Phishing Website Detection Using Machine Learning

Phishing attacks remain one of the most prevalent and dangerous cyber threats, tricking unsuspecting users into revealing sensitive information through fraudulent websites. As phishing techniques grow more sophisticated, machine learning has emerged as a powerful tool for detecting and preventing these attacks. This presentation explores how machine learning algorithms can be leveraged to identify phishing websites with high accuracy, providing an additional layer of protection for users and organizations. We'll examine the data collection process, feature selection, model development, and evaluation metrics used in building effective phishing detection systems.

by Stephen Sam

Understanding Phishing Attacks

Phishing attacks are a form of social engineering that exploits human psychology to trick victims into divulging confidential information. These attacks typically involve creating fake websites that mimic legitimate ones, often with subtle differences that can be difficult for users to detect. The goal is to capture login credentials, financial information, or other sensitive data.

Traditional methods of phishing prevention, such as user awareness training and blacklisting known phishing URLs, have limitations. Machine learning offers a more dynamic and adaptive approach to identifying potential threats.

1. **User Receives Phishing Email**

   Attacker sends a deceptive email containing a link to a fraudulent website.

2. **User Clicks Malicious Link**

   The victim is directed to a convincing replica of a legitimate website.

3. **Sensitive Information Entered**

   User unwittingly submits login credentials or other sensitive data.

4. **Attacker Captures Information**

   The phisher collects the submitted data for fraudulent use.

Data Collection and Preparation

To develop an effective machine learning model for phishing detection, a diverse and representative dataset is crucial. This project utilized a combination of legitimate URLs from the University of New Brunswick dataset and phishing URLs from PhishTank, an open-source service. A total of 10,000 URLs were collected, evenly split between legitimate and phishing examples.

Data preparation involves cleaning the collected URLs, removing duplicates, and ensuring a balanced representation of both classes. This step is critical for training a model that can generalize well to new, unseen examples.

Legitimate URLs

5,000 URLs randomly selected from the UNB dataset, representing a wide range of legitimate websites across various domains and industries.

Phishing URLs

5,000 URLs obtained from PhishTank, regularly updated to include the latest phishing attempts and techniques used by attackers.

Data Cleaning

Removal of duplicate entries, normalization of URL formats, and verification of labels to ensure data quality and consistency.

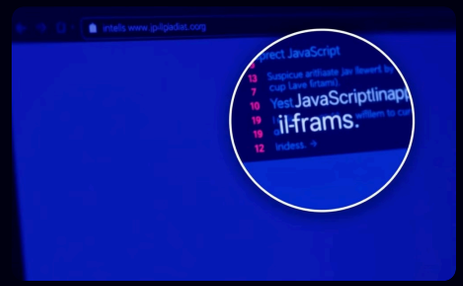
Feature Extraction and Selection

Address Bar Features

Includes IP address presence, URL length, use of URL shortening services, and presence of suspicious symbols or subdomains.

Domain Features

Examines DNS records, domain age, website traffic, and the expiration date of the domain.



HTML & JavaScript Features

Analyzes webpage content for suspicious elements like iframes, disabled right-clicks, and status bar customization.

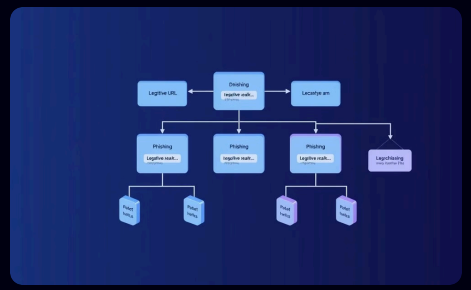
Feature Engineering

Creation of additional derived features to capture complex patterns and relationships within the data.

Feature extraction is a critical step in developing an effective phishing detection model. The features selected for this project fall into three main categories: Address Bar based Features, Domain based Features, and HTML & JavaScript based Features. These features were chosen based on their ability to distinguish between legitimate and phishing websites.

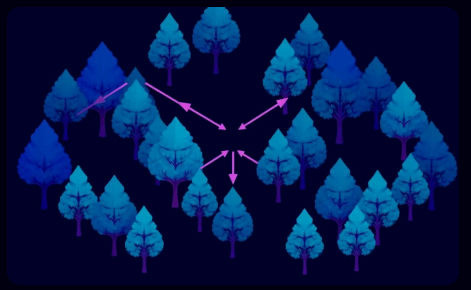
By carefully selecting and extracting these features, we create a rich representation of each URL that machine learning algorithms can use to make accurate predictions. The feature extraction process involves analyzing various aspects of the URL structure, domain properties, and webpage content.

Machine Learning Models



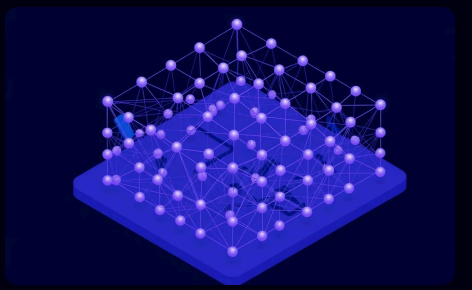
Decision Tree

A simple yet interpretable model that makes decisions based on a series of feature-based rules.



Random Forest

An ensemble method that combines multiple decision trees to improve accuracy and reduce overfitting.



Multilayer Perceptrons

A type of artificial neural network capable of learning complex, non-linear relationships in the data.

XGBoost

An advanced implementation of gradient boosting, known for its high performance and efficiency.

This project explores several supervised machine learning models to classify URLs as either phishing or legitimate. The chosen models represent a diverse range of algorithms, each with its own strengths and characteristics. By comparing multiple models, we can identify the most effective approach for phishing detection.

The models are trained on the prepared dataset, using the extracted features as input. Each model learns to recognize patterns associated with phishing websites and make predictions on new, unseen examples.

Model Evaluation and Comparison

To assess the performance of each machine learning model, we use a combination of evaluation metrics, with a primary focus on accuracy. The dataset is split into training and testing sets, allowing us to measure how well each model generalizes to unseen data. This evaluation process helps identify the most effective model for phishing website detection.

The results show that XGBoost outperforms other models, achieving the highest accuracy on both training and test datasets. This suggests that XGBoost is particularly well-suited for capturing the complex patterns associated with phishing websites.

Model	Train Accuracy	Test Accuracy
XGBoost	0.868	0.857
Multilayer Perceptrons	0.866	0.854
AutoEncoder	0.810	0.810
Random Forest	0.820	0.809
Decision Tree	0.816	0.803
SVM	0.806	0.786

Practical Implementation

The successful development of a machine learning model for phishing detection opens up several avenues for practical implementation. One of the most effective ways to leverage this technology is through the creation of browser extensions or plugins. These tools can provide real-time protection for users by analyzing URLs as they browse the web.

Additionally, the model can be integrated into existing security infrastructure, such as email filters or network monitoring systems, to provide an additional layer of protection against phishing attacks at an organizational level.

1

Model Integration

Incorporate the trained XGBoost model into a browser extension or security tool.

2

URL Analysis

Extract features from visited URLs in real-time for classification.

3

Risk Assessment

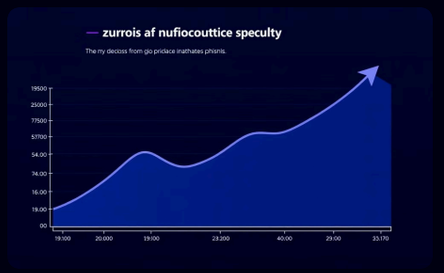
Determine the likelihood of a website being phishing based on model predictions.

4

User Alert

Notify users of potential phishing attempts and provide safety recommendations.

Future Directions and Conclusion



Continuous Model Updates

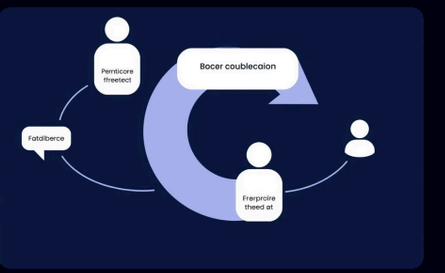
Implement systems for regular retraining of models with new phishing examples to adapt to evolving attack techniques.

Explainable AI

Develop methods to interpret model decisions, providing insights into why a website is classified as phishing.

Cross-Platform Integration

Expand the implementation to cover various platforms and devices, including mobile browsers and email clients.



User Feedback Loop

Incorporate user feedback mechanisms to improve model accuracy and reduce false positives over time.

The field of phishing detection using machine learning is rapidly evolving, with new techniques and challenges emerging regularly. Future work in this area could focus on improving model accuracy through advanced feature engineering, exploring deep learning architectures, or developing ensemble methods that combine multiple models for enhanced performance.

In conclusion, this project demonstrates the effectiveness of machine learning techniques in detecting phishing websites. By leveraging a diverse set of features and comparing various algorithms, we've developed a robust system capable of identifying potential threats with high accuracy. As phishing attacks continue to pose a significant risk to online security, the integration of machine learning-based detection methods into everyday browsing experiences will play a crucial role in protecting users and organizations from these sophisticated threats.