C-Sentinel
Your Cyber Guardian

# Phishing Website Detection using Machine Learning Algorithms -Supervised Learning-

Haneen Alhomoud HaneenAlhomoud@gmail.com

## Introduction

Phishing attack is a simplest way to obtain sensitive information from innocent users. Aim of the phishers is to acquire critical information like username, password and bank account details.

This project will use machine learning algorithms to detect phishing URLs by extracting and analyzing various features of legitimate and phishing URLs. Using different classification algorithms such as Logistic Regression, Tree, KNNeighbors, Random Forest, Support vector machine (SVM) and NaiveBayes. Aim of the project is to detect phishing URLs as well as narrow down to best machine learning algorithm by comparing accuracy rate, precision and recall rate of each algorithm.

## Background of Problem Statement :

This code is used for a binary classification model (phishing website or not) using Python Scikit-Learn that trains on the data and calculates the accuracy, precision, recall and f2. More than one classification algorithms is used to train a model on the phishing and legitimate website data set.

## Data

The data set is provided both in the following URL https://www.kaggle.com/dnyaneshsatpute/phishing-webiste-detection/data in csv file:

- A collection of website URLs for 11000+ websites (data point)
- 30 columns to identify a phishing website or not (0 or 1):

## Dataset Description:

'UsingIP', 'LongURL', 'ShortURL', 'Symbol@', 'Redirecting//', 'PrefixSuffix-', 'SubDomains', 'HTTPS', 'DomainRegLen', 'Favicon', 'NonStdPort', 'HTTPSDomainURL', 'RequestURL', 'AnchorURL', 'LinksInScriptTags', 'ServerFormHandler', 'InfoEmail', 'AbnormalURL', 'WebsiteForwarding', 'StatusBarCust', 'DisableRightClick', 'UsingPopupWindow', 'IframeRedirection', 'AgeofDomain', 'DNSRecording', 'WebsiteTraffic', 'PageRank', 'GoogleIndex', 'LinksPointingToPage', 'StatsReport', 'result'

Brief Description of the features in data set

- UsingIP : { -1,1 }
- LongURL : { 1,0,-1 }
- ShortURL : { 1,-1 }
- Symbol@ : { 1,-1 }
- Redirecting// : { -1,1 }
- PrefixSuffix- { -1,1 }
- SubDomains : { -1,0,1 }
- HTTPS : { -1,1,0 }
- DomainRegLen : { -1,1 }
- Favicon : { 1,-1 }
- NonStdPort : { 1,-1 }
- HTTPSDomainURL : { -1,1 }
- RequestURL : { 1,-1 }
- AnchorURL : { -1,0,1 }
- Result : { 0,1 }

### Table: Detailed Description of Data Columns

| | |
|---|---|
| **UsingIP** | If an IP address is used as an alternative of the domain name in the URL, such as "http://125.98.3.123/fake.html", users can be sure that someone is trying to steal their personal information. Sometimes, the IP address is even transformed into hexadecimal code as shown in the following link "http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html" |
| **LongURL** | Phishers can use long URL to hide the doubtful part in the address bar. For example: http://federmacedoadv.com.br/3f/aze/ab51e2e319e51502f416dbe46b773a5e/?cmd=_home&dispatch=11004d58f5b74f8dc1e7c2e8dd4105e811004d58f5b74f8dc1e7c2e8dd4105e8@phishing.website.html To ensure accuracy of our study, we calculated the length of URLs in the dataset and produced an average URL length. The results showed that if the length of the URL is greater than or equal 54 characters then the URL classified as phishing. By reviewing our dataset we were able to find 1220 URLs lengths equals to 54 or more which constitute 48.8% of the total dataset size. Rule: IF We have been able to update this feature rule by using a method based on frequency and thus improving upon its accuracy. |
| **ShortURL** | URL shortening is a method on the "World Wide Web" in which a URL may be made considerably smaller in length and still lead to the required webpage. This is accomplished by means of an "HTTP Redirect" on a domain name that is short, which links to the webpage that has a long URL. For example, the URL "http://portal.hud.ac.uk/" can be shortened to "bit.ly/19DXSk4". |
| **Symbol@** | Using "@" symbol in the URL leads the browser to ignore everything preceding the "@" symbol and the real address often follows the "@" symbol. |
| **Redirecting//** | The existence of "//" within the URL path means that the user will be redirected to another website. An example of such URL's is: "http://www.legitimate.com//http://www.phishing.com". We examin the location where the "//" appears. We find that if the URL starts with "HTTP", that means the "//" |

| | |
|---|---|
| | should appear in the sixth position. However, if the URL employs “HTTPS” then the “//” should appear in seventh position. |
| **PrefixSuffix-** | The dash symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name so that users feel that they are dealing with a legitimate webpage. For example http://www.Confirme-paypal.com/. |
| **SubDomains** | Let us assume we have the following link: http://www.hud.ac.uk/students/. A domain name might include the country-code top-level domains (ccTLD), which in our example is “uk”. The “ac” part is shorthand for “academic”, the combined “ac.uk” is called a second-level domain (SLD) and “hud” is the actual name of the domain. To produce a rule for extracting this feature, we firstly have to omit the (www.) from the URL which is in fact a sub domain in itself. Then, we have to remove the (ccTLD |
| **HTTPS** | The existence of HTTPS is very important in giving the impression of website legitimacy, but this is clearly not enough. The authors in (Mohammad, Thabtah and McCluskey 2012) (Mohammad, Thabtah and McCluskey 2013) suggest checking the certificate assigned with HTTPS including the extent of the trust certificate issuer, and the certificate age. Certificate Authorities that are consistently listed among the top trustworthy names include: “GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster and VeriSign”. Furthermore, by testing out our datasets, we find that the minimum age of a reputable certificate is two years |
| **DomainRegLen** | Based on the fact that a phishing website lives for a short period of time, we believe that trustworthy domains are regularly paid for several years in advance. In our dataset, we find that the longest fraudulent domains have been used for one year only. |
| **Favicon:** | A favicon is a graphic image (icon) associated with a specific webpage. Many existing user agents such as graphical browsers and newsreaders show favicon as a visual reminder of the website identity in the address bar. If the favicon is loaded from a domain other than that shown in the address bar, then the webpage is likely to be considered a Phishing attempt. |
| **NonStdPort** | This feature is useful in validating if a particular service (e.g. HTTP) is up or down on a specific server. In the aim of controlling intrusions, it is much better to merely open ports that you need. Several firewalls, Proxy and Network Address Translation (NAT) servers will, by default, block all or most of the ports and only open the ones selected. If all ports are open, phishers can run almost any service they want and as a result, user information is threatened. |
| **HTTPSDomainURL** | The phishers may add the “HTTPS” token to the domain part of a URL in order to trick users. For example, http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/. |
| **Request URL** | Request URL examines whether the external objects contained within a webpage such as images, videos and sounds are loaded from another domain. In legitimate webpages, the webpage address and most of objects embedded within the webpage are sharing the same domain |
| **AnchorURL** | An anchor is an element defined by the < a > tag. This feature is treated exactly as “Request URL”. However, for this feature we examine: 1.If the < /a >< a > tags and the website have different domain names. This is similar to request URL feature. 2.If the anchor does not link to any webpage, e.g.: A.< /a >< a href=""“#”"> B.< /a >< a href=""“#content”"> C.< /a >< a href=""“#skip”"> D.< /a >< a href=""“JavaScript"> |
| **LinksInScriptTags** | Given that our investigation covers all angles likely to be used in the webpage source code, we find that it is common for legitimate websites to use tags to offer metadata about the HTML document; |
| **ServerFormHandler** | SFHs that contain an empty string or “about:blank” are considered doubtful because an action should be taken upon the submitted information. In addition, if the domain name in SFHs is different from the domain name of the webpage, this reveals that the webpage is suspicious because the submitted information is rarely handled by external domains. |
| **InfoEmail** | Web form allows a user to submit his personal information that is directed to a server for processing. A phisher might redirect the user’s information to his personal email. To that end, a server-side script language might be used such as “mail()” function in PHP. One more client-side function that might be used for this purpose is the “mailto:” function. |
| **AbnormalURL** | This feature can be extracted from WHOIS database. For a legitimate website, identity is typically part of its URL. |
| **WebsiteForwarding** | The fine line that distinguishes phishing websites from legitimate ones is how many times a website has been redirected. In our dataset, we find that legitimate websites have been redirected one time max. On the other hand, phishing websites containing this feature have been redirected at least 4 times. |
| **StatusBarCust** | Phishers may use JavaScript to show a fake URL in the status bar to users. To extract this feature, we must dig-out the webpage source code, particularly the “onMouseOver” event, and check if it makes any changes on the status bar. |
| **DisableRightClick** | Phishers use JavaScript to disable the right-click function, so that users cannot view and save the webpage source code. This feature is treated exactly as “Using onMouseOver to hide the Link”. Nonetheless, for this feature, we will search for event “event.button==2” in the webpage source code and check if the right click is disabled. |
| **UsingPopupWindow** | It is unusual to find a legitimate website asking users to submit their personal information through a pop-up window. On the other hand, this feature has been used in some legitimate websites and its main goal is to warn users about fraudulent activities or broadcast a welcome announcement, though no personal information was asked to be filled in through these pop-up windows. |

| | |
|---|---|
| **IframeRedirection** | IFrame is an HTML tag used to display an additional webpage into one that is currently shown. Phishers can make use of the “iframe” tag and make it invisible i.e. without frame borders. In this regard, phishers make use of the “frameBorder” attribute which causes the browser to render a visual delineation. |
| **AgeofDomain** | This feature can be extracted from WHOIS database (Whois 2005). Most phishing websites live for a short period of time. By reviewing our dataset, we find that the minimum age of the legitimate domain is 6 months. |
| **DNSRecording** | For phishing websites, either the claimed identity is not recognized by the WHOIS database (Whois 2005) or no records founded for the hostname (Pan and Ding 2006). If the DNS record is empty or not found then the website is classified as “Phishing”, otherwise it is classified as “Legitimate”. |
| **WebsiteTraffic** | This feature measures the popularity of the website by determining the number of visitors and the number of pages they visit. However, since phishing websites live for a short period of time, they may not be recognized by the Alexa database (Alexa the Web Information Company., 1996). By reviewing our dataset, we find that in worst scenarios, legitimate websites ranked among the top 100,000. Furthermore, if the domain has no traffic or is not recognized by the Alexa database, it is classified as “Phishing”. Otherwise, it is classified as “Suspicious”. |
| **PageRank** | PageRank is a value ranging from “0” to “1”. PageRank aims to measure how important a webpage is on the Internet. The greater the PageRank value the more important the webpage. In our datasets, we find that about 95% of phishing webpages have no PageRank. Moreover, we find that the remaining 5% of phishing webpages may reach a PageRank value up to “0.2”. |
| **GoogleIndex** | This feature examines whether a website is in Google’s index or not. When a site is indexed by Google, it is displayed on search results (Webmaster resources, 2014). Usually, phishing webpages are merely accessible for a short period and as a result, many phishing webpages may not be found on the Google index. |
| **LinksPointingToPage** | The number of links pointing to the webpage indicates its legitimacy level, even if some links are of the same domain (Dean, 2014). In our datasets and due to its short life span, we find that 98% of phishing dataset items have no links pointing to them. On the other hand, legitimate websites have at least 2 external links pointing to them. |
| **StatsReport** | Several parties such as PhishTank (PhishTank Stats, 2010-2012), and StopBadware (StopBadware, 2010-2012) formulate numerous statistical reports on phishing websites at every given period of time; some are monthly and others are quarterly. In our research, we used 2 forms of the top ten statistics from PhishTank: “Top 10 Domains” and “Top 10 IPs” according to statistical-reports published in the last three years, starting in January2010 to November 2012. Whereas for “StopBadware”, we used “Top 50” IP addresses. |
| **result** | 1 means legitimate 0 is suspicious -1 is phishing |

# Algorithms

1. Problem Understanding
2. Data Collecting
3. Dataset Exploration and Cleansing
    - Null Values
    - structural errors
    - Outliers
    - Duplicated rows
    - Exploratory Data Analysis (EDA)
4. Modeling
    - Split data (Train, Valid, Test)
    - Baseline models
        - Logistic Regression
    - Models
        - Extra Tree
        - Random Forest
        - KNN
        - Support vector machine (SVM)

    - Chosen (best) Classification Models
5. Feature Engineering
    - Up-sampling:
        - Random
        - SMOTE
    - Down-sampling:
        - TomekLinks
        - KNN
    - Random Forest Hyperparameter Tuning Using Random Search
6. Graphs and Plots
    - Precision and Recall plots
    - Confusion Matrices
    - Roc Curve
7. Insights
8. Pickling
    - Train Models and Save to Pickled Files
    - Read Models from Pickled Files
9. Conclusion

# Tools:

- Technologies: Python, Jupiter Notebook
- Libraries: NumPy, Pandas, Matplotlib, Seaborn, sklearn, pickle, beautifultable, imblearn

# Communication:

## Charts:

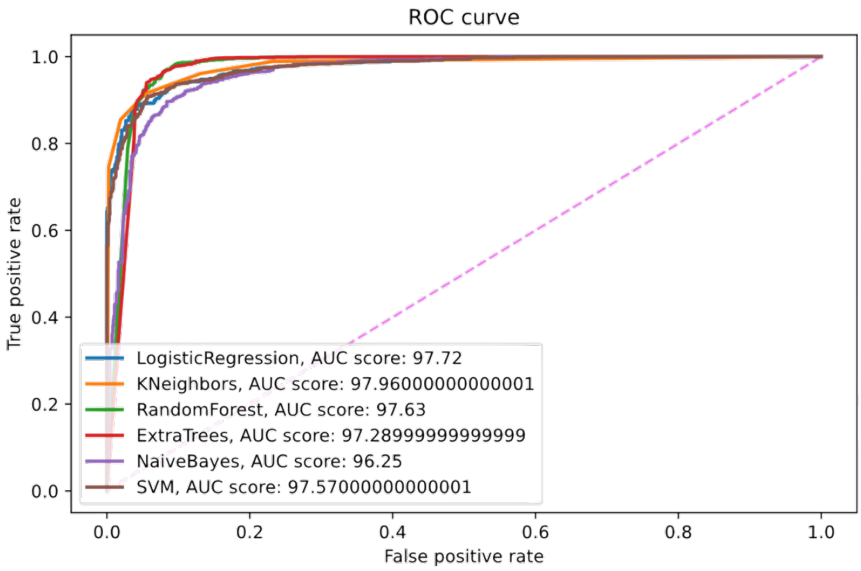



**Fig 1.Graph shows the Area Under The Curve of the Roc Curve**

# Presentation snips:

-SUPERVISED LEARNING-

## DETECTING PHISHING WEBSITES USING MACHINE LEARNING CLASSIFICATION

C-Sentinel
Your Cyber Guardian

Prepared by: Haneen Alhomoud

## what is a website?

**WEBSITE**

collection of many web pages or digital files that are written using HTML(HyperText Markup Language)

URL: https://blog.hubspot.com/marketing/

Scheme | Subdomain | Second-level domain | Top-level domain | Subdirectory



## Problem Description

**C-Sentinel**

Cyber security company that is now looking for trustworthy and steady techniques for phishing attacks websites detection.

**Phishing**

Phishing attack is a simplest way to obtain sensitive information from innocent users. Aim of the phishers is to acquire critical information like username, password and bank account details



## Dataset



## Dataset

**30 Features**

- UsingIP
- LongURL
- ShortURL
- Symbol@
- Redirecting//
- PrefixSuffix
- SubDomains
- HTTPS
- DomainRegLen
- Favicon
- NonStdPort
- HTTPSDomainURL
- RequestURL
- AnchorURL
- LinksInScriptTags
- ServerFormHandler
- InfoEmail
- AbnormalURL
- WebsiteForwarding
- StatusBarCust
- DisableRightClick
- UsingPopupWindow
- IframeRedirection
- AgeofDomain
- DNSRecording
- WebsiteTraffic
- PageRank
- GoogleIndex
- LinksPointingToPage
- StatsReport

**Target**

- Result



## Dataset Exploration (Data Cleansing)

**Data Obtained from:**

kaggle

12,000 datapoint

**Data values:**

1 : true
0 : false
-1 : suspicious

**Outliers**
None

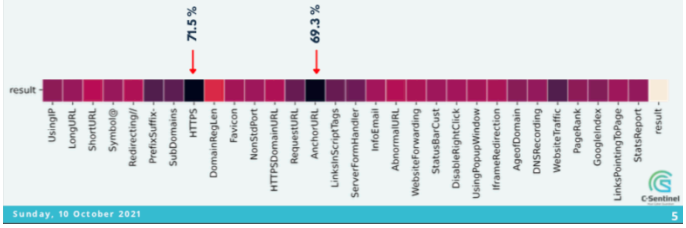
**Duplicated rows**
None

**Null Values**
None



## Dataset Correlations

- **HTTPS:** Hypertext transfer protocol secure is the secure version of HTTP, which is the primary protocol used to send data between a web browser and a website
- **AnchorURL:** An anchor is an element defined by the <a> tag



## Upsamling

| | Training f2 score | Validation f2 score |
|---|---|---|
| | 99% | 95.4% |
| Random | 99.2 | 95.5 |
| SMOTE | 99.1 | 95.3 |



## Downsamling

| | Training f2 score | Validation f2 score |
|---|---|---|
| | 99% | 95.4% |
| TomekLinks | 99.1 | 95.5 |
| KNN | 99.9 | 95.7 |



## Random Forest Hyperparameter Tuning (Random Search)

| | | |
|---|---|---|
| Training f2 score | 99.92% | 99.95% |
| Validation f2 score | 95.74% | 95.6% |



## Conclusion and Future Work

- KNN downsamling technique gave the best score
- RandomForest Classifier gave the best score

**Future Work**

- Use regex to obtain all URL features
- Apply more experiment and try more models
- Autoamated API phishing web detection



# Modeling - Classification

## Baseline Models

**Training**

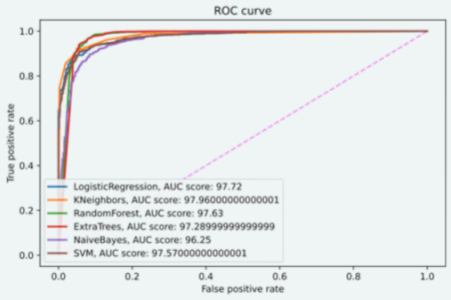
| | Accuracy | Precision | Recall | F2 |
|---|---|---|---|---|
| LogisticRegression | 93.2 | 93.4 | 91.1 | 91.6 |
| KNeighbors | 96.1 | 95.8 | 95.3 | 95.4 |
| RandomForest | 99.1 | 99.1 | 99.0 | 99.0 |
| ExtraTrees | 90.1 | 89.5 | 89.7 | 89.7 |
| NaiveBayes | 93.1 | 93.3 | 90.6 | 91.1 |

**Validation**

| | Accuracy | Precision | Recall | F2 |
|---|---|---|---|---|
| LogisticRegression | 92.5 | 92.5 | 90.1 | 90.5 |
| KNeighbors | 93.4 | 93.1 | 91.6 | 92 |
| RandomForest | 96.7 | 97.2 | 95 | 95.4 |
| ExtraTrees | 96.6 | 97.3 | 94.7 | 95.2 |
| NaiveBayes | 92.6 | 93.2 | 89.4 | 90.2 |

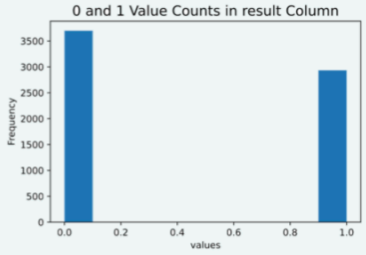


## Baseline Models Roc Curve






## Feature Engineering ( Handle Imbalancement)




Legitimate websites : 3,699

Phishing websites : 2,934