Title: Detection of phishing URL with Machine learning

Contents

[Lists of figures 4](#_Toc123648985)

[List of Tables 5](#_Toc123648986)

[Abstract 6](#_Toc123648987)

[List of Abbreviations 7](#_Toc123648988)

[Word 7](#_Toc123648989)

[Abbreviation 7](#_Toc123648990)

[Chapter 1: Introduction 8](#_Toc123648991)

[1.1 Background on Phishing: 8](#_Toc123648992)

[1.1.1 Phishing attacks in progress: 9](#_Toc123648993)

[1.1.2 What are phishing attacks? 10](#_Toc123648994)

[1.1.3 Use case 11](#_Toc123648995)

[1.1.4 Types of Phishing 11](#_Toc123648996)

[1.1.5 A real-world example of phishing URLs 16](#_Toc123648997)

[1.2 Motivation 17](#_Toc123648998)

[1.3 Problem in the Current Solutions 18](#_Toc123648999)

[1.4 Research Questions 19](#_Toc123649000)

[1.5 Problem statements 20](#_Toc123649001)

[1.6 Aims and Objectives 21](#_Toc123649002)

[1.7 Summary of the proposed solution 21](#_Toc123649003)

[1.8 Applications of the proposed mechanism in the real world 22](#_Toc123649004)

[Chapter 2: LITERATURE REVIEWS 22](#_Toc123649005)

[2.1 Static and Dynamic Features Approaches based 23](#_Toc123649006)

[2.2 Feature-based approaches: 24](#_Toc123649007)

[2.3 Domain-based information approached 25](#_Toc123649008)

[2.4 Dynamic feature-based solutions 28](#_Toc123649009)

[2.5 Comparison among existing mechanisms 29](#_Toc123649010)

[2.6 Research Gaps 31](#_Toc123649011)

[Chapter 3: Methodology 32](#_Toc123649012)

[3.1 Research Methodology followed (Qualitative/Quantitative) 32](#_Toc123649013)

[3.2 Proposed Framework 33](#_Toc123649014)

[3.2.1 Data collection 34](#_Toc123649015)

[3.2.2 Data preprocessing 34](#_Toc123649016)

[3.2.2 Feature extraction 34](#_Toc123649017)

[3.2.3 Machine learning models 35](#_Toc123649018)

[6. Ridge Classifier 39](#_Toc123649019)

[7. Artificial neural network: 39](#_Toc123649020)

[3.2.4 Data Splitting 40](#_Toc123649021)

[3.2.4 Train Model 41](#_Toc123649022)

[3.2.4 Evaluation of Prediction Model 41](#_Toc123649023)

[3.3 Tools and Techniques: 42](#_Toc123649024)

[3.3.1 Overview of Machine learning ML 43](#_Toc123649025)

[Chapter 4 Implementation 45](#_Toc123649026)

[4.1 Collection of Dataset 45](#_Toc123649027)

[4.1.1 Collection of Phishing URLS: 45](#_Toc123649028)

[4.1.2 Authentic/Legitimate URLs 47](#_Toc123649029)

[4.2 Feature Extraction 47](#_Toc123649030)

[4.2.1 Address Bar Based Features 48](#_Toc123649031)

[4.2.2 HTML & Javascript based Features 48](#_Toc123649032)

[4.2.3 Domain based Features 49](#_Toc123649033)

[4.3 Implementation of Machine learning Techniques: 51](#_Toc123649034)

[4.3.1 Data visualization: 51](#_Toc123649035)

[4.2.2 Data preprocessing: 52](#_Toc123649036)

[4.2.3 Train/test: 53](#_Toc123649037)

[4.2.4 Evaluation 53](#_Toc123649038)

[Chapter 5: Results and Discussion 56](#_Toc123649039)

[5.1 Results of ML: 56](#_Toc123649040)

[5.1.1 Ridge Classifier: 56](#_Toc123649041)

[5.1.2 XGBoost: 56](#_Toc123649042)

[5.1.3 Decision Tree: 57](#_Toc123649043)

[5.1.4 MLP 58](#_Toc123649044)

[5.1.5 Random Forest 58](#_Toc123649045)

[5.1.6 SVM 59](#_Toc123649046)

[5.1.7 Artificial neural network ANN 60](#_Toc123649047)

[5.2 Result comparison 61](#_Toc123649048)

[5.2. 1 Compare the generated results with state-of-the-art similar work: 61](#_Toc123649049)

[5.3 Discussion: 62](#_Toc123649050)

[Chapter 6 Conclusion & Future Work 63](#_Toc123649051)

[References 65](#_Toc123649052)

[Appendix 73](#_Toc123649053)

[Get phishing data 73](#_Toc123649054)

[Legitimate URLs: 74](#_Toc123649055)

[Feature Extractions 74](#_Toc123649056)

[Final version of CSV 75](#_Toc123649057)

[Machine learning 75](#_Toc123649058)

[Drop Domain & Status columns 75](#_Toc123649059)

[Data pre processing 76](#_Toc123649060)

[Divide the data into test and train 76](#_Toc123649061)

[ML implementation 77](#_Toc123649062)

[1. Ridget Classifier 77](#_Toc123649063)

[2. XGBoost 77](#_Toc123649064)

[3. Decision tree 78](#_Toc123649065)

[4. MLP 78](#_Toc123649066)

[5. Random forest 78](#_Toc123649067)

[6. SVM 79](#_Toc123649068)

[7. ANN 80](#_Toc123649069)

[Compare ML algorithms 81](#_Toc123649070)

# Lists of figures

[Figure 1 Scam of AOL users 9](#_Toc121945272)

[Figure 2 forms of phishing attacks. 11](#_Toc121945273)

[Figure 3 Phishing in organization 12](#_Toc121945274)

[Figure 4 Example of SMS phishing 14](#_Toc121945275)

[Figure 5 Example of Pharming 14](#_Toc121945276)

[Figure 6 SQL injection vulnerability detection 24](#_Toc121945277)

[Figure 7 Feature Extraction 25](#_Toc121945278)

[Figure 8 Proposed Framework 32](#_Toc121945279)

[Figure 9 Random Forest Flowchart 34](#_Toc121945280)

[Figure 10 SVM 35](#_Toc121945281)

[Figure 11 MLP block diagram 36](#_Toc121945282)

[Figure 12 training vs test 38](#_Toc121945283)

[Figure 13 True vs False negative/positive 39](#_Toc121945284)

[Figure 14 Confusion matrix calculation formula 39](#_Toc121945285)

[Figure 15 Machine Learning vs Traditional Programming 41](#_Toc121945286)

[Figure 16 downloading phishing URLs from Phish Tank. 43](#_Toc121945287)

[Figure 17 Downloaded Phishing URL Data 43](#_Toc121945288)

[Figure 18 8 Sample Collection 44](#_Toc121945289)

[Figure 19 legitimate URLs 44](#_Toc121945290)

[Figure 20 Random Legitimate URLs from 4000 Samples 44](#_Toc121945291)

[Figure 21 Feature Extraction on Phishing URLs 47](#_Toc121945292)

[Figure 22 Feature extraction on legitimate URLs 47](#_Toc121945293)

[Figure 23 New dataset After Marge 47](#_Toc121945294)

[Figure 24 Data Information 47](#_Toc121945295)

[Figure 25 Bar Plot for Data Visualization 48](#_Toc121945296)

[Figure 26 Histogram Visualization 48](#_Toc121945297)

[Figure 27 Confusion Matrix 49](#_Toc121945298)

[Figure 28 Data After Dropping Column 49](#_Toc121945299)

[Figure 29 Test and Train 49](#_Toc121945300)

[Figure 30 Ridge Classifier 51](#_Toc121945301)

[Figure 31 XGBoost 51](#_Toc121945302)

[Figure 32 Important Features of XGboost 52](#_Toc121945303)

[Figure 33 Decision tree 52](#_Toc121945304)

[Figure 34 Importance Features 53](#_Toc121945305)

[Figure 35 MLP 53](#_Toc121945306)

[Figure 36 Random Forest 54](#_Toc121945307)

[Figure 37 Importance Features of RF 54](#_Toc121945308)

[Figure 38 Confusion Matrix of SVM 55](#_Toc121945309)

[Figure 39 binary accuracy of ANN 55](#_Toc121945310)

[Figure 40 combine output 56](#_Toc121945311)

[Figure 41 Roc curved 57](#_Toc121945312)

[Figure 42 Precision vs Recall 58](#_Toc121945313)

# List of Tables

Table 1.1 Problem in the Current Solutions

Table 2.1: Comparison among existing mechanisms

Table 3.1 Tool specifications

Table 4.1 Results of All models

Table 5.1 Compare results with state of the art

# Abstract

Internet technologies have spread widely in recent years, not only in information-based websites but also in online social networking and online banking, making people's life simpler. As a result of this expansion, computer networks are exposed to a wide range of security risks from across the world. One of these major risks is "phishing," which seeks to fool its victims into getting their private information such as usernames, passwords, social security numbers, financial information, and credit card numbers through bogus e-mails, websites, or both. Detection of phishing attacks is difficult since it is a semantics-based assault that relies on user weaknesses rather than network flaws. In this research aimed to use a machine learning based algorithms to evaluate a unique collection of features which may be utilized to predict phishing URLs promptly and accurately using an ML method.

# List of Abbreviations

|  |  |
| --- | --- |
| Word | Abbreviation |
| USL | Unsupervised Learning |
| AOL | America Online |
| NB | Naïve Bayes |
| ZTA | Zero day attack |
| RF | Random forest |
| SL | Supervise learning |
| DNS | Domain Name Servers |
| URL | Uniform Resource Locator. |
| ML | Machine Learning. |
| ELM | Extreme learning machine |
| APWG | Anti-Phishing Working Group |
| SVM | Support vector machine |
| MBL | Model-based learning |
| ROC | Receiver Operating Characteristics |
| UL | Unsupervised Learning |
| MLP | Multi layer perception |
| BL | Batch learning |
| USAID | United States Agency for International Development |
| MAE | Mean absolute errors |

# Chapter 1: Introduction

In this chapter, I have discussed the background of phishing attacks, motivation, problem statement, research questions, aim & objectives.

## 1.1 Background on Phishing:

Although (Rekouche, n.d.), and his associates planned to deceive America Online AOL workers into exposing confidential info in 1994; they had no idea that this sort of assault, subsequently dubbed phishing, might evolve into the world's finest privacy danger in 2022.

In 1994, Rekouche was recruited by a hacker named Dave Lusby, who had established a technique to acquire access to America Online AOL members' credit card details. They both persuaded other cybercriminals to join their ring and began duping America Online AOL subscribers. Thus launched the first phishing assault in history. New, susceptible AOL customers were usually applied to these online conversations during their initial few online accounts.

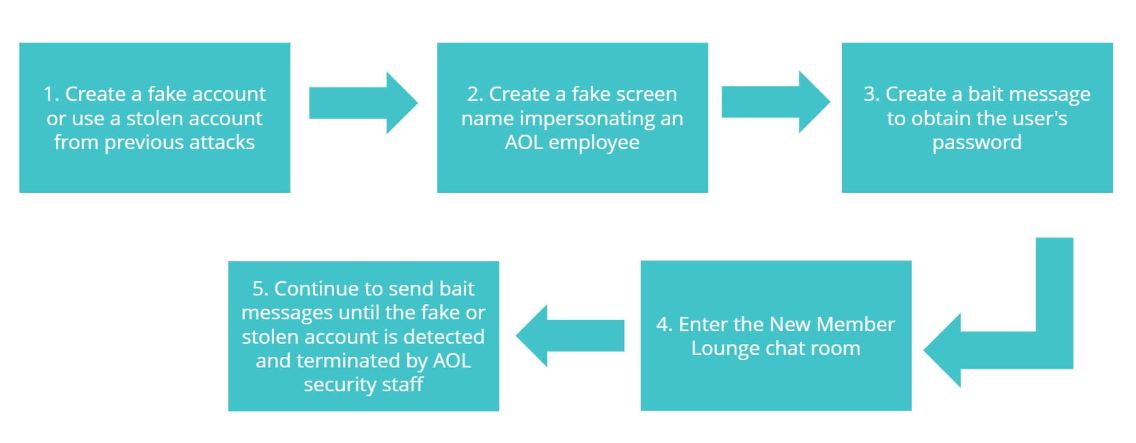


Figure 1 Scam of AOL users

In current history, the rapid growth of global online activity has led to an increase in online services in sectors such as e-commerce, online networks, and e-government. It has raised the number of data transactions on the internet, such as sensitive data exchanges. The availability and dissemination of such data have led attackers to develop phishing. This hacking tactic allows them to access the data and mimic the targets to carry out undesirable actions. Many researchers have offered many phishing schemes based on the scenario.

### 1.1.1 Phishing attacks in progress:

Phishers may compile the basic details of a casualty's personal and professional background using available sources, incredibly informal groups. These sources collect the possible casualty's names, jobs, email addresses, interests, and pastimes. When this information is obtained, the phisher can utilize it to construct a trustworthy false message.

The victim regularly gets communications from what seem to be eminent persons or organizations. Links to rebel web pages or malicious links are used to conduct attacks. Attackers could make false websites that appear to be run by trustworthy companies, including the victim's bank, workplace, or company. Attackers attempt to collect sensitive data from some of these services, such as login credentials or payment information.

Certain phishing communications might be easy to spot due to awkward wording and incorrect use of fonts, logos, and artwork. However, a lot of internet criminals are growing better at making their correspondences appear authentic, and they are employing sophisticated marketing techniques to monitor and improve the development of their communications.

### 1.1.2 What are phishing attacks?

Phishing is a tactic used by scammers to get sensitive data from their targets, such as usernames and passwords, by disguising themselves as reliable organizations. Phishing is a technique for collecting personal data using deceptive emails and websites. Phishing aims to trick email recipients into thinking they may click a link or install an attachment because the message contains something they need, such as a request form their account or a letter reorganization from a coworker.

These methods may entice you to open a file, click a link, complete a form, or respond with personal information. That logic requires you to be always alert, which may be stressful. Though there are several forms of phishing assaults, in general, phishing is a hacking effort to obtain the data of a person. Email – and, more lately, SMS messaging – is the primary weapons in this form of cyber attack. Typically, the hacker will send a message disguised as a reputable institution (through a bot), leading the target into a false feeling of security. The purpose of this attack is to deceive the recipient into thinking the message is from a reputable source, causing them to click a malicious link or download an attachment inside the transmission.

Hackers sent these emails without connection to the organizations they claim to represent. The forms they give the victim to fill out are meant to acquire sensitive data and deliver it directly to the hacker for their purpose. It's a risk for the consumer and the website that hosts the scam site. This fraud is created by breaking into other people's servers and sending emails utilizing their resources. If a hacker sends a phishing email using their help, it may be tracked back to them. This implies that if your website is on the server used by the hacker, it might harm your business in various ways.

At its foundation, phishing is a reasonably simple assault, comparable to the forms of hacking that have been prevalent on the internet since its inception. This fraud may significantly harm organizations, sites, or reputations despite its simplicity.

### 1.1.3 Use case

A phishing attempt could begin with an email that looks to be from your bank. The email address seems to be from your bank and is formatted similarly to any other email you receive. The logo is present, and everything appears to be in order.

According to the email, you must verify certain information on your bank account owing to a database breach. The URL that invites you to click is also from your bank's website. Everything is in order. The difficulty is that none of this is legal.

If you look attentively, you'll see that the reply email looks nothing like your bank's official email accounts. When you hover your mouse over the URL they offer, you'll see that the link you'd click on is not the URL mentioned but rather another email address or a bogus website.

Then there's the logic side of things. If there were a data breach, your bank would instruct you to visit their official website, input your login credentials securely, and change any passwords to protect your personal information.

Hackers are now using social media to obtain information as well. Like the bank example, fraudsters will construct and distribute bogus login pages for Instagram, Facebook, and other networks. Once they access your profile, they can take your personal information and utilize it for future fraud.

### 1.1.4 Types of Phishing

The two components of phishing attacks are technological fraud and social engineering.

An external file that holds a picture, illustration, etc.
Object name is pone.0258361.g001.jpg

Figure 2 forms of phishing attacks.

In the social engineering layer, there are attackers, victims, and those who send fake emails with forged links. Acquiring this email from a reputable and legitimate business is the first step in the process. It collects sensitive data, including user IDs, passwords, and banks detail.

The second layer concerns a fake website. The victim is taken to a fake website with a fake email that looks similar to the actual website. This layer employs key/screen logger methods, XSS, session hijacking, malware phishing, and DNS poisoning. These layers transmit the collected information and allow attackers to access the victim's PC or the original website remotely. Most frequently hacked websites are depicted in the figure 3.

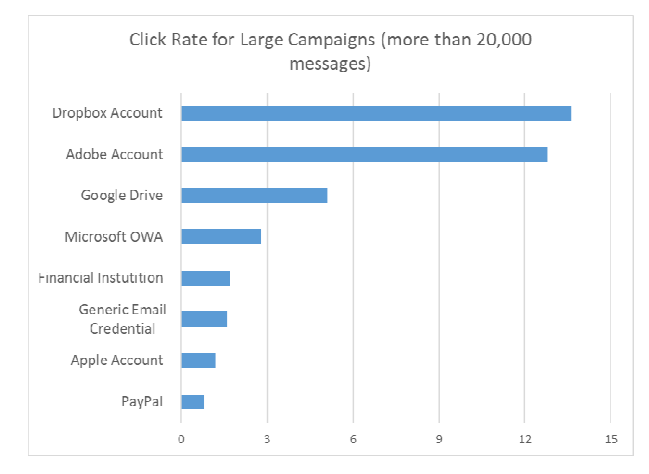


Figure 3 Phishing in organization

Email, quick chat, and texting are common methods of phishing. It is a devious method of convincing people to submit confidential material. Some other type of deception is the installation of malware/ransomware on a machine. In either case, the criminal has access to personally identifiable details.

This issue is becoming vexing since intruders may apply a variety of approaches. The following are the most common phishing scenarios:

1. **Phishing in emails**

The most common phishing case involves harmful emails sent to individuals posing as employees of legitimate businesses. With this assault, also known as email phishing, a large number of registered website users may be accessed remotely.

1. **Clone Phishing**

Clone phishing typically works since this attacker may claim that the first email was sent with a faulty link, making it necessary to send it again. Because they would be familiar with the corporate name, the recipient would not be wary of the sender. Clone phishing occasionally involves a feeling of urgency, such as a time restriction to take benefit of a promotion or a notification on the URLs that your account will be cancelled if you do not change your login details soon. Of course, the second kind of scenario, masquerade has always been undertaken for safety reasons

.

1. **Spear phishing**

Spear phishing emails need a lot of time to produce since the phisher needs to gather data from several sources. Therefore, it should be no surprise that this type of malicious assault is shared on social networking sites like LinkedIn, where the phisher may use social engineering techniques.

1. **SMS Phishing (Smishing)**

The advent of mobile technology brought about many advantages in communication and online banking. At the same time, it opened up a new point of contact for unscrupulous individuals to commit more crimes. One such is smishing, where cybercriminals lure victims through text messaging to:

* Visit rogue websites
* Download malicious apps
* Contact tech support

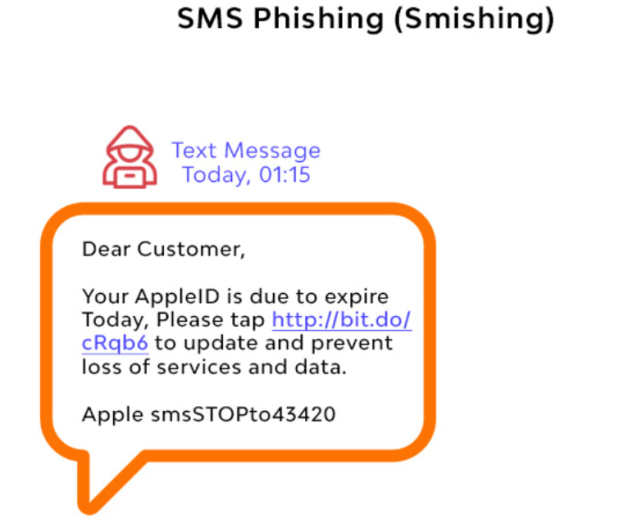


Figure 4 Example of SMS phishing

1. **Pharming**

As a more sophisticated kind of phishing, fraudsters often use pharming instead of more straightforward scams.

Pharming is when someone targets someone else by installing and executing DNS-based bespoke malicious programs. The fraudster poisons the DNS cache as part of the attack on the DNS (Domain Name System). Although less common, attacking the DNS server might jeopardize millions of online users' URL queries.

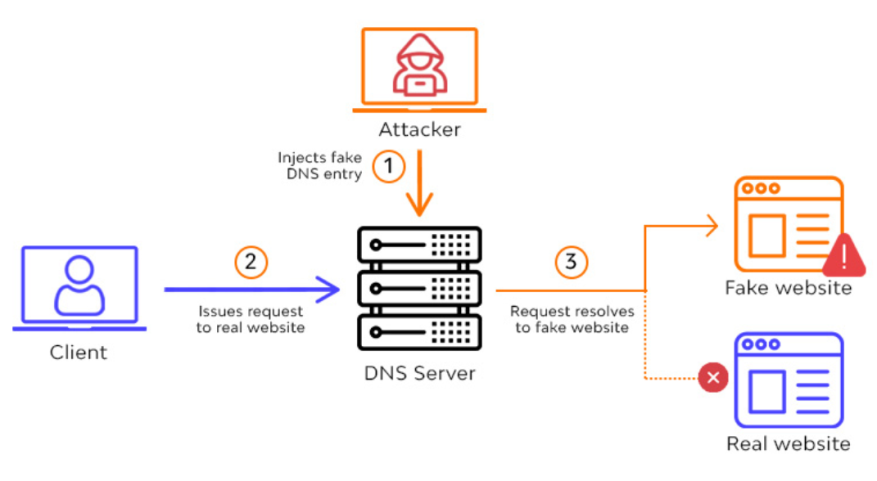


Figure 5 Example of Pharming

1. **Domain Spoofing**

Using a company's domain, an attacker can pose as the organization, the business, or one of its workers using the phishing method known as domain spoofing. Attackers use strategies to emulate another person, group, or entity to carry out hostile actions. Simple spoofing techniques include email addresses, websites, and phone numbers. In contrast, more sophisticated spoofing techniques use faked IP or DNS to trick users into opening attachments, providing sensitive data, or clicking malicious links.

The most well-known or well-liked email accounts, sites, and other online organizations are the ones that cybercriminals pick to impersonate. By lowering the amount of uncertainty and resistance, it can now capitalize on people's inherent trustworthiness. Several different kinds of fraud are referred to be domain spoofing:

* We are utilizing an email account with the recipient's website address as the sender's address to spoof the domain.
* Attackers could employ an IDN given rise attack or HTTPS spoofing, which uses visually identical web addresses.

1. **URL Phishing**

Not all links are what they seem to be. Hackers have made considerable efforts to build convincing and authentic websites. This frequently involves impersonating a well-known firm, like Microsoft. The goal of malicious actors impersonating trustworthy websites is to get end users to input their credentials. Therefore, URL phishing serves as a cover for assaults that gather credentials.

When done correctly, URL phishing may result in the theft of usernames, passwords, credit cards, and other sensitive data. The most popular ones frequently demand that users register into their bank or email accounts. Without adequate protections, end users and businesses might quickly become victims.

### 1.1.5 A real-world example of phishing URLs

These threats are one area of cyber that shows no evidence of slowing down. Instead, it seems that the exact opposite is true. From Jan to the most recent Google research, phishing URLs increased by 3506. Another study by Check Point Research found that 64% of firms have fallen victim to phishing attempts in the previous year. More research by Verizon has verified that 78% of cyber-espionage instances involve phishing.

These statistics were created using precise data from actual phishing assaults that occurred over time. The most prominent cases are as follows:

* The USAID looked to be the source of certain attack emails discovered by the Microsoft Threat Intelligence Center (MTIC) in May. (USAID). A legitimate sender email address that matched the typical Constant Contact Service was included with the talk when it came. The spear phishing emails used election fraud to mislead users into clicking a link that ultimately routed them to NOBELIUM-controlled infrastructure. The victim's computer was downloaded via that infrastructure with a malicious ISO file.

The Hacker News reported on a spam campaign run by the APT-C-36 threat actor many months later. The threat actor deceived users into opening infected PDFs or Word documents using assault emails that seemed legitimate contact from Columbian government entities. Shortened URLs in such files led recipients to a website hosting the remote access malware BitRAT.

* US giants Facebook and Google reportedly lost $100 million between 2013 and 2015 due to a sophisticated wire fraud campaign. The perpetrator established a fake company to impersonate the Taiwanese Quanta Computer enterprise.

The operation could go unnoticed for a long time by using faked company seals and fake supporting documentation for transactions. Evaldas Rimasauskas, a Lithuanian apprehended in 2017, received a five-year prison term after being recognized as the attacker.

* (Seals, 2021), published a story in June 2021 about a vishing operation that sent emails purporting to be Geek Squad yearly protection service renewal notices. The emails directed recipients to contact a phone number using branding from Geek Squad. If receivers cooperated, they were transferred to a "billing department," which tried to steal the callers' personal and credit card information. A few months later, Adrian Mitan, a 36-year-old Romanian national, was given a 140-month prison term by the Eastern District of Kennedy's U.S. Attorney's Office. After Mitan admitted guilt to three different counts, this thing happened. One of the accusations was connected to an American-targeted vishing operation.

## 1.2 Motivation

Many aspects of our everyday lives, including communication, coordination, commerce, banking, registrations, and applications, have been moved from the physical world to the virtual one due to the rapid development of internet users. As a result, flawed individuals and attackers also migrated to this domain and may easily commit threats and crimes in the dark. Using the "Cyber Security" idea, technology must be handled wisely and structured to secure the confidentiality and safety of cyber information.

One of the most destructive gaps in Internet user security is "identity theft," or "phishing urls on real time." Attackers utilize malicious web pages that pose as authentic webpages in this sort of crime to gather sensitive information from victims, such as login details, financial data, etc. The email's content entices the recipient to click on the address, which may also be hidden as a hypertext link. This URL leads the victim to a fake site that looks just like a genuine website, such as an email site, a social engineering site, or a banking website.

Although expert users are susceptible to phishing scams and can become victims, a dynamic help and security system is required to identify phishing attacks. Blocklists and safelists are typically employed as phishing detection techniques. This is a useful preventative measure that immediately determines phishing URLs from genuine ones. Furthermore, as highlighted in (Khonji, 2013), between 47-83% of phishing web pages are banned in 12 hours, sufficient time for most individuals to fall for the scam. Additionally, 63% of phishing efforts are completed in the first two hours. Blocklists and allowlists are, therefore, useless, especially against ZTA.

A dynamic and practical algorithm that can learn the structure of genuine websites and identify abnormal ones must be built to counteract this attack. To determine if a URL is legitimate or phishing, we created a classification system in this research.

## 1.3 Problem in the Current Solutions

The problem in the current solution are discussed in Table 1.1.

Table 1.1 Problem in the Current Solutions

|  |  |  |
| --- | --- | --- |
| Techniques | Problems | References |
| Email based techniques | Focus on evaluating emails using a variety of criteria. However, phishing emails have substantially improved to counteract these strategies, making them ineffective in the current environment. | (Kovač, 2022), (John Yearwood, 2010), (Jingguo Wang, 2012) |
| content-based approaches | Develop classifiers and conduct in-depth content analyses to identify phishing websites. These pieces use information gathered from web page content and outside sources like search engines and DNS servers. However, due to the massive amount of training, develop classifiers and conduct in-depth content analyses to identify phishing websites. These pieces use information gathered from web page content and outside sources for example web search DNS.  Numerous factors included in these strategies, such as URL-related characteristics, are also insufficient in capturing the phishing issue. A significant problem with most of these algorithms is that they use biased datasets and feature designs that seem to work well with them. | (Mohamed, 2022), (Krishnamurthi, 2014), (Samuel Marchal, 2016), (Castaño, 2021), (Pais., 2018) |
| URL-based approaches | Consider many factors based on the target URL, including hostname information like IP address, domain age, DNS characteristics, geographic attributes, URL length, page rank, number of dots in the URL, and the presence of special characters. Even though the assumption of these tactics is correct, namely that the URL is a strong indicator of phishing scams, sophisticated URL structure alterations invalidate numerous lexical properties revealed by these techniques. For example, websites like Amazon & Google generate today's lengthy or sometimes non-alphabetic URLs. These lessen the linguistic similarity of absolute URLs. URL-based techniques will likely fail due to this inadvertent bias against the utilised datasets. | (Mohammed Al-Janabi, 2017), (Salahdine, 2021), (Tupsamudre, 2019), (Baykara, n.d.) |
|  |  |  |

## 1.4 Research Questions

These Research Questions (RQ) follow the study's purpose and context. Which are mentioned below:

RQ1: How does Phishing URLs look like in the real world?

RQ2: How model predict if a URL link is legitimate or phishing?

RQ3: How should the performance of a URL detector be evaluated?

RQ4: Why do state-sponsored hackers regularly use phishing to breach and get a permit to access intellectual property/sensitive data on high-priority public and private website URLs?

RQ5: How many types of feature selection strategies are being used?

RQ6: ML models are based on datasets. The datasets in phishing tanks are updating regularly. What will be the effect of newly added data on the already trained models?

However, Research question 1-2 assist in developing an ML-based phishing detection system for securing URLs from phishing. However, RQ3 specifies the importance of the performance evaluation of a phishing technique. Whereas RQ4 address the motivation behind phishing.

## 1.5 Problem statements

Phishing is a type of exploitation technique that includes obtaining sensitive information from internet users while masquerading as a genuine business. It is a developing concern since it uses human vulnerabilities, not system ones. To counteract human weakness, internet users must be taught and trained to differentiate between simple & phishing URLs. On the other hand, assuming users analyze the URLs & investigate correspondingly is illogical because accuracy depends on the user's knowledge and competence. Furthermore, attackers are always developing new phishing methods by exploiting flaws in current systems, rendering training sessions and anti-phishing technologies worthless. Human education and training are the primary options for minimizing the effect; however, this alone is insufficient. As a result, we want a more effective and exact technique for identifying phishing URLS.

Because phishing URLs are so important and common in real-time, an optimal solution for rapid and effective URL detection should have the following design details:

* It cannot rely on databases of known or suspected phishing URLs derived from complaints filed by persons or software.
* When phishing URL techniques develop, ML has the edge over heuristics. It can keep the greatest detectability by updating the prediction rules through data re-collection and re-training. As a result, it is a superior course of action. On the other hand, the new technique is tough to hold and is prone to high error rates when changes occur.
* High predictive performance & minimal misclassification rates are required. In this domain, accuracy levels between 99.9-100% and misclassification levels around 0- 1% are selected.
* It must maintain real-time detection, which means that the user's overall web surfing experience must not suffer due to the extra time needed to assess whether a URL will conduct a phishing webpage.
* It ought to implement fresh prediction characteristics. This is due to the possibility that the hackers have already mastered predictive features employed by current detection methods and has created procedures to go around the solutions.

Since, the phishing tank datasets are updating regularly that is why the already trained models are not capable to identity the newly added phishing attacks particularly, if the new attacks have new features. The focus of this current work is to apply ML models to large and updated datasets and to improve the performance.

## 1.6 Aims and Objectives

This study aims to examine, propose, and evaluate a unique collection of features which may be utilised to predict phishing URLs promptly and accurately using an ML method.

To meet the aims of the project, the research illustrates aims have been defined:

* + Dataset gathering & pre-processing
  + Identifying the most relevant sets of characteristics and applying them to develop a unique ML-based prediction model for detecting phishing URLs.
  + To assess the accuracy and efficiency of the result prediction models.

## 1.7 Summary of the proposed solution

In this research, provide a series of approaches that target different types of phishing sites to mitigate and enhance the performance of the existing techniques in real-time. Researchers focus on selecting dangerous URLs from the vast database of URLs, in contradiction to the majority of past techniques. As a result, my research recommends an ML-based solution for URL identification. The rest of the thesis is organized as follows:

In chapter 1, I introduce the concept of Phishing URLs and the study's objective. The background of the research. In chapter 2, I discuss literature review and present the related review studies. Similarly, I compare these with my study and find gaps analysis. In chapter 3, I discuss the proposed methodology. Similarly, chapter 4 results and discussion are presented. Chapter 5, concludes the study with its future direction.

## 1.8 Applications of the proposed mechanism in the real world

Tools for data analysis are widely available because of ML. Recently; it has demonstrated inspiring results in the struggle against phishing URLs in real-time. ML methods are valued for spotting phishing assaults. Different ML approaches have been used to protect customers against phishing devices. The phishing site may be identified based on specific crucial features, including URL & Domain Identity, security, and other factors. My solution will be helpful if a user completes an online transaction and makes a payment through the website. Many e-commerce businesses may operate this programme to defend the entire transaction process. In Chrome, users may add extension files. The consumer can confidently buy things from the internet market using this extension file.

Chapter 2: LITERATURE REVIEWS

In this chapter, I discuss the previous and existing works on malicious URL detection with different approaches. Machine learning is becoming increasingly popular in these fields. Based on the characteristics used, existing malicious website detection approaches may be grouped into the following categories:

1. Static and Dynamic Features Approaches based
2. Feature-based approaches: depend on features collected from the URL, page content, and other sources.
3. Domain-based information approached: HTML DOM structure (such as WHOIS and DNS records)
4. Dynamic feature-based solutions, on the other hand, are primarily concerned with analyzing behaviors gathered when the page is loaded and displayed or studying system activity Logs when certain programs are run.

## 2.1 Static and Dynamic Features Approaches based

(O'Mara, 2021), Evaluated many conventional and ensemble-based models and offered the best models and model settings with high prediction accuracy. The practicality of evading phishing classifiers was then studied by perturbing static and dynamic factors using AI productive models and evaluating both conventional and ensemble classifiers. The results show that it is more difficult to avoid phishing classifiers that depend on dynamic content characteristics, which provide robustness against evasion strategies. According to their findings, evaluating static and dynamic characteristics of websites has a significant potential for application in adversarial learning to design phishing attacks.

(AlEroud, 2020), proposed a way for creating URL-based phishing instances using Generative Adversarial Networks. These scenarios are used to deceive Blackbox's ml-based phishing detection techniques. Although when sophisticated approaches, such as intra-URL similarities, are utilized to create Blackbox phishing analyzers, the samples produced can trick them. They tested their approach on real-world phishing datasets. According to the findings, GAN networks are particularly adept at creating adversarial phishing instances that can trick basic and advanced ml phishing detection algorithms.

(Yang, 2022), proposed DSM is built on a combination of static and dynamic data. The model investigates the effects of susceptibility and dynamic characteristic variables (history, demographic, and events) on static characteristic factors (design changes and eye tracking). A hybrid LSTM & LightGBM model for the prediction applied was used to identify user vulnerability. Lastly, they assessed the DSM's capacity to forecast using a survey questionnaire of 1150 people and an eye-tracking experiment on 50 participants. Based on the experimental results, the DSM outperforms individual feature prediction in terms of correct prediction rate.

(da Silva, 2022) Offer a rule-based model approach for phishing prediction called piracema.io. They identified static & dynamic characteristics based on the LR results, considering their relevance, connection, and similarity. The work employs a statistical technique to examine the prediction modelling across the proposed progressive depth and adherent acting strategies as evidence of concept. As a consequence, the author proposal's qualitative information, offering contributions, dangers, and limits, as well as ideas for future work for the model.

## 2.2 Feature-based approaches:

(Cheng, 2022), Provides a simple detection approach for identifying harmful domain names based on AdaBoost, particularly emphasizing proactively detecting dangerous domain names by investigating aberrant WHOIS information. The suggested approach may be used by domain name registries and registrars as the first line of defense to discover rogue domains during domain user registration. Substantial studies on a large-scale database show that the suggested technique performs well on various fraudulent web addresses.

(Mithra Raj, 2022), investigated eight existing machine learning classification algorithms to prevent suspicious web pages.  According to authors, XGBoost had the most remarkable accuracy of 96.71%. As a result of the execution results on the malicious site dataset, XGBoost outperforms the other classification algorithms in categorizing websites as phishing or legitimate.

(Hassan, 2021), Proposed a SQL injection technique. It employs feature selection approaches & a feed-forward NN to optimize the dataset's output. Finally, the NN model is used with the dataset to forecast the SQLi risk of the target web app.

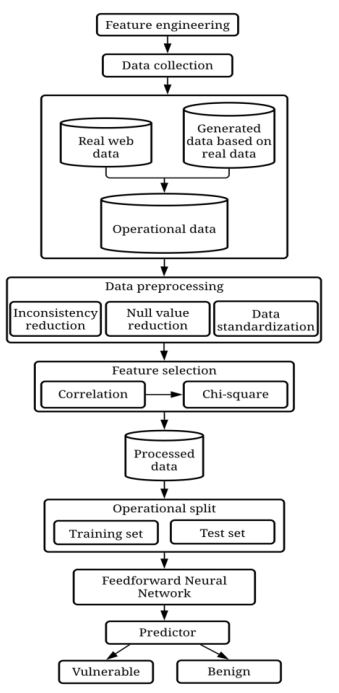


Figure 6 SQL injection vulnerability detection

Not only has the feed-forward network method been used in the feature. Not just in the selecting process but also in the detecting process. 98.04% is provided by our solution. With over 1,850 recorded datasets, it indicates outstanding efficiency in addition to other machine-learning solutions.

(Thabtah, 2017), proposed a more systematic method for determining cut-off ranks for features rated by IG and Chi-Square. Chi-Square & IG were used to benchmark the features for phishing website detection. A threshold-based rule set is offered. The cut-off ranking is represented as two features with at least a 50% difference in IG and Chi-Square values. If a cut-off rank falls below the recommended minimum value of the filter measure.

2.3 Domain-based information approached

(Dattaa, 2022), proposed Ml algorithms that have been used to detect phishing URLs. The best-suited strategy was developed & refined using another ML approach, yielding about 97.0% testing accuracy.

An interactive & responsive web framework was designed to make this project more user-friendly. The characteristics of phishing domains and the qualities that separate these domains from anti-phishing domains are detailed here. The phishing URLs in the body of the messages are made to look to go to the fraudulent organization by using that company's logos and other genuine information.

(Aung, 2019) , study concentrated on details characteristics using a neural network-based that takes into account a combination of domain-based & path-based characteristics. Then, the authors compare our findings to earlier articles, collecting suggestions for a more effective testing method. Eliminate the bottleneck caused by manually created features. By selecting information-rich characteristics, we may target them. They spoke some profound things. The authors use tokenization techniques to use two alternative embeddings in the model layers. They use LSTM after each dropout layer embedding, concatenating the outputs vector & then conducting batch-normalization of a thick coating.

Because relevant words/brand names are most commonly found in the domain portion, they tokenize URLs to create a list of terms. The features extracted in this case are phrase characteristics combined with non-alphanumeric letters. The feature extraction is shown in Figure 7.

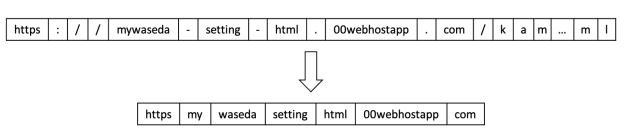


Figure 7 Feature Extraction

Lastly, they corresponded the implementation evaluation with URLNet by applying variables similar to previous work, which improved by around 5 to 1% using the sample. In (Shirazi, 2019), the authors offer a simple but effective method for simulating assaults by directly manipulating hostile samples. They suppose that perhaps the intruder is unfamiliar with the parameters, learning techniques, or datasets used only for training. They did some tests using four currently accessible datasets from the Online platform. The results of our tests show that phishing identification measures are sensitive to adversarial learning techniques. Modifying a single parameter reduced the recognition rate for malicious scams to 70%. Whenever four characteristics were changed, the recognition rate decreased to 0%. This indicates that every dataset of phishing, identified accurately by a base classifier may be bypassed by altering only four feature variables, a small exertion for an intruder to receive such a large payout. By each sample, we create the idea of vulnerability level, which evaluates the number of factors that may be modified and the cost of each modification. A statistic like this will enable us to evaluate different defensive algorithms.

(Gupta, 2021), improve the detection accuracy of harmful URL identification by creating and constructing a cyber threat algorithmic misuse URL - based detection algorithm based on two-stage ensemble methods. Cyber threat intelligence-based characteristics are derived from online searches to boost detection accuracy. Cybersecurity analysts and user reports worldwide can supply critical information about harmful websites. Cyber threat intelligence characteristics derived from Google searches & Whois websites are employed to boost detection performance. In addition, the study suggested a two-stage ensemble learning model that combines the random forest (RF) method for pre-classification with the multilayer perceptron (MLP) algorithm for final decision-making. The trained Classification method has replaced the three trained random forest classifiers' general voting mechanism for decision-making. The uncertain output of the random forest's weak classifiers was collected & utilized as feed for the MLP classifier for sufficient classification. The results reveal that the retrieved CTI-based elements with the organization mean to beat the classifier model in previous investigations. Compared to the old URL-based model, the suggested CTI-based detection algorithm improved accuracy by 7.8% and reduced false-positive rates by 6.7%.

## 2.4 Dynamic feature-based solutions

(Rao, 2020), Proposed a technique inexpensive feature-based ML approach for detecting malicious websites. The extracted features are gathered from many sources, including URLs, source code, or 3rd party services, before input into the supervised ml classifier. The model achieved a significant efficiency of 99% using an RF method, with a TPR & a TNR of 99%. The ML-based strategy, on the other hand, achieved a high degree of accuracy. Despite the great precision of the ML-based technique, the procedure may collapse when phishing pages hosted on compromised networks (PSHCS) are recognised due to the use of 3rd party services such as search engines and webpage ranking algorithms. They presented two strategies, one with third-party support and another without.

The first offers a unique heuristic technique for spotting fake registrants, phishing URLs & sites hosted on compromised servers based on the TWSVM. The semantic relationship between the home website and the troublesome site is calculated using the suspicious site's home page. This strategy may fail if the correct web page of the accused site is not obtained.

The options presented by the authors rely on the website's source code and third-party services that require the page to be loaded to determine the website's state. As a result, the client-side reaction time to the identification process may be delayed. Furthermore, because the webpage is certain to be seen, there is a larger chance of malware being downloaded inadvertently via the website.

As a consequence, they created two minimal URL-based techniques. These methods will be used for the first phishing URL assessment without accessing the suspect site. The first approach is a web application that identifies TFIDF features.

The earlier described solutions either employed content or URLs to identify phishing, but they lacked information about the target URL of the planned malicious website. To that end, we introduced a compact visual, physical resemblance solution that keeps finger the trends of banned phishing sites alongside their legal target domains. The approach also incorporates heuristic capabilities for detecting phishing sites that target non-whitelisted lawful websites.

(Chiew, 2019), Proposed a unique dynamic feature selection methodology for machine learning-based phishing detection techniques In the first step of HEFS, an innovative Cumulative Distribution Function gradient approach is utilized to produce multiply effects feature subsets, which are then input into a data concerns ensemble to produce secondary component subsets. A function awareness ensemble extracts baseline features from secondary feature subsets in the second step.

(Das Guptta, 2022), they create a new dataset to undertake tests using primary classification by machine learning approaches. Their testing results reveal that the suggested phishing detection strategy is more successful than conventional techniques, with a detection performance of 99.01% using the XG Boost method. Therefore, whenever HEFS is used with a Classiﬁer, the p values properly differentiate 94.6% of phish or allow URLs while employing just 21% of the original dataset. A second experiment used various models utilising the RF from among ten.

## 2.5 Comparison among existing mechanisms

I choose a few research studies to compare with the suggested URL detector based on the relevant work and performance. These studies were selected because they utilized ml techniques and had an average accuracy of 90%.

**Table 2.1: Comparison among existing mechanisms**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Authors** | **Contributions** | **Limitations** |
| 1 | (Basit, 2021) | SVM, RF, ANN, C4.5, k-NN, and DT were employed as classification techniques, whereas feature extraction with ANOVA was used. | Machine learning algorithms are faster as compare to scenario-based environment |
| 2. | (Odeh, 2021) | RF, SVM, NB, & Ada Boosting. Author also implement deep learning and compare result with ML and highlight automated solution to phishing websites. | ML have low accuracy and overfitting as compare to Deep learning |
| 3. | (Gandotra E., 2021) | Implemented a variety of categorization techniques to find dangerous URLs. | The results of the studies showed that the system's effectiveness was superior to that of other ML techniques. It cannot handle more significant amounts of data. |
| 4 | (Purbay M., 2021) | used a variety of ML techniques to categorize URLs. | They evaluated the effectiveness of several ML techniques. Regardless, there were no comments concerning the algorithms' retrieval power. |
| 5 | (J. Kumar, 2020) | Presented a URL analyzer based on a collection of blocked URLs. Lexical feature method was also used to distinguish between dangerous and trustworthy websites. | Using an outdated dataset could affect how well the detector performs with real-time URLs. |
| 6 | (da Silva, 2022) | Proposed  prediction based on a decision tree with based on 30 features & 25 rules | Work on only phishing pages |
| 7 | (Das Guptta, 2022) | Present hybrid feature-based anti-phishing technique that extracts features from client-side URL & hyperlink metadata | Only work on search engine based features |
| 8 | (Cheng, 2022) | Proposed AdaBoost to identify malicious domain names | Work on  various malicious domain names. |
| 9 | (Mithra Raj, 2022) | Proposed XGBoost, RF, adaboost, decision trees KNN, SVM, LR & naïve bayes (NB) to detect malicious websites | XGBoost performs better for detecting phishing or legitimate |
| 10 | (Dattaa, 2022) | Proposed LR & Multinomial Naïve techniques that is mostly utilized in NLP | RF algorithm & black list method to build the phishing detection system as a scalable web accommodation |

## 2.6 Research Gaps

The majority of methods rely on browsers to detect phishing URLs. Due to this dependence on browsers, enrolled domains are classified as phishing sites, which reduces the TNR. Phishers place their phishing website on less reputable and safe websites. It implies that data from compromised websites are being exchanged on the darknet. These websites are difficult to delete since most anti-phishing programs consider them legitimate. As a result, an independent or upgraded search engine-based detection approach from a third party is necessary. Another flaw is that Phishing URLs that run embedded objects like HTML or scripts can be identified. As a reason, identifying phishing sites that use embedded objects remains difficult. Previous research has concentrated on developing or testing behavioral models rather than forecasting users' susceptibility to phishing emails, or the current model suggests the potential of exposure prediction.

Furthermore, phishing sites use java script to make their content dynamic, allowing them to avoid anti-phishing systems that rely on static source code of URLs. The most well-known algorithms retrieve textual characteristics from the static source code of a website.

Exploring and identifying live phishing sites with this kind of dynamic content are thus a relevant research area. Ensemble learning is repeatedly used for classification but may also use for feature extraction. In conclusion, one of the research goals is to analyze the study and develop a more efficient feature selection ensemble for selecting the appropriate number of features.

# Chapter 3: Methodology

This Section delves further into the study technique & architecture phishing URL detection investigations in depth. My research show how ML algorithms work and the proposed framework, which includes the design architecture, essential systems, and standard components. The historical chronology of our investigations, major design decisions, ethical issues, & our test-driven development method are all examined.

## 3.1 Research Methodology followed (Qualitative/Quantitative)

A quantitative approach is used. The primary objective is to offer to users a safe environment for detecting Phishing URLs at real time. In addition, an experimental investigation is carried out in which a phishing URL detection system is employed. In addition, literature-based analysis is carried out to gather required knowledge about the issue, collect background information, and identify research gaps.

## 3.2 Proposed Framework

A phishing URL is a common social engineering method that impersonates reliable URLs and internet services. This research aims to train ML models & ANN on the phishing website dataset. To create a data sample, phishing vs real URLs are gathered, and needed URL & site information features are retrieved from them. Each model's performance level is evaluated and contrasted. Figure 8 depicts our suggested structure.

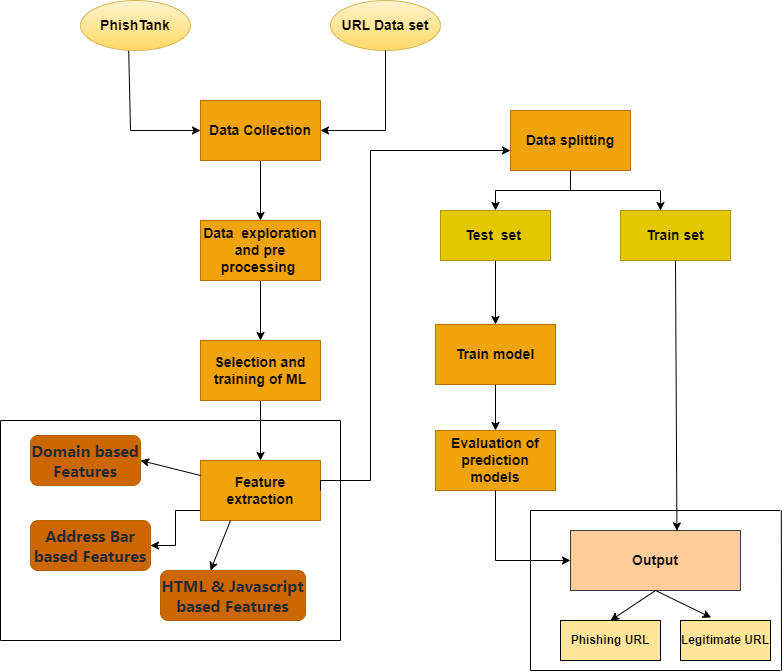


Figure 8 Proposed Framework

Every classification has a unique collection of phishing features & attributes. The given characteristics are extracted for each URL, and acceptable input ranges are found. The risk of each phishing URL is subsequently assigned to one of these levels. The input values range from 0 to 10, whereas the output numbers range from 0 to 4000. Extracted features are binary digits 0 - 1, indicating whether or not the characteristic is available.

I next performed data training, dividing the data into 70/30 before stimulating the ML model for training. And a suitable ML algorithm for the dataset. ML algorithms have received much attention. Following that, we use regression and classification algorithms such as SVM, RF, ridge classifier, MLP, XGbosst, Neural network & DT to anticipate the accuracy of identifying the phishing URL, is yielding the desired result.

Following that, I will test the data and determine the forecast accuracy, which will be greater than the current system. I'll now look at the various classifier/regression challenges & talk about the suggested framework we employed in this study.

### 3.2.1 Data collection

I got Phishing URLs for this study from the open-source website Phish Tank (admin, 2020). This site gives a collection of phishing URLs in several formats, such as CSV, JSON, & others, updated hourly. This dataset comprises four thousand phishing URLs used to train machine learning algorithms.

The other dataset was obtained from the open datasets of the University of New Brunswick (dataset, 2021). This dataset contains regular URLs, phishing, fraud, malware, or URL defacement. The URL dataset is being evaluated out of these types for this research. This dataset provides 4,000 unique random

### 3.2.2 Data preprocessing

The initial stage in ml for model construction is data preprocessing. Data preprocessing is how ML algorithms may use raw data. In the actual world, data is irregular; preprocessing organizes disorganized information for the data visualization stage. Data preparation comprises data cleansing to prepare the data using a machine learning model. These are the conclusions we reached after employing preprocessing data procedures

### 3.2.2 Feature extraction

Certain features appear to be more important than others in getting an accurate categorization in any ML method. When paired with other prominent or minor features, these features can deliver exceptional results. The difficulty emerges when we must determine which characteristics from a collection are the most significant and which combination of features gives us near-flawless classification accuracies.

The below-mentioned category of features is extracted from the preprocessing URL dataset:

1. Address Bar-based Features (In this category, 9 features are extracted.)
2. Domain-based Features (4 features are extracted.)
3. HTML & Javascript-based Features (4 features are extracted.)

### 3.2.3 Machine learning models

Detecting & recognizing Phishing URLs is a difficult and ever-changing subject.  ML has been widely employed to produce automated solutions in various fields.

It is evident from the dataset preparation stage that this is SL ML approach. Apply the classification and regression techniques listed below, depending on the application and nature of the dataset. We can't tell whether algorithms are better or worse because they're used for various things. Each classifier & regression has its unique method of operation & type. I go through each one in detail.

#### Random ForestRF

The RF method is a popular classification & regression algorithm. This classification approach is similar to the ensemble classification method. Regression and other tasks are carried out by generating a set of DT at the training data level and during class output, which may be classified or predicted by smaller trees. This classifier accuracy technique of overfitting the training data set. Figure 9 (admin, 2021) show the flow of random forest.

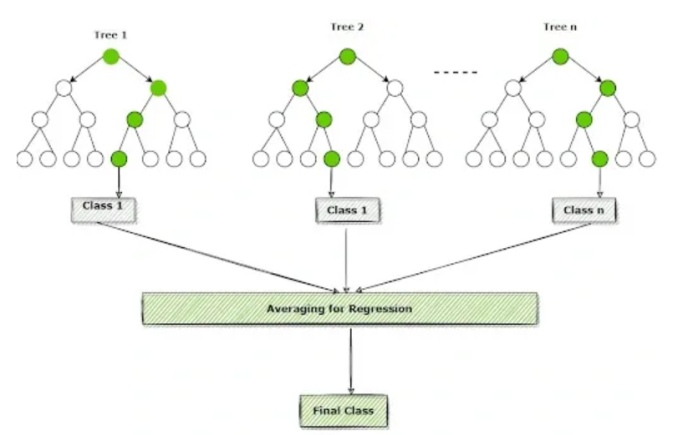
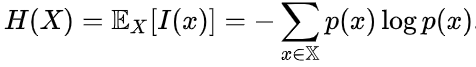


Figure 9 Random Forest Flowchart

#### Decision tree

Using the DT strategy is extremely straightforward in comparison to other categorization approaches. To solve the issue, the decision tree method employs a tree. The interior nodes of the tree contain attributes, whereas the leaf nodes represent class labels.

The general equation is mentioned below.



#### XGBoost

It was created **by PhD student Tianqi Chen, University of Washington. XGBoost is a very efficient, adaptable, & compact ML technique that leverages distributed gradient boosting.** (Vito, 2017)**.**

#### ****Support Vector Machine****

SVMs might be used to solve binary classification difficulties. However, as the number of computationally complex multiclass issues increases, multiple binary classifiers are built and blended to build SVMs capable of executing such multiclass classifications by binary means.

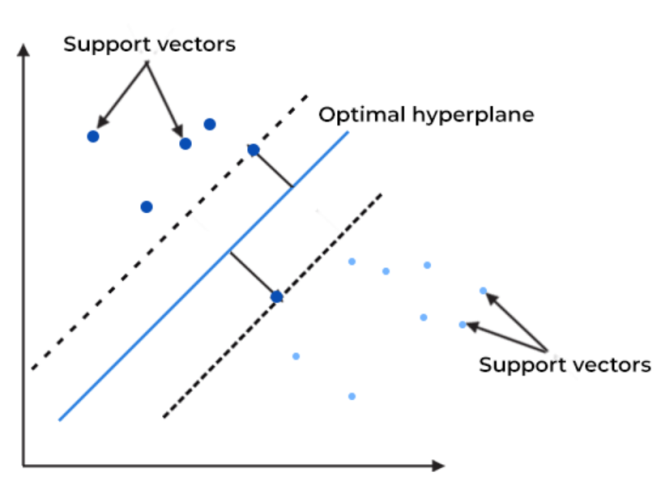


Figure 10 SVM

#### Multilayer Perceptrons MLP

The Perceptron actually identifies data sets that can be separated linearly. They run into significant concerns with data sets that do not follow this pattern, as demonstrated by the XOR issue. The XOR problem illustrates a collection of four points that are not linearly separable for any categorization of four points.

The Perceptron comprises two completely linked layers: input and output. MLPs have the same input and output layers but may have numerous hidden layers in between, as shown in Figure 11.

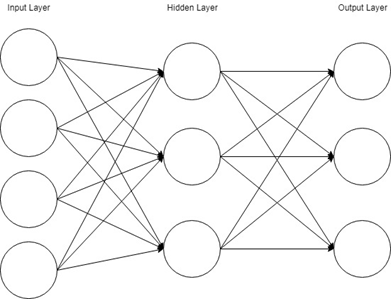


Figure 11 MLP block diagram

The MLP algorithm is as follows:

* The MLP, like the Perceptron, pushes inputs forward by taking the input's dot product and the weights between the input layer and the hidden layer (W­­­H). At the buried layer, this dot product produces a value. We do not, however, advance this number as we would with a perceptron.
* MLPs employ activation functions at each computed layer. There are several activation procedures to consider: rectified linear units (ReLU), sigmoid function, & tanh.
* Once the estimated output at the hidden layer has been pushed via the activation function, take the dot product with the relevant weights and send it to the next layer in the MLP.
* The calculations will be operated at the output layer for either a backpropagation technique that corresponds to the activation function chosen for the MLP (in the event of training) or a decision based on the output (in the case of testing).

### Ridge Classifier

Ridge classification is a machine-learning approach for analyzing linear discriminant models. It is a type of regularization in which model coefficients are penalized to prevent overfitting. Overfitting is a typical problem in machine learning in which a model becomes too complicated and catches noise in the data rather than the underlying signal. This might result in poor generalization performance when dealing with new data. Ridge classification tackles this issue by including a penalty term into the cost function that discourages complexity. As a result, the model is more generalizable to new data (Tertytchny, 2020).

The ridge method can achieve by introducing a cost function penalty term that discourages complexity. The penalty term is usually the amount of the squared coefficients of the model's components. This causes the coefficients to stay modest, preventing overfitting. The penalty term can be changed to adjust the amount of regularization. A higher penalty leads to more regularization and lower coefficient values. This can be useful when training data are scarce. However, if the punishment time is too long, under fitting might occur.

***Y = XB + e***

The Ridge classifier's loss function is not a cross-entropy loss like Logistic Regression. The loss function is instead mean square loss with L2 penalty. It uses the Ridge regression technique to solve binary classification problems in the following way:

* Transform the specified variable to +1 and -1 as needed.
* Train a Ridge model with a mean square loss function and L2 regularization (ridge) as the penalty term.
* If the expected value is less than zero, the predicted class label is -1; otherwise, the scheduled class label is +1.

### 7. Artificial neural network:

Artificial neural networks are programs derived from the mind neural nets that adapt & respond to input, enabling them to make judgments based on that information like humans do. As a starting point for training, ANNs necessitate a pool (Brownlee, 2020).. The more data there is, hence more interconnections a NN can create & develop. As an ANN develops, it can consistently provide accurate outputs based on the answer a user is looking for. Deep neural networks are artificial neural networks (ANNs) with hidden coatings between input and output. DNNs are used by developers when creating an intelligent application with deep learning features. Further deep learning methods, such as image recognition and natural language processing, are built on artificial neural networks.

Once the model has been trained, it is critical to analyze and test the classifier so that we will utilize its potential. The results are obtained after using the categorization; indeed, the URLs were classed as phishing or real. The Phishing URLs are legal and are banned in the database.

### 3.2.4 Data Splitting

An ML algorithm operates in two stages when dealing with datasets. We normally divide the data between the testing and training phases by 80-20%. In Python ML, we employ supervised learning to divide a dataset into tests and training data as shown in Figure 12.

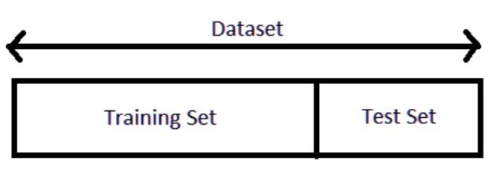


Figure 12 training vs test

### 3.2.4 Train Model

The most critical phase in ML is training. During training, you feed a stable structure to your ML model, which looks for patterns and accurately predicts them. Consequently, it get from dataset and can complete the assigned goal. The model improves in guessing over time as it is trained.

### 3.2.4 Evaluation of Prediction Model

I used seven of the dataset's most reliable ML methods for the dataset and evaluated each classification model's accuracy, precision score, recall, & f1 score. The confusion matrix, which typically measures true positives tp, tn negatives, FP false positives, or fn false negatives, is used to build these evaluation criteria.

To assess the classification capabilities of my models, the predictions with the lowest validation loss were chosen & used to classify the data in the test set.

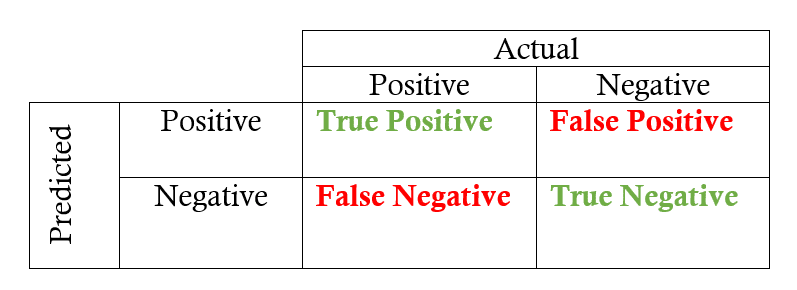
Figure 13 depicts and evaluates the classification results for each model using a confusion matrix.

Figure 13 True vs False negative/positive

By constructing confusion metrics that gauge the effectiveness of anomaly detection per class using the values from the matrix, we further assess the optimal model for each neural network. Accuracy is the percentage of predictions that are accurate across all samples (O'Reilly, 2014).

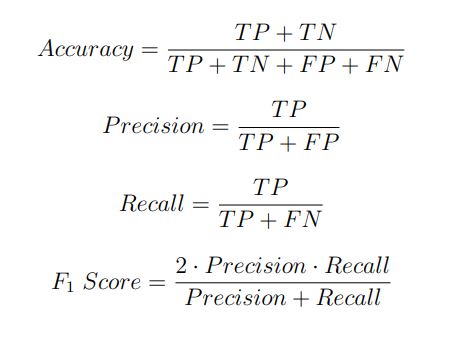


Figure 14 Confusion matrix calculation formula

## 3.3 Tools and Techniques:

I use Python, which is an analyzed high-level programming language. It follows a rigid design philosophy that stresses code readability due to the effective use of spacing. Python is becoming more popular among data scientists & software engineers. Python is more than a calculating tool; it can also be used to build a website, libraries, (GUIs), scientific analyses, and even applications and video games.

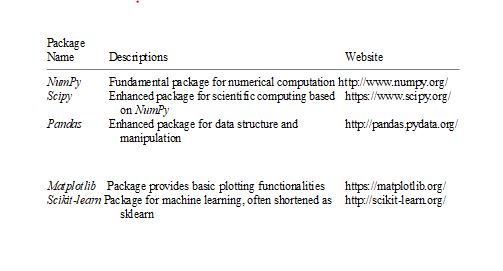
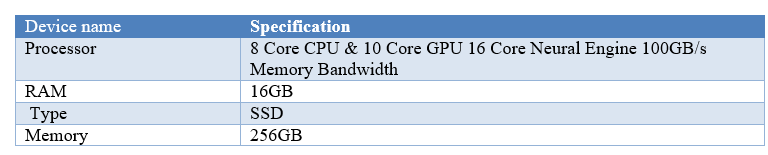


Figure 14 Python library

**Table 3.1 Tool specifications**



### 3.3.1 Overview of Machine learning ML

Artificial intelligence AI has already accomplished feats once thought to be science fiction:

* Self-driving cars.
* Computer vision models evaluate cyber-attacks more appropriately than a researcher.
* Predictive business analysis help businesses comprehend their customers stronger than their business.

#### What distinguishes ML from existing computing?

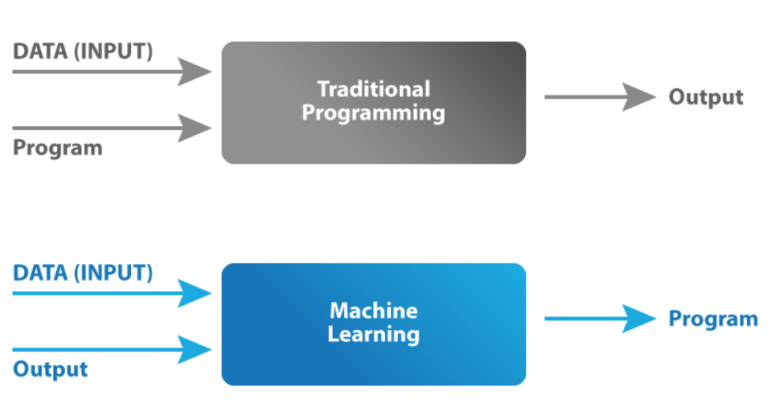
An equation of the patterns buried in data is a machine-learning model. The algorithm for machine learning finds some controlling structure when it has been programmed (or created or fitted) to the training data. This controlling framework is codified into rules, which may be used to anticipate outcomes in novel circumstances. Therefore, the model would be able to infer the relationships inside it if we train it on specific training data but then apply it to new data.

Figure 15 ML vs Traditional Programming

#### What is the Significance of Machine Learning?

Machine learning, specifically with significant developments, may provide new chances to your organization, regardless of industry. One of the reasons machine learning has gained so much attention is that it is being utilized to enable advancements in apparently unrelated domains such as:

* Image processing (for example, facial recognition);
* Sound processing (for example, automatically captioning films with subtitles); \Text processing (for example, translating between natural languages);
* Time series analysis (for example, projecting future energy consumption);
* Numerical simulation (for example, estimating a fair price for a house or the probability that a particular customer will buy a specific product).

Any of these domains expand into additional subfields (for example, the same algorithms used for facial recognition can also detect cancer in X-Rays), and, more tellingly, very roughly comparable algorithms can be used across all of these fields, implying that advances in one algorithm can lead to advances in many. This is true for every profession, including medicine, marketing, banking, and trading.

Massive (and rising) datasets enable machine learning in these domains. Machine learning advancements uncover more of the value in this data, increasing the value of the data itself. As more attention is paid to data, machine learning models improve, creating a virtuous loop.

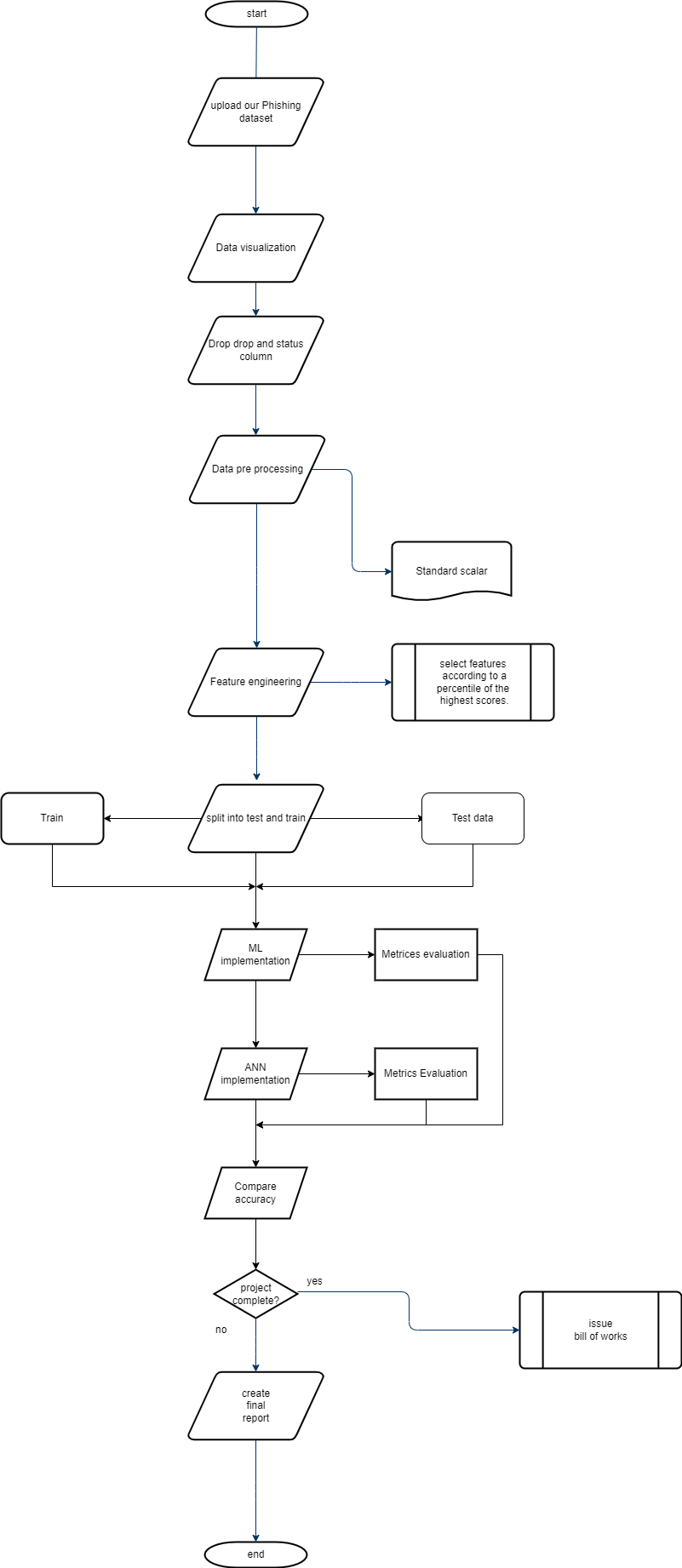
1. **How Machine learning work in Phishing?**

Most people are conscious of phishing, which occurs when an attacker sends a fraudulent email that appears to be authentic. Phishing emails or URLs that appear to be from trustworthy institutions (banks, Amazon, Netflix, etc.) urge the receiver to enter into their account to resolve the issue. Attackers can obtain login passwords and other personal information by creating a website that looks like the actual site.

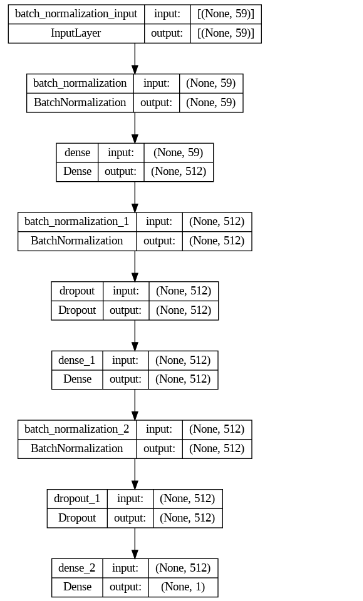
Phishing emails are a more specific type of phishing email. Instead of using a generic pretext (such as an email from a well-known bank), spear phishers investigate their chosen targets and customize their malicious emails to that victim. An easy example is when an attacker utilizes the business logo and the name of a CEO to create an attack that appears to be the CEO directing an employee to do something for them. These phishing emails are more likely to fool their recipients since they appear more natural and intimate.

Detecting phishing URLS is integral to a company's cyberdefense plan. However, as phishing emails get more complex, they become increasingly difficult to detect.

**Flow chart of complete code**



Flow chart for ANN:



**Step 1**: Initialization Set all the weights and threshold levels of the network to random numbers uniformly distributed inside a small range.

**Step 2:** Activation Activate the artificial neural network by applying inputs and desired outputs (a)Calculate the actual outputs of the neurons in the hidden layer:

(1) where n is the number of inputs of neuron j in the hidden layer, and relu , sigmoid is the activation function.

(b)Calculate the actual outputs of the neurons in the output layer

(2) where m is the number of inputs of neuron k in the output layer.

**Step 3:** Weight training Update the weights in the back-propagation network propagating backward the errors associated with output neurons.

(a) Calculate the error gradient for the neurons in the output layer:

(3) Where

(4) Calculate the weight corrections

(5) Update the weights at the output neurons:

(6) (b) Calculate the error gradient for the neurons in the hidden layer

(7) Calculate the weight corrections

(8) Update the weights at the hidden neurons

**Step 4:** Iteration Increase iteration p by one, go back to Step 2 and repeat the process until the selected error criterion is satisfied.

# Chapter 4 Implementation

## 4.1 Collection of Dataset

In this research, consider a lot of legitimate (0) & phishing URLs(1) for our research. The open source tool Phish Tank makes it relatively simple to collect phishing URLs. This website provides an hour of updating phishing Websites in many formats like CSV, JSON, and many others. Go to https://www.phishtank.com/developer info.php to get the data.

### 4.1.1 Collection of Phishing URLS:

The URLs are collected from the Phish Tank using the provided link. The wget application is used to get a CSV file containing phishing URLs. That whenever a dataset is acquired, it is loaded into a DataFrame. Figure 16 depicts the end outcome.

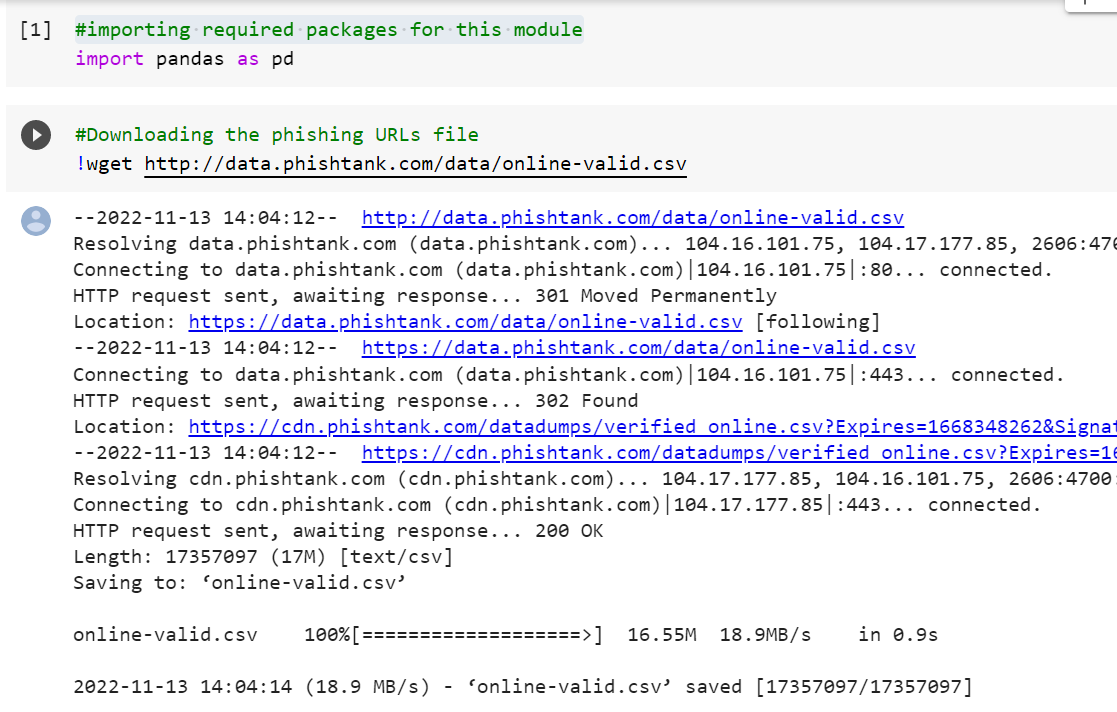


Figure 16 downloading phishing URLs from Phish Tank.

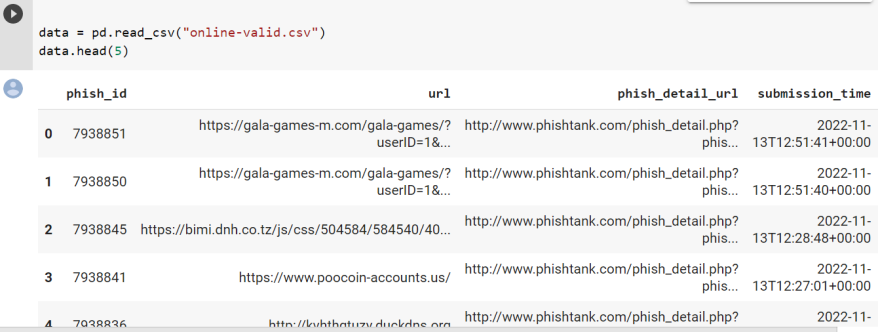


Figure 17 Downloaded Phishing URL Data

As a result, the data contains millions of phishing URLs. The issue is that the aforementioned data is updated regularly. Despite going into the possibility of data imbalance, I'm thinking about a margin of 11,000 phishing URLs and 4000 legal URLs. As a result, 4000 samples were chosen at random from the aforementioned dataframe.

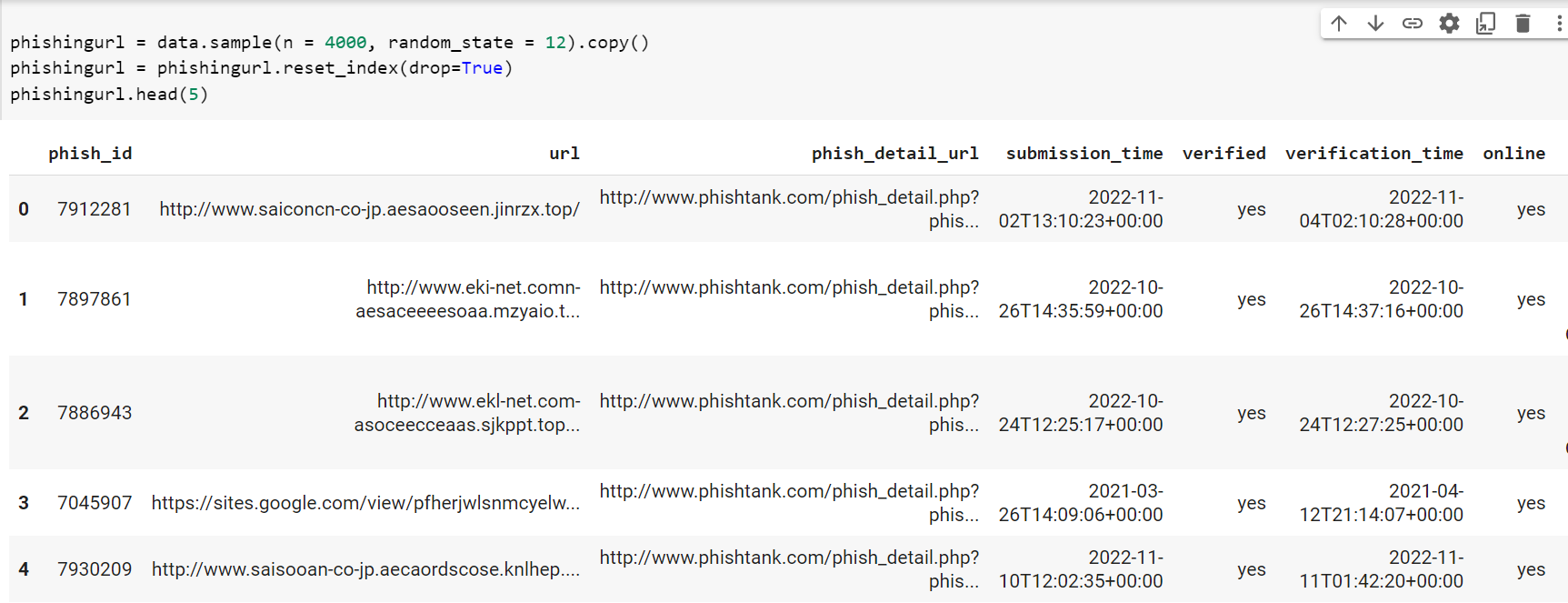


Figure 18 Sample Collection

### 4.1.2 Authentic/Legitimate URLs

Once we load the legi.CSV and we got to know the legitimated URLs list.



Figure 19 legitimate URLs

As previously indicated, 4000 genuine URLs are arbitrarily chosen from the given dataframe.

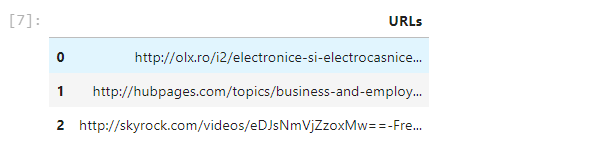


Figure 20 Random Legitimate URLs from 4000 Samples

## 4.2 Feature Extraction

Next step is to extracted the features from the URLs.

The extracted features are categorized into

1. Address 9 Bar based Features(ABBF)
2. HTML & 8 Javascript based Features(H&JBF)
3. Domain based Features(DBF)

### 4.2.1 Address Bar Based Features

All of those are features obtained from the URL themselves, for example, whether it has an IP address, uses a URL shortening service like URL shortener or Bitly, or employs redirection. Additional characteristics might include:

* Ip
* Domain
* "@" in URL
* Existence of HTTPS/HTTP
* length & Depth of URL
* Prefix or Suffix "-" in Domain
* URL shortener

### 4.2.2 HTML & Javascript based Features

In this category has several characteristics that may be extracted. The following were chosen from among them for this purpose.

* Phishing scams block the well‐characterized with JavaScript, preventing visitors from seeing and saving the website code. This functionality is precisely the same as "Using onMouseOver to conceal the Links." However, for this functionality, will examine the page source code for the occurrence "event.button==2" to see if the right click is disabled.

If the reply is null then there is no on mouseover, the support added to this feature is 1 for phishing and 0 for legitimate.

* IFrame is an HTML feature that displays another site within the one that is now displayed. Phishing scams can employ the "iframe" tag to make the iframe hidden, namely the lacking frame boundaries. Phishing scams use the "frame border" property to induce the browsers to generate a graphical distinction in this regard.

If the iframe is vacant or even no response is detected, the support added to this feature is 1 for phishing and 0 for legitimate.

* Phishing scams might need JavaScript to display a bogus URL inside the status bar. To retrieve this functionality, must examine the page code base, specifically the "onMouseOver" trigger, to see whether it modifies the status bar. Whereas if answer is null or there is no mouse over, the support added to this feature is 1 for phishing and 0 for legitimate.

### 4.2.3 Domain based Features

This area has several characteristics that may be extracted, which are mentioned below.

* **Age and End period of Domains**

Such feature is available through the WHOIS repository. Some phishing sites only exist for a short space of time.  In this, the minimum age of the lawful domains is deemed to be twelve months. Age in this context refers to the period between birth or expiry.  Whereas if domain is older than twelve months, the support added to this feature is 1 for phishing and 0 for legitimate. Same for end period, the remaining domain time is determined for this feature by subtracting the expiry time from the current time. The end term evaluated for the genuine domain is 6 months/ fewer. Whereas if domain is older than six months, the support added to this feature is 1 for phishing and 0 for legitimate.

* **DNS records**

Inside the scenario of phishing sites, whichever the stated identity is not recognized by the WHOIS repository whereas no records for the hostname are found. If the DNS record is missing/null, the support added to this feature is 1 for phishing and 0 for legitimate.

* **Web traffics**

A capability assesses the appeal of a website by counting the amount of visitors or the pages they view. If the domain rank is 1000000, the support added to this feature is 1 for phishing and 0 for legitimate. The extracted features are shown in the Figure 21.

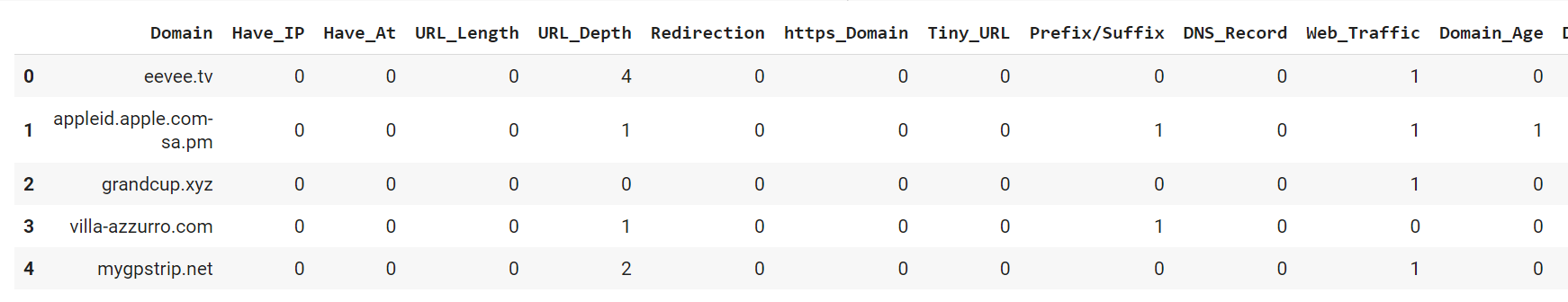


Figure 21 Feature Extraction on Phishing URLs

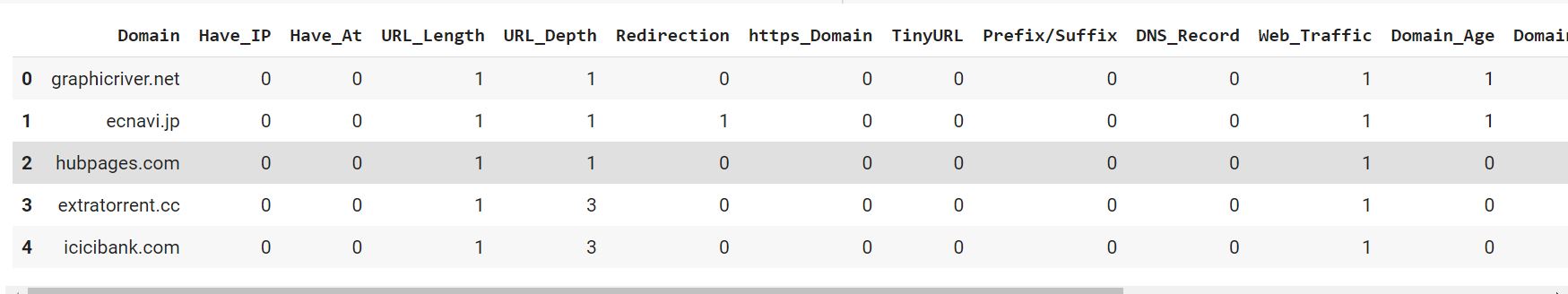


Figure 22 Feature extraction on legitimate URLs

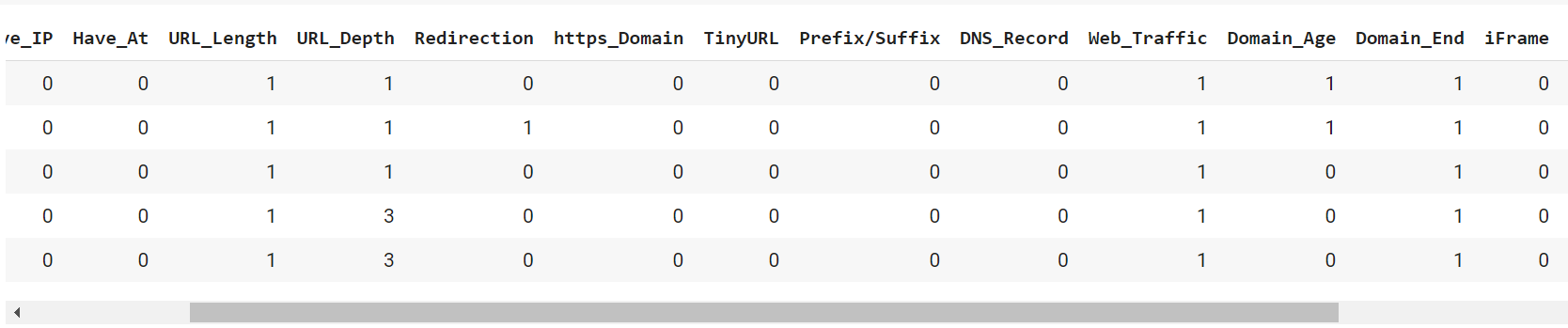
Now, integrate them into a single data frame.

Figure 23 New dataset After Marge

## 4.3 Implementation of Machine learning Techniques:

First I import all libraries and the data we collect from legitimate phishing urls. After loading analyzing the information of our collective dataset.

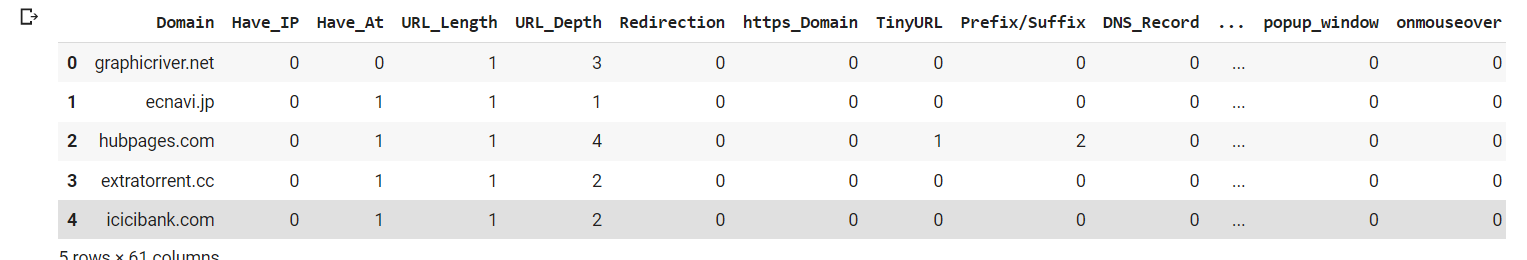


Figure 24 Data Information

### 4.3.1 Data visualization:

In this visualized the collected dataset and checked which column was irrelevant to our research. In this we use bar graph.

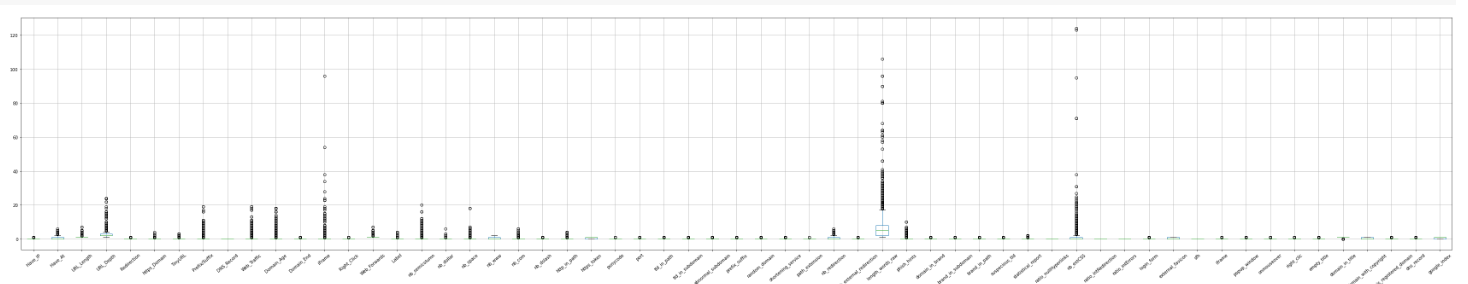


Figure 25 Bar Plot for Data Visualization

Also use histogram method for visualization

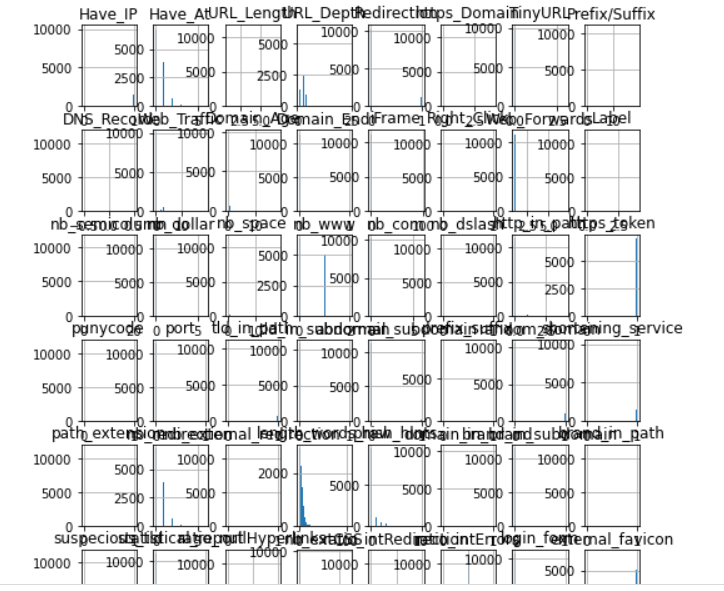


Figure 26 Histogram Visualization

Now we create a confusion matrix for our dataset.

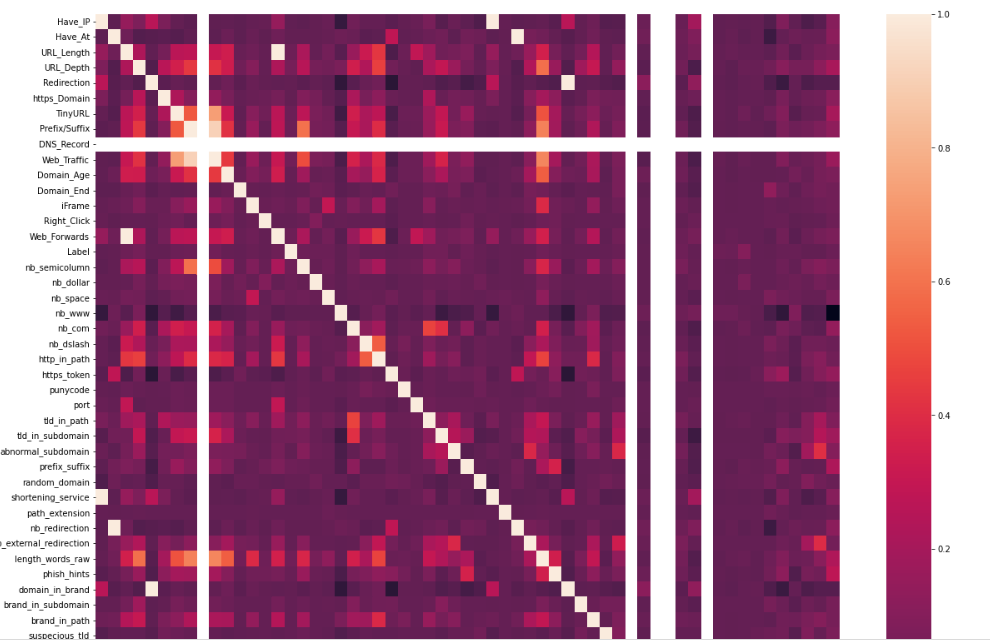


Figure 27 Confusion Matrix

### 4.2.2 Data preprocessing:

Now clean the data using data preprocessing procedures and convert it for usage in the models. Except for the 'Domain' columns, the above result demonstrates that most data is 0/1. As a result, the 'Domain' column has been removed from the dataset. This gives us 16 features & a target column.

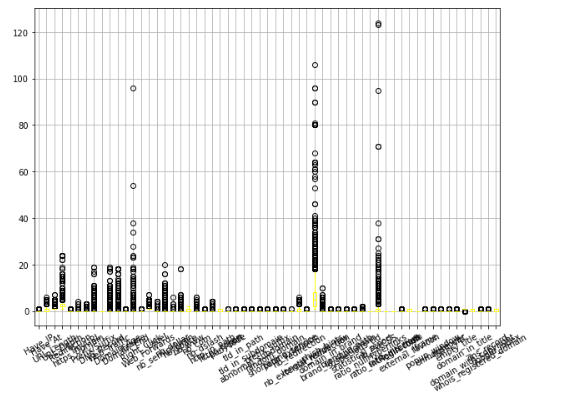


Figure 28 Data After Dropping Column

The graph above demonstrates that the data has no missing data. This extensively preprocesses & prepares the data for training.

### 4.2.3 Train/test:

Splitting the dataset into train/test sets into 80:20 split.



Figure 29 Test and Train

### 4.2.4 Evaluation

I evaluate all machine learning model by using accuracy, precision, F1score Recall and ROC and AUCscore formula.

Table 4.1 All models Results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **accuracy** | **Precision** | **F1-score** | **Recall** | **Rc ROC**  **Auc score** |
| Decision Tree | 90.2% | 87.7% | 90% | 92.4% | 90.2% |
| Ridge Classifier | 88.8% | 87.7% | 88.7% | 89% | 88.8% |
| XGB | 93.7% | 94.2% | 93.7% | 93.2% | 93.6% |
| MLP | 93.3% | 93.3% | 93.4% | 93.5% | 93.3% |
| Random forest | 90.9% | 90.1% | 90.8% | 91.6% | 90.9% |
| SVM | 90.68% | 90.0% | 90.6% | 91.3% | 90.6% |
| Neural Network | 96.3% | 96.2% | 96.4% | 96.8% | 96.3% |

# 

# Chapter 5: Results and Discussion

## 5.1 Results of ML:

In this section, I describe the results by evaluating the accuracy rate of the different models mentioned in the comparative study.

### 5.1.1 Ridge Classifier:

Using the Scikit-learn library’s, the train data was used to implement a ridge classifier model. The model performance accuracy against the testing set was 79% the confusion matrix shown in Figure 30.

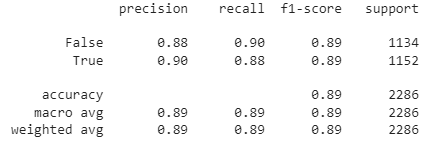


Figure 30 Ridge Classifier

### 5.1.2 XGBoost:

Using the Scikit-learn library’s, the training set was used to build a XGboost classification model. The model performance accuracy against the testing set was 85% the confusion matrix shown in Figure 31 and 32.

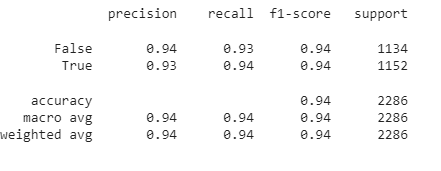


Figure 31 XGBoost

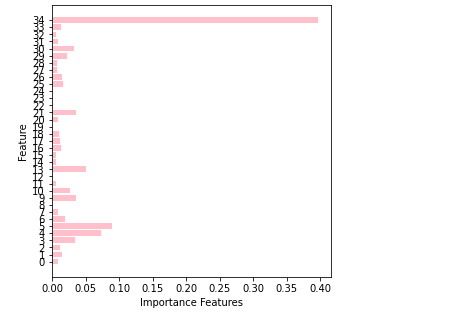


Figure 32 Important Features of XGboost

### 5.1.3 Decision Tree:

Using the Scikit-learn library’s, the training set was used to build a decision tree model. The model performance accuracy against the testing set was 81% the confusion matrix shown in Figure 33 and 34.

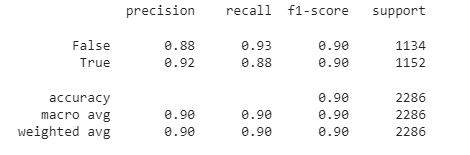


Figure 33 Decision tree

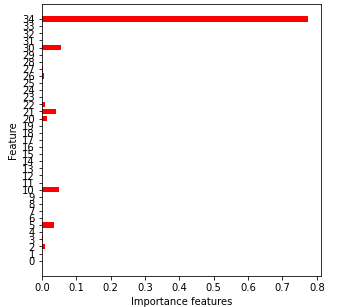


Figure 34 Importance Features

### 5.1.4 MLP

Using the Scikit-learn library’s, the training set was used to build a MLP classification model. The model performance accuracy against the testing set was 85.2% the confusion matrix shown in Figure 35.

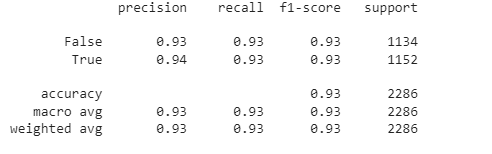


Figure 35 MLP

### 5.1.5 Random Forest

Using the Scikit-learn library’s, the training set was used to build a MLP classification model. The model performance accuracy against the testing set was 81% the confusion matrix shown in Figure 36 and 37.

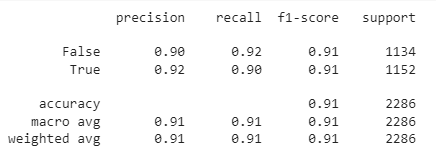


Figure 36 Random Forest

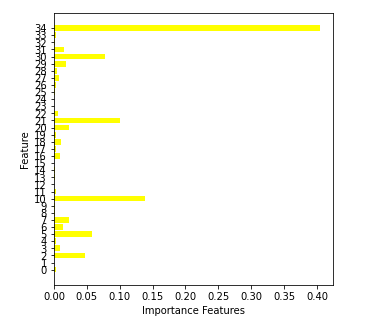


Figure 37 Importance Features of RF

### 5.1.6 SVM

The training dataset was employed to develop the Svms using the Scikit-learn toolkit. The model's efficiency versus the testing dataset was shown below.

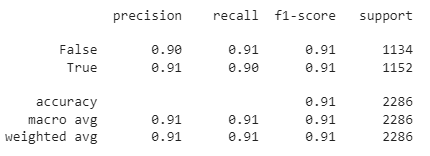


Figure 38 SVM

### 5.1.7 Artificial neural network ANN

I assigned 35 the input shape; In this case, we used Keras sequential() to build our model, adding three layers: input, hidden, or outer. Use 512 neurons with relu as the activation function for the hidden and input layers, plus sigmoid for the final layer.

Now construct the model using adam optimizer & pick loss as binary cross-entropy whilst binary accuracy as a metric.

I choose batch size 12, epochs 200, and callback early halting to match the model. For visualizing binary accuracy, use matplotlib.

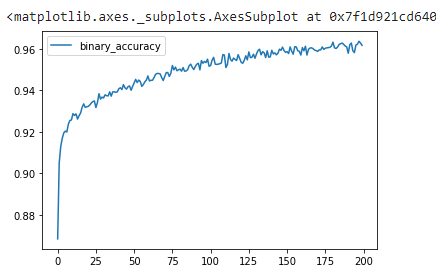
****

Figure 39 binary accuracy of ANN

# 5.2 Result comparison

By comparing all accuracy results, got ANN as high accuracy from above all ML models. The results are shown in Figure 40.

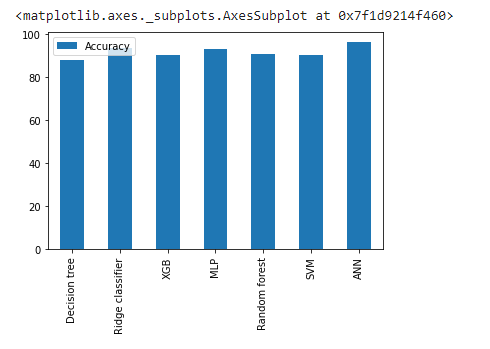


Figure 40 combine output

### 5.2. 1 Compare the generated results with state-of-the-art similar work:

Inside this section, contrast my findings with past studies.

Table 5.1 Compare results with state of the art

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No | **Papers** | **Methods** | **Dataset** | **Accuracy** |
| 1 | (Das Guptta, 2022), | RF | Opensource  dataset | 94% |
| 2 | Our proposed method | ANN | Collect dataset  Phish tank & Urls dataset | 96.3% |
| 3 | (Kim, 2022) | Random forest | Open source dataset | 89% |

On this dataset, my technique consistently outperforms state-of-the-art detection algorithms in various practical classes. In our dataset, we used 61 features for train and testing. Our research indicated that training machine learning algorithms with large amounts of data enhance their execution. If we use large data in future our accuracy result may improve. Injecting fake data into the training set improved the performance of the classification algorithm. Furthermore, learning methods with some new datasets were far more robust to experiment assaults.

## 5.3 Discussion:

I assess the efficacy of machine learning systems in detecting Phishing URLs in real time. ANN is an ideal place to start when developing data-driven prediction models. I detected with 85.5% accuracy. The output of the implementation is depicted in the image below 38.

The binary classification judgement is centered on whether the material is phishing or real throughout this examination. A structure known as a confusion matrix can aid in determining the classifiers' conclusions.

The Precision-Recall Curves, AUC-ROC Curve,  (MAE, or Accuracy are the evaluation metrics used in this evaluation. Now plot the Roc to analyse the results.

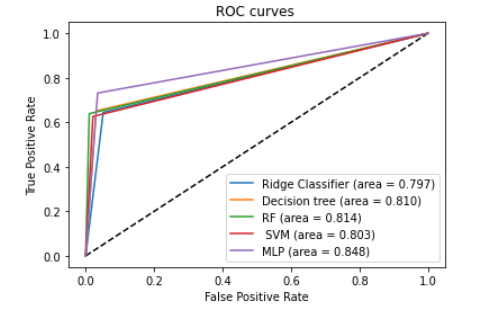


Figure 41 Roc curved

In Figures 41 and 42 show the precision-recall and ROC curves, respectively of other Machine learning algorithms.

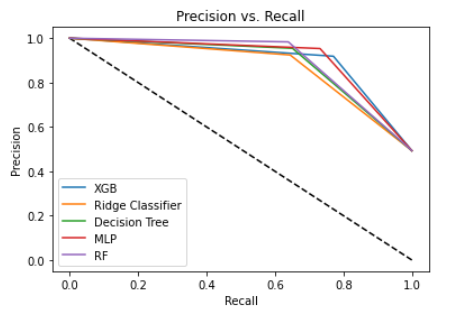


Figure 42 Precision vs Recall

# Chapter 6 Conclusion & Future Work

Because of the increasing usage of the Internet in our daily lives, cybercriminals target their victims via online platform. One of the most common attacks is "phishing URLS," which involves the creation of a fake URLs to gain the users' sensitive information, such as user and password, on financial websites using social networking facilities. The malicious web URLS is constructed similarly to a simple legitimate URLs. As a result of its semantic structure, which exploits people's weaknesses, identification of these URLs is a fairly easy difficulty to solve.

The software can only be utilized as a support mechanism for this sort of detection and prevention of phishing. These technologies, in particular, adopt an allowlist/blocklist technique to counter this attack. However, because they are Dynamic algorithms, they can detect new types of assaults in the networks System. As a result, we advocate the usage of Machine Learning in real-time as an efficient approach.

An ML-based system is used to categories incoming Phishing URLs. The outcomes of the experiments demonstrate that all ANN Artificial neural network algorithms yield High accuracy rates as compared to other support vector machine SVM, random forestRF, Decision Tree, Ridge, MLP and XGBoost. The Artificial neural network ANN produces the optimum answer with around 96.3%.

In the future, techniques for feature selection will be able to reduce data dimensionality while improving classification efficiency. Various textual aspects may also be analyzed to increase the efficacy of the detection procedure.

# References

Abbas, Q. M. U. Z. a. M. A., 2022. A CNN-RNN Based Fake News Detection Model Using Deep Learning.. *In 2022 International Seminar on Computer Science and Engineering Technology (SCSET).*

admin, 2019. [Online]   
Available at: https://machinelearningmastery.com/logistic-regression-for-machine-learning/

admin, 2020. [Online]   
Available at: https://www.modernhealthcare.com/safety-quality/coronavirus-outbreak-live-updates-covid-19

admin, 2020. [Online]   
Available at: https://www.tensorflow.org/resources/learn-ml?gclid=CjwKCAjw14uVBhBEEiwAaufYx6s9PiSWabrobrggALUr2CRPqaq2P3ymG30E3s04oPeuMLEIUXLZCBoC5TUQAvD\_BwE

admin, 2020. [Online]   
Available at: https://phishtank.org/developer\_info.php

admin, 2021. [Online]   
Available at: https://www.turing.com/kb/random-forest-algorithm

admin, 2021. *Evaluating Information: Fake news in the 2016 US Elections.* [Online]   
Available at: https://libraryguides.vu.edu.au/evaluating\_information\_guide/fakenews2016

admin, 2022. [Online]   
Available at: https://www.statista.com/topics/1164/social-networks/#topicHeader\_\_wrapper

Ahmad, I. M. Y. S. Y. a. M. O. A., 2020. Fake news detection using machine learning ensemble methods.. *Complexity 2020.*

AlEroud, A. a. G. K., 2020. Bypassing detection of URL-based phishing attacks using generative adversarial deep neural networks. *In Proceedings of the Sixth International Workshop on Security and Privacy Analytics, pp. 53-60..*

Alloghani, M. D. A.-J. J. M. A. H. a. A. J. A., 2020. A systematic review on supervised and unsupervised machine learning algorithms for data science.. *Supervised and unsupervised learning for data science.*

Anon., 2019. *fake and real news.* [Online]   
Available at: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

Anon., n.d. [Online]   
Available at: https://www.kaggle.com/datasets?search=fake+news

Anon., n.d. [Online]   
Available at: https://scikit-learn.org/stable/modules/svm.html#:~:text=Support%20vector%20machines%20(SVMs)%20are,than%20the%20number%20of%20samples.

Anon., n.d. [Online]   
Available at: https://medium.datadriveninvestor.com/expand-your-dataset-without-machine-learning-d82d8174ac11

Anon., n.d. [Online]   
Available at: https://www.python.org/

Aung, E. S. a. H. Y., 2019. Phishing URL Detection using Information-rich Domain and Path Features.

Ayodele, T. O., 2010. Types of machine learning algorithms.. *New advances in machine learning 3.*

Bali, A. P. S. M. F. S. C. a. M. G., 2019. Comparative performance of machine learning algorithms for fake news detection.. *In International conference on advances in computing and data sciences.*

Barla, N., 2022. Understanding Representation Learning With Autoencoder.

Basit, A. M. Z. X. L. A. R. J. Z. J. a. K. K., 2021. A comprehensive survey of AI-enabled phishing attacks detection techniques. *Telecommunication Systems 76, no. 1.*

Baykara, M. a. Z. Z. G. ". o. p. a., n.d. In 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-5. IEEE. *2018.*

Bhandari, A., 2020. [Online]   
Available at: https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/

Blake, A., 2018. A new study suggests fake news might have won Donald Trump the 2016 election. Issue https://www.washingtonpost.com/news/the-fix/wp/2018/04/03/a-new-study-suggests-fake-news-might-have-won-donald-trump-the-2016-election/.

Brownlee, J., 2019. [Online].

Brownlee, J., 2020. [Online]   
Available at: https://machinelearningmastery.com/types-of-classification-in-machine-learning/

Brownlee, J., 2020. Autoencoder Feature Extraction for Classification.

Bustillo, A. R. R. A. R. M. a. D. Y. P., 2020. Improving the accuracy of machine-learning models with data from machine test repetitions. *Journal of Intelligent Manufacturing.*

Buzea, M. C. S. T.-M. a. T. R., 2022. Automatic Fake News Detection for Romanian Online News. *Information 13.*

Bzdok, D. M. K. a. N. A., 2018. Machine learning: supervised methods.. *Nature methods 15.1.*

Castaño, F. F. E. A. E. C. D. &. S.-P. M. (. S. o. t. A. C.-b. a. H. P. D. a. p. a., 2021. State of the Art: Content-based and Hybrid Phishing Detection. *rXiv preprint arXiv:2101.12723.*

Chauhan, V. K. K. D. a. A. S., 2019. Problem formulations and solvers in linear SVM: a review.. *Artificial Intelligence Review 52, no. 2 .*

Cheng, Y. T. C. Z. Z. K. L. a. Y. D., 2022. Detecting malicious domain names with abnormal whois records using feature-based rules.. *The Computer Journal 65, no. 9.*

Chen, S. G. I. W. L. L. a. X. M., 2020. A novel selective naïve Bayes algorithm.. *Knowledge-Based Systems 192.*

Chiew, K. L. C. L. T. K. W. K. S. Y. a. W. K. T., 2019. A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Information Sciences 484.*

da Silva, C. M. R. B. J. T. F. E. L. F. a. V. C. G., 2022. Piracema. io: A rules-based tree model for phishing prediction.. *Expert Systems with Applications 191 .*

Das Guptta, S. K. T. S. H. A. D. A. a. I. H. S., 2022. Modeling Hybrid Feature-Based Phishing Websites Detection Using Machine Learning Techniques. *Annals of Data Science.*

dataset, 2021. [Online]   
Available at: https://www.unb.ca/cic/datasets/url-2016.html

Dattaa, S. S. S. a. P. K., 2022. A Trustworthy Swift Weapon to Detect the Phishing URLs by Machine Learning Approaches.

edpresso, 2020. [Online]   
Available at: https://www.educative.io/edpresso/data-normalization-in-python

Elhadad, M. K. K. F. L. a. F. G., 2019. Fake news detection on social media: a systematic survey.. *IEEE Pacific Rim conference on communications, computers and signal processing (PACRIM). IEEE, 2019.*

Fayaz, M. A. K. M. B. a. S. U. K., 2022. Machine learning for fake news classification with optimal feature selection.. *Soft Computing (2022).*

Gandotra E., G. D., 2021. An Efficient Approach for Phishing Detection using Machine Learning. *Algorithms for Intelligent Systems, Springer, Singapore.*

Gilda, S., 2017. Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection.. *In 2017 IEEE 15th student conference on research and development (SCOReD).*

Gupta, B. B. K. Y. I. R. K. P. A. C. a. X. C., 2021. A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment.. *Computer Communications 175 .*

Hassan, M. M., 2021. SQL Injection Vulnerability Detection Using Deep Learning: A Feature-based Approach. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI).*

Hassan, M. M. R. B. A. a. T. G., 2021. SQL Injection Vulnerability Detection Using Deep Learning: A Feature-based Approach.. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI) 9, no. 3 .*

Hodge, V. J., 2018. A Survey of Outlier Detection Methodologies. *university of new york.*

ibm, 2021. [Online]   
Available at: https://www.ibm.com/cloud/learn/unsupervised-learning

J. Kumar, A. S. B. J. B. R. a. B. S. B., 2020. Phishing Website Classification and Detection Using Machine Learning,. *2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020,.*

J. Scott Brennen, 1. F. M. S. a. R. K. N., 2021. Beyond (Mis)Representation: Visuals in COVID-19 Misinformation. *premed,* Issue https://journals.sagepub.com/doi/10.1177/1940161220964780.

Jain, A. A. S. H. K. a. A. K. G., 2019. A smart system for fake news detection using machine learning.. *In 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), vol..*

Jingguo Wang, T. H. R. C. A. V. a. H. R. R., 2012. Research article: Phishing susceptibility: An investigation into the processing of a targeted spear phishing email.. *IEEE Trans. on Professional Communication, 55(4):345–362.*

John Yearwood, M. M. a. A. B., 2010. Profiling phishing emails based. *In Proc. of IEEE/ACM Advances in Social Network Analysis and.*

Kadhim, A. I., 2019. Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review 52, no. 1.*

Kaliyar, R. K. A. G. a. P. N., 2019. Multiclass fake news detection using ensemble machine learning.. *2019 IEEE 9th International Conference on Advanced Computing (IACC). IEEE.*

Khanam, Z. B. N. A. H. S. a. M. R., 2021. Fake news detection using machine learning approaches.. *IOP Conference Series: Materials Science and Engineering, vol. 1099, no. 1, p. 012040. IOP Publishing.*

Khonji, M. I. Y. &. J. A., 2013. Phishing detection: a literature survey. *IEEE Communications Surveys & Tutorials.*

Kovač, A. D. I. a. S. S., 2022. An overview of machine learning algorithms for detecting phishing attacks on electronic messaging services.. *In 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO) (pp. 954-961). IEEE..*

Krishnamurthi, R. G. a. I., 2014. A comprehensive and efficacious architecture for detecting phishing webpages.. *Computers and Security, 40:23–37,.*

Kumar, A. K. P. S. S. K. a. L. V., 2022. Image Classification in Python Using Keras.. *In Proceedings of Data Analytics and Management, pp. 541-556. Springer, Singapore, .*

Lahby, M. S. A. W. Y. a. Y. A., 2022. Online Fake News Detection Using Machine Learning Techniques: A Systematic Mapping Study.. *Combating Fake News with Computational Intelligence Techniques .*

Li, H. H. J. D. W. a. B. H., 2018. An improved KNN algorithm for text classification.. *In 2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC).*

Malhotra, P. a. S. K. M., 2022. Fake news detection using supervised machine learning techniques. *Journal of Information and Optimization Sciences 43.1.*

Manzoor, S. I. a. J. S., 2019. Fake news detection using machine learning approaches: A systematic review.". *2019 3rd international conference on trends in electronics and informatics (ICOEI)..*

Mithra Raj, M. a. J. A. A. J., 2022. Website Phishing Detection Using Machine Learning Classification Algorithms. *In International Conference on Applied Informatics, pp. 219-233. Springer, Cham.*

Mohamed, G. J. V. M. M. J. A. a. M. E., 2022. An Effective and Secure Mechanism for Phishing Attacks Using a Machine Learning Approach.. *Processes 2022, 10, 1356..*

Mohammed Al-Janabi, E. d. Q. a. P. A., 2017. Using supervised machine learning algorithms to detect suspicious urls in online social networks.. *In IEEE/ACM Advances in Social Network Analysis and Mining (ASONAM), pages 1104–1111.*

News, T. P. o. F., 2021. GordonPennycook. *Trends in Cognitive Sciences.*

Nicholson, S., 2008. Misunderstanding and Misinformation When Interpreting The Written Text. Issue https://www.researchgate.net/publication/236946508\_Misunderstanding\_and\_Misinformation\_When\_Interpreting\_The\_Written\_Text.

Odeh, A. I. K. a. E. A., 2021. Machine learningtechniquesfor detection of website phishing: a review for promises and challenges.. *In 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0813-0818. IEEE, .*

Ofcom, 2020. *Half of UK Adults Exposed to False Claims about Coronavirus..* [Online]   
Available at: https://www.ofcom.org.uk/about-ofcom/latest/features-and-news/half-of-uk-adults-exposed-to-false-claims-about-coronavirus

Okken, B., 2022. Python Testing with pytest. Pragmatic Bookshelf.

O'Mara, A. A. I. &. A. A. (. S., 2021. Generative Adverserial Analysis of Phishing Attacks on Static and Dynamic Content of Webpages.. *In 2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom) (pp. 1657-1662). IEEE..*

O'Reilly, C. A. G. M. A. I. a. S. R., 2014. Anomaly detection in wireless sensor networks in a non-stationary environment.. *EEE Communications Surveys & Tutorials 16, no. 3.*

Pais., R. S. R. a. A. R., 2018. Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Computing and Applications,.*

Patel, H. H. a. P. P., 2018. Study and analysis of decision tree based classification algorithms. *International Journal of Computer Sciences and Engineering 6, no. 10.*

Pennycook, G. C. T. a. R. D. G., 2018. Prior exposure increases perceived accuracy of fake news.. *J. Exp. Psychol.*

Pineda, M. E., 2020. [Online]   
Available at: https://www.profolus.com/topics/benefits-limitations-of-machine-learning/

plivo, 2019. [Online]   
Available at: https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-to-use-it-e3c6186d008f

Praveena, K. N. a. R. M., 2022. A Survey on Early Prediction of Autism Spectrum Disorder Using Supervised Machine Learning Methods.. *Technology Enabled Ergonomic Design. Springer, Singapore.*

Purbay M., K. D., 2021. Split Behavior of Supervised Machine Learning Algorithms for Phishing URL Detection. *Lecture Notes in Electrical Engineering, vol. 683,.*

Rao, R. S., 2020. Study on Website Phishing and their Countermeasures..

Ray, S., 2019. A quick review of machine learning algorithms. *In 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon), pp. 35-39. IEEE, .*

Reis, J. C. A. C. F. M. A. V. a. F. B., 2019. "Explainable machine learning for fake news detection.. *In Proceedings of the 10th ACM conference on web science, pp. 17-26..*

Reis, J. C. A. C. F. M. A. V. a. F. B., 2019. Explainable machine learning for fake news detection.. *In Proceedings of the 10th ACM conference on web science, pp.*

Rekouche, K., n.d. *early phishing.* [Online]   
Available at: https://arxiv.org/ftp/arxiv/papers/1106/1106.4692.pdf

Romero, C. a. S. V., 2020. Educational data mining and learning analytics: An updated survey.. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 10.3 .*

R, S. E., 2021. [Online]   
Available at: https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/#:~:text=Random%20forest%20is%20a%20Supervised,average%20in%20case%20of%20regression.

Sahoo, K. S. B. K. T. K. N. S. R. B. B. M. K. a. D. B., 2008. An evolutionary SVM model for DDOS attack detection in software defined networks.. *IEEE Access 8.*

Salahdine, F. Z. E. M. a. N. K., 2021. Phishing Attacks Detection A Machine Learning-Based Approach. *In 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), pp. 0250-0255. IEEE.*

Samuel Marchal, K. S. N. S. a. N. A., 2016. Know your phish: Novel techniques for detecting phishing sites and their targets. *In Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on, pages 323–333. IEEE,.*

SCHAEFFER, K., 2020. *Nearly three-in-ten Americans believe COVID-19 was made in a lab.* [Online]   
Available at: https://www.pewresearch.org/fact-tank/2020/04/08/nearly-three-in-ten-americans-believe-covid-19-was-made-in-a-lab/

Seals, T., 2021. *Geek Squad Vishing Attack Bypasses Email Security to Hit 25K Mailboxes.* [Online]   
Available at: https://threatpost.com/geek-squad-vishing-bypasses-email-security/167014/

Seif, G., 2021. [Online]   
Available at: https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

seldon, 2021. [Online]   
Available at: https://www.seldon.io/machine-learning-regression-explained#:~:text=Regression%20is%20a%20technique%20for,used%20to%20predict%20continuous%20outcomes.

Shirazi, H. B. B. I. R. a. C. A., 2019. Adversarial sampling attacks against phishing detection.. *In IFIP Annual Conference on Data and Applications Security and Privacy, pp. 83-101. Springer,.*

Singh, V. R. D. D. S. K. R. a. I. G., 2017. Automated fake news detection using linguistic analysis and machine learning.. *In International conference on social computing, behavioral-cultural modeling, & prediction and behavior representation in modeling and simulation (SBP-BRiMS), pp. 1-3..*

Tarbani, N., 2021. [Online]   
Available at: https://www.analyticsvidhya.com/blog/2021/04/how-the-gradient-boosting-algorithm-works/

Tertytchny, G. N. N. a. M. K. M., 2020. Classifying network abnormalities into faults and attacks in IoT-based cyber physical systems using machine learning. *Microprocessors and Microsystems 77.*

Thabtah, F. a. N. A., 2017. Deriving correlated sets of website features for phishing detection: a computational intelligence approach.. *Journal of Information & Knowledge Management 15, no. 04 .*

Tupsamudre, H. A. K. S. a. S. L., 2019. Everything is in the name–a url based approach for phishing detection. *In International symposium on cyber security cryptography and machine learning, pp. 231-248. Springer, Cham.*

Vito, L. d., 2017. LinXGBoost: Extension of XGBoost to Generalized Local Linear Models. *arXiv:1710.03634.*

Wang, W. Y., 2017. liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648.*

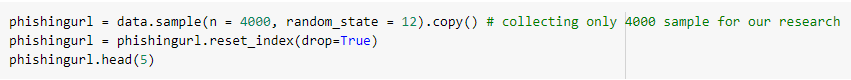
Yang, R. K. Z. B. W. C. W. a. X. W., 2022. Prediction of Phishing Susceptibility Based on a Combination of Static and Dynamic Features.. *Mathematical Problems in Engineering 2022.*

# Appendix

## Get phishing data







## Legitimate URLs:

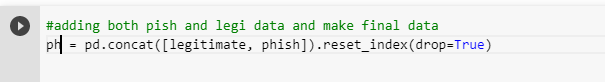


## Feature Extractions



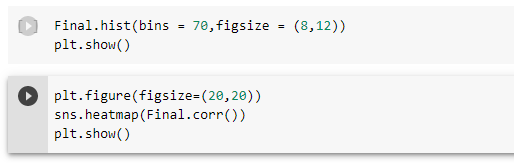


# Final version of CSV

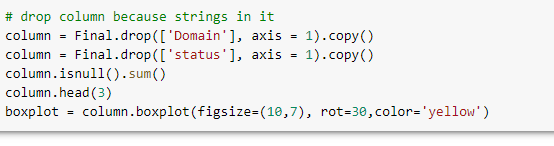


# Machine learning

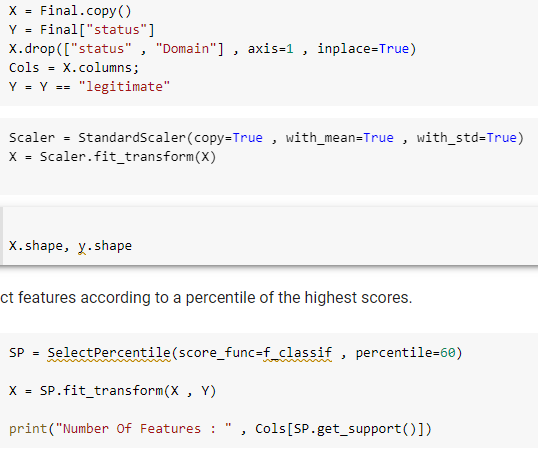




# Drop Domain & Status columns



# Data pre processing

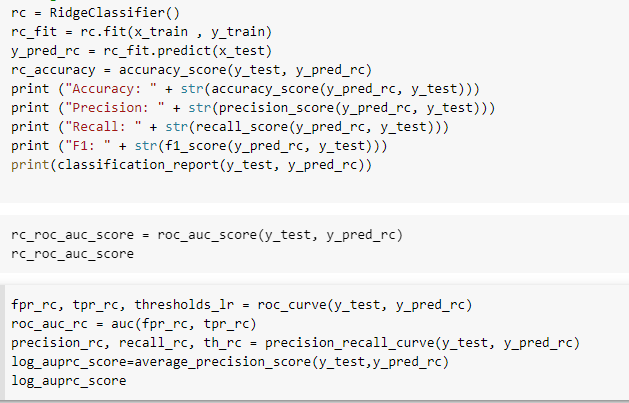


# Divide the data into test and train

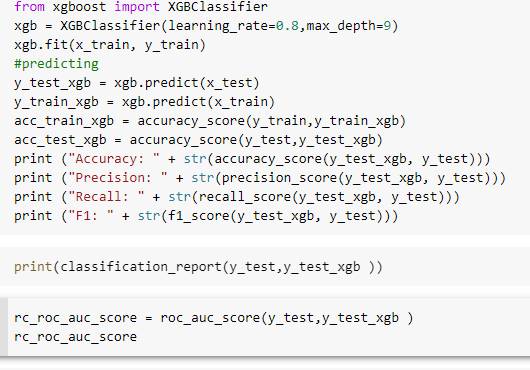


# ML implementation

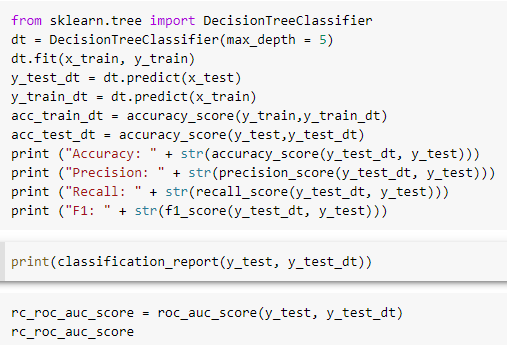
## ****Ridget Classifier****



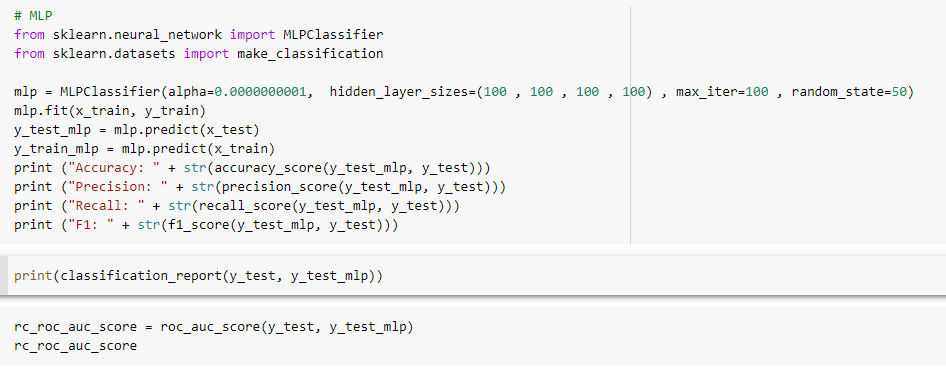
## XGBoost



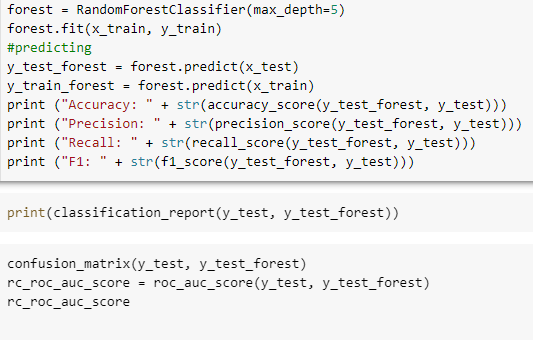
## Decision tree



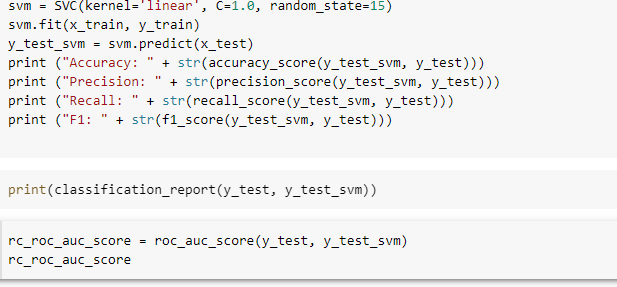
## MLP



## Random forest

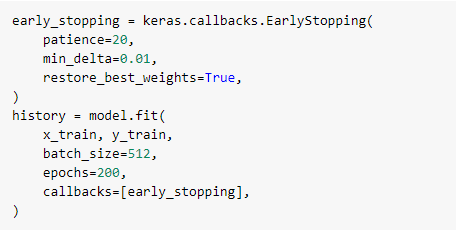


## SVM



## ANN







# Compare ML algorithms

