

```
import pandas as pd

df = pd.read_csv("/content/news.csv")
```

```
df.head()
df.shape
df['label'].value_counts()
```

```
count
label
-----
REAL    3171
FAKE    3164

dtype: int64
```

```
import numpy as np
import re

from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score,
                           f1_score

import matplotlib.pyplot as plt
import seaborn as sns
```

```
df['label'] = df['label'].map({'REAL': 0, 'FAKE': 1})
```

```
df['content'] = df['title'] + " " + df['text']
```

```
def preprocess_text(text):
    text = text.lower()
    text = re.sub(r'^\w\s]', '', text)
    return text
```

```
df['clean_content'] = df['content'].apply(preprocess_text)
```

```
vectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)

X = vectorizer.fit_transform(df['clean_content'])
y = df['label']
```

```
print("Feature matrix shape:", X.shape)
```

```
Feature matrix shape: (6335, 88742)
```

```
X_train, X_test, y_train, y_test = train_test_split(  
    X, y, test_size=0.2, random_state=42  
)
```

```
model = MultinomialNB()  
model.fit(X_train, y_train)
```

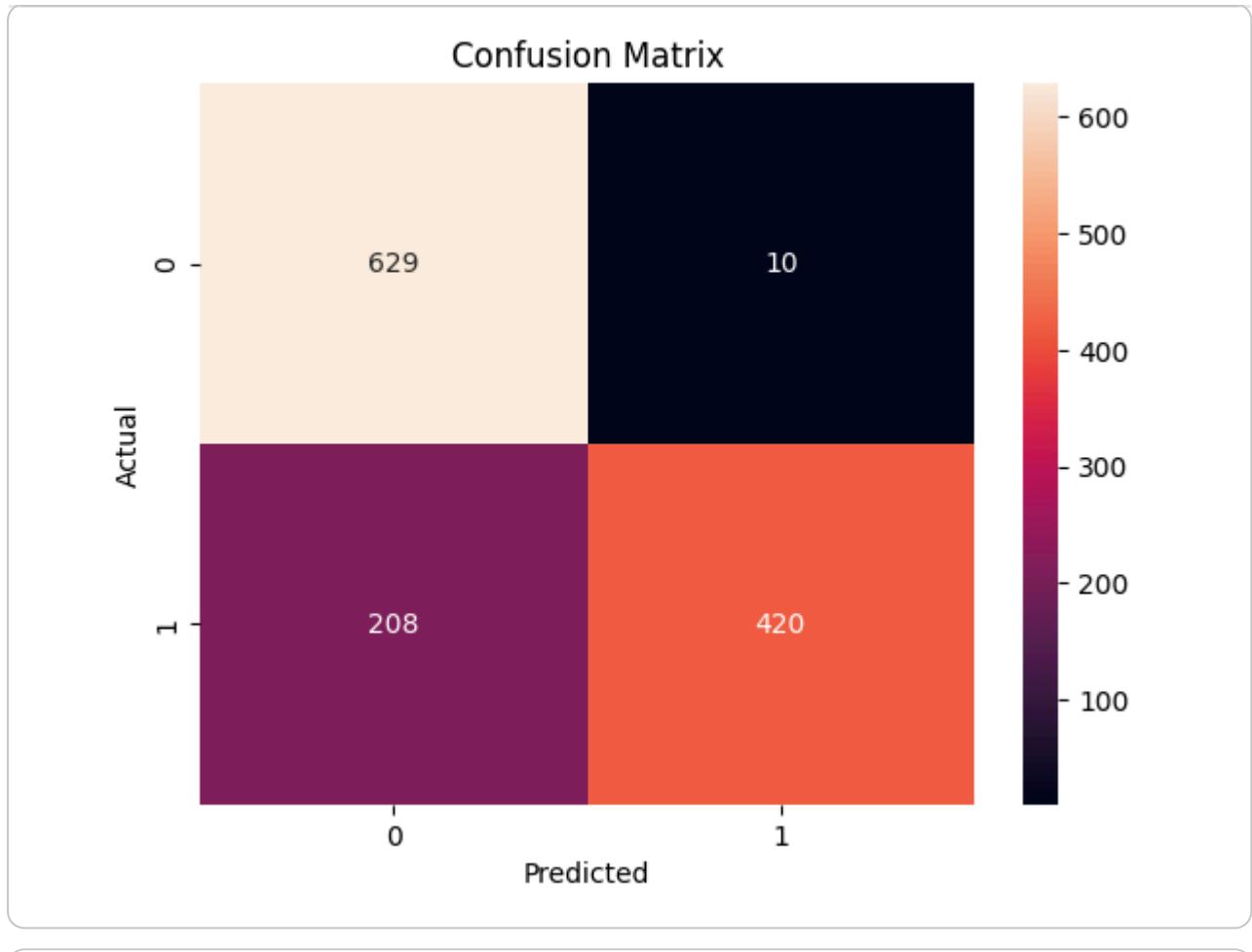
▼ MultinomialNB ⓘ ⓘ
MultinomialNB()

```
y_pred = model.predict(X_test)
```

```
accuracy = accuracy_score(y_test, y_pred)  
precision = precision_score(y_test, y_pred)  
recall = recall_score(y_test, y_pred)  
f1 = f1_score(y_test, y_pred)  
  
print("Accuracy:", accuracy)  
print("Precision:", precision)  
print("Recall:", recall)  
print("F1 Score:", f1)
```

```
Accuracy: 0.8279400157853196  
Precision: 0.9767441860465116  
Recall: 0.6687898089171974  
F1 Score: 0.7939508506616257
```

```
cm = confusion_matrix(y_test, y_pred)  
  
plt.figure()  
sns.heatmap(cm, annot=True, fmt='d')  
plt.xlabel("Predicted")  
plt.ylabel("Actual")  
plt.title("Confusion Matrix")  
plt.show()
```



Start coding or generate with AI.