Phishing Website Classification

Abhilash Antony

Introduction…

A phishing website is a fraudulent website designed to mimic a legitimate one to deceive users into divulging sensitive information, such as usernames, passwords, credit card numbers, or other personal details.

Phishing websites often look very similar to legitimate websites, using similar logos, fonts, and layouts to create a sense of trust.

These sites may use URLs that look similar to a legitimate site's URL but have slight variations, such as misspellings or additional characters. For example, a phishing site may use **www.faceb00k.com** instead of **www.facebook.com**.

The main goal of a phishing website is to steal sensitive information for malicious purposes, including Financial Gain, Identity Theft, Account Compromise and Distribution of Malware.

Such phishing websites can be persuasive, but they can be detected by carefully checking the URL for slight misspellings or unusual characters and ensuring the site uses HTTPS for secure communication. Look out for poor grammar or spelling mistakes, be cautious of urgent messages prompting quick action, and always verify the source by manually entering the website address instead of clicking on suspicious links.

Not all users are observant enough to distinguish phishing websites from genuine ones. So, it is imperative to build a model that can accurately classify phishing sites, helping protect users from online threats and safeguarding their sensitive information.

In this project, we explore a dataset related to a classification task that aims to differentiate between phishing and legitimate activities based on various features. The dataset consists of multiple predictor variables, each representing a distinct characteristic or behaviour that may indicate whether an activity is phishing or legitimate. The target variable, labelled as 'Result', has two classes: 'Phishing' and 'Legitimate'.

To accomplish this task, we will build and evaluate several machine learning models. The primary goal is to develop a model that accurately predicts the class of each observation based on the provided features. We will start by pre-processing the data, followed by the implementation of various classifiers such as Logistic Regression, Support Vector Machine (SVM), Random Forest, and Gradient Boosting. We will also perform hyperparameter tuning to optimize model performance. The effectiveness of these models will be evaluated using metrics like accuracy, precision, recall, and F1 score to determine the best model for this classification task. The detailed report, dataset and codes can be accessed from my GitHub Repository.

About the Dataset…

The dataset contains 2,456 entries and 31 columns, each representing different features of a website that are helpful in distinguishing between phishing and legitimate websites. Below is a list of the columns along with a brief explanation of each:

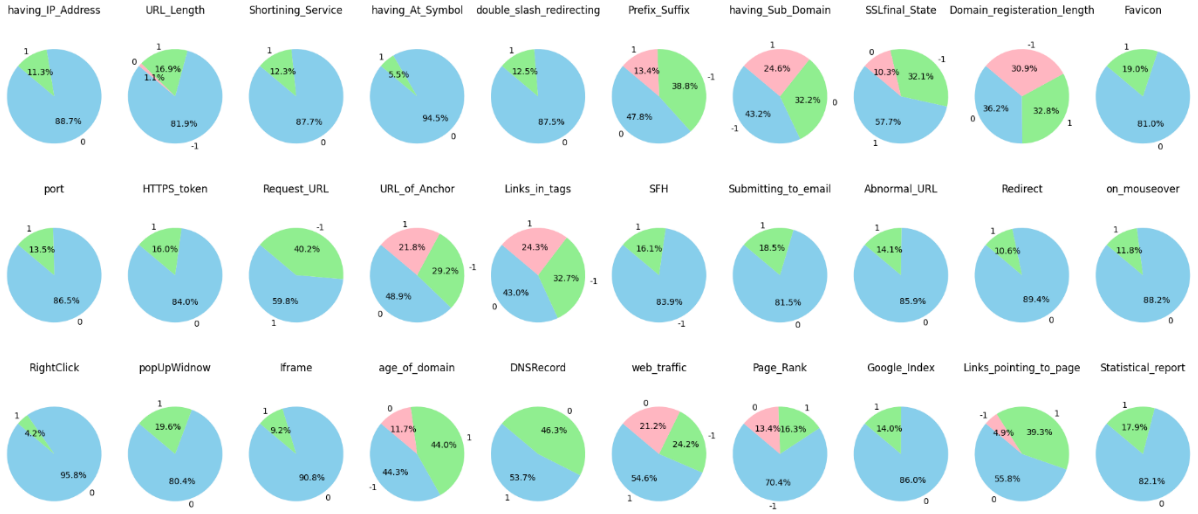
1. **having_IP_Address**: Indicates whether the website URL contains an IP address.
2. **URL_Length**: The length of the URL.
3. **Shortining_Service**: Whether the URL uses a shortening service
4. **having_At_Symbol**: Indicates the presence of an '@' symbol in the URL, which is a common trait of phishing sites.
5. **double_slash_redirecting**: If the '//' appears more than once in the URL path.
6. **Prefix_Suffix**: Indicates if the domain name contains hyphens.
7. **having_Sub_Domain**: Tracks whether the domain has multiple sub-domains.
8. **SSLfinal_State**: The SSL certificate validity of the website.
9. **Domain_registeration_length**: The registration length of the domain.
10. **Favicon**: Whether the favicon is loaded from the same domain.
11. **port**: Checks whether uncommon ports are open on the website.
12. **HTTPS_token**: Whether the URL contains "https" as part of the domain name.
13. **Request_URL**: Whether the majority of the objects are from the same domain.
14. **URL_of_Anchor**: Percentage of anchor tags linking to different domains.
15. **Links_in_tags**: Presence of links in meta, script, and link tags.
16. **SFH**: Server Form Handler, indicating where the form data is submitted.
17. **Submitting_to_email**: Whether the form sends user data to an email instead of a server.
18. **Abnormal_URL**: Whether the website's URL is abnormal.
19. **Redirect**: Number of times a user is redirected.
20. **on_mouseover**: If the status bar changes on mouse hover.
21. **RightClick**: Indicates if right-click is disabled on the website.
22. **popUpWidnow**: Whether pop-up windows are used.
23. **Iframe**: Checks if the website uses iframe tags.
24. **age_of_domain**: The age of the domain.
25. **DNSRecord**: Whether DNS records exist for the domain.
26. **web_traffic**: Website traffic rank based on popularity.
27. **Page_Rank**: Google's page rank (High or Low).
28. **Google_Index**: Whether the website is indexed by Google.
29. **Links_pointing_to_page**: Number of links pointing to the page.
30. **Statistical_report**: Classification of websites based on statistical data
31. **Result**: The target label indicating whether the website is legitimate or phishing.

These columns provide valuable insights into website behavior and characteristics, allowing machine learning models to effectively detect phishing sites.

Exploratory Analysis…

The dataset contains information relevant to identifying phishing websites. Upon initial inspection using *df.info*() and *df.describe(),* it was confirmed that the dataset includes several categorical and numerical features that can be utilized for model building. A further check for missing values using *df.isnull*().sum() revealed that there are no missing values in any of the columns. This completeness ensures that the data is ready for analysis and model development without the need for imputation or data cleaning. Moving forward, we will use this dataset to build a machine learning model to classify websites as either phishing or legitimate based on the provided features.

The dataset displays a moderate class imbalance, with approximately 55.46% of the instances labelled as '0' (legitimate) and 44.54% labelled as '1' (phishing). This imbalance is not severe, indicating that the dataset has a relatively even distribution of both classes, which allows for building a machine learning model without significant bias towards either class.



Key Findings from the data distribution:

- Most legitimate websites (~88.7%) do not use an IP address in their URL, whereas 11.3% do, which could indicate phishing.
- Approximately 81.9% of websites have a normal URL length, with abnormal lengths potentially signalling phishing attempts.
- About 12.3% of websites use URL shortening services, which can sometimes be associated with phishing.
- Around 5.5% of websites contain an '@' symbol, a possible indicator of phishing websites.
- A majority of websites (57.7%) have a valid SSL certificate, while 32.1% do not, which could be a potential sign of phishing.
- There is a mix of domain registration lengths, reflecting a combination of phishing and legitimate sites.
- About 81% of sites have a legitimate favicon, which can help determine the legitimacy of a website.
- Around 85.9% of websites do not have abnormal URLs, making this a critical feature in phishing detection.
- Higher values in page rank, Google index, and web traffic are typically associated with legitimate websites.

Model Building and Evaluation…

We will now proceed to train our model using a variety of classifiers to identify which one performs best for this task. By evaluating multiple models, including logistic regression, SVM, Random Forest, we can determine the most effective approach for distinguishing between legitimate and phishing websites based on the features in our dataset. This comprehensive evaluation will help us select the model that provides the highest accuracy and robustness for our phishing detection system.

1. Using Logistic Regression:

Logistic Regression is simple, effective, and efficient for binary classification tasks. It’s designed for problems with two classes, like 'Phishing' and 'Legitimate'. The results are straightforward and you can easily see how features affect the outcome. It provides the probability of each class, which helps in understanding the confidence of predictions. It’s fast to train and works well with large datasets. It works well with categorical features when they are encoded properly.

The evaluation metrics for the logistic regression model indicate strong performance in identifying phishing websites. The model achieved an accuracy of 0.95, meaning it correctly classified 95% of the cases. The precision of 0.96 suggests that when the model predicts a website is phishing, it is correct 96% of the time. The recall of 0.92 indicates that the model correctly identifies 92% of all actual phishing websites. The F1 Score, which balances precision and recall, is 0.94, demonstrating that the model effectively balances the ability to identify phishing websites while minimizing false positives. Overall, these metrics suggest that the logistic regression model is a reliable choice for this task.

2. Using Support Vector Classifier:

Support Vector Machines can be more suitable for this task. SVM works well with high-dimensional data. It finds the optimal boundary (hyperplane) that best separates classes, which can lead to better accuracy. SVM can use different kernel functions (like RBF) to handle non-linear relationships between features, potentially capturing complex patterns that Logistic Regression might miss. SVM's kernel functions allow it to model more complex decision boundaries than the linear boundary of Logistic Regression. In some cases, SVM can achieve higher accuracy by better separating the classes, especially in cases with non-linear relationships.

The evaluation metrics for the SVM (Support Vector Machine) model indicate excellent performance in identifying phishing websites. With an accuracy of 0.97, the model correctly classified 97% of the cases. The precision of 0.97 shows that when the model predicts a website is phishing, it is correct 97% of the time. The recall of 0.96 indicates that the model successfully identifies 96% of all actual phishing websites. The F1 Score of 0.96, which balances both precision and recall, further demonstrates that the SVM model maintains a strong balance between detecting phishing websites and minimizing false positives. These results suggest that the SVM model is slightly more effective than Logistic Regression for our task.

3. Using Random Forest Classifier:

Random Forest is well-suited for this task due to its ability to handle both numerical and categorical features, manage large datasets, and deal with missing values. Its ensemble approach, which combines multiple decision trees, improves predictive accuracy and reduces the risk of overfitting compared to Logistic Regression, which might not capture complex patterns in the data. Unlike SVM, which may struggle with very large datasets and requires careful tuning of kernel parameters, Random Forest is more robust and less sensitive to hyperparameter settings. Additionally, Random Forest provides feature importance insights, which can be valuable for understanding the factors driving predictions. Overall, its versatility, robustness, and ease of use make Random Forest a strong candidate, potentially outperforming both Logistic Regression and SVM in various scenarios.

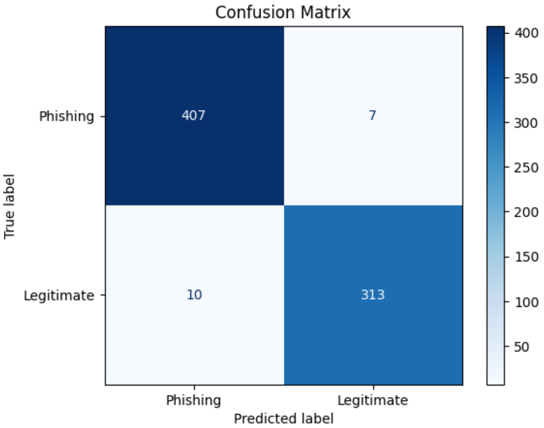
The evaluation metrics for the Random Forest model show exceptional performance in detecting phishing websites. With an accuracy of 0.97, the model correctly classifies 97% of cases. The precision of 0.97 indicates that when the model predicts a website is phishing, it is correct 97% of the time. Similarly, a recall of 0.97 shows that the model successfully identifies 97% of all actual phishing websites. The F1 Score of 0.97, which balances precision and recall, confirms the model's effectiveness in maintaining a high detection rate while minimizing false positives. Compared to the Logistic Regression and SVM models, the Random Forest model provides a slight edge, achieving consistently high scores across all metrics, making it the best-performing model among the three.

4. Random Forest with Optimized Parameters:

To further enhance the performance of the Random Forest classifier, I performed some hyperparameter tuning. This process involved adjusting the model's parameters to find the optimal combination that maximizes its accuracy and generalization capability on the dataset.

This process involved fitting the Random Forest model 1,620 times across different combinations of hyperparameters using 5-fold cross-validation. The best hyperparameters identified through this search were:

- **n_estimators**: 100 (number of trees in the forest)
- **max_depth**: None (no limit on the depth of each tree)
- **max_features**: 'log2' (logarithm of the number of features is considered at each split)
- **min_samples_leaf**: 1 (minimum number of samples required to be at a leaf node)
- **min_samples_split**: 2 (minimum number of samples required to split an internal node)



The evaluation metrics for the refined Random Forest model with the tuned hyperparameters reveal impressive performance. The model achieved a precision of 0.98 for both the legitimate (class 0) and phishing (class 1) classes. This indicates that the model is highly accurate in distinguishing between the two classes, minimizing false positives. The recall scores are also high, with 0.98 for class 0 and 0.97 for class 1. This demonstrates that the model effectively identifies the majority of true instances for both classes, ensuring that phishing attempts are largely detected.

The F1 scores, which combine precision and recall, are 0.98 for class 0 and 0.97 for class 1. These values reflect a strong balance between precision and recall, showing that the model performs well across both metrics. Additionally, the overall accuracy of 0.98 signifies that the model correctly classifies 98% of the samples in the dataset.

Conclusions…

Overall, the final Random Forest model with tuned hyperparameters offered the best performance compared to the models tested so far. Its superior accuracy, precision, recall, and F1 scores highlight its effectiveness in detecting phishing websites and make it the most reliable choice among the models evaluated.

The high accuracy and balanced precision and recall indicate that this model is highly reliable for real-world phishing detection. By correctly classifying 98% of the instances, it significantly enhances the security of users against phishing attempts.

Its robustness, coupled with fine-tuning, ensures accurate detection, minimizing the risk of false positives or missed phishing attempts. Thus, it emerges as the top choice for this classification task, offering both reliability and efficiency in protecting users from online threats.

By utilizing this model, organizations and individuals can better safeguard sensitive information from phishing attacks, enhancing the overall security of their digital activities.

However, it's important to consider that while the Random Forest model shows exceptional results, real-world applications should also account for possible limitations and ensure continuous evaluation and updates to address evolving threats.