Spam filtering

Spam filtering in email is the process of identifying and filtering out unwanted, irrelevant, or malicious emails. This is typically done through machine learning techniques, where models are trained to classify emails as spam or not spam based on various features. Python using the Naive Bayes algorithm, one of the common methods for spam filtering. This example uses the `sklearn` library to train a spam filter with a sample dataset, then tests it with new emails to determine if they're spam.

Code:

```
pip install sklearn pandas

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.feature_extraction.text import CountVectorizer

from sklearn.naive_bayes import MultinomialNB

from sklearn.metrics import accuracy_score, classification_report


# Sample email dataset with labels (1 for spam, 0 for non-spam)

data = {

  "text": [

    "Win a free iPhone! Click here to claim your prize!",

    "Hello, let's schedule a meeting for tomorrow.",

    "Congratulations! You've won a lottery. Claim your reward now!",

    "Can we discuss the project proposal later?",

    "Free money! Click to get your cash prize!",

    "Let's grab lunch tomorrow."

  ],

  "label": [1, 0, 1, 0, 1, 0]

}


# Convert data to a DataFrame

df = pd.DataFrame(data)


# Split the dataset into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(df["text"], df["label"], test_size=0.3,
random_state=42)
```

```
# Vectorize the email text data
vectorizer = CountVectorizer()
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)

# Train a Naive Bayes model
model = MultinomialNB()
model.fit(X_train_vec, y_train)

# Test the model with the test set
y_pred = model.predict(X_test_vec)

# Calculate accuracy and display classification report
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy * 100:.2f}%")
print(classification_report(y_test, y_pred))

# Test with a new email
new_email = ["Get a free trip to Hawaii!"]
new_email_vec = vectorizer.transform(new_email)

# Predict if it's spam or not
is_spam = model.predict(new_email_vec)[0]
print(f"Is the new email spam? {'Yes' if is_spam else 'No'}")
```

OUTPUT:

```
Accuracy: 100.00%
Precision    Recall  F1-Score
0         1.00     1.00    1.00
1         1.00     1.00    1.00
Is the new email spam? Yes
```