PHISHING DOMAIN DETECTION

(Machine Learning)

LOW LEVEL DESIGN

iNeuron.ai

July 2023

INTRODUCTION

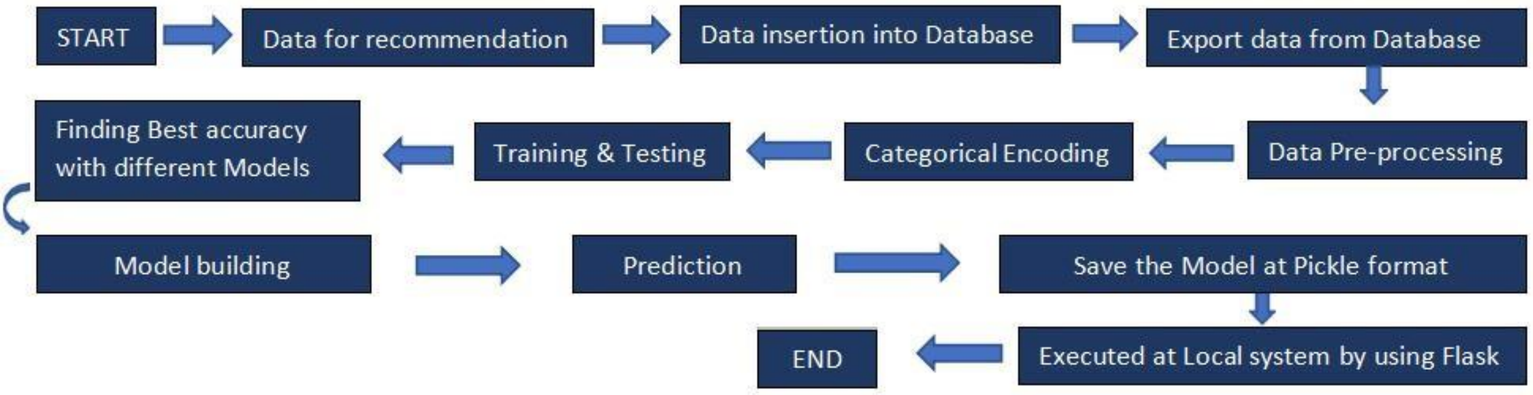
What is Low-Level design document -

- LLD stands for Low-Level Design Document. It is a document that describes the internal logical design of a program.
- In the case of phishing domain detection, the LLD would describe how the program would use the features of domain names (such as the domain name itself, the top-level domain, the registrar, etc.) to classify them as phishing or legitimate.
- The LLD would also describe the different modules of the program, and how they would interact with each other.
- The purpose of the LLD is to provide a detailed plan for the program, so that the programmer can directly code the program from the document.

SCOPE -

- LLD is a detailed description of how a software system will be implemented.
- It is a step-by-step process that starts with the high-level design of the system and then breaks it down into smaller and smaller components.
- Each component is then described in detail, including its data structures, algorithms, and interfaces.
- The goal of LLD is to provide a complete and accurate description of the software system so that it can be implemented without ambiguity.

ARCHITECTURE -


START
Data for recommendation
Data insertion into Database
Export data from Database
Data Pre-processing
Categorical Encoding
Training & Testing
Finding Best accuracy with different Models
Model building
Prediction
Save the Model at Pickle format
Executed at Local system by using Flask
END

ARCHITECTURE DESCRIPTION -

1. DATA DESCRIPTION-

The phishing dataset consists of two variants: a full variant and a small variant.

The full variant has 88,647 instances, of which 58,000 are legitimate websites and 30,647 are phishing websites. The full variant has 111 features, which are used to determine whether a website is legitimate or phishing.

The small variant has 58,645 instances, of which 27,998 are legitimate websites and 30,647 are phishing websites. The small variant also has 111 features.

Both variants of the phishing dataset can be used as input for machine learning algorithms to detect phishing websites.

The choice of which variant to use depends on the specific needs of the application. If accuracy is the most important factor, then the full variant should be used. However, if computational resources are limited, then the small variant may be a better choice. Here, we have used the full dataset for accuracy.

2. DATA INSERTION INTO DATABASE -

a) Database creation and connection-Create a database with name phishingDB. If the database is already created, open the connection to the database.

b) Table creation in the database.

c) Insertion of files in the table.

3. EXPORT DATA FROM DATABASE -

Data export from database: To be used for model training and data pre-processing, the data in a stored database is exported as a CSV file.

4. DATA PRE-PROCESSING -

The phishing dataset can be prepared for machine learning by converting the domain names to a structured format, creating new features, standardizing the features, encoding the categorical features, and splitting the dataset into a training set and a test set.

- Converting the domain names to a structured format makes it easier for the machine learning model to understand the data.

- Creating new features can help the machine learning model to learn more about the data and make better predictions.

- Standardizing the features ensures that all of the features have a similar scale, which makes it easier for the machine learning model to learn from the data.

- Encoding the categorical features converts categorical features into numerical values so that the machine learning model can understand them.

- Splitting the dataset into a training set and a test set allows the machine learning model to be trained and evaluated on different data.

5. ML ALGORITHM -

The best model is derived from classification using the entire ML procedure.

6. CATEGORICAL ENCODING -

Since none of the datasets supplied were categorical, they were all not required to be transformed.

7. TRAINING AND TESTING DATASET -

80% of the dataset has been trained in this case, and 20% has been tested.

8. FINDING ACCURACY WITH DIFFERENT MODEL -

All the supervised machine learning algorithm were used to classify the output such as Logistic regression, Decision tree, Random forest, Gradient boosting classifier, XG Boost Classifier, AdaBoost etc. and found accuracy with every models.

9. MODEL BUILDING -

Model building was done with the highest accuracy possible after accuracy was verified using various models, and the model was then saved in pickle format.

10. WEB FRAMEWORK -

It has been tested on a local machine using the flask API.

Conclusion -

This application runs on the internet. The user interface was created using Flask. We can look up a domain name and determine whether a link is harmful or safe to use.