Phishing Domain Detection

Objective:

Development of a predictive model for monitoring fraud URLs used for phishing purpose. The model will determine whether a URL is malicious one or legitimate one.

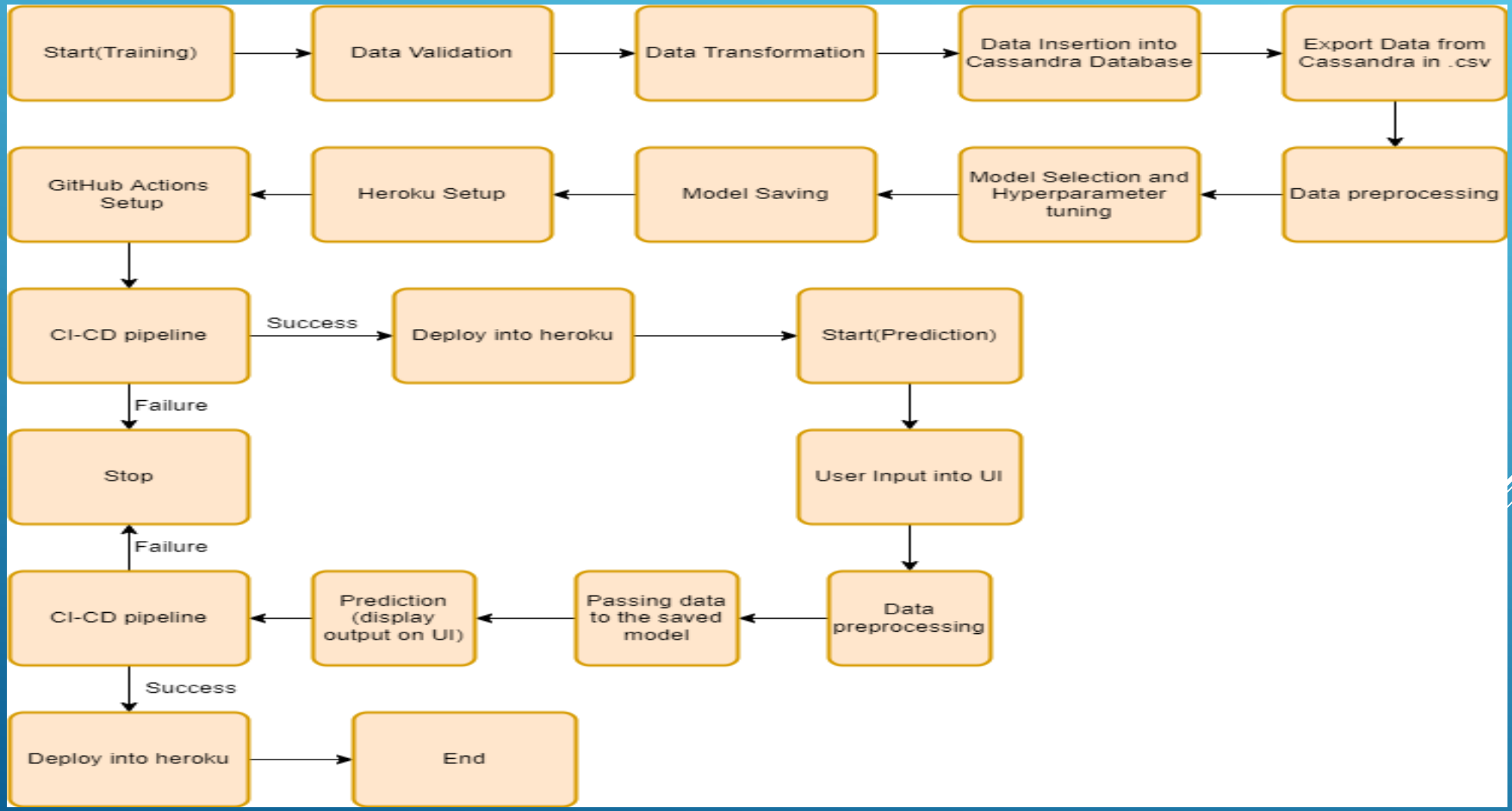
Benefits:

- Detection of upcoming frauds.
- Gives better insight of customers base.
- Helps in easy flow for managing resources.
- Manual inspection if fraud is identified .

Data Sharing Agreement :

- Number of Columns(74)
- Column names
- Column data type

Architecture



Data Validation and Data Transformation :

- Number of Columns – Validation of number of columns present in the files, and if it doesn't match then the file is moved to "Bad_Data_Folder.“
- Name of Columns - The name of the columns is validated and should be the same as given in the schema file. If not, then the file is moved to "Bad_Data_Folder".
- Data type of columns - The data type of columns is given in the schema file. It is validated when we insert the files into Database. If the datatype is wrong, then the file is moved to "Bad_Data_Folder".
- Null values in columns - If any of the columns in a file have all the values as NULL or missing, we discard such a file and move it to "Bad_Data_Folder".

Data Insertion in Database:

- Table creation :- Table name “phishing_training_good_raw_data" is created in the database for inserting the files. If the table is already present then new files are inserted in the same table.
- Insertion of files in the table - All the files in the "Good_Data_Folder" are inserted in the above-created table. If any file has invalid data type in any of the columns, the file is not loaded in the table

Model Training:

- Data Export from Db :

  The accumulated data from db is exported in csv format for model training

- Data Preprocessing
  - Performing EDA to get insight of data like identifying distribution , outliers ,trend among data etc.
  - Check for null values in the columns. If present impute the null values.
  - Encode the categorical values with numeric values.
  - Perform Standard Scalar to scale down the values.

we find the best model for the preprocessed Data. By using algorithms “SVC” "XGBoost“, “Random-Forest“, “Descision-Tree“, “Logistic-Regression“, “Naïve-Bayes“ For the preprocessed data we used all these algorithms with the hyper tunning. We calculate the AUC and accuracy scores for both models and select the model with the best score. Similarly, the model is selected for the data.The models is saved for use in prediction

Prediction:

- The user inputs the given fields from UI.
- We perform data pre-processing techniques on it.
- model is loaded and is used to predict the data .
- Once the prediction is done the output is displayed on the UI for the user to verify.

Q & A:

Q1) What's the source of data?

The The dataset available for this application is present on https://data.mendeley.com/datasets/72ptz43s9v/1

Q 2) What was the type of data?

The data was numerical values.

Q 3) What's the complete flow you followed in this Project?

Refer slide 5th for better Understanding

Q 4) After the File validation what you do with incompatible file or files which didn't pass the validation?

Files like these are moved to the Achieve Folder and a list of these files has been shared with the client and we removed the bad data folder.

Q 5) How logs are managed?

We are using different logs as per the steps that we follow in validation and modeling like File validation log , Data Insertion ,Model Training log , prediction log etc.

Q 6) What techniques were you using for data pre-processing?

- Removing unwanted attributes
- Visualizing relation of independent variables with each other and output variables
- Checking and changing Distribution of continuous values
- Removing outliers
- Cleaning data and imputing if null values are present.
- Converting categorical data into numeric values.
- Scaling the data

Q 7) How training was done or what models were used?

- Before diving the data in training and validation set we performed clustering over fit to divide the data into clusters.
- As per cluster the training and validation data were divided.
- The scaling was performed over training and validation data
- Algorithms like SVM , XGBoost were used based on the recall final model was used for each cluster and we saved that model .

Q 8) How Prediction was done?

The testing files are shared by the client .We Perform the same life cycle till the data is clustered .Then on the basis of cluster number model is loaded and perform prediction. In the end we get the accumulated data of predictions.