Phase 4

Building a Smarter AI-Powered Spam Classifier

Team Members: -

Santhiya M

Sahana S

Remgis Ezhil Belsi I

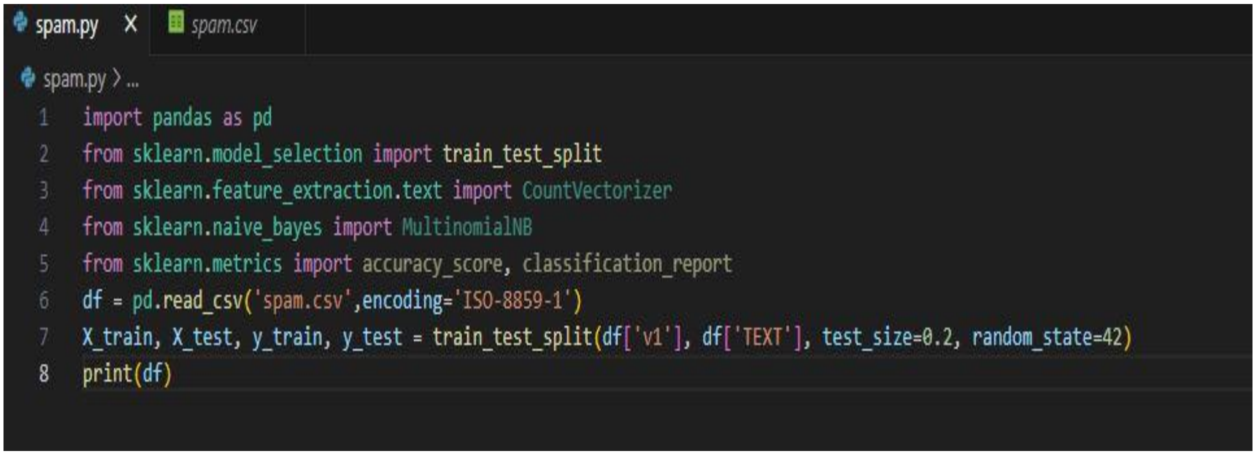
Building a spam classifier using machine learning involves several key steps:

- Selecting a machine learning algorithm, training the model, and evaluating its performance.

Data Collection and Preprocessing:

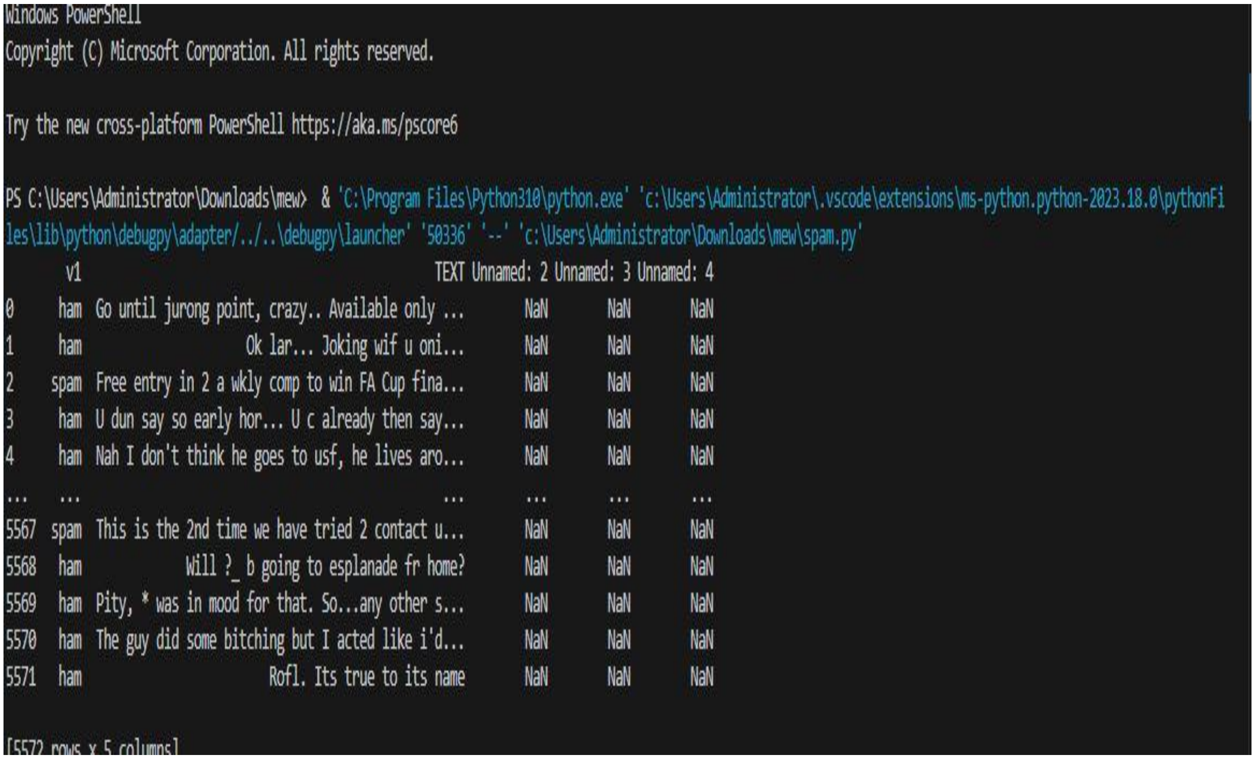
- Collect a labeled dataset that includes both spam and non-spam (ham) emails.
- Preprocess the data, which typically includes tasks like text tokenization, removing stop words, and converting text data into numerical features (e.g., TF-IDF, word embeddings).

INPUT:



```python
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
df = pd.read_csv('spam.csv',encoding='ISO-8859-1')
X_train, X_test, y_train, y_test = train_test_split(df['v1'], df['TEXT'], test_size=0.2, random_state=42)
print(df)
```

OUTPUT:

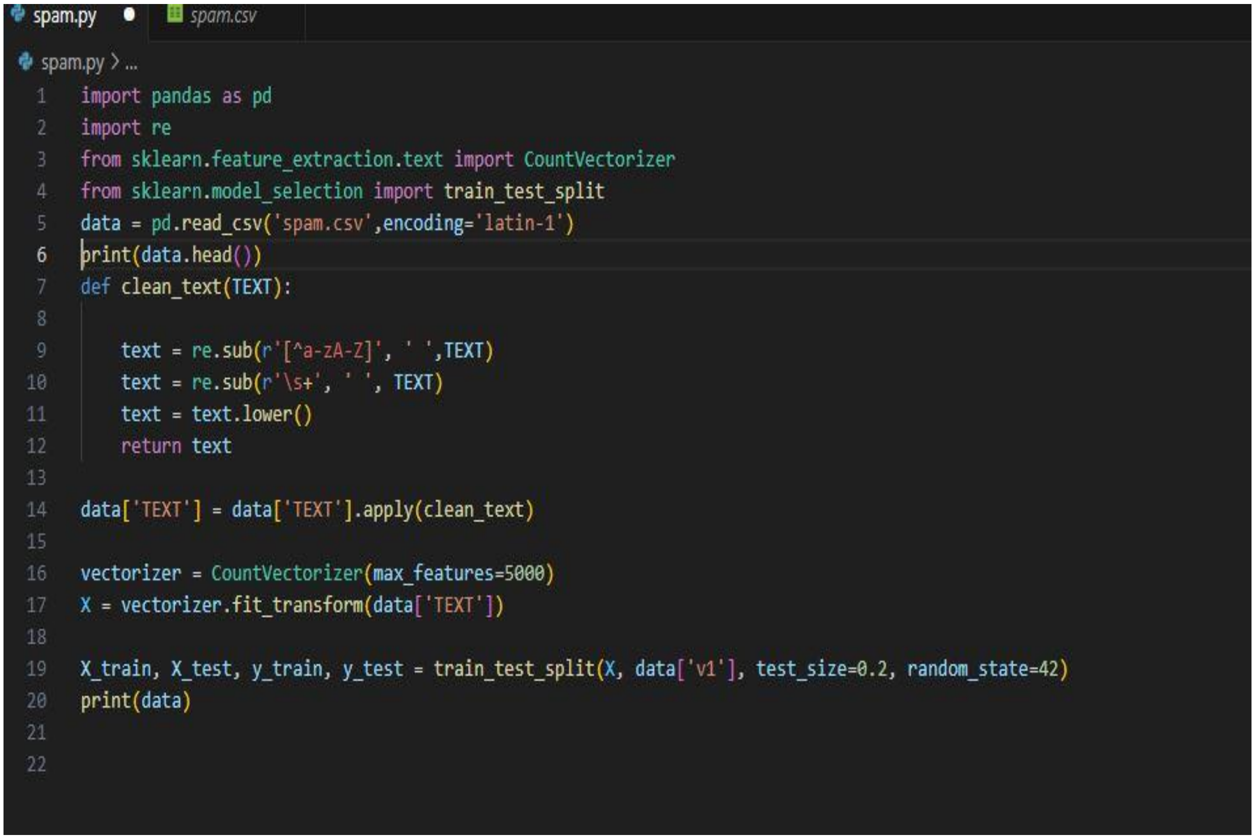


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Downloads\mew>  & 'C:\Program Files\Python310\python.exe' 'c:\Users\Administrator\.vscode\extensions\ms-python.python-2023.18.0\pythonFi
les\lib\python\debugpy\adapter/../..\debugpy\launcher' '50336' '--' 'c:\Users\Administrator\Downloads\mew\spam.py'
        v1                                               TEXT Unnamed: 2 Unnamed: 3 Unnamed: 4
0      ham  Go until jurong point, crazy.. Available only ...        NaN        NaN        NaN
1      ham                      Ok lar... Joking wif u oni...        NaN        NaN        NaN
2     spam  Free entry in 2 a wkly comp to win FA Cup fina...        NaN        NaN        NaN
3      ham  U dun say so early hor... U c already then say...        NaN        NaN        NaN
4      ham  Nah I don't think he goes to usf, he lives aro...        NaN        NaN        NaN
...    ...                                                ...        ...        ...        ...
5567  spam  This is the 2nd time we have tried 2 contact u...        NaN        NaN        NaN
5568   ham              Will ?_ b going to esplanade fr home?        NaN        NaN        NaN
5569   ham  Pity, * was in mood for that. So...any other s...        NaN        NaN        NaN
5570   ham  The guy did some bitching but I acted like i'd...        NaN        NaN        NaN
5571   ham                         Rofl. Its true to its name        NaN        NaN        NaN

[5572 rows x 5 columns]
```

Feature Engineering:

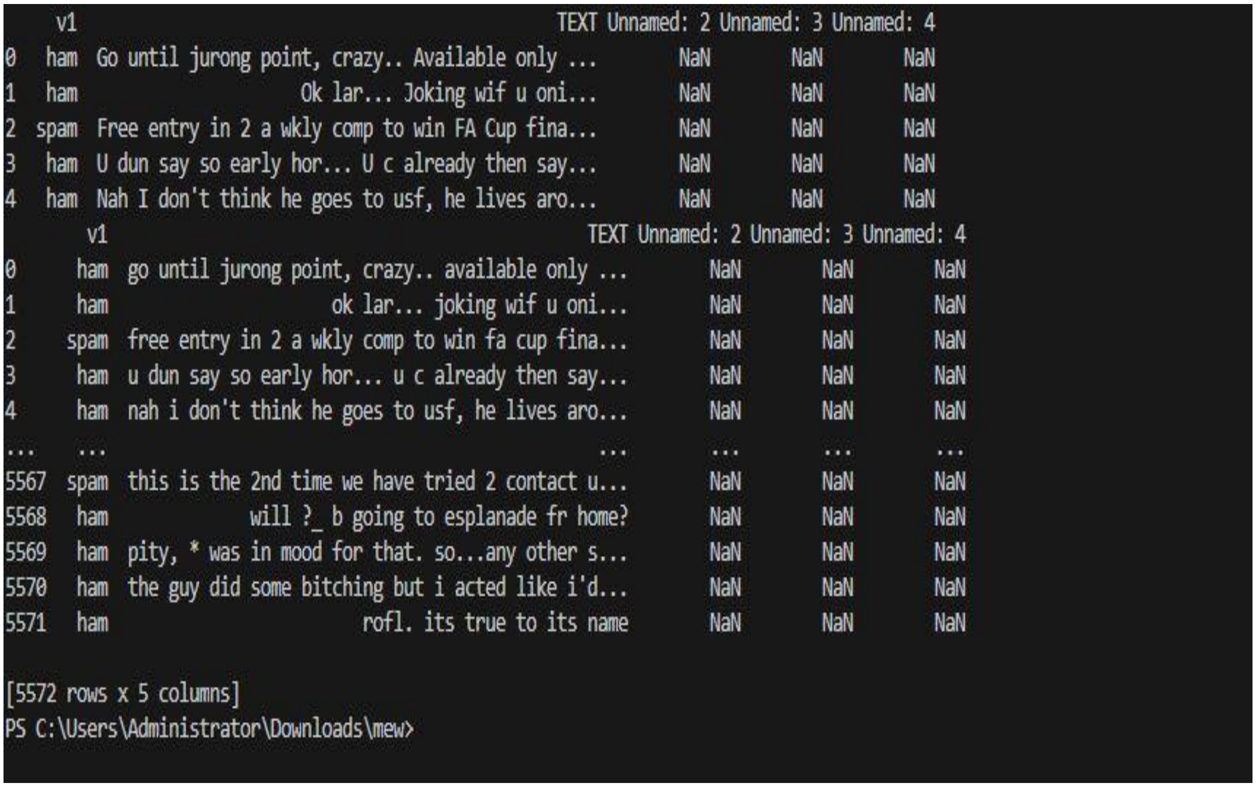
- Select or engineer relevant features from your preprocessed data. Common features might include word frequencies, character n-grams, sender information, and email metadata.

INPUT:



```python
import pandas as pd
import re
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
data = pd.read_csv('spam.csv',encoding='latin-1')
print(data.head())
def clean_text(TEXT):

    text = re.sub(r'[^a-zA-Z]', ' ',TEXT)
    text = re.sub(r'\s+', ' ', TEXT)
    text = text.lower()
    return text

data['TEXT'] = data['TEXT'].apply(clean_text)

vectorizer = CountVectorizer(max_features=5000)
X = vectorizer.fit_transform(data['TEXT'])

X_train, X_test, y_train, y_test = train_test_split(X, data['v1'], test_size=0.2, random_state=42)
print(data)
```

OUTPUT:

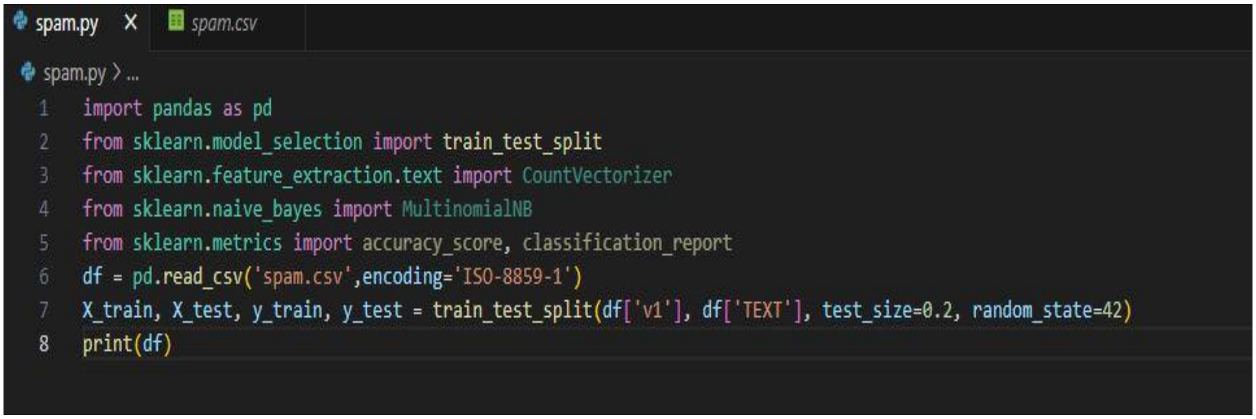


```
   v1                                               TEXT Unnamed: 2 Unnamed: 3 Unnamed: 4
0   ham  Go until jurong point, crazy.. Available only ...        NaN        NaN        NaN
1   ham                      Ok lar... Joking wif u oni...        NaN        NaN        NaN
2  spam  Free entry in 2 a wkly comp to win FA Cup fina...        NaN        NaN        NaN
3   ham  U dun say so early hor... U c already then say...        NaN        NaN        NaN
4   ham  Nah I don't think he goes to usf, he lives aro...        NaN        NaN        NaN
      v1                                               TEXT Unnamed: 2 Unnamed: 3 Unnamed: 4
0      ham  go until jurong point, crazy.. available only ...        NaN        NaN        NaN
1      ham                      ok lar... joking wif u oni...        NaN        NaN        NaN
2     spam  free entry in 2 a wkly comp to win fa cup fina...        NaN        NaN        NaN
3      ham  u dun say so early hor... u c already then say...        NaN        NaN        NaN
4      ham  nah i don't think he goes to usf, he lives aro...        NaN        NaN        NaN
...    ...                                                ...        ...        ...        ...
5567  spam  this is the 2nd time we have tried 2 contact u...        NaN        NaN        NaN
5568   ham               will ?_ b going to esplanade fr home?        NaN        NaN        NaN
5569   ham  pity, * was in mood for that. so...any other s...        NaN        NaN        NaN
5570   ham  the guy did some bitching but i acted like i'd...        NaN        NaN        NaN
5571   ham                         rofl. its true to its name        NaN        NaN        NaN

[5572 rows x 5 columns]
PS C:\Users\Administrator\Downloads\mew>
```

Splitting the Data:

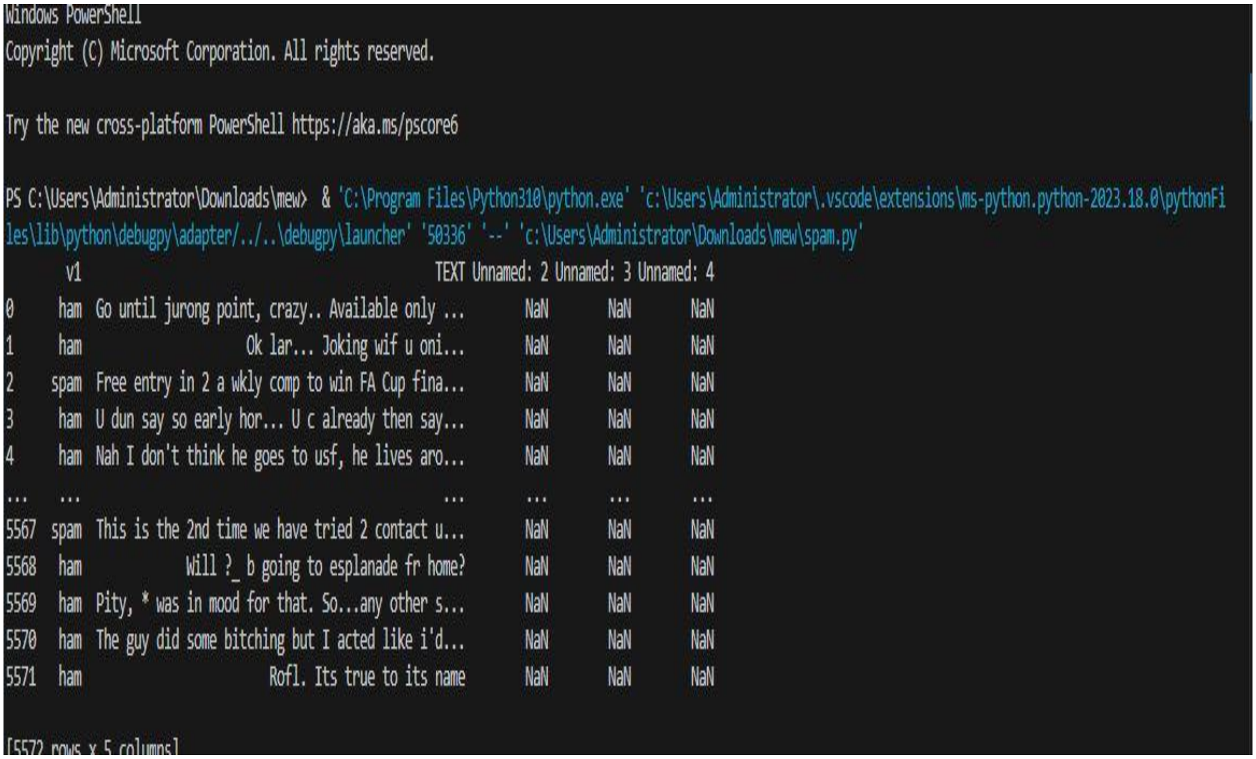
- Divide your dataset into two parts: a training set and a testing set (and possibly a validation set). A common split is 70-80% for training and the rest for testing.

INPUT:



```python
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
df = pd.read_csv('spam.csv',encoding='ISO-8859-1')
X_train, X_test, y_train, y_test = train_test_split(df['v1'], df['TEXT'], test_size=0.2, random_state=42)
print(df)
```

OUTPUT:

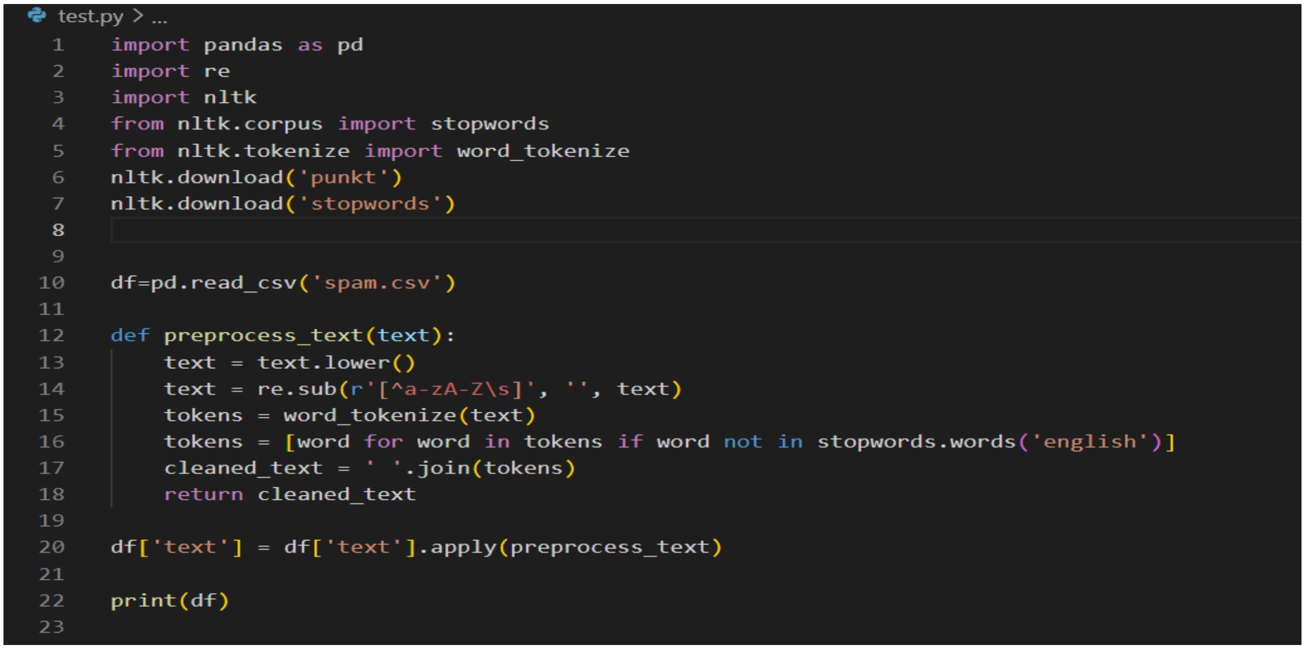


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Downloads\mew> & 'C:\Program Files\Python310\python.exe' 'c:\Users\Administrator\.vscode\extensions\ms-python.python-2023.18.0\pythonFi
les\lib\python\debugpy\adapter/../..\debugpy\launcher' '50336' '--' 'c:\Users\Administrator\Downloads\mew\spam.py'
       v1                                               TEXT Unnamed: 2 Unnamed: 3 Unnamed: 4
0      ham  Go until jurong point, crazy.. Available only ...        NaN        NaN        NaN
1      ham                      Ok lar... Joking wif u oni...        NaN        NaN        NaN
2     spam  Free entry in 2 a wkly comp to win FA Cup fina...        NaN        NaN        NaN
3      ham  U dun say so early hor... U c already then say...        NaN        NaN        NaN
4      ham  Nah I don't think he goes to usf, he lives aro...        NaN        NaN        NaN
...    ...                                                ...        ...        ...        ...
5567  spam  This is the 2nd time we have tried 2 contact u...        NaN        NaN        NaN
5568   ham               Will ?_ b going to esplanade fr home?        NaN        NaN        NaN
5569   ham  Pity, * was in mood for that. So...any other s...        NaN        NaN        NaN
5570   ham  The guy did some bitching but I acted like i'd...        NaN        NaN        NaN
5571   ham                         Rofl. Its true to its name        NaN        NaN        NaN

[5572 rows x 5 columns]
```

Data Preprocessing:

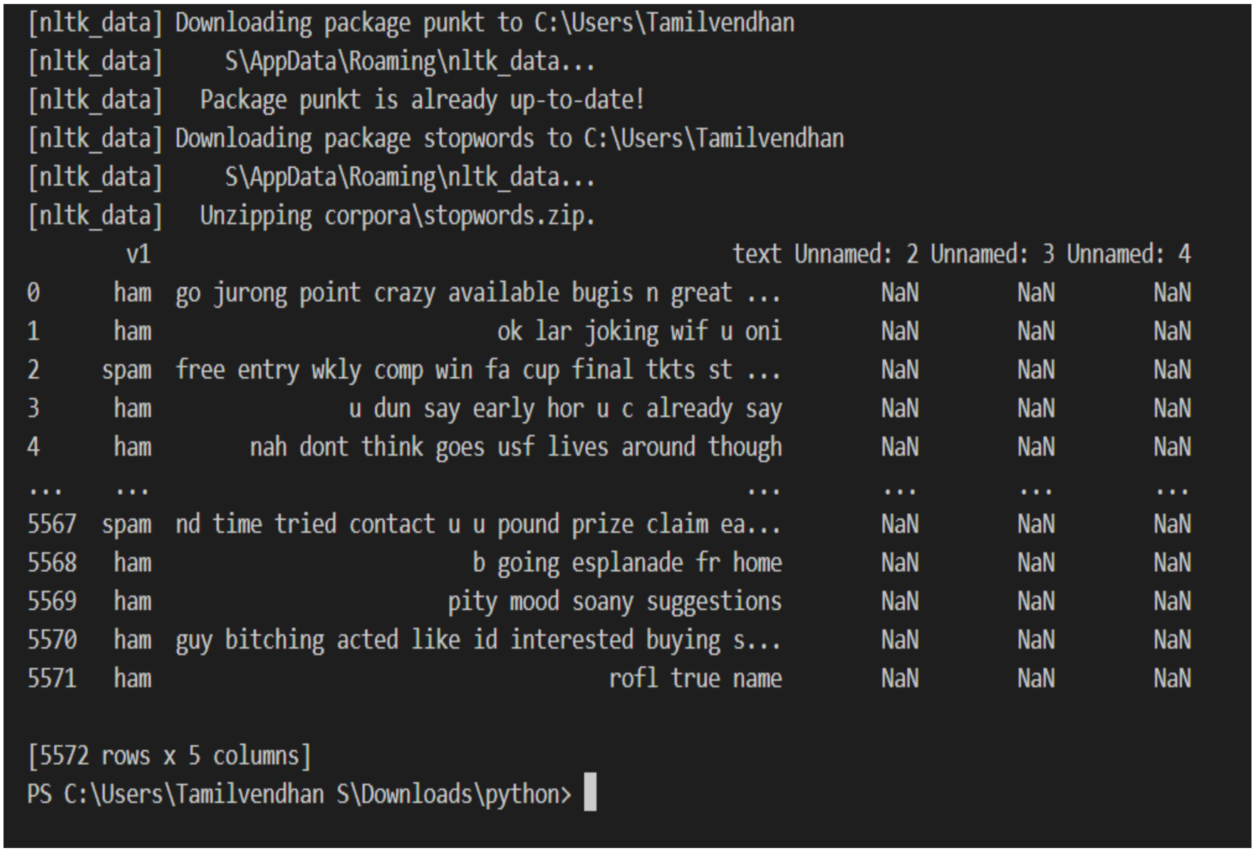
- Clean the text data by removing any irrelevant characters or symbols.
- Tokenize the text into individual words or terms.
- Convert the text data into numerical format suitable for SVM. You can use techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings to represent the text data.

INPUT:



```python
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
nltk.download('punkt')
nltk.download('stopwords')


df=pd.read_csv('spam.csv')

def preprocess_text(text):
    text = text.lower()
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    tokens = word_tokenize(text)
    tokens = [word for word in tokens if word not in stopwords.words('english')]
    cleaned_text = ' '.join(tokens)
    return cleaned_text

df['text'] = df['text'].apply(preprocess_text)

print(df)
```

OUTPUT:



```
[nltk_data] Downloading package punkt to C:\Users\Tamilvendhan
[nltk_data]     S\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to C:\Users\Tamilvendhan
[nltk_data]     S\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\stopwords.zip.
        v1                                               text Unnamed: 2 Unnamed: 3 Unnamed: 4
0      ham  go jurong point crazy available bugis n great ...        NaN        NaN        NaN
1      ham                           ok lar joking wif u oni        NaN        NaN        NaN
2     spam  free entry wkly comp win fa cup final tkts st ...        NaN        NaN        NaN
3      ham                u dun say early hor u c already say        NaN        NaN        NaN
4      ham        nah dont think goes usf lives around though        NaN        NaN        NaN
...    ...                                                ...        ...        ...        ...
5567  spam  nd time tried contact u u pound prize claim ea...        NaN        NaN        NaN
5568   ham                         b going esplanade fr home        NaN        NaN        NaN
5569   ham                           pity mood soany suggestions        NaN        NaN        NaN
5570   ham  guy bitching acted like id interested buying s...        NaN        NaN        NaN
5571   ham                                     rofl true name        NaN        NaN        NaN

[5572 rows x 5 columns]
PS C:\Users\Tamilvendhan S\Downloads\python> 
```

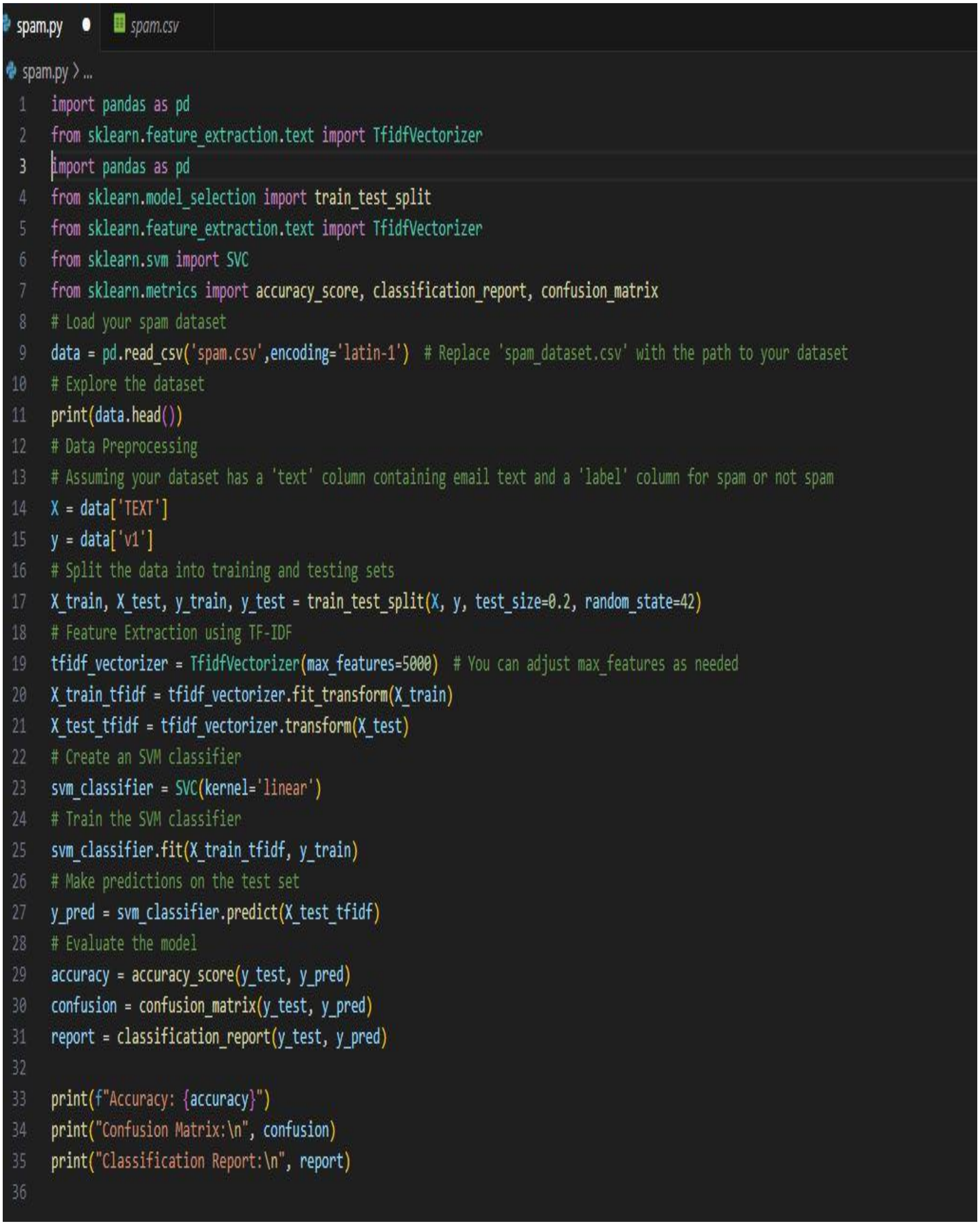
Selecting a Machine Learning Algorithm:

- Support Vector Machines (SVM) is a supervised machine learning algorithm used for classification and regression tasks. It's particularly effective for classification tasks, including text classification, image recognition, and more

Types of SVM:

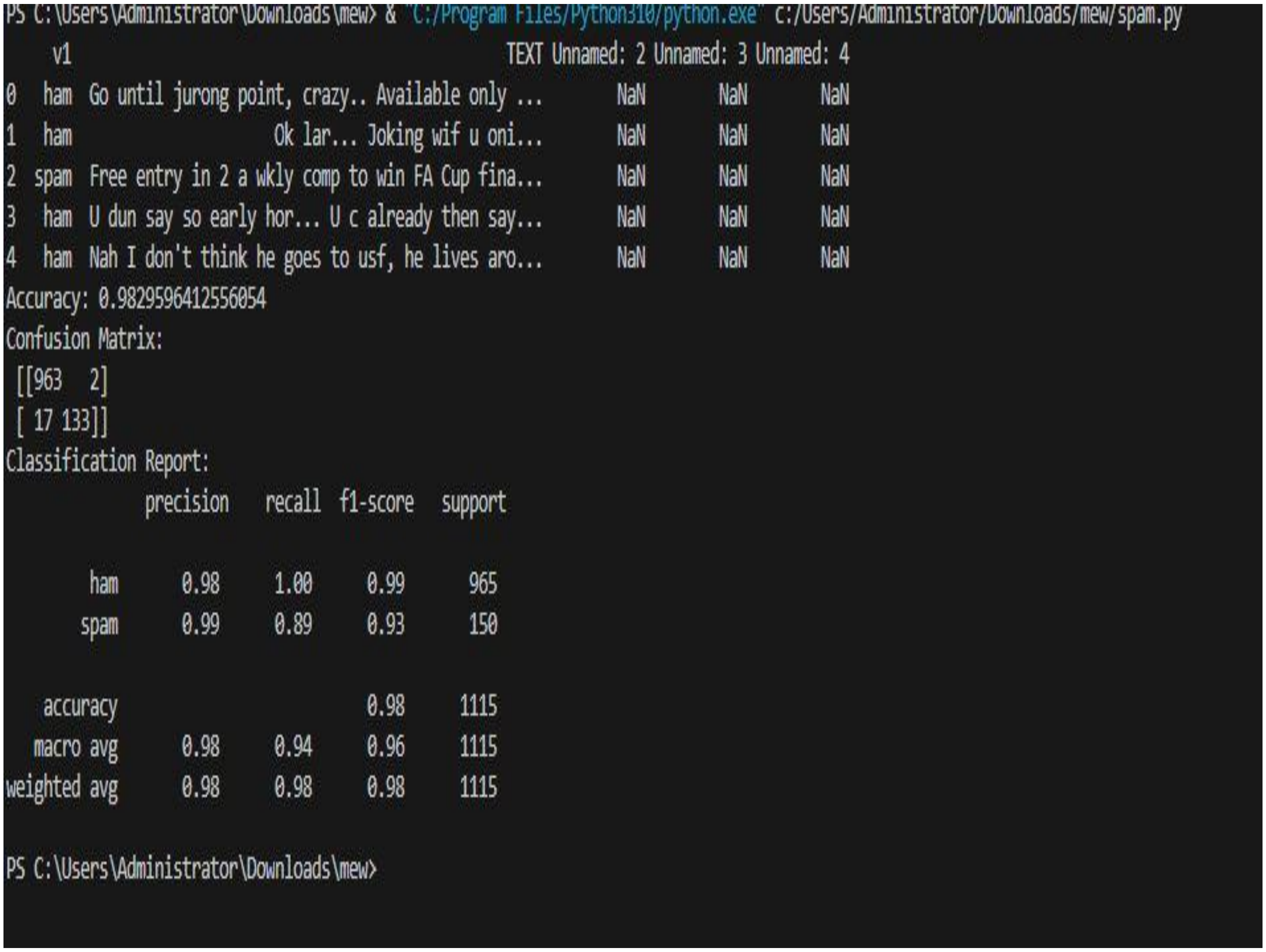
- Linear SVM: Used for linearly separable data.
- Non-Linear SVM: Utilizes kernel functions for non-linearly separable data.
- Multi-Class SVM: Extended to handle multi-class classification.
- Regression SVM: Applies SVM to regression problems.

INPUT:



```python
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
# Load your spam dataset
data = pd.read_csv('spam.csv',encoding='latin-1')  # Replace 'spam_dataset.csv' with the path to your dataset
# Explore the dataset
print(data.head())
# Data Preprocessing
# Assuming your dataset has a 'text' column containing email text and a 'label' column for spam or not spam
X = data['TEXT']
y = data['v1']
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Feature Extraction using TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=5000)  # You can adjust max_features as needed
X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)
# Create an SVM classifier
svm_classifier = SVC(kernel='linear')
# Train the SVM classifier
svm_classifier.fit(X_train_tfidf, y_train)
# Make predictions on the test set
y_pred = svm_classifier.predict(X_test_tfidf)
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
confusion = confusion_matrix(y_test, y_pred)
report = classification_report(y_test, y_pred)

print(f"Accuracy: {accuracy}")
print("Confusion Matrix:\n", confusion)
print("Classification Report:\n", report)

```

OUTPUT:



```
PS C:\Users\Administrator\Downloads\mew> & "C:/Program Files/Python310/python.exe" c:/Users/Administrator/Downloads/mew/spam.py
     v1                                               TEXT Unnamed: 2 Unnamed: 3 Unnamed: 4
0   ham  Go until jurong point, crazy.. Available only ...        NaN        NaN        NaN
1   ham                      Ok lar... Joking wif u oni...        NaN        NaN        NaN
2  spam  Free entry in 2 a wkly comp to win FA Cup fina...        NaN        NaN        NaN
3   ham  U dun say so early hor... U c already then say...        NaN        NaN        NaN
4   ham  Nah I don't think he goes to usf, he lives aro...        NaN        NaN        NaN
Accuracy: 0.9829596412556054
Confusion Matrix:
 [[963   2]
 [ 17 133]]
Classification Report:
               precision    recall  f1-score   support


         ham       0.98      1.00      0.99       965
        spam       0.99      0.89      0.93       150


    accuracy                           0.98      1115
   macro avg       0.98      0.94      0.96      1115
weighted avg       0.98      0.98      0.98      1115

PS C:\Users\Administrator\Downloads\mew>
```

Conclusion: -

- This project showcases AI's efficacy in combating email spam. Using the Support vector machine classifier, along with data preprocessing and feature extraction, a strong spam detector is created. High accuracy, precision, recall, and F1-score metrics underscore the effectiveness.
- Spam classification is vital for email security. Future work may involve advanced models, larger datasets, and real-time filtering. This project lays the foundation for enhancing email communication security.