High Level Document

Phishing Domain Detection

Project by – Tejas Muley

Content

1.Introduction

1.1 Why this High-Level Design Documentation

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project

List and describe the non-functional attributes like:

- Security
- Reliability
- Maintainability
- Portability
- Reusability
- Application compatibility
- Resource utilization
- Serviceability

1.2 Scope

This software system will be a web application, this system will be designed to predicts weather the website is real or malicious/fake based on the user's input in which there are several categories to fill in like the Length of the URL, Length of the Directory, Length of the file, Activation Domain Time and Expiration Domain Time.

Based on these features model will predict weather the website is real or fake. We make sure that all the given features should be available at that time in order to get the optimum utilization and earn maximum profits by the company.

2. General Description

2.1 Product Perspective

This Phishing Domain Detection System is a machine learning based model which will predict weather the website is real or malicious on the basis of user’s input values.

2.2 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.

The main goal is to predict whether the domains are real or malicious.

2.3 Proposed Solution

This system requires features like Length of URL, Length of Directory, Length of File, Activation Domain Time and Expiration Domain Time for phishing. Based on these features the system will predict weather the website is real or malicious.

2.4 Further Improvements

As the data is contains only values of URL, not the URL link, so our main aim is to complete this use case with machine learning algorithm as a best optimized solution, In future if we are expected to get URL sites and different categories, if needed we might use deep-learning algorithm to get best solution.

2.5 Data Requirements

Data requirements completely depend on our problem statement.

2.6 Tool Used

1) Flask 2) Pandas 3) Numpy 4) Sklearn 5) Matplotlib / Seaborn 6) Github 7) Visual Studio

8) Jupyter Notebook 9) pickle 10) Azure

2.7 Constraints

We will be using only few features (Selection of features using Feature Selection method).This system only predict weather the website is real or fake.

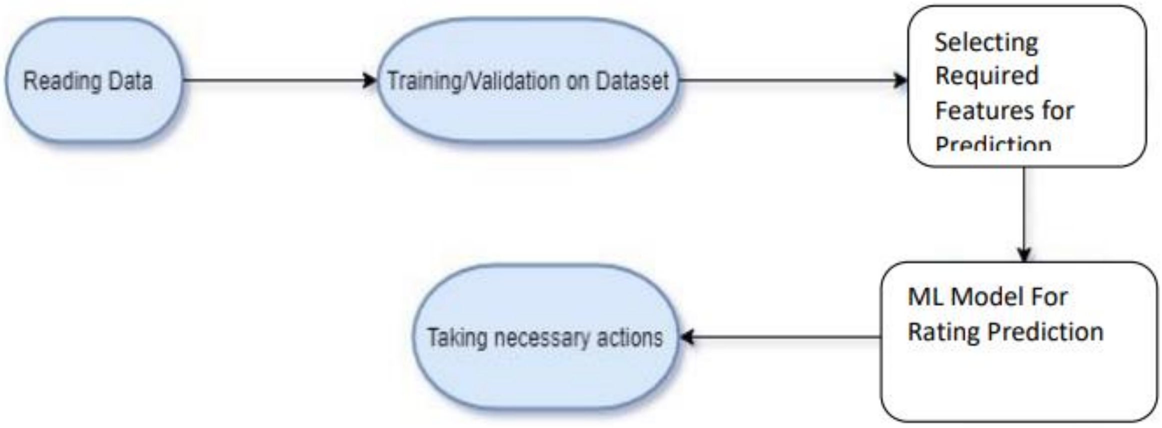
2.8 Assumptions

The values in dataset are scrapped using web scrapping already. Our role is only to apply various Data Pre-processing, Feature Selection, Feature Scaling, Model Building and Deployment of model using flask on Azure.

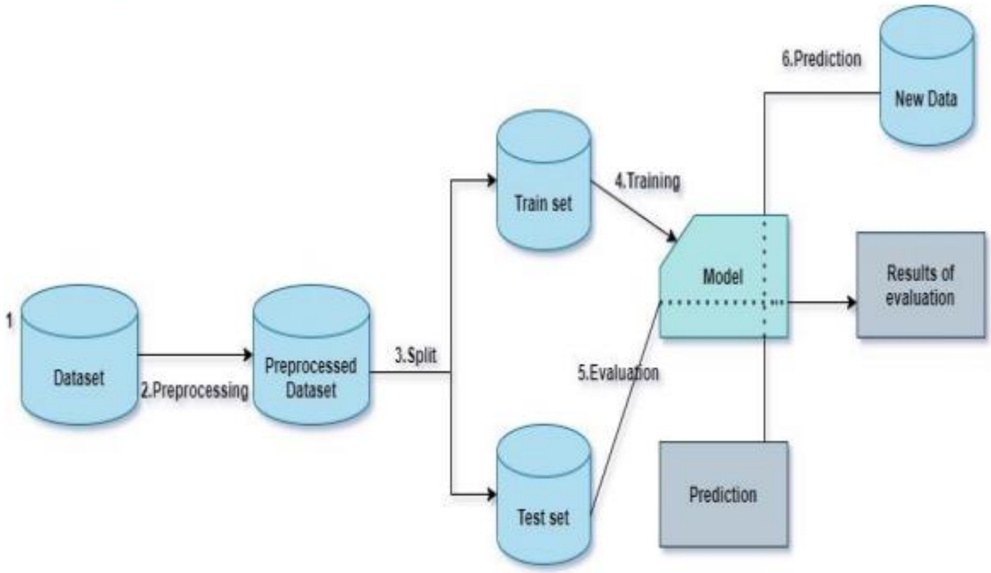
3.Design Details

3.1 Process Flow

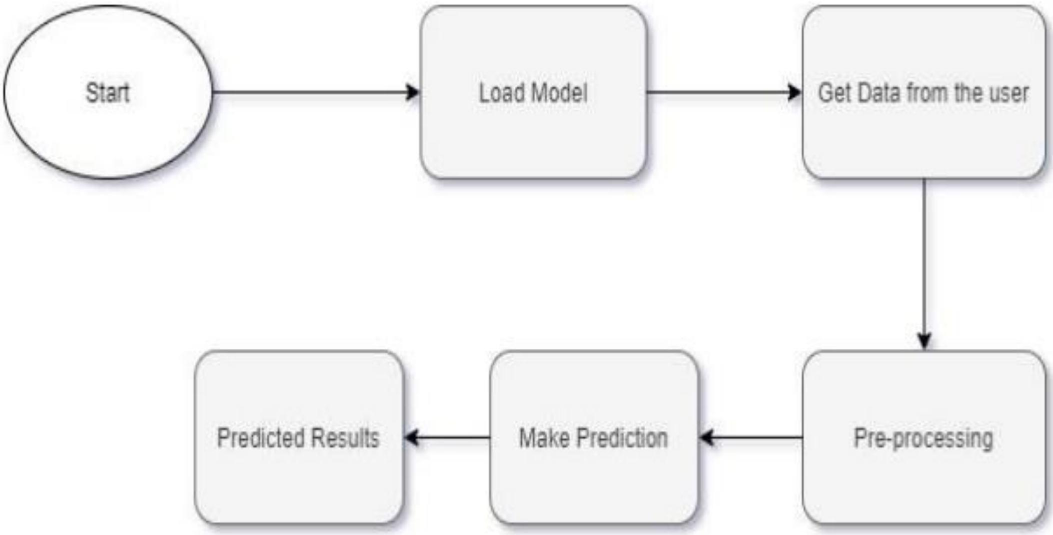
Based on the use-case, we will use a machine learning base model. Below is the process flow diagram is as shown below.


Reading Data
Training/Validation on Dataset
Selecting Required Features for Prediction
ML Model For Rating Prediction
Taking necessary actions

3.2 Model Training and Evaluation



3.3 Deployment Process



3.4 Event log

The system should log every event so that the user will know what process is running internally. Initial Step-step description:

- The system identifies at what step logging required
- The system should be able to log each and every system flow.
- Developer can choose logging method. We chose File logging.
- System should not hang as we have used file logging. Logging is used just because we can easily debug issues so logging is mandatory to do.

3.5 Error Handling

Should error be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage.

4. Performance

4.1 Reusability

The code written and the components used has the ability to be reused with no problems if there is similar problem statement.

4.2 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

4.3 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

4.4 Deployment

Deployment has been done using flask on cloud. We used Azure for cloud deployment.

5. Conclusion

This Phishing Domain Detection is used in-order to avoid phishing. According to the different parameters by the URL, we as an users should have an idea that how our personal information can leaked through phishing. This system helps to check whether the site is real or malicious, so that we can avoid and get alert either to provide information on that site or not.