In [1]:
```
pip install pandas scikit-learn
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandas in c:\programdata\anaconda3\lib\site-package
s (2.0.3)
Requirement already satisfied: scikit-learn in c:\programdata\anaconda3\lib\site-p
ackages (1.3.0)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\programdata\anaconda3
\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\programdata\anaconda3\lib\site-p
ackages (from pandas) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\programdata\anaconda3\lib\site
-packages (from pandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in c:\programdata\anaconda3\lib\site-
packages (from pandas) (1.24.3)
Requirement already satisfied: scipy>=1.5.0 in c:\programdata\anaconda3\lib\site-p
ackages (from scikit-learn) (1.11.1)
Requirement already satisfied: joblib>=1.1.1 in c:\programdata\anaconda3\lib\site-
packages (from scikit-learn) (1.2.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\programdata\anaconda3\li
b\site-packages (from scikit-learn) (2.2.0)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-packa
ges (from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [4]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score
```

In [5]:
```
data = pd.read_csv('data.csv')
```

In [23]:
```
# Replace labels
data['label'] = data['label'].replace({'good': 'benign', 'bad': 'malicious'})
```

In [33]:
```
data
```

Out[33]:

	url	label	features
0	diaryofagameaddict.com	malicious	{'url_length': 22, 'num_digits': 0, 'num_speci...
1	espdesign.com.au	malicious	{'url_length': 16, 'num_digits': 0, 'num_speci...
2	iamagameaddict.com	malicious	{'url_length': 18, 'num_digits': 0, 'num_speci...
3	kalantzis.net	malicious	{'url_length': 13, 'num_digits': 0, 'num_speci...
4	slightlyoffcenter.net	malicious	{'url_length': 21, 'num_digits': 0, 'num_speci...
...	...	...	...
420459	23.227.196.215/	malicious	{'url_length': 15, 'num_digits': 11, 'num_spec...
420460	apple-checker.org/	malicious	{'url_length': 18, 'num_digits': 0, 'num_speci...
420461	apple-iclods.org/	malicious	{'url_length': 17, 'num_digits': 0, 'num_speci...
420462	apple-uptoday.org/	malicious	{'url_length': 18, 'num_digits': 0, 'num_speci...
420463	apple-search.info	malicious	{'url_length': 17, 'num_digits': 0, 'num_speci...

420464 rows × 3 columns

In [7]:
```
def extract_url_features(url):
    return {
        'url_length': len(url),
        'num_digits': sum(c.isdigit() for c in url),
        'num_special_chars': sum(c in '!@#$%^&*()-_=+[{]}|;:",<.>/?' for c in url),
        'https_present': int('https' in url)
    }
```

In [8]:
```
# Apply feature extraction
data['features'] = data['url'].apply(extract_url_features)
features_df = pd.DataFrame(data['features'].tolist())
```

In [10]:
```
# Define the features (X) and target variable (y)
X = features_df
y = data['label']  # Assuming 'label' column contains 'malicious' or 'benign'
```

In [11]:
```
# Split the data into training and test sets (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
```

In [12]:
```
# Initialize the classifier
clf = RandomForestClassifier(n_estimators=100, random_state=42)
```

In [13]:
```
# Train the classifier
clf.fit(X_train, y_train)
```

Out[13]:
```
RandomForestClassifier(random_state=42)
```

In [14]:
```
# Make predictions on the test set
y_pred = clf.predict(X_test)
```

In [15]:
```
# Evaluate the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
Accuracy: 0.862081267168492
              precision    recall  f1-score   support

         bad       0.76      0.33      0.46     14964
        good       0.87      0.98      0.92     69129

    accuracy                           0.86     84093
   macro avg       0.81      0.65      0.69     84093
weighted avg       0.85      0.86      0.84     84093
```

In [16]:
```
# Load the new unlabeled dataset
unlabeled_data = pd.read_csv('filtered_file.csv')
```

In [17]:
```
unlabeled_data
```

Out[17]:

	url
0	https://www.google.com
1	https://www.youtube.com
2	https://www.facebook.com
3	https://www.baidu.com
4	https://www.wikipedia.org
...	...
450171	http://ecct-it.com/docmmmnn/aptgd/index.php
450172	http://faboleena.com/js/infortis/jquery/plugin...
450173	http://faboleena.com/js/infortis/jquery/plugin...
450174	http://atualizapj.com/
450175	http://writeassociate.com/test/Portal/inicio/I...

450176 rows × 1 columns

In [18]:
```
# Extract features from the new URLs
unlabeled_data['features'] = unlabeled_data['url'].apply(extract_url_features)
unlabeled_features_df = pd.DataFrame(unlabeled_data['features'].tolist())
```

In [25]:
```
# Predict labels
predictions = clf.predict(unlabeled_features_df)
unlabeled_data['predicted_label'] = predictions
```

In [26]:
```
# Replace predicted labels
unlabeled_data['predicted_label'] = unlabeled_data['predicted_label'].replace({'goo
```

In [27]:
```
unlabeled_data
```

Out[27]:

	url	features	predicted_label
0	https://www.google.com	{'url_length': 22, 'num_digits': 0, 'num_speci...	malicious
1	https://www.youtube.com	{'url_length': 23, 'num_digits': 0, 'num_speci...	malicious
2	https://www.facebook.com	{'url_length': 24, 'num_digits': 0, 'num_speci...	malicious
3	https://www.baidu.com	{'url_length': 21, 'num_digits': 0, 'num_speci...	malicious
4	https://www.wikipedia.org	{'url_length': 25, 'num_digits': 0, 'num_speci...	benign
...	...	...	...
450171	http://ecct-it.com/docmmmnn/aptgd/index.php	{'url_length': 43, 'num_digits': 0, 'num_speci...	benign
450172	http://faboleena.com/js/infortis/jquery/plugin...	{'url_length': 159, 'num_digits': 21, 'num_spe...	benign
450173	http://faboleena.com/js/infortis/jquery/plugin...	{'url_length': 147, 'num_digits': 20, 'num_spe...	benign
450174	http://atualizapj.com/	{'url_length': 22, 'num_digits': 0, 'num_speci...	benign
450175	http://writeassociate.com/test/Portal/inicio/I...	{'url_length': 143, 'num_digits': 9, 'num_spec...	benign

450176 rows × 3 columns

In [29]:
```
# If you have true labels
true_labels = pd.read_csv('urldata.csv')
true_labels['label'] = true_labels['label'].replace({'good': 'benign', 'bad': 'mali
unlabeled_data = pd.merge(unlabeled_data, true_labels, on='url')
```

In [31]:
```
# Calculate accuracy
if 'label' in unlabeled_data.columns:
    accuracy = accuracy_score(unlabeled_data['label'], unlabeled_data['predicted_la
    print(f'Accuracy on unlabeled data: {accuracy:.2f}')
else:
    print("True labels are not available for accuracy calculation.")
```

```
Accuracy on unlabeled data: 0.07
```