**Detecting Phishing Websites Using Machine Learning**

Group-20

1. Mohammad Aftab Hussain Shaik
2. Ram Nisanth Simhadri
3. William Nsambu
4. Jai Kumar Aragonda

**Motivation**

The problem we aim to address is the pervasive and ever-evolving threat of phishing attacks on the internet. Phishing attacks have become a significant concern due to their potential to compromise personal and financial information, resulting in substantial financial losses, identity theft, and significant data breaches. The importance of tackling this problem is multi-fold:

1. Protection of Personal Data: Phishing attacks can lead to the unauthorized access of sensitive personal information, such as login credentials, financial details, and identity-related data. Preventing such attacks is crucial for safeguarding individual privacy and security.

2. Financial Security: Phishing attacks are a direct threat to financial security. They can result in the loss of money, unauthorized access to bank accounts, and credit card fraud, causing substantial financial harm to victims.

3. Data Breach Prevention: By thwarting phishing attacks, we can mitigate the risk of data breaches. Data breaches not only harm individuals but also impact organizations' reputations and financial stability.

4. Preventing Identity Theft: Phishing attacks can be instrumental in identity theft. Preventing these attacks helps in reducing the instances of identity theft, which can have long-lasting consequences for victims.

5. Cybersecurity Education: Addressing phishing attacks raises awareness about cybersecurity practices and helps educate users about the dangers they might encounter online.

**Task Definition:**

The technical aspects of the problem we are addressing involve the development of a machine learning-based system for the prediction of phishing websites. The primary technical steps are as follows:

1. Data Collection: Gather a dataset of website URLs, labelling them as either legitimate or phishing. This dataset serves as the foundation for training and testing the machine learning models.

2. Feature Extraction: Extract relevant features from the URLs and their associated content. These features may include URL structure, content analysis, page rank, and other parameters that can help differentiate between legitimate and phishing websites.

3. Machine Learning Models: Implement and train various machine learning algorithms, such as Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), Random Forest, Decision Tree, XGBoost Classifier, and Naïve Bayes. These models are trained using the labeled dataset to learn patterns and characteristics of phishing websites.

4. Model Evaluation: Assess the performance of these models using appropriate metrics such as accuracy, precision, recall, and F1-score. This step is essential for selecting the best-performing algorithm.

5. User-Facing Front End: Develop a user-friendly interface (front end) where users can input website links.

6. Prediction: Extract parameters from the user-entered website links and feed them into the trained machine learning model to predict whether the entered website is legitimate or a potential phishing website.

7. Output Display: Display the model's prediction to the user, indicating whether the website is safe or poses a risk of phishing.

Concrete Input/Output Pairs:

For this problem, here are concrete input/output pairs:

Input: A website URL (e.g., "http://www.example.com")

Output: "Legitimate Website" (if the URL is from a legitimate source) or "Phishing Website" (if the URL is identified as malicious)

This process is intended to empower users to make informed decisions about whether to interact with a given website, thereby enhancing their online safety and security. It's a proactive approach to tackle a critical issue in today's digital landscape.

Approach:

1. Baseline:

The baseline for our project involves implementing a simple and commonly used method for identifying phishing websites. This initial approach will serve as a foundation for evaluating the performance of our more advanced machine learning models and algorithms. By comparing the results of our proposed models with this baseline, we aim to ascertain the effectiveness and superiority of our approach.

*Baseline Methodology:*

The baseline methodology for predicting phishing websites is based on analysing a set of predefined features commonly associated with malicious websites. This includes but is not limited to:

* **URL Analysis**: Inspecting the structure and components of the URL, such as the presence of suspicious domains, subdomains, or IP addresses.
* **Content Analysis**: Scanning the content of the web page for known phishing keywords, scripts, or embedded objects that are indicative of malicious intent.
* **HTTP Header Examination**: Checking for irregularities in the HTTP headers, including server responses, content type, and encoding.
* **SSL Certificate Validation**: Verifying the legitimacy of SSL certificates to determine whether the website is using a secure connection.
* **Blacklist Comparison**: Cross-referencing the website with known phishing website blacklists.

By having this baseline in place, we ensure that we have a basic, functional system for identifying potential phishing websites. This will help us rigorously assess the effectiveness of our advanced machine learning techniques and their ability to outperform a simpler approach.

2. Proposed Methods:

We plan to implement the following techniques:

- Machine Learning Models: As described in the problem statement, we will use various machine learning algorithms such as Logistic Regression, KNN, SVC, Random Forest, Decision Tree, XGBoost Classifier, and Naïve Bayes for predicting phishing websites. Each model will be trained and tested on the dataset.

- Feature Engineering: We will extract and engineer features from website URLs and content. These features will help the machine learning models distinguish between legitimate and phishing websites.

- User-Facing Front End: Develop a user-friendly interface (front end) using web technologies (e.g., Flask and HTML) for users to input website links.

- Model Integration: Integrate the trained machine learning models with the front end to enable real-time predictions based on user-provided URLs.

Extra-Credit Ideas:

- Ensemble Methods: We can explore ensemble learning techniques to combine the predictions of multiple machine learning models, potentially enhancing overall accuracy.

- Active Learning: Implement active learning strategies to improve the model's performance incrementally by focusing on difficult-to-classify instances.

**Evaluation Metric:**

We will evaluate our method using common machine learning metrics, including accuracy, precision, recall, and F1-score. The choice of metric depends on the problem domain, and we will provide context to understand what constitutes a good or bad score.

The baseline's performance will be compared with these metrics to ascertain if our proposed methods provide a significant improvement.

**Data:**

Our dataset consists of website URLs, categorized as either legitimate or phishing. It is essential for training and testing our models.

We have taken this dataset from Kaggle. The provided dataset includes 11430 URLs with 87 extracted features. The dataset is designed to be used as benchmarks for machine learning-based phishing detection systems. Features are from three different classes: 56 extracted from the structure and syntax of URLs, 24 extracted from the content of their correspondent pages, and 7 are extracted by querying external services. The dataset is balanced, it contains exactly 50% phishing and 50% legitimate URLs.

**Experiments:**

Quantitative experiments will include:

- Cross-validation: We'll use techniques like k-fold cross-validation to assess the model's robustness.

- Hyperparameter tuning: Experiment with different model configurations to optimize performance.

**Qualitative Analysis:**

Qualitative analysis will involve assessing the usability and user-friendliness of the front end. We will gather user feedback and conduct user testing to evaluate the effectiveness of our interface.

**Plan:**

**Team Roles:**

- Role 1: Responsible for data collection, preprocessing, and feature engineering.

- Role 2: Focusing on the implementation of machine learning models.

- Role 3: Developing the user-facing front end.

- Role 4: Handling experiment setup and data analysis.

**Timeline:**

-We are aiming to complete the project by this month end (Oct 31st)

This timeline is subject to adjustments as needed during the project's progress to ensure its success and qualitys