Mini Project 9: Sentiment Analysis

Step 1: Import library

```
import pandas as pd

import numpy as np
```

Step 2: Import Data

```
df=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/Financial%20Market%20News.csv',encoding="ISO-8859-1")

df.head()
```

Saved successfully!   ✕

	Date	Label	News 1	News 2	News 3	News 4	News 5	News 6	News 7	News 8	...	News 16	
0	01-01-2010	0	McIlroy's men catch cold from Gudjonsson	Obituary: Brian Walsh	Workplace blues leave employers in the red	Classical review: Rattle	Dance review: Merce Cunningham	Genetic tests to be used in setting premiums	Opera review: La Bohème	Pop review: Britney Spears	...	Finland 0 - 0 England	F r
1	02-01-2010	0	Warning from history points to crash	Investors flee to dollar haven	Banks and tobacco in favour	Review: Llama Farmers	War jitters lead to sell-off	Your not-so-secret history	Review: The Northern Sinfonia	Review: Hysteria	...	Why Wenger will stick to his Gunners	E l
2	03-01-2010	0	Comment: Why Israel's peaceniks feel betrayed	Court deals blow to seizure of drug assets	An ideal target for spooks	World steps between two sides intent on war	What the region's papers say	Comment: Fear and rage in Palestine	Poverty and resentment fuels Palestinian fury	Republican feud fear as dissident is killed	...	FTSE goes upwardly mobile	
	04-		£750,000-	Newcastle	Brown sent to the	Tourists wary of	Canary Wharf			Review: New		More cash	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4101 entries, 0 to 4100
Data columns (total 27 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Date      4101 non-null   object
 1   Label     4101 non-null   int64
 2   News 1    4101 non-null   object
 3   News 2    4101 non-null   object
 4   News 3    4101 non-null   object
```

```
                                                object
                                                object
 7      News 6      4101 non-null      object
 8      News 7      4101 non-null      object
 9      News 8      4101 non-null      object
 10     News 9      4101 non-null      object
 11     News 10     4101 non-null      object
 12     News 11     4101 non-null      object
 13     News 12     4101 non-null      object
 14     News 13     4101 non-null      object
 15     News 14     4101 non-null      object
 16     News 15     4101 non-null      object
 17     News 16     4101 non-null      object
 18     News 17     4101 non-null      object
 19     News 18     4101 non-null      object
 20     News 19     4101 non-null      object
 21     News 20     4101 non-null      object
 22     News 21     4101 non-null      object
 23     News 22     4101 non-null      object
 24     News 23     4100 non-null      object
 25     News 24     4098 non-null      object
 26     News 25     4098 non-null      object
dtypes: int64(1), object(26)
memory usage: 865.2+ KB
```

```
df.shape
```

```
(4101, 27)
```

```
df.columns
```

```
Index(['Date', 'Label', 'News 1', 'News 2', 'News 3', 'News 4', 'News 5',
       'News 6', 'News 7', 'News 8', 'News 9', 'News 10', 'News 11', 'News 12',
       'News 13', 'News 14', 'News 15', 'News 16', 'News 17', 'News 18',
       'News 19', 'News 20', 'News 21', 'News 22', 'News 23', 'News 24',
       'News 25'],
      dtype='object')
```

Saved successfully!   ✕

```
' '.join(str(x) for x in df.iloc[1,2:27])
```

```
'Warning from history points to crash Investors flee to dollar haven Banks and tobacco in favour Revie
w: Llama Farmers War jitters lead to sell-off Your not-so-secret history Review: The Northern Sinfonia
Review: Hysteria Review: The Guardsman Opera: The Marriage of Figaro Review: The Turk in Italy Deutsch
e spells out its plans for diversification Traders' panic sends oil prices skyward TV sport chief leav
es home over romance Leader: Hi-tech twitch Why Wenger will stick to his Gunners Out of luck England h
it rock bottom Wilkinson out of his depth Kinsella sparks Irish power play Brown banished as Scots reb
```

```
df.index
```

```
RangeIndex(start=0, stop=4101, step=1)
```

```
len(df.index)
```

```
4101
```

```
news=[]
for row in range(0,len(df.index)):
  news.append(' '.join(str(x) for x in df.iloc[row,2:27]))
```

```
type(news)
```

```
list
```

```
news[0]
```

Gudjonsson Obituary: Brian Walsh Workplace blues leave employers in the red Classical review: Rattle Dance review: Merce Cunningham Genetic tests to be used in setting premiums Opera review: La Bohème Pop review: Britney Spears Theatre review: The Circle Wales face a fraught night Under-21 round-up Smith off to blot his copybook Finns taking the mickey Praise wasted as Brown studies injury options Ireland wary of minnows Finland 0 - 0 England Healy a marked man Happy birthday Harpers & Queen Win unlimited access to the Raindance film festival Labour pledges £800m to bridge nor

```
X=news
```

```
type(X)
```

```
    list
```

Step 4: Text Conversion

```
from sklearn.feature_extraction.text import CountVectorizer
```

```
cv=CountVectorizer(lowercase=True,ngram_range=(1,1))
```

```
X=cv.fit_transform(X)
```

```
X.shape
```

```
    (4101, 48527)
```

```
y=df['Label']
```

Saved successfully!   ✕

```
(4101,)
```

Step 5: Splitting Data

```
from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,stratify=y,random_state=2529)
```

Step 6: Creating Model

```
from sklearn.ensemble import RandomForestClassifier

model=RandomForestClassifier(n_estimators=200)
```

Step 7: Training Model

```
model.fit(X_train,y_train)

    RandomForestClassifier(n_estimators=200)
```

Step 8: Prediction Model

Saved successfully!   ✕

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

```
y_pred.shape
```

```
(1231,)
```

```
y_pred
```

```
array([1, 1, 0, ..., 1, 0, 1])
```

Step 9: Accuracy

```
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
```

```
confusion_matrix(y_test,y_pred)
```

```
array([[152, 429],
       [165, 485]])
```

```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.48      0.26      0.34       581
           1       0.53      0.75      0.62       650

    accuracy                           0.52      1231
   macro avg       0.51      0.50      0.48      1231
weighted avg       0.51      0.52      0.49      1231
```

Saved successfully!   ✕

`0.5174654752233956`

Link of the same: https://colab.research.google.com/drive/1qv6fLW3IX-k8cv4RuOM_lNGU40PnkOEO?usp=sharing