High Level Design (LLD)

Phishing Domain Detection (Machine Learning)

By

Kavitha Narsapur

Abstract

Phishing is a type of fraud in which an attacker impersonates a reputable company or person in order to get sensitive information such as login credentials or account information via email or other communication channels. Phishing is popular among attackers because it is easier to persuade someone to click a malicious link that appears to be authentic than it is to break through a computer's protection measures. Phishing attacks are done via emails, text messages, or websites. Phishing websites have the same look as legitimate sites. However, their backend is designed to collect sensitive information that is inputted by the victim. Thus this project aims to detect phishing domains using machine learning models. Different models like Logistic Regression classifier, Decision Tree classifier, Random Forest classifier, K-Nearest Neighbor classifier, eXtreme Gradient Boosting classifier and Naïve Bayes classifier were built. Out of all the classifiers, Random forest classifier resulted in best accuracy of 97.16%and F1 score of 0.9717.

Introduction

Why this High-Level Design Document?

This High-level Design (HLD) Document indicates all the necessary steps that were carried out prior to building machine learning model as Data pre-processing, Exploratory data analysis, Feature selection and Data balancing. After data preparation, the document indicates which models were built and tested on test data and also describes model deployment.

Scope

The LLD documentation presents the structure of the system, such as the application architecture (layers), application flow (Navigation), and technology architecture. The LLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system. This software system will be a Web application. This system will be designed to detect phishing sites.

General Description

Introduction

Phishing domain detection is a technique used to detect whether a site is legitimate or not by using machine learning models.

Problem Statement

Phishing is a type of fraud in which an attacker impersonates a reputable company or person in order to get sensitive information such as login credentials or account information via email or other communication channels. Phishing is popular among attackers because it is easier to persuade someone to click a malicious link that appears to be authentic than it is to break through a computer's protection measures. Phishing attacks are done via emails, text messages, or websites. Phishing websites have the same look as legitimate sites. However, their backend is designed to collect sensitive information that is inputted by the victim. Thus this project aims to detect phishing domains using machine learning models.

Approach

This project involves data pre-processing like Data cleaning, Exploratory data analysis, Feature selection using Extra tree classifier, fixing data imbalance using SMOTE (Synthetic Minority Oversampling Technique), Model building and Model deployment. Different models like Logistic Regression classifier, Decision Tree classifier, Random Forest classifier, K-Nearest Neighbor classifier, eXtreme Gradient Boosting classifier and Naïve Bayes classifier were built and then tested on to test data.

Dataset overview

This data consists of a collection of legitimate as well as phishing website instances. Each website is represented by the set of features which denote, whether website is legitimate or not. Data can serve as an input for machine learning process. In this project the two variants of the Phishing dataset are presented.

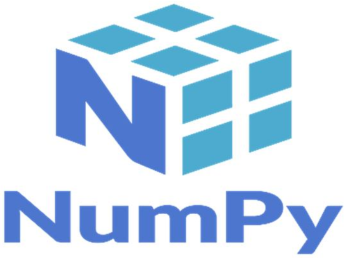
Full variant:
Total number of instances: 88,647
Number of legitimate website instances (labeled as 0): 58,000
Number of phishing website instances (labeled as 1): 30,647
Total number of features: 111

Small variant:
Total number of instances: 58,645
Number of legitimate website instances (labeled as 0): 27,998
Number of phishing website instances (labeled as 1): 30,647
Total number of features: 111

Further Improvements

The performance of machine learning models can be further enhanced by tuning various hyperparameters of the models.

Tools used


NumPy

pandas

matplotlib

seaborn

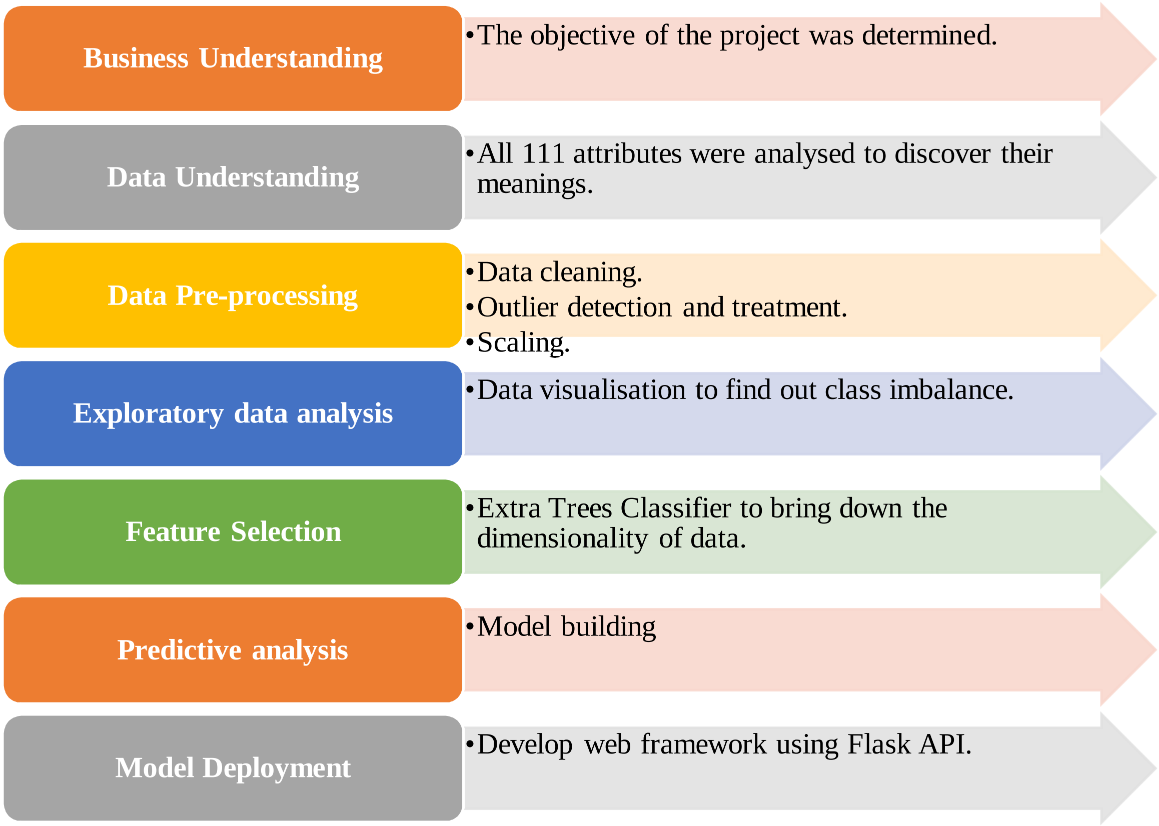
scikit
learn

Pickle

Flask

python™

Architecture



Business Understanding

- The objective of the project was determined.

Data Understanding

- All 111 attributes were analysed to discover their meanings.

Data Pre-processing

- Data cleaning.
- Outlier detection and treatment.
- Scaling.

Exploratory data analysis

- Data visualisation to find out class imbalance.

Feature Selection

- Extra Trees Classifier to bring down the dimensionality of data.

Predictive analysis

- Model building

Model Deployment

- Develop web framework using Flask API.

Conclusion

Out of all the tested models, Random forest classifier performed the best with highest accuracy of 97.16% and f1-score of 0.9717. Its highest accuracy means that the model is able to capture 97% of malicious websites correctly. It’s high accuracy and precision makes it an effective candidate for use in real-world scenarios.