE, 16(10), e0258361. https://doi.org/10.1371/journal.pone.0258361
6. Faris, H., & Yazid, S. (2021). Phishing Web Page Detection Methods: URL and HTML Features Detection. In Proceedings of the IEEE International Conference on Internet of Things and Intelligence Systems.
7. Geng G.G., Yan Z.W., Zeng Y., & Jin X.B. (2018). RRPhish: Anti-phishing via mining brand resources request. IEEE International Conference on Consumer Electronics.
8. Gupta, B. B., Yadav, K., Razzak, I., Psannis, K., Castiglione, A., & Chang, X. (2021). A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment. Computer Communications, 175(April), 47-57.
9. Hannousse, A., Kathrine G.J.W., Praise P.M., Rose A.A., & Kalaivani E.C. (2019). Variants of phishing attacks and their detection techniques. Proceedings of the International Conference on Trends in Electronics and Informatics.
10. Hong, J. (2012). The State of Phishing Attacks. Communications of the ACM, 55(1), 74-81. doi: 10.1145/2063176.2063197.
11. InnoTech Today. (2021, March 16). FBI: 12x Surge in Phishing Over the Last 5 Years. Retrived from https://bit.ly/3j7tK4C
12. Jain, A.K., Gupta, B.B. (2018). PHISH-SAFE: URL features-based phishing detection system using machine learning. In Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 729-734. https://doi.org/10.1007/978-981-10-8536-9_44.
13. Jordan MI, Mitchell TM. Machine Learning: Trends Perspectives and Prospects. Science 2020; 349(6245):255-260.
14. Kahksha & Naaz, S.(2018). Machine Learning Algorithm to Predict Survivability in Breast Cancer Patients.International Journal of Computer Science Engineering(IJCSE),10(4),97–101
15. Kahksha, S. N. (2021). Detection of Phishing Websites using Machine Learning Approach. Department of Computer Science and Engineering, School of Engineering Sciences and Technology, New Delhi – 110062, India.
16. Kathrine, G.J.W., Praise, P.M., Rose, A.A., Kalaivani, E.C. (2019). Variants of phishing attacks and their detection techniques. Proceedings of the International Conference on Trends in Electronics and Informatics (ICOEI 2019) (pp. 255-259). https://doi.org/10.1109/ICOEI.2019.8862697
17.
Korkmaz, M. (2020). Feature Selections for the Classification of Web pages to Detect Phishing

Attacks: A Survey. In HORA 2020 - 2nd International Congress on Human-Computer Interaction, Optimization and Robotic Applications Proceedings.

18. Krombholz, K., et al. (2015). The impact of psychological persuasion on security decisions: An empirical analysis. Proceedings of the 24th USENIX Security Symposium.
19. Kumar, A., & Srivastava, S. (2018). A review of phishing detection techniques. International Journal of Computer Science and Mobile Computing, 7(6), 1-10.
20. Nakamura, A., Dobashi, F., 2019. Proactive phishing sites detection. In: Proceedings - 2019 IEEE/WIC/ACM International Conference on Web Intelligence, WI, pp. 443–448. https://doi.org/10.1145/3350546.3352565..
21. Paliath, S., Abu Qbeitah, M., Aldwairi, M., 2020. PhishOut: effective phishing detection using selected features. IEEE.
22. Phriendly Phishing. (2019). The Impacts of a Phishing Attack. Retrieved from https://www.phriendlyphishing.com/blog/the-impacts-of-a-phishing-attack.
23. Ponemon Institute (2020). Cost of a Data Breach Report 2020: Global Overview. Retrieved from https://www.ibm.com/security/data-breach.
24. Qabajeh, I., Thabtah, F., Chiclana, F. (2018). A recent review of conventional vs. automated cybersecurity anti-phishing techniques. Computer Science Review, 29, 44-55. https://doi.org/10.1016/j.cosrev.2018.05.003
25. Ramana, A. V., Rao, K. L., & Rao, R. S. (2021). Stop-Phish: an intelligent phishing detection method using feature selection ensemble. Social Network Analysis and Mining, 11(1), 1-9.
26. Sarker, I.H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. Springer Nature Singapore Pte Ltd.
27. Sharma, A., Kumar, V., & Singh, S. K. (2019). Phishing Detection Techniques: A Review. International Journal of Advanced Research in Computer Science and Software Engineering, 9(5), 1-7.
28. Sindhu, S., Patil, S.P., Sreevalsan, A., Rahman, F., Saritha, A.N. (2020). Phishing detection using random forest, SVM and neural network with backpropagation. In Proceedings of the International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), pp. 391-394. https://doi.org/10.1109/ICSTCEE49637.2020.9277256.
29. Wang, Y., Liang, X., & Zhang, Y. (2017). A survey on email spam filtering techniques. Journal of Network and Computer Applications, 88, 1-26.
30. Williams,G.J.(2009).Rattle: a data mining GUI for R.The R Journal,1(2),45-55
31. Wright, C. S. (2008). The IT Regulatory and Standards Compliance Handbook: How to Survive Information Systems Audit and Assessments. Elsevier.
32. Yang, L., Zhang, J., Wang, X., Li, Z., Li, Z., He, Y. (2021). An improved ELM-based and data preprocessing integrated approach for phishing detection considering comprehensive features. Expert Systems with Applications, 165, 113863. https://doi.org/10.1016/j.eswa.2020.113863.
33. Yang, Y. (2019). Effective phishing detection using machine learning approach. (Master's thesis). Case Western Reserve University, Department of Electrical Engineering and Computer Science.
34. Zabihimayvan, M., & Doran, D. (2019). Fuzzy rough set feature selection to enhance phishing attack detection. IEEE International Conference on Fuzzy Systems.
35. Zhang, Y., Li, X., & Zhang, Y. (2020). A survey on phishing detection techniques. Journal of Network and Computer Applications, 168, 102736.
36. Zhao, Y. (2012). R and data mining: Examples and case studies. Academic Press.
37. Zuraiq, A.A., Alkasassbeh, M. (2019). Review: Phishing Detection Approaches. In 2019 2nd International Conference on New Trends in Computing Sciences (pp. 1-6). https://doi.org/10.1109/ICTCS.2019.8923069