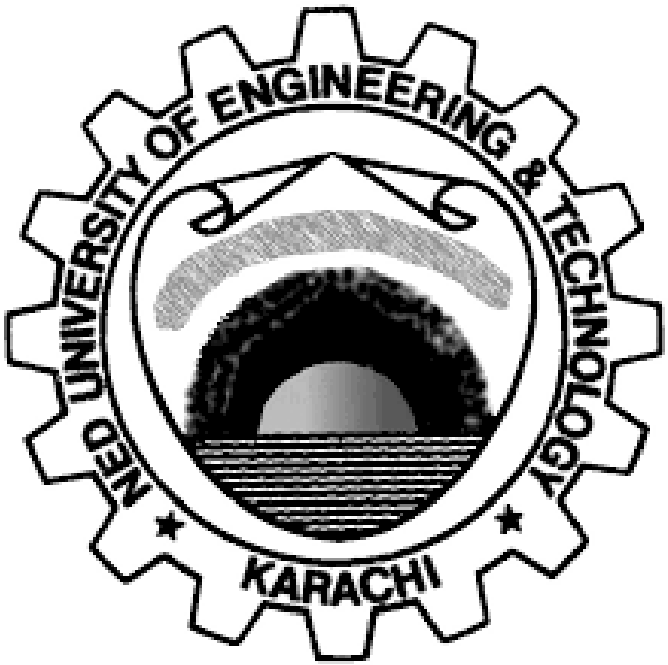
MACHINE LEARNING

CS-324

OPEN ENDED LAB



FRAUD DETECTION FOR TWITTER ACCOUNTS

NAME	ROLL NUMBER
Rayyan Sajid	CS-21034
Waliya Noor	CS-21043
Maaz Bin Tariq	CS-21112

NED University of Engineering and Technology

FRAUD DETECTION FOR TWITTER ACCOUNTS

INTRODUCTION

The primary objective of this project is to detect fraudulent Twitter accounts using machine learning algorithms. The dataset was sourced from Kaggle and includes various features that can help identify whether an account is fraudulent or not. The steps followed in this project include data collection, preprocessing, exploratory data analysis (EDA), feature engineering, model building, model evaluation, and the creation of a user-friendly interface for model interaction.

DATA COLLECTION

The dataset (twitter_human_bots_dataset) used in this project was obtained from Kaggle. It contains a comprehensive set of features about Twitter accounts, including user profile details, followers count, friends count, status count, average tweets per day and account age in days, etc. The dataset has 37438 records and 20 features, with a target variable indicating whether an account is fake or not.

DATA PREPROCESSING AND EXPLORATORY DATA ANALYSIS (EDA)

Data preprocessing involved cleaning the dataset by handling missing values and encoding categorical variables. Missing values were removed, and categorical variables were transformed into numerical values. Visualizations such as count plots and pair plots were used to understand the distributions and patterns in the data. EDA was conducted to gain deeper insights into the data. Heat maps were used to examine the distribution of variables, while a correlation matrix helped identify relationships between features.

MODEL BUILDING

Three machine learning algorithms were used to build predictive models: Logistic Regression, K-Nearest Neighbors (KNN), and Decision Trees. Each model was implemented using Python packages such as scikit-learn and also manually to understand the underlying mechanics.

1. Logistic Regression:
    - Implemented using scikit-learn's LogisticRegression class.
    - Manually implemented by computing the sigmoid function and gradient descent.
2. K-Nearest Neighbors (KNNs):
    - Implemented using scikit-learn's KNeighborsClassifier.
    - Manually implemented by calculating the Euclidean distance and majority voting.
3. Decision Trees:
    - Implemented using scikit-learn's DecisionTreeClassifier.
    - Manually implemented by constructing a tree based on information gain.

MODEL EVALUATION

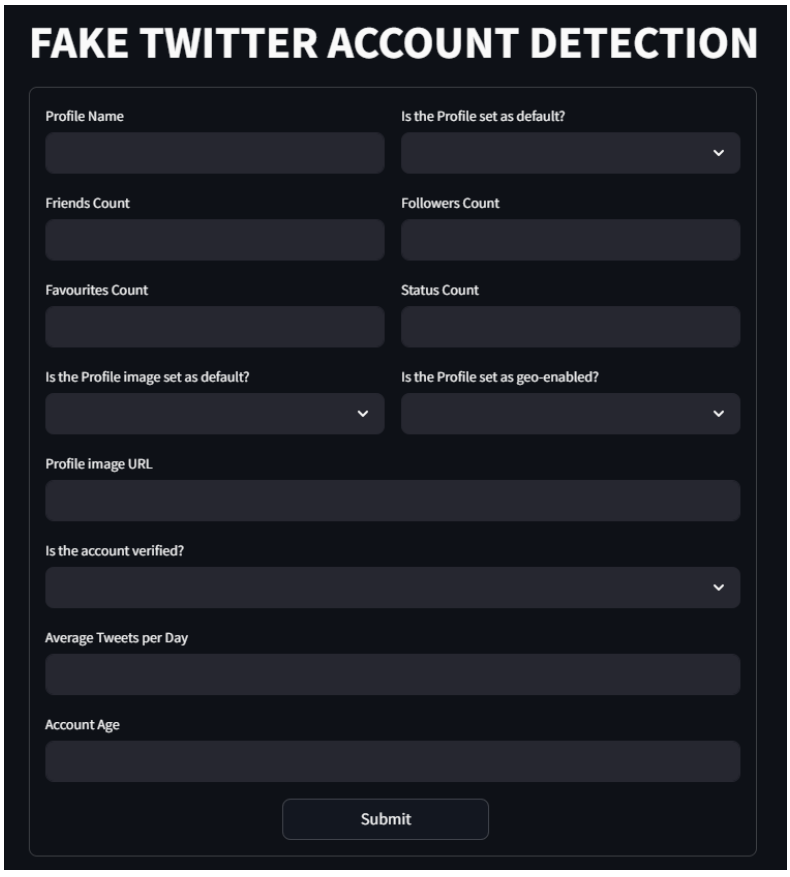
The models were evaluated using various metrics such as accuracy, precision, recall, F1 score, and ROC-AUC. The performance of each model was compared to identify the best-performing one.

1. Logistic Regression:
   Accuracy = 76% Precision = 74%
   Recall = 71% F1 Score = 72%
2. K-Nearest Neighbors (KNNs):
   Accuracy = 85% Precision = 84%
   Recall = 82% F1 Score = 83%
3. Decision Trees:
   Accuracy = 83% Precision = 80%
   Recall = 81% F1 Score = 81%

K-Nearest Neighbors emerged as the best-performing model based on these metrics.

USER-FRIENDLY INTERFACE

A user-friendly interface was created using Streamlit, allowing users to compare the performance of multiple algorithms and predict the likelihood of an account being fraudulent using an unknown sample. This interface enhances the usability and accessibility of the model.



Compare Algorithms

Choose the Algorithm for prediction:

Decision Trees × Logistic Regression × k-Nearest Neigh... ×

Enter the Number of neighbours:

5

Start Prediction

Decision Trees

IT'S NOT A FAKE ACCOUNT

Classification Report

	Human	Bot	accu
precision	0.8751	0.7333	0.8
recall	0.864	0.7519	0.8
f1-score	0.8695	0.7425	0.8
support	5,001	2,487	0.8

The accuracy is 82.678952991452 99%

Logistic Regression

IT'S NOT A FAKE ACCOUNT

Classification Report

	Human	Bot	accu
precision	0.7957	0.6779	0.7
recall	0.8698	0.5509	0.7
f1-score	0.8311	0.6078	0.7
support	5,001	2,487	0.7

The accuracy is 76.388888888888889%

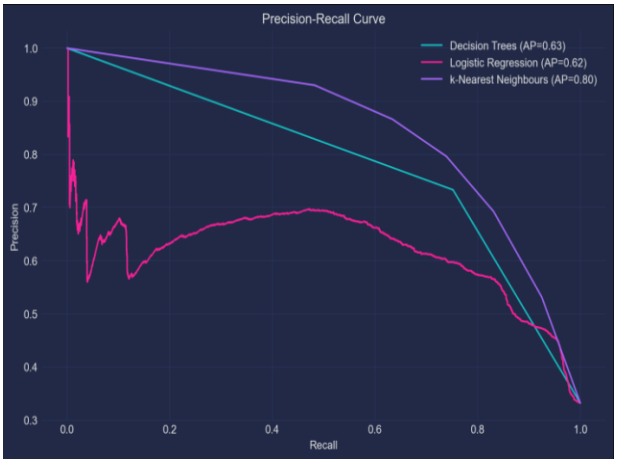
k-Nearest Neighbours

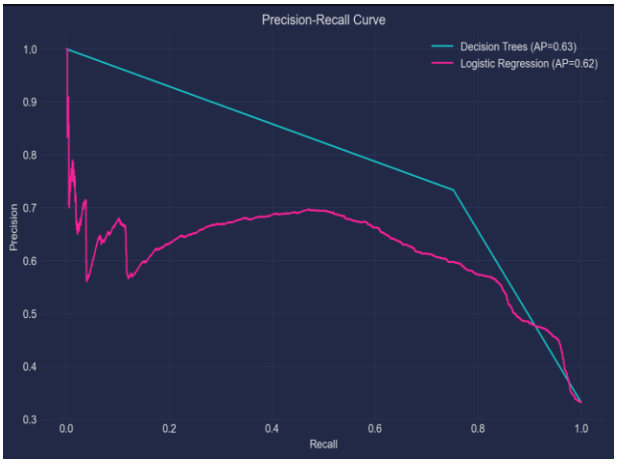
IT'S NOT A FAKE ACCOUNT

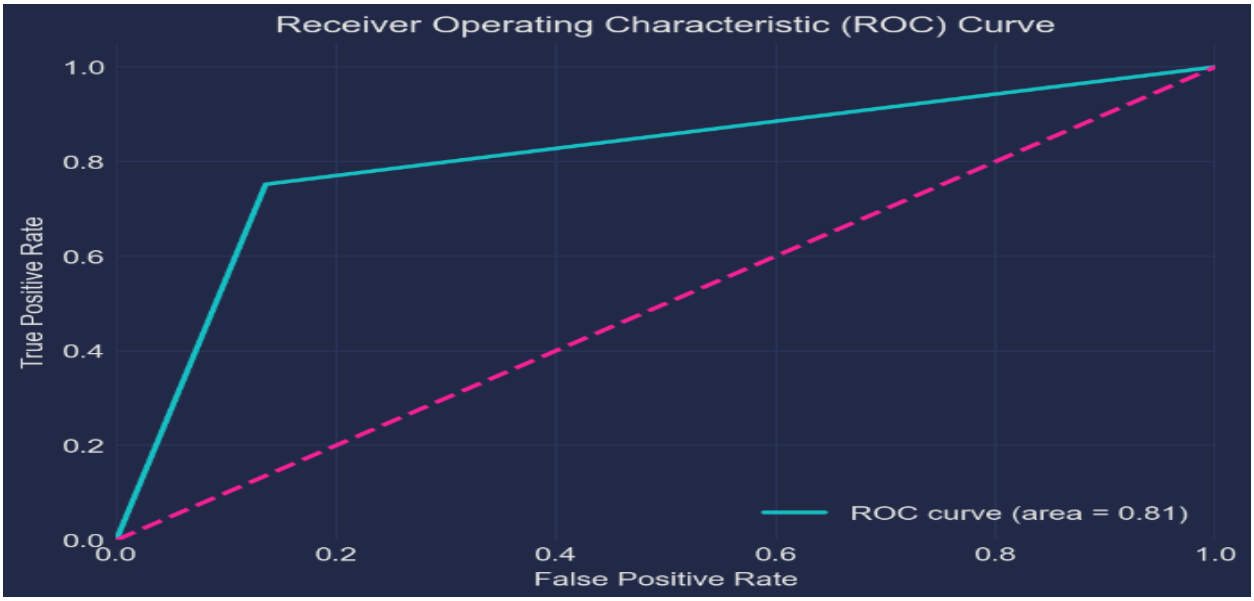
Classification Report

	Human	Bot	accu
precision	0.8747	0.7964	0.8
recall	0.906	0.739	0.8
f1-score	0.8901	0.7666	0.8
support	5,001	2,487	0.8

The accuracy is 85.056089743589 75%







The accuracy is 82.6789529914529 9%

Confusion Matrix

	0	1
0	4,321	680
1	617	1,870

Classification Report

	Human	Bot	accuracy	macro avg	weighted avg
precision	0.8751	0.7333	0.8268	0.8042	0.828
recall	0.864	0.7519	0.8268	0.808	0.8268
f1-score	0.8695	0.7425	0.8268	0.806	0.8273
support	5,001	2,487	0.8268	7,488	7,488

CONCLUSION

The project successfully demonstrated the application of machine learning algorithms to detect fraudulent Twitter accounts. Future work could explore more advanced algorithms and larger datasets for improved performance.