Collect data for good and bad sites

Clean the scraped data (HTMLprocessing.py)

Extract features (TF-IDF) (HTMLprocessing.py)

XGBoost selects features itself

Split data into training and testing sets (Train.py)

Train the models (Train.py)

Evaluate the models: accuracy, precision, recall, F1 score, ROC and DET plot, tSNE (Train.py)

Use models to predict for new data (verify.py)

Generate\_features.py

def main - add\_all\_sites, make\_train\_test, feature\_names.append, save to features.csv

HTMLprocessing.py

A screenshot of a computer

Description automatically generated A screenshot of a computer program

Description automatically generated A screenshot of a computer program

Description automatically generated

Train.py

Class Model

def feature\_importance

def xgboost - train, predict, save

param = {'max\_depth': 3, 'eta': 1, 'verbosity': 1, 'objective': 'binary:logistic'}

def sklearn – fit, predict, save

single tree, random forest, neural net

def main

add\_all\_sites, make\_train\_test, predict using sklearn and xgboost, print metrics and plots