Web Phishing Detection

Submitted by

Parvathy AJ	19MIA1048
Tharani Kumar	19MIA1033
Nithya Sharma	19MIA1028

J Component - Report

CSE4036 - MACHINE LEARNING
INTEGRATED MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

TABLE OF CONTENTS

SECTION 1

ABSTRACT

Phishing is one of the biggest dangers to a user's personal information and data on the internet. This project aims to use a dataset containing features and data regarding websites and use it to predict whether a site is a Phishing website or not. Several machine learning models were used for the prediction, such as Logistic Regression, KNN, Random Forest, Decision Tree, SVM Sigmoid and SVM rbf. The analysis was done on Jupyter Notebook in Python and the findings have been presented.

SECTION 2

INTRODUCTION

Phishing is a type of social engineering attack, where an individual(s) tricks the victim into revealing sensitive information or acquiring harmful software by posing as a legitimate source using fraudulent tactics. It provides a hacker with the scale and capability to target hundreds, if not thousands, of people at once.

Cybercriminals utilize social engineering to persuade their victims to run harmful files on their computers, click on a link to an infected website, or unwittingly give their personal information to criminals.

Phishing schemes entail sending emails or SMS that appear to be from trusted sources. They may appear to be from a reputable business or law enforcement agency, but they are actually malware. These messages are particularly meant to use fear and intimidation methods to get the victim to open the email. When someone opens it, dangerous software is installed on their computer, and the cybercriminal is now in your system.

Sending mails with embedded URLs is a common social engineering technique. When someone clicks on the link, they are sent to a phishing website. A phishing email might include a malicious attachment packed with vulnerabilities, frequently claiming that the file is an unpaid invoice that requires attention.

Nowadays, phishing attacks have evolved to the point that they now often completely mirror the site being attacked, allowing the attacker to watch everything the victim does while exploring the site and easily bypass any extra security barriers the victim might have in place.

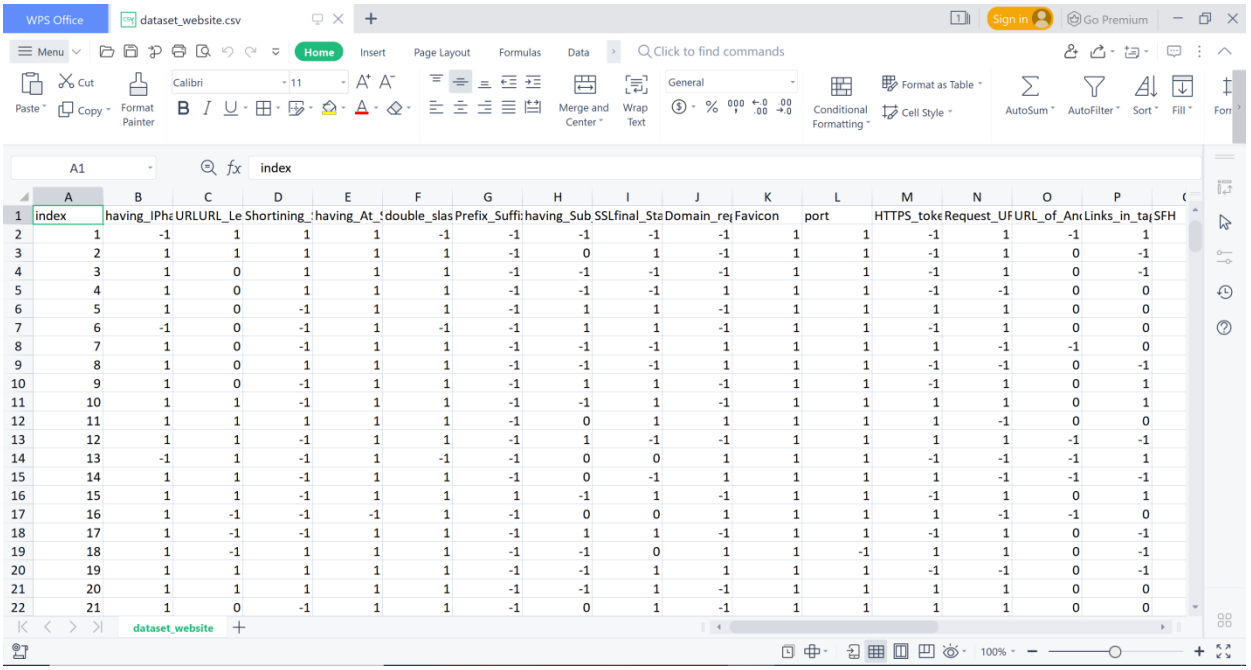
If you fall for phishing and click on a website or attachment that contains malware, your computer will be infected, and is prone to all sorts of terrible consequences. For example, you may become a victim of ransomware, which encrypts all of your files and demands a huge sum to unlock them (with no guarantee that will happen, even if you do pay out). At times, the malevolent party will have your username and password – maybe even your bank account information – and will be able to get into your account, even altering the password to lock you out the next time you try to log in.

It is highly necessary to come up with an effective solution to protect against phishing attacks, by detection of phishing websites and making sure a potential victim is aware that the link they might be entering could put them at risk. In this project, various machine learning models have been built and tested to deduce which yields the best accuracy in detecting a phishing website.

SECTION 3

DATASET

The dataset used in the project is a collection of 11,057 websites and is used for the purpose of phishing detection. It contains thirty website features, which have been proven to be reliable and effective in predicting phishing. The data present in the dataset is categorical, and has three values -1, 0 and 1, which correspond to Phishing, Suspicious and Legitimate. The various features in the dataset are explained below.



index	having_IPhe	URLURL_Le	Shortining_	having_At_	double_slas	Prefix_Suffi	having_Sub	SSLfinal_Sta	Domain_re	Favicon	port	HTTPS_toke	Request_UF	URL_of_Anc	Links_in_ta	SFH
1	-1	1	1	1	-1	-1	-1	-1	-1	1	1	-1	1	-1	1	
2	1	1	1	1	1	-1	0	1	-1	1	1	-1	1	0	-1	
3	1	0	1	1	1	-1	-1	-1	-1	1	1	-1	1	0	-1	
4	1	0	1	1	1	-1	-1	-1	1	1	1	-1	-1	0	0	
5	1	0	-1	1	1	-1	1	1	-1	1	1	1	1	0	0	
6	-1	0	-1	1	-1	-1	1	1	-1	1	1	-1	1	0	0	
7	1	0	-1	1	1	-1	-1	-1	1	1	1	1	-1	-1	0	
8	1	0	1	1	1	-1	-1	-1	1	1	1	-1	-1	0	-1	
9	1	0	-1	1	1	-1	1	1	-1	1	1	-1	1	0	1	
10	1	1	-1	1	1	-1	-1	1	-1	1	1	1	1	0	1	
11	1	1	1	1	1	-1	0	1	1	1	1	1	-1	0	0	
12	1	1	-1	1	1	-1	1	-1	-1	1	1	1	1	-1	-1	
13	-1	1	-1	1	-1	-1	0	0	1	1	1	-1	-1	-1	1	
14	1	1	-1	1	1	-1	0	-1	1	1	1	1	-1	-1	-1	
15	1	1	-1	1	1	1	-1	1	-1	1	1	-1	1	0	1	
16	1	-1	-1	-1	1	-1	0	0	1	1	1	1	-1	-1	0	
17	1	-1	-1	1	1	-1	1	1	-1	1	1	-1	1	0	-1	
18	1	-1	1	1	1	-1	-1	0	1	1	-1	1	1	0	-1	
19	1	1	1	1	1	-1	-1	1	1	1	1	-1	-1	0	-1	
20	1	1	1	1	1	-1	-1	1	-1	1	1	1	1	0	0	
21	1	0	-1	1	1	-1	0	1	-1	1	1	1	1	0	0	

Sample of Dataset

3.1 Using IP Address

If a website contains the IP address of the website in the URL instead of the domain name, it is certain that someone is using this as a way to steal information from the user. At times, the IP address would be transformed into hexadecimal code as well. If a website has the IP address in the URL, it is classified as Phishing (-1) and if not, it is classified as Legitimate (1).

3.2 Using Long URLs

Phishers use longer URLs to hide the suspicious part of the URL. In the

dataset, a URL length lesser than 54 is classified as Legitimate (1), length between 54 and 75 is classified as Suspicious (0) and length greater than 75 is classified as Phishing (-1). This feature rule was arrived upon to ensure accuracy using a method based on frequency.

3.3 Using URL Shortening Services

URL shortening is a means of reducing the length of a URL while still directing to the desired webpage on the "World Wide Web." This is done by using a "HTTP Redirect" on a short domain name that redirects to a webpage with a lengthy URL f a website uses an URL shortening service, its classified as Phishing (-1) and if not, it's classified as Legitimate (1).

3.4 Using @ Symbol

When the "@" sign is used in a URL, the browser ignores anything before the "@" symbol, and the genuine address is commonly found after the "@" symbol. If a website has an @ symbol, its classified as Phishing (-1) and if not, it's classified as Legitimate (1).

3.5 Redirecting using "//"

If the URL path contains the character "//," the visitor will be routed to another website. If the URL begins with "HTTP," the "//" must be placed in the sixth position and if it begins with "HTTPS," , the "//" should occur in the seventh position. So, any URL where the "//" appears in the eighth position or higher is classified as Phishing (-1) and the others are classified as Legitimate (1).

3.6 Adding Prefix or Suffix Separated by "-"

In genuine URLs, the dash sign is rarely used. Phishers tend to add prefixes or suffixes to the domain name separated by (-) to make visitors believe they are dealing with a legitimate website. If a website has a "-" symbol, its classified as Phishing (-1) and if not, it's classified as Legitimate (1).

3.7 Sub Domain & Multi Domain

In order to create a rule to extract this feature, the (www.) was removed from the URL, which is a subdomain in and of itself. Then, if the (ccTLD) exists, it must be removed. Finally, all of the dots were added up. The URL is categorised

as Suspicious (0) if the number of dots is larger than one as this means that it has one subdomain. If there are more than two dots, it is considered Phishing (-1) since it will have several subdomains. Otherwise, if the URL does not contain any subdomains, we will mark the feature as "Legitimate (1).

3.8 .HTTPS (Hyper Text Transfer Protocol with Secure Sockets Layer)

The presence of HTTPS is critical in conveying the authenticity of a website, yet it is certainly insufficient. "GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster, and VeriSign" are frequently named among the most trustworthy certificate authorities. Furthermore, it was discovered that a trustworthy certificate must be at least two years old after examining datasets. If the website uses HTTPS, has a trusted issuer and the age of the certificate is greater than or equal to 1, it is classified as Legitimate (1). If it uses HTTPS and the issuer is not trusted, it is marked as Suspicious (0). Otherwise, it is classified as Phishing (-1).

3.9 Domain Registration Length

Trustworthy domains are consistently purchased for several years in advance, based on the fact that phishing websites only exist for a limited time. It was discovered that the longest fake domains were only utilized for one year in the database. If the domain registration expires in a year or lesser, it is classified as Phishing (-1). Otherwise, it is Legitimate (1).

3.10 Favicon

A favicon (short for "favourite icon") is a visual image (icon) that is connected with a certain webpage. Many existing user agents, such as graphical browsers and newsreaders, display the favicon in the address bar as a visual reminder of the website's identity. If the favicon is loaded in from a different domain then the webpage, then it is considered Phishing (-1) and otherwise, its Legitimate (1).

3.11 Using Non-Standard Port

This feature is important for determining whether a server-side service is functional. It is much preferable to only open ports that you require in order to control incursions.Several firewalls, proxy servers, and Network Address Translation (NAT) servers will block all or most of the ports by default, only

allowing access to the ones that have been specified. Phishers may launch nearly any service they want if all ports are open, putting user information at risk. If the port # is of the preferred status to phishers, its considered Phishing (-1) and if not, it is Legitimate (1).

3.12 Presence of "HTTPS" Token in the Domain Part of the URL

Phishers might add HTTPS token int he domain part of the URL to trick the victims. If a domain contains "HTTPS", its classified as Phishing (-1) and if not, it's classified as Legitimate (1).

3.13 Request URL:

If external objects like images, videos and sounds embedded in the webpage along with the webpage address do not share the same domain, then it is certain that the website is not safe. If the objects in website do not share same domain, then it is classified as Phishing (-1) and if they do, it is classified as Legitimate (1).

3.14 URL of Anchor:

This is similar to Request URL where, if the <a> tag and the website have different domains, the website is not safe. If the percentage of URL Anchor is less than 31%, then the website is classified as Legitimate (1), if it is between 31% and 67% it is classified as Suspicious (0) and if it is greater than 67%, the website is classified as Phishing (-1).

3.15 Links in <Meta>, <script> and <link> tags:

Websites use <Meta>, <Script> and <Link> tags to offer metadata, create a client side script and retrieve other web resources respectively. If the percentage of Links in these tags are very high (>81%), then the website is classified as Phishing (-1), if it is between 17% and 81%, it is classified as Suspicious (0) and if the percentage is very low (<17%), it is classified as Legitimate (1).

3.16 Server Form Handler (SSH):

If the SFH of a website contains empty strings or has a different domain name from that of the website, the website is not considered safe. If the SFH has an empty string or has an 'about:blank', the website is classified as Phishing (-1), If the SFH refers to different domain, the website is classified as Suspicious (0)

Otherwise, the website is classified as Legitimate (1).

3.17 Submitting Information to Email:

Using ‘mail()’ or ‘mailto:’ in the web form to allow the user to submit personal information can mean that the phisher is redirecting the user’s information to their personal email. If the website has ‘mail()’ or ‘mailto:’, then the website is classified as Phishing (-1) and if its got neither of them, the website is classified as Legitimate (1).

3.18 Abnormal URL

Legitimate websites usually have identity as part of its URL. If the host name of the website is not included in the URL, the website is classified as Phishing (-1) and if the website has its host name included in its URL, it is classified as Legitimate (1).

3.19 Website Forwarding

Legitimate websites are found to be redirected less times compared to that of Phishing websites. If a website has been redirected more than 4 times, it is classified as Phishing (-1), if it has been redirected between 2 and 4 times, it is classified as Suspicious (0) and if the website has less than one redirect, it is classified as Legitimate (1).

3.20 Status Bar Customization

Phishers show fake URL in the status bars for users with the help of JavaScript. If there are changes in the status bar when the mouse pointer is moved onto the URL, the website is classified as Phishing (-1) and if there are no changes in the status bar, the website is classified as Legitimate (1).

3.21 Disabling Right Click

Phishing websites usually have their right click function disabled preventing users to save the webpage source code. If the right click function of a website is disabled, it is classified as Phishing (-1) and if the right click function is enabled, it is classified as Legitimate (1).

3.22 Using Pop-up Window

Phishers enable users to submit their personal information through pop-up windows to steal their information. If the pop-up window of the website has text fields, the website is classified as Phishing (-1) and if it doesn't, the website is classified as Legitimate (1).

3.23 IFrame Redirection:

Phishers use Iframe tag to display an invisible, additional webpage on the existing one. If the website uses Iframe, it is classified as Phishing (-1) and if the website does not Iframe tag, it is classified as Legitimate (1).

3.24 Age of Domain

This information may be obtained from the WHOIS database. The majority of phishing websites are often only active for a short time. The authentic domain has a minimum age of 6 months If the domain's age is greater than or equal to six months, it is considered Legitimate (1) and otherwise, it's Phishing (-1).

3.25 DNS Record

In the case of phishing websites, the stated identity is either not recognized by the WHOIS database or no records for the host name have been identified. The website is categorised as Phishing (-1) if the DNS record is empty or not discovered; otherwise, it is rated as Legitimate (1).

3.26 Website Traffic

This function determines the number of visitors and the number of pages they view to determine the popularity of the website. However, because phishing websites only exist for a brief time, the Alexa database may not recognize them. Legitimate websites were rated among the top 100,000 in the worst-case scenario and so, if a website is ranked here, it is considered Legitimate (1). Furthermore, it is categorised as Phishing (-1) if the domain has no traffic or is not recognized by the Alexa database. It is classed as Suspicious (0) otherwise.

3.27 Page Rank

PageRank is a number that ranges from 0 to 1. The goal of PageRank is to determine the importance of a webpage on the Internet. The more important a webpage is, the higher its PageRank value. In our research, we discovered that around 95% of phishing websites had no PageRank. Furthermore, we discovered that the remaining 5% of phishing URLs can have a PageRank of up to "0.2."

3.28 Google Index

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. If a website it is classified as Legitimate (1) and if it not, it's considered Phishing (-1).

3.29 Number of Links Pointing to Page

The number of links pointing to a webpage indicates the legitimacy of the site. We discovered that 98 percent of phishing sites have no connections linking to them in our datasets, owing to their short lifespan. Legitimate websites, on the other hand, have at least two external links going to them. If the number of links pointing to a website is 0, it is considered Phishing (-1) and if it is 1 or 2, then it is considered Suspicious (0). Otherwise, it's considered Legitimate (1).

3.30 . Statistical-Reports Based Feature

Several organizations, including PhishTank (PhishTank Stats, 2010-2012) and StopBadware (StopBadware, 2010-2012), produce a variety of statistics reports on phishing websites as Top 10/Top 100 lists at different intervals of time. If the website belongs to a list of Top Phishing IPs or Top Phishing domains, it is classified as Phishing (-1) and otherwise, it's classified as Legitimate (1).

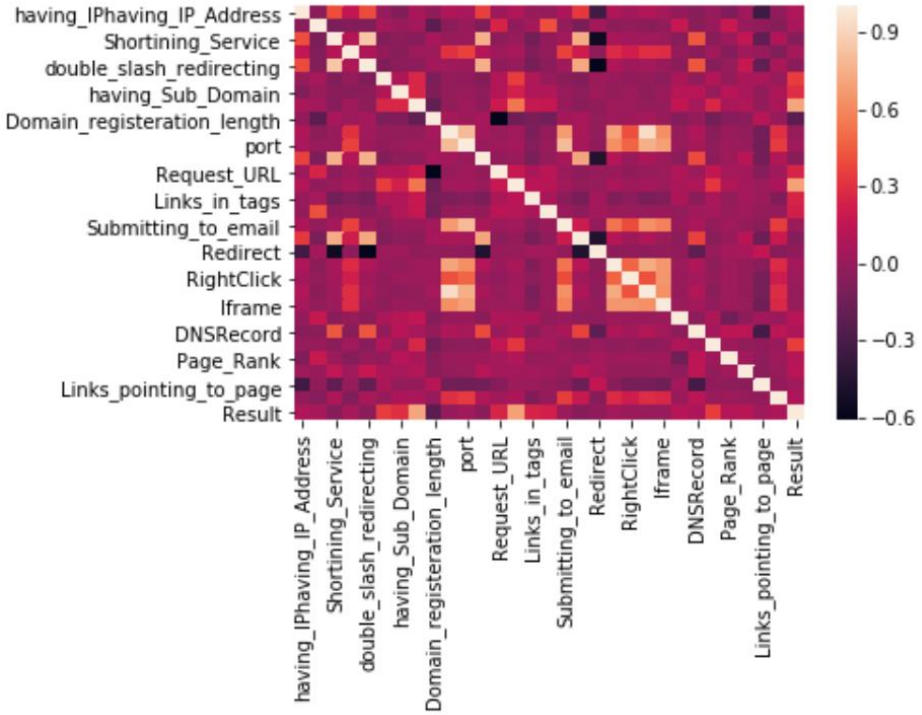
SECTION 4

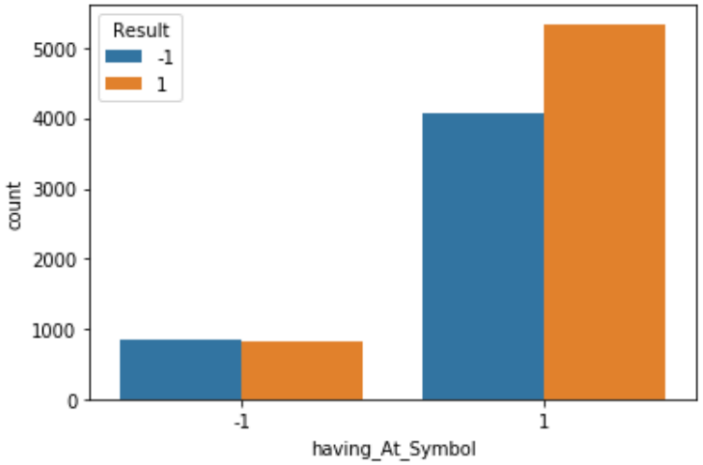
DATA PREPROCESSING & DESCRIPTION

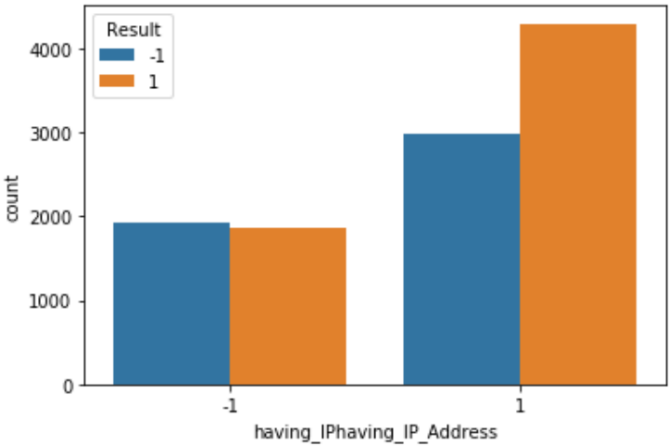
The data type of the data in the dataset was found to be int64. The data was checked for null values and it was found that there are no null values. From the result column, the number of phishing websites (-1) was found to be 4898 and the number of legitimate websites (1) was found to be 6157. The column “Index” was dropped as it was not important to predict the target variable.

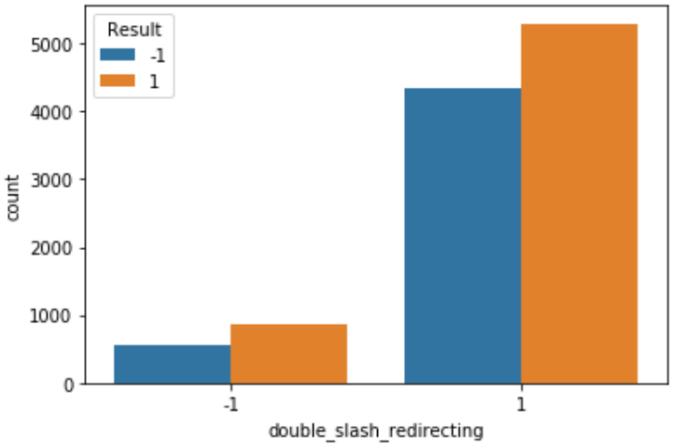
A description of the dataset was generated. The count of every column in the dataset was found to be 11,055, as there was no missing data. The mean and standard deviation of the target was found to be 0.113885 and 0.993539 respectively, which indicates that the data has a higher level of dispersion.The correlations between all the features of the dataset were found, as well as the correlation between each feature and the target variable. The result had the highest correlation with the feature SSLfinal_State with a value of 0.714741.

Various visualizations of the dataset were done to better understand and depict the data, which are shown below. These include a heatmap, which shows the correlation between two features using colors and intensity and the lighter the hue, the higher the correlation. Countplots were also used, which help us see the number of phishing and legitimate websites based on each feature.









The dataset was split into the independent variables and dependant variable which were named x and y respectively. The very last column, “Result” was taken as y and the remaining columns were taken as x.

The data was then split into training and testing sets with a test size of 0.2, meaning that 80% of the data was taken for training and the remaining 20% was taken for testing.

SECTION 5

MACHINE LEARNING MODELS

5.1 LOGISTIC REGRESSION

The logistic model (or logit model) is used in statistics to represent the likelihood of a specific class or event, such as pass/fail, win/lose, alive/dead, or healthy/sick, existing. This may be used to represent a variety of occurrences, such as identifying whether a picture contains a cat, dog, lion, or other animal. Each identified object in the image would be given a probability between 0 and 1, with the total adding up to one.

Logistic Regression yielded an accuracy of 91.68% and upon performing cross validation, an accuracy of 92.54% was obtained.

```
print(classification_report(y_test, y_predLR))
```

```
              precision    recall  f1-score   support

          -1       0.92      0.89      0.91      1014
           1       0.91      0.94      0.92      1197

    accuracy                           0.92      2211
   macro avg       0.92      0.91      0.92      2211
weighted avg       0.92      0.92      0.92      2211
```

```
confusion_matrix(y_test,y_predLR)
```

```
array([[ 905,  109],
       [  75, 1122]], dtype=int64)
```

5.2 KNN

The k-nearest neighbours algorithm (k- NN) is a non-parametric approach for classification and regression in pattern recognition. The input is the k closest training instances in the feature space in both situations.

KNN with k = 15, 25, 45, and 105 was attempted and the best accuracy of 93.44% was obtained by KNN using 15 as the k value. After performing cross validation, an accuracy of 94.72% was acquired.

```
print(classification_report(y_test, y_predKnn1))
```

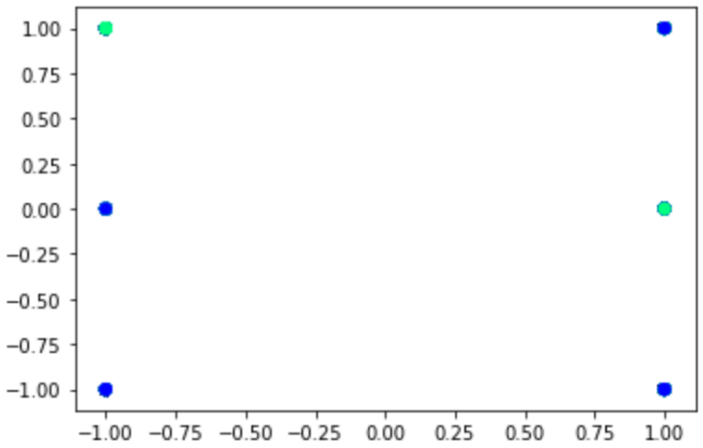
```
              precision    recall  f1-score   support

          -1       0.95      0.90      0.93      1014
           1       0.92      0.96      0.94      1197

    accuracy                           0.93      2211
   macro avg       0.94      0.93      0.93      2211
weighted avg       0.94      0.93      0.93      2211
```

```
confusion_matrix(y_test,y_predKnn1)
```

```
array([[ 917,   97],
       [  48, 1149]], dtype=int64)
```

5.3 SVM

SVM is a supervised machine learning algorithm that can be used for regression and classification. SVM creates a hyperplane which divides the data into classes. This is done with the help of a kernel trick which is used to transform the data. In the below scatter plot, it can be observed that the data points are not linearly separable. Hence linear kernel would not be optimal for transforming the data.



5.3.1 SVM RBF

SVM RBF (Radial Basis Kernel) is a kernel function that is used to transform non linear data. In RBF, the boundaries are hypothesized to be curve shaped. This model yielded an accuracy of 94.07% .

```
print(classification_report(y_test, y_predSVM1))
```

```
              precision    recall  f1-score   support

          -1       0.95      0.92      0.93      1014
           1       0.93      0.96      0.95      1197

    accuracy                           0.94      2211
   macro avg       0.94      0.94      0.94      2211
weighted avg       0.94      0.94      0.94      2211
```

```
confusion_matrix(y_test,y_predSVM1)
```

```
array([[ 933,   81],
       [  50, 1147]], dtype=int64)
```

5.3.2 SVM Polynomial

SVM Polynomial kernel is used when there is a similarity of training samples in feature space over polynomials of the original variables. This model yielded an accuracy of 94.53% and a cross validation performed on it gave an accuracy of 95.36%

```
print(classification_report(y_test, y_predSVM2))
```

```
              precision    recall  f1-score   support

          -1       0.83      0.80      0.81      1014
           1       0.84      0.86      0.85      1197

    accuracy                           0.83      2211
   macro avg       0.83      0.83      0.83      2211
weighted avg       0.83      0.83      0.83      2211
```

```
confusion_matrix(y_test,y_predSVM2)
```

```
array([[ 811,  203],
       [ 167, 1030]], dtype=int64)
```

5.4 Decision Tree

A decision tree is a decision-making aid that employs a tree-like model of decisions and their potential results, such as chance event outcomes, resource costs, and utility. We strive to construct a condition on the features at each step or node of a decision tree used for classification to segregate all the labels or classes included in the dataset to the fullest purity.

The decision tree model yielded an accuracy of 91.49% and after performing cross validation on the model, an accuracy of 91.79% was obtained.

```
print(classification_report(y_test, y_predDT))
```

```
              precision    recall  f1-score   support

          -1       0.91      0.90      0.91      1014
           1       0.92      0.93      0.92      1197

    accuracy                           0.91      2211
   macro avg       0.91      0.91      0.91      2211
weighted avg       0.91      0.91      0.91      2211
```

```
confusion_matrix(y_test,y_predDT)
```

```
array([[ 913,  101],
       [  87, 1110]], dtype=int64)
```

5.5 Random Forest

Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in a forest of trees.

The random forest model with 100 estimators and a maximum depth of 2 using the gini criterion gave the best accuracy, which was 92.45%

	n_estimators	max_depth	criterion	cross_val_score	accuracy
0	100	2	gini	[0.9163369135104579, 0.9214245336348219, 0.906...	0.924469
1	100	2	entropy	[0.9129353233830846, 0.9222071460877431]	0.918137
2	50	5	gini	[0.9249208502939846, 0.9224332881049299]	0.914971
3	50	5	entropy	[0.9206241519674355, 0.9305744007236545]	0.919946

```
Evaluating the Algorithm:

oobscore: 0.9139529624604251
cross val: [0.91633691 0.92142453 0.90616167 0.93159977 0.91742081]

accuracy: 0.924468566259611

Confusion Matrix:
 [[ 879  135]
 [  32 1165]]
```

5.6 Model Comparison

The table below shows the accuracy obtained from each model.

	Model	Test Score
2	SVM	0.953595
1	KNN	0.947172
0	Logistic Regression	0.925373
4	Random Forest	0.924469
3	Decision Tree	0.917867

Model SVM (Polynomial) gave the best accuracy with a value of 95.36% and decision tree had the least accuracy among the models with a value of 91.79%.

SECTION 6

CONCLUSION

The purpose of this project was to analyse the data regarding various websites and come up with the best-suited classification model to identify legitimate websites. Phishing attacks have increased exponentially over the years and this has led to many victims who have been manipulated by phishers impersonating a trusted person and companies. Implementing Machine Learning algorithms on websites can help the user identify legitimate websites from phishing. Hence the prediction models need to be accurate to correctly classify the websites.

All the models that were built had an accuracy above 90%. The performances of the models were further estimated using Cross Validation as well. SVM using kernel trick as polynomial had the best accuracy among the models (95.3%). This was followed by KNN with k=15 (94.7%), Logistic Regression (92.5%), and Random Forest (92.4%). Decision Tree model had the least accuracy among them all (91.7%).

SECTION 7

REFERENCES

1. McCluskey L., Mohammad R., Thabtah F. (2015) Phishing Websites Features. University of Huddersfield Repository.

2. V. Gomes, J. Reis and B. Alturas, "Social Engineering and the Dangers of Phishing," 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), 2020, pp. 1-7, doi: 10.23919/CISTI49556.2020.9140445.

3. Basnet R., Mukkamala S., Sung A.H. (2008) Detection of Phishing Attacks: A Machine Learning Approach. In: Prasad B. (eds) Soft Computing Applications in Industry. Studies in Fuzziness and Soft Computing, vol 226. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-77465-5_19

4. Jakobsson, Markus. (2005). Modeling and Preventing Phishing Attacks. 89. 10.1007/11507840_9.

5. Miyamoto D., Hazeyama H., Kadobayashi Y. (2009) An Evaluation of Machine Learning-Based Methods for Detection of Phishing Sites. In: Köppen M., Kasabov N., Coghill G. (eds) Advances in Neuro-Information Processing. ICONIP 2008. Lecture Notes in Computer Science, vol 5506. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-02490-0_66