Date of Submission : December 18,2021

WEBSITE PHISHING DETECTION

PATTERN RECOGNITION

**** Project Report ****

Kajal -S20190020215
Chittoor Vamsi- S20190020208
Rithik kumar-S20190020248

,
Team:01

Abstract:
A phishing website is a common social engineering method that mimics trustful uniform resource locators (URLs) and webpages. The objective of this project is to train machine learning models and deep neural networks on the dataset created to predict phishing websites. Both phishing and benign URLs of websites are gathered to form a dataset and from them required URL and website content-based features are extracted. The performance level of each model is measures and compared.

1. Introduction

1.1. What is phishing?

Phishing is the most commonly used social engineering and cyber attack.Through such attacks, the phisher targets naïve online users by tricking them into revealing confidential information, with the purpose of using it fraudulently.

1.2. Solution

1.In order to avoid getting phished, users should have awareness of phishing websites.
2.Have a blacklist of phishing websites which requires the knowledge of the website being detected as phishing.
3.Detect them in their early appearance, using machine learning and deep neural network algorithms.
4.Of the above three, the machine learning based method is proven to be the most effective than the other methods.
5.Even then, online users are still being trapped into revealing sensitive information in phishing websites.

2. Mutual Information

Mutual information(MI) between two random variables is a non-negative value,which measures the dependency between the variables .It is equal to zero if and only if two random variables are independent ,and higher values mean higher dependency.

3. Project Implementation

Below mentioned are the steps involved in the completion of this project:

• Collect dataset containing phishing and legitimate websites from the open source platforms.

• Write a code to extract the required features from the URL database.

• Analyze and preprocess the dataset by using EDA techniques.

• Divide the dataset into training and testing sets.

• Run selected machine learning and deep neural network algorithms like SVM, Random Forest, Autoencoder on the dataset.

• Write a code for displaying the evaluation result considering accuracy metrics.

• Compare the obtained results for trained models and specify which is better.

3.1. Data Collection

• Legitimate URLs are collected from the dataset provided by University of New Brunswick,
https://www.unb.ca/cic/datasets/url-2016.html

• From the collection, 5000 URLs are randomly picked.

• Phishing URLs are collected from opensource service called PhishTank . This service provide a set of phishing URLs in multiple formats like csv, json etc. that gets updated hourly.

• From the obtained collection, 5000 URLs are randomly picked.

3.2. Feature extraction

The following category of features are selected:

3.2.1 Address Bar based Features:-

1.Domain of the url -
we can drop this as it does not have much role in classification

2.IP address in url -
Checks for the presence of IP address in the URL. URLs may have IP address instead of domain name. If an IP address is used as an alternative of the domain name in the URL, we can be sure that someone is trying to steal personal information with this URL.
If the domain part of URL has IP address, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

3.'@' symbol in url -
Checks for the presence of '@' symbol in the URL. Using "@" symbol in the URL leads the browser to ignore everything preceding the "@" symbol and the real address often follows the "@" symbol.
If the URL has '@' symbol, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

4.Length of url -
Computes the length of the URL. Phishers can use long URL to hide the doubtful part in the address bar. In this project, if the length of the URL is greater than or equal 54 characters then the URL classified as phishing otherwise legitimate.
If the length of URL $>=$ 54 , the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

5.Depth of url -
Computes the depth of the URL. This feature calculates the number of sub pages in the given url based on the '/'.
The value of feature is a numerical based on the URL.

6.Redirection '//' in url -
Checks the presence of "//" in the URL. The existence of "//" within the URL path means that the user will be redirected to another website. The location of the "//" in URL is computed. 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.
If the "//" is anywhere in the URL apart from after the protocal, thee value assigned to this feature is 1 (phishing) or else 0 (legitimate).

7.'http/https' in domain name -
Checks for the presence of "http/https" in the domain part of the URL. The phishers may add the "HTTPS" token to the domain part of a URL in order to trick users.
If the URL has "http/https" in the domain part, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

8.Using url shortening service -
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.
If the URL is using Shortening Services, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

9.Prefix or suffix "-" in domain -
Checking the presence of '-' in the domain part of URL. 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.
If the URL has '-' symbol in the domain part of the URL, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

3.2.2 Domain based Features:-

1.DNS Record -
For phishing websites, either the claimed identity is not recognized by the WHOIS database or no records founded for the hostname. If the DNS record is empty or not found then, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

2.Website Traffic -
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".
If the rank of the domain < 100000, the vlaue of this feature is 1 (phishing) else 0 (legitimate).

3.Age of Domain -
This feature can be extracted from WHOIS database. Most phishing websites live for a short period of time. The minimum age of the legitimate domain is considered to be 12 months for this project. Age here is nothing but different between creation and expiration time.
If age of domain > 12 months, the vlaue of this feature is 1 (phishing) else 0 (legitimate).

4.End Period of Domain -
This feature can be extracted from WHOIS database. For this feature, the remaining domain time is calculated by finding the different between expiration time current time. The end period considered for the legitimate domain is 6 months or less for this project.
If end period of domain > 6 months, the vlaue of this feature is 1 (phishing) else 0 (legitimate).

3.2.3 HTML and Java script based Feature:-

1.Iframe Redirection -
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.
If the iframe is empty or repsonse is not found then, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

2.Disabling Right click -
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
If the response is empty or onmouseover is found then, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

3.Website forwarding -
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.
If the response is empty or onmouseover is not found then, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

4.Status bar Customization -
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.

3.3. Machine learning models:

This is a supervised machine learning task. There are two major types of supervised machine learning problems, called classification and regression.
This data set comes under classification problem, as the input URL is classified as phishing (1) or legitimate (0). The machine learning models (classification) considered to train the dataset in this notebook are:

- Random Forest -

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.
As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.
The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

- XGBoost -

XGBoost stands for “Extreme Gradient Boosting”. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements Machine Learning algorithms under the Gradient Boosting framework. It provides a parallel tree boosting to solve many data science problems in a fast and accurate way.

- Support Vector Machines -

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.
The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

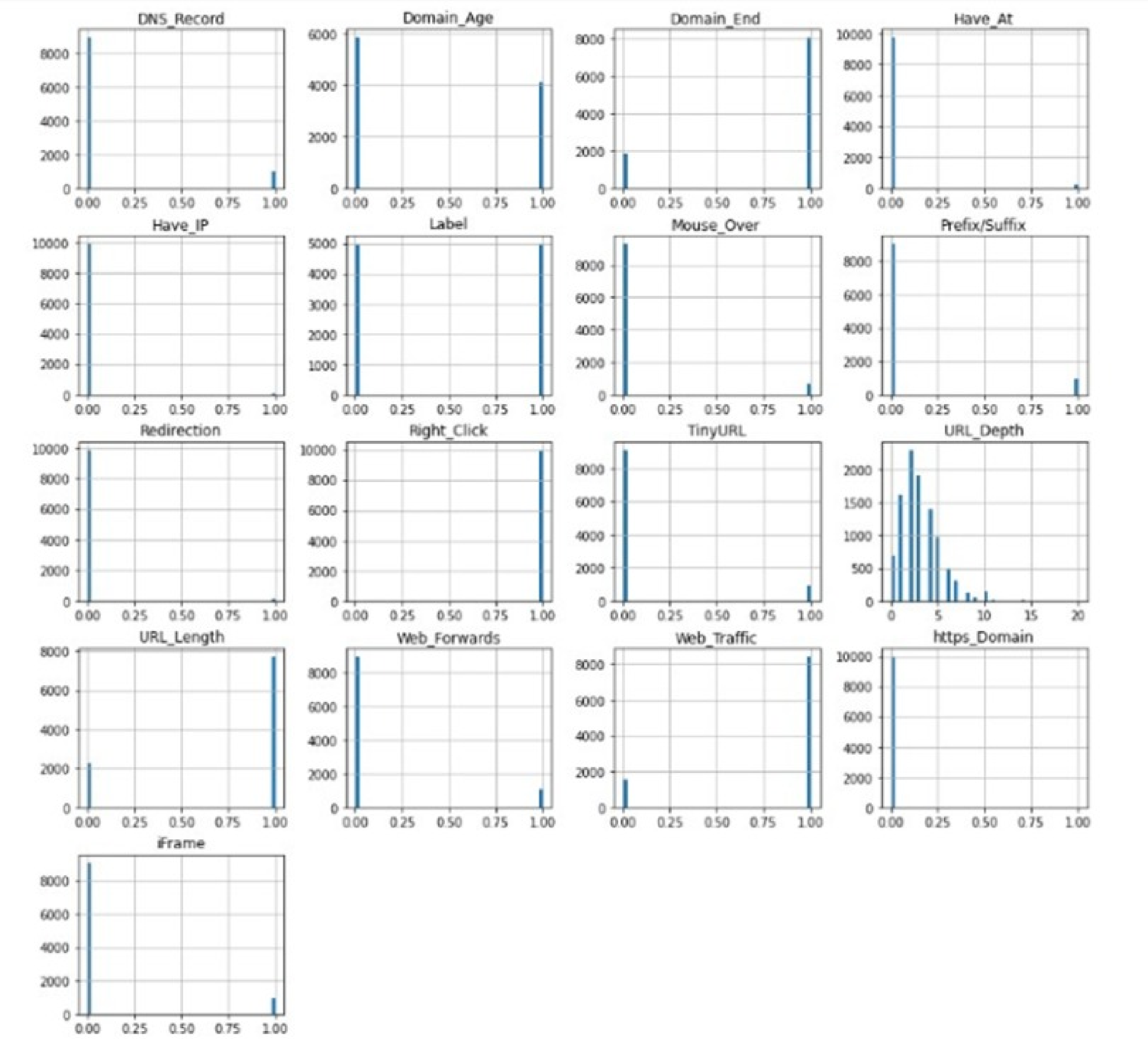
3.4. Model Evaluation:

- The models are evaluated, and the considered metric is accuracy.

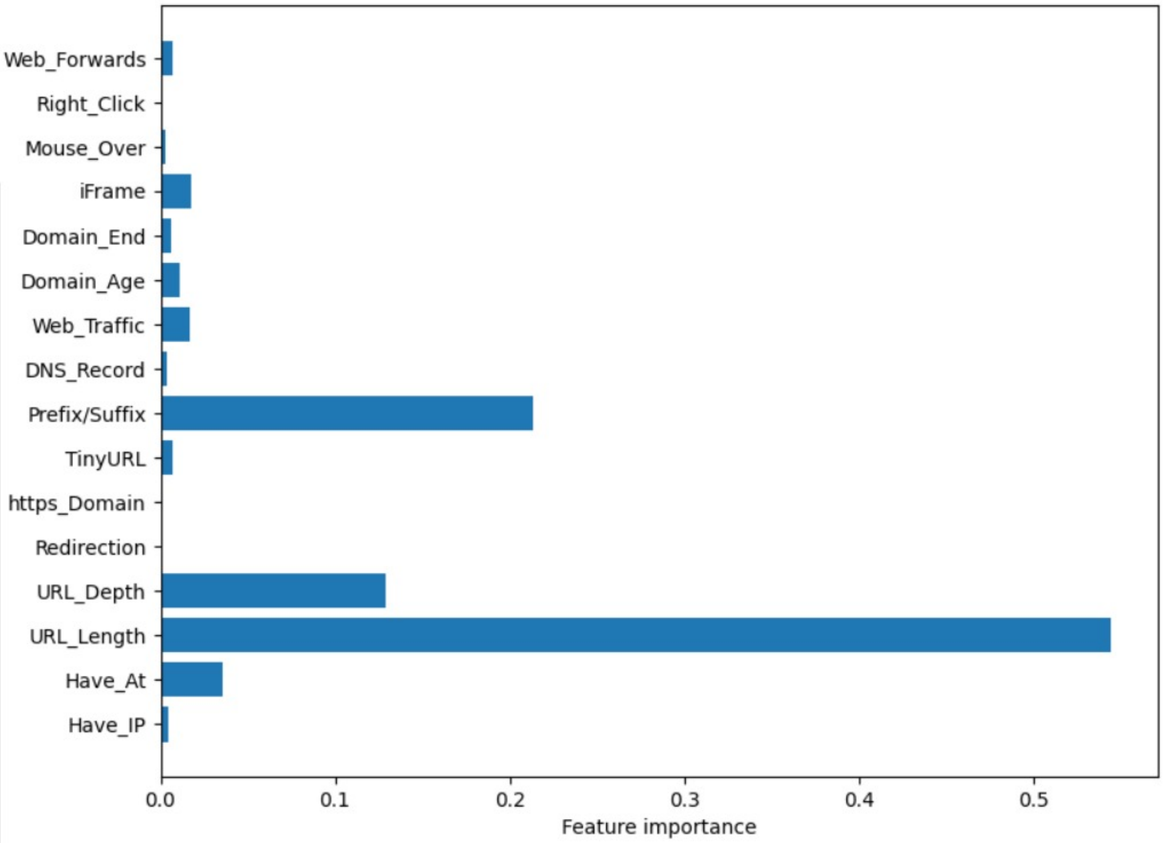
- The best model is selected for further use based on train and test accuracies.

4. Experimental Results

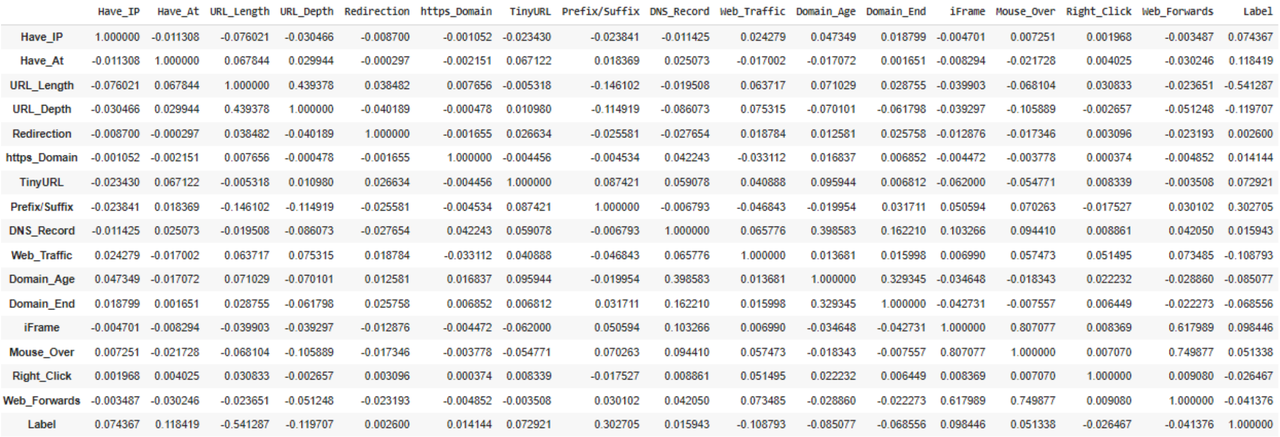
4.1. Feature Distribution



4.2. Feature importance

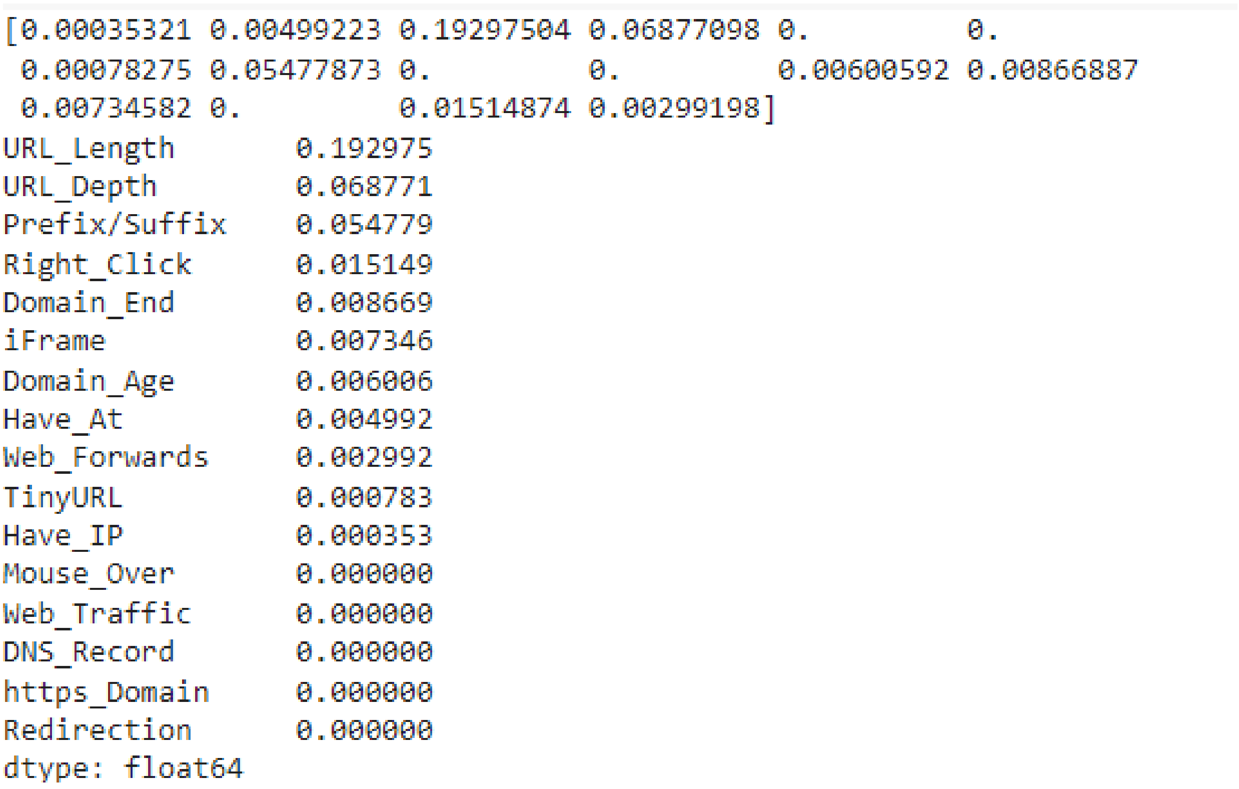


4.3. Correlation Matrix

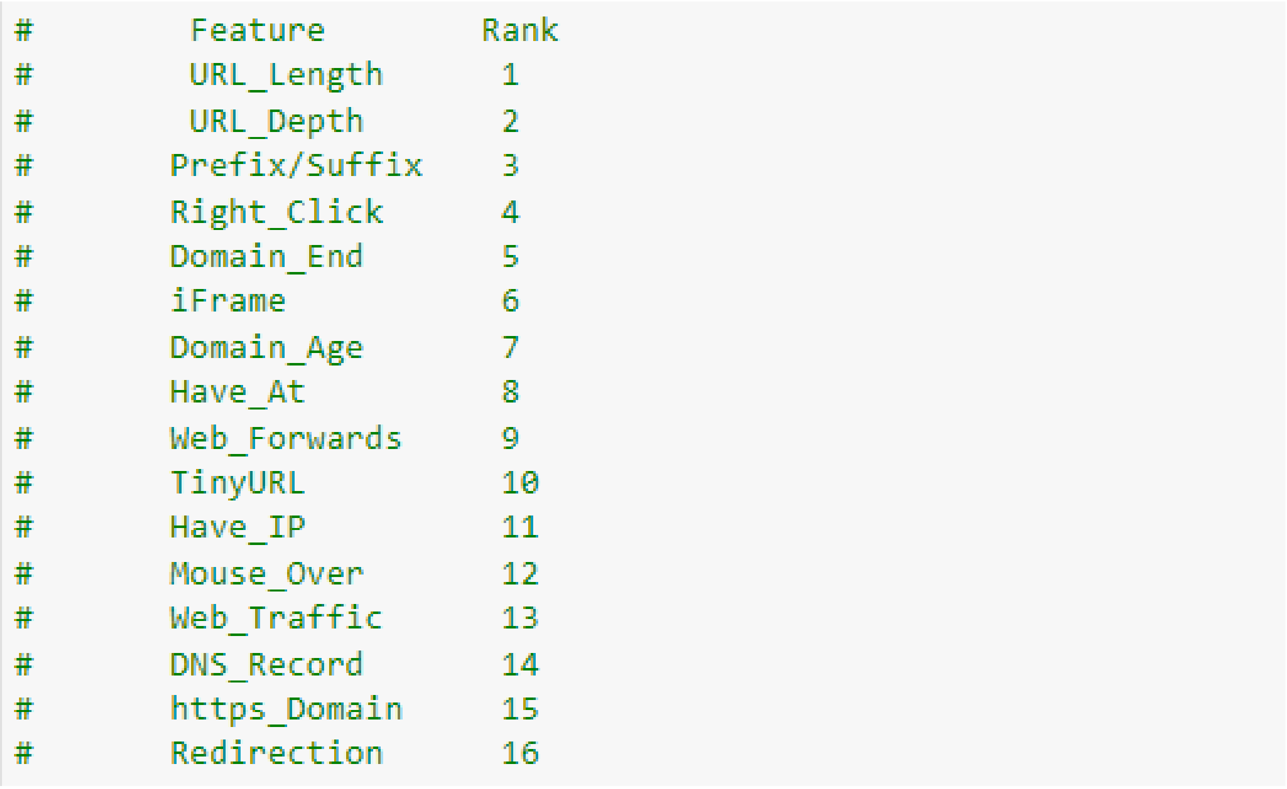


	Have_IP	Have_At	URL_Length	URL_Depth	Redirection	https_Domain	TinyURL	Prefix/Suffix	DNS_Record	Web_Traffic	Domain_Age	Domain_End	iFrame	Mouse_Over	Right_Click	Web_Forwards	Label
Have_IP	1.000000	-0.011308	-0.076021	-0.030466	-0.008700	-0.001052	-0.023430	-0.023841	-0.011425	0.024279	0.047349	0.018799	-0.004701	0.007251	0.001968	-0.003487	0.074367
Have_At	-0.011308	1.000000	0.067844	0.029944	-0.000297	-0.002151	0.067122	0.018369	0.025073	-0.017002	-0.017072	0.001651	-0.008294	-0.021728	0.004025	-0.030246	0.118419
URL_Length	-0.076021	0.067844	1.000000	0.439378	0.038482	0.007656	-0.005318	-0.146102	-0.019508	0.063717	0.071029	0.028755	-0.039903	-0.068104	0.030833	-0.023651	-0.541287
URL_Depth	-0.030466	0.029944	0.439378	1.000000	-0.040189	-0.000478	0.010980	-0.114919	-0.086073	0.075315	-0.070101	-0.061798	-0.039297	-0.105889	-0.002657	-0.051248	-0.119707
Redirection	-0.008700	-0.000297	0.038482	-0.040189	1.000000	-0.001655	0.026634	-0.025581	-0.027654	0.018784	0.012581	0.025758	-0.012876	-0.017346	0.003096	-0.023193	0.002600
https_Domain	-0.001052	-0.002151	0.007656	-0.000478	-0.001655	1.000000	-0.004456	-0.004534	0.042243	-0.033112	0.016837	0.006852	-0.004472	-0.003778	0.000374	-0.004852	0.014144
TinyURL	-0.023430	0.067122	-0.005318	0.010980	0.026634	-0.004456	1.000000	0.087421	0.059078	0.040888	0.095944	0.006812	-0.062000	-0.054771	0.008339	-0.003508	0.072921
Prefix/Suffix	-0.023841	0.018369	-0.146102	-0.114919	-0.025581	-0.004534	0.087421	1.000000	-0.006793	-0.046843	-0.019954	0.031711	0.050594	0.070263	-0.017527	0.030102	0.302705
DNS_Record	-0.011425	0.025073	-0.019508	-0.086073	-0.027654	0.042243	0.059078	-0.006793	1.000000	0.065776	0.398583	0.162210	0.103266	0.094410	0.008861	0.042050	0.015943
Web_Traffic	0.024279	-0.017002	0.063717	0.075315	0.018784	-0.033112	0.040888	-0.046843	0.065776	1.000000	0.013681	0.015998	0.006990	0.057473	0.051495	0.073485	-0.108793
Domain_Age	0.047349	-0.017072	0.071029	-0.070101	0.012581	0.016837	0.095944	-0.019954	0.398583	0.013681	1.000000	0.329345	-0.034648	-0.018343	0.022232	-0.028860	-0.085077
Domain_End	0.018799	0.001651	0.028755	-0.061798	0.025758	0.006852	0.006812	0.031711	0.162210	0.015998	0.329345	1.000000	-0.042731	-0.007557	0.006449	-0.022273	-0.068556
iFrame	-0.004701	-0.008294	-0.039903	-0.039297	-0.012876	-0.004472	-0.062000	0.050594	0.103266	0.006990	-0.034648	-0.042731	1.000000	0.807077	0.008369	0.617989	0.098446
Mouse_Over	0.007251	-0.021728	-0.068104	-0.105889	-0.017346	-0.003778	-0.054771	0.070263	0.094410	0.057473	-0.018343	-0.007557	0.807077	1.000000	0.007070	0.749877	0.051338
Right_Click	0.001968	0.004025	0.030833	-0.002657	0.003096	0.000374	0.008339	-0.017527	0.008861	0.051495	0.022232	0.006449	0.008369	0.007070	1.000000	0.009080	-0.026467
Web_Forwards	-0.003487	-0.030246	-0.023651	-0.051248	-0.023193	-0.004852	-0.003508	0.030102	0.042050	0.073485	-0.028860	-0.022273	0.617989	0.749877	0.009080	1.000000	-0.041376
Label	0.074367	0.118419	-0.541287	-0.119707	0.002600	0.014144	0.072921	0.302705	0.015943	-0.108793	-0.085077	-0.068556	0.098446	0.051338	-0.026467	-0.041376	1.000000

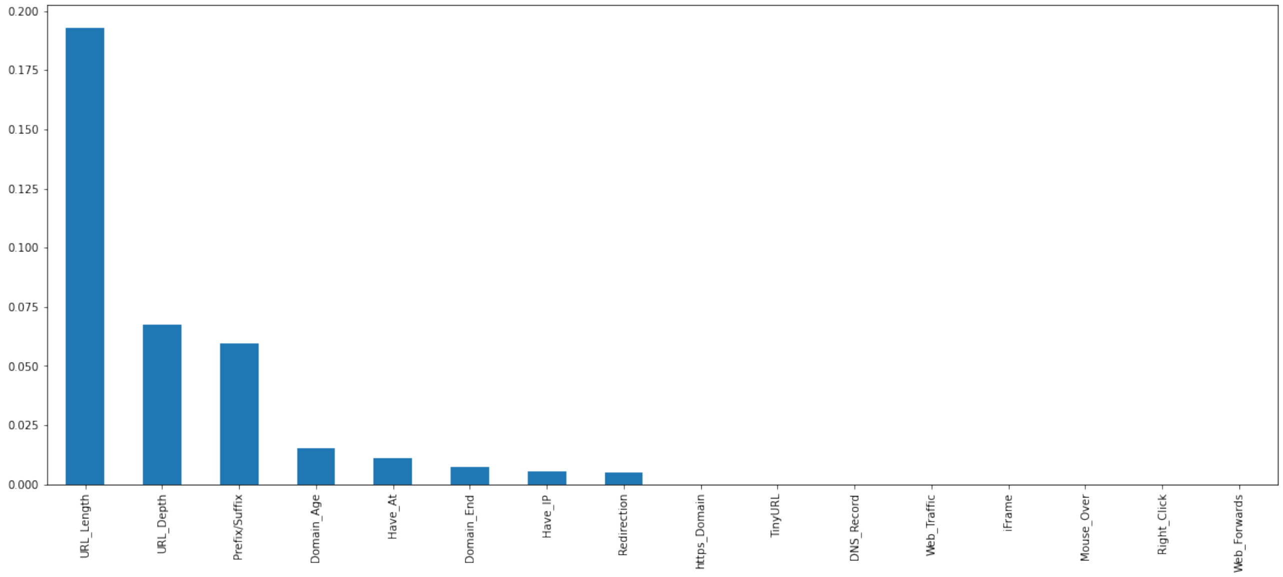
4.4. Rank Features Based on Mutual Information



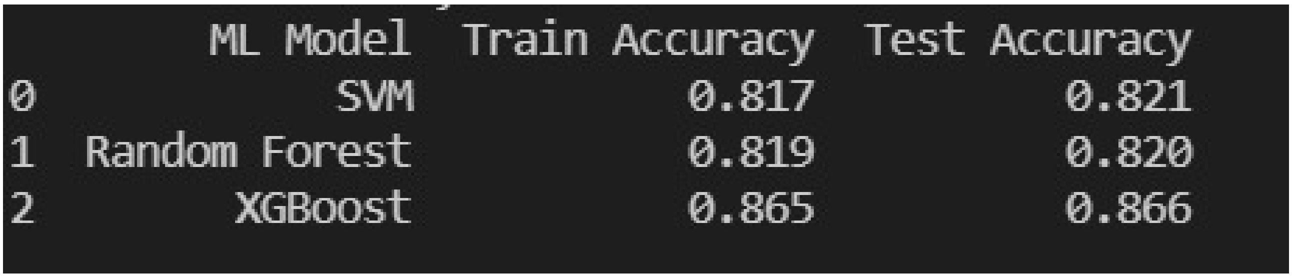
```
[0.00035321 0.00499223 0.19297504 0.06877098 0.         0.
 0.00078275 0.05477873 0.         0.         0.00600592 0.00866887
 0.00734582 0.         0.01514874 0.00299198]
URL_Length       0.192975
URL_Depth        0.068771
Prefix/Suffix    0.054779
Right_Click      0.015149
Domain_End       0.008669
iFrame           0.007346
Domain_Age       0.006006
Have_At          0.004992
Web_Forwards     0.002992
TinyURL          0.000783
Have_IP          0.000353
Mouse_Over       0.000000
Web_Traffic      0.000000
DNS_Record       0.000000
https_Domain     0.000000
Redirection      0.000000
dtype: float64
```



```
#         Feature        Rank
#         URL_Length       1
#         URL_Depth        2
#        Prefix/Suffix     3
#        Right_Click       4
#        Domain_End        5
#        iFrame            6
#        Domain_Age        7
#        Have_At           8
#        Web_Forwards      9
#        TinyURL           10
#        Have_IP           11
#        Mouse_Over        12
#        Web_Traffic       13
#        DNS_Record        14
#        https_Domain      15
#        Redirection       16
```



4.5. Accuracies



```
        ML Model  Train Accuracy  Test Accuracy
0            SVM           0.817          0.821
1  Random Forest           0.819          0.820
2        XGBoost           0.865          0.866
```

Figure 1: Model accuracies

5. Conclusion

Thus we have built a models to predict the phishing websites based on url features.The accuracy depends on the problem we are solving and the dataset available.With doing proper exploration of data we got highest accuracy of 86.6 percentage with XGBoost model.This could be improved further if we have more data.For the above it is clear that the XGBoost model gives better performance. The model is saved for further usage.Now we can use this model in real world scenario to find phising websites.

6. Link for Project Repository

Repository link

7. Contributions towards the Project

7.1. Kajal :

Feature extraction from raw website URLs and rank features based on mutual information

7.2. Vamsi :

Applying ML models to the dataset and rank features based on mutual information

7.3. Rithik :

Data preprocessing and rank features based on mutual information