```
 Title: Detection Powered by Natural Language Processing: Exposing the Truth with Advanced Fake News

mport pandas as pd
rom sklearn.model_selection import train_test_split
rom sklearn.feature_extraction.text import TfidfVectorizer
rom sklearn.linear_model import PassiveAggressiveClassifier
rom sklearn.metrics import accuracy_score, confusion_matrix

 Load dataset (example: 'news.csv' with columns 'text' and 'label')
f = pd.read_csv('news.csv')

 Features and labels
 = df['text']
 = df['label']  # 'FAKE' or 'REAL'

 Split the dataset
_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=7)

 Vectorize text data
ectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)
_train_vec = vectorizer.fit_transform(X_train)
_test_vec = vectorizer.transform(X_test)

 Model - PassiveAggressiveClassifier is good for online learning
odel = PassiveAggressiveClassifier(max_iter=50)
odel.fit(X_train_vec, y_train)

 Predict and evaluate
_pred = model.predict(X_test_vec)
core = accuracy_score(y_test, y_pred)

rint(f"Model Accuracy: {round(score * 100, 2)}%")
rint("Confusion Matrix:")
rint(confusion_matrix(y_test, y_pred))
```