Phishing detection using machine learning

Group Members:

Wasatullah Nizamani 18K-1263

Zeerak Ibrahim 18K-1197

Jaafar Bin Farooq 18K-1294

Abstract:

Our project aimed to create a ML model that detects phishing websites and helps people in staying away from such websites.

To train our model we got our dataset from UCI machine learning repository. A collection of website URLs for 11000 websites. Each sample has 30 website parameters and a class label identifying it as a phishing website or not (1 or -1). Of which following are the numbers of Legitimate and Phishing Websites.

Legitimate websites: 4898

Phishing Websites: 6157

Feature Selection:

There were total of 30 features. Features which were more related to the Result were selected Correlation of each feature with result was calculated Features with correlation between -0.02 to 0.02 were dropped.

Total Features used : 24

Features dropped: 6
['Iframe','Favicon','popUpWidnow','index','RightClick','Submitting_to_email']

Null values in selected features were dropped

Following data distribution was done:

Dataset was split in 80:20 ratio

80 % for training

20% for testing

We used following ML models: Decision tree, Random Forest K Nearest Neighbor

SVM , XLGBoost

Accuracies:

XGB Classifier 96.715328

Random Forest: 96.669708

Decision Tree: 96.122263

SVM: 94.799270

KNN : 94.479927

XGBoost had the highest accuracy and the best performance while KNN was the worst in terms of accuracy and performance.

Introduction:

Phishing attacks are used to steal the confidential information of a user. Fraud websites appear like genuine websites with the logo and graphics of the genuine website. This project aims to detect fraud or phishing websites using machine learning techniques. This problem is important because naive users don't have an idea of phishing websites and they fall prey to bad guys and their information is stolen when they enter their login password details or maybe credit card credentials on fake phishing websites.

Summarize related work:

There are detailed accounts on how a phishing website conventionally looks. They have certain properties which are hidden from the eye of a layman and can only be detected through an in-depth study of these properties. These are the properties on which various websites were tested and experts lay down a pattern that distinguishes phishing websites from legitimate ones.

https://www.temjournal.com/content/102/TEMJournalMay2021_947_953.pdf

Summarize the methodology that you have followed for evaluating your implementation:

Following ML models were used Decision tree, Random Forest K Nearest Neighbor, SVM, XLGBoost

XGB Classifier 96.715328

Random Forest: 96.669708

Decision Tree: 96.122263

SVM: 94.799270

KNN : 94.479927

XGBoost had the highest accuracy and the best performance while KNN was the worst in terms of accuracy and performance. While Random Forest was the 2nd Best.

List contributions of your project:

Wasatullah Nizamani: Data gathering + presentation + report

Zeerak Ibrahim : Train ML Models + report

Jaafar Bin Farooq: Data preprocessing + report

II. Existing System (7%)

There is no existing system however our project can predict if you provide a URL whether it is a phishing website or not with over 96 per cent accuracy.

III. Related Work (5%)

https://www.temjournal.com/content/102/TEMJournalMay2021_947_953.pdf.

IV. Adversary Model (5%)

It is the same.

V. System Design (20%)

We knew it was a classification problem initially. We just had to pick the machine learning model for our project. We tried and tested the models which we thought were best suited. We then compared and choose the best one.

VI. System Implementation (20%)

We used python libraries such as Pandas and NumPy for data preprocessing. We also used advanced python libraries such as keras, sklearn and tensorflow for machine learning to import and implement the machine learning models. For plotting the results on a graph for better visualization, we used matplotlib.

VII. System Evaluation

1. (A. Evaluation Methodology) (8%)

The models are evaluated, and the considered metric is accuracy

2. (B. Results of the evaluation) (6%)

Model	**Accuracy(%age)**
XGB Classifier:	96.715328
Random Forest:	96.669708
Decision Tree:	96.122263
SVM:	94.799270
KNN :	94.479927

3. (C. Discussion of the evaluation results) (4%)

XGBoost model gives better performance, KNN model gives the worst performance.

VII. Discussion (10%)

Through this project, one can know a lot about the phishing websites and how they are differentiated from legitimate ones. If we have the features based on which our model has been trained we can predict if the website was a phishing website or a legitimate website. However we currently have not applied any method through which we can extract the features from the url. This implementation cannot be used by everyone, only people with enough knowledge about these features can use it to predict the class.

The motivation behind this project is to be able to train machine learning models on the dataset created to predict phishing websites. So we can use the model to predict the phishing websites to prevent phishing attacks. Due to phishing attacks any online users can be tricked into revealing confidential information, which the attacker can use in any fraudulent activity.

There are detailed accounts on how a phishing website conventionally looks. They have certain properties which are hidden from the eye of a layman and can only be detected through an in-depth study of these properties. These are the properties on which various websites were tested and experts lay down a pattern that distinguishes phishing websites from legitimate ones.

For evaluation each model was trained and tested on the exact same dataset. For every model the training accuracy,testing accuracy,confusion matrix,precision,recall and f1 score was calculated. The models were then evaluated and based on the testing accuracy the best model was selected.

Developers can further use this project by implementing ways to extract features from the url and inputting those features into our trained model. Which would make this easy to use for everyone.

VIII. Conclusion (5%)

Based on the results we can conclude that the XLGBoost Classifier obtained the best baseline accuracy for further comparison. Through this project, one can know a lot about the phishing websites and how they are differentiated from legitimate ones. This project can classify phishing websites from legitimate ones given there are enough features.

This project can be further improved by using a better dataset that consists of information of more and updated urls. Also by using GUI that inputs url and predicts whether it is a phishing website or a legitimate website.