In [2]:
```python
1  import pandas as pd
2  import numpy as np
3  import matplotlib.pyplot as plt
4  import seaborn as sns
```

In [3]:
```python
1  data = pd.read_csv("fake_news.csv")
2  data.head()
```

Out[3]:

| | id | title | author | text | label |
|---|---|---|---|---|---|
| **0** | 0 | House Dem Aide: We Didn't Even See Comey's Let... | Darrell Lucus | House Dem Aide: We Didn't Even See Comey's Let... | 1 |
| **1** | 1 | FLYNN: Hillary Clinton, Big Woman on Campus - ... | Daniel J. Flynn | Ever get the feeling your life circles the rou... | 0 |
| **2** | 2 | Why the Truth Might Get You Fired | Consortiumnews.com | Why the Truth Might Get You Fired October 29, ... | 1 |
| **3** | 3 | 15 Civilians Killed In Single US Airstrike Hav... | Jessica Purkiss | Videos 15 Civilians Killed In Single US Airstr... | 1 |
| **4** | 4 | Iranian woman jailed for fictional unpublished... | Howard Portnoy | Print \nAn Iranian woman has been sentenced to... | 1 |

In [4]:
```python
1  data.shape
```

Out[4]: (20800, 5)

In [5]:
```python
1  data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20800 entries, 0 to 20799
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   id      20800 non-null  int64
 1   title   20242 non-null  object
 2   author  18843 non-null  object
 3   text    20761 non-null  object
 4   label   20800 non-null  int64
dtypes: int64(2), object(3)
memory usage: 812.6+ KB
```

In [6]:
```python
1  data.isna().sum()
```

Out[6]:
```
id          0
title     558
author   1957
text       39
label       0
dtype: int64
```

In [7]:
```python
1  data = data.drop(['id'] , axis=1)
```

```
In [8]:    1  data = data.fillna('')
```

```
In [9]:    1  data['content'] = data['author'] +''+ data['title']+''+data['text']
```

```
In [10]:   1  data = data.drop(['title','author','text'], axis=1)
```

```
In [11]:   1  data.head()
```

Out[11]:

|   | label | content |
|---|-------|---------|
| **0** | 1 | Darrell LucusHouse Dem Aide: We Didn't Even Se... |
| **1** | 0 | Daniel J. FlynnFLYNN: Hillary Clinton, Big Wom... |
| **2** | 1 | Consortiumnews.comWhy the Truth Might Get You ... |
| **3** | 1 | Jessica Purkiss15 Civilians Killed In Single U... |
| **4** | 1 | Howard PortnoyIranian woman jailed for fiction... |

```
In [12]:   1  data['content'] = data['content'].apply(lambda x: " ".join(x.lower() fo
```

```
In [13]:   1  data['content'] = data['content'].str.replace('[^\w\s]','')
```

```
C:\Users\nihar\AppData\Local\Temp\ipykernel_8128\3643324700.py:1: FutureWa
rning: The default value of regex will change from True to False in a futu
re version.
  data['content'] = data['content'].str.replace('[^\w\s]','')
```

```
In [14]:   1  import nltk
           2  nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\nihar\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[14]: True

```
In [15]:   1  from nltk.corpus import stopwords
           2  stop = stopwords.words('english')
           3  data['content'] = data['content'].apply(lambda x:" ".join(x for x in x.
```

In [16]:
```python
!pip install textblob
```

Requirement already satisfied: textblob in c:\users\nihar\anaconda3\lib\site-packages (0.19.0)
Requirement already satisfied: nltk>=3.9 in c:\users\nihar\anaconda3\lib\site-packages (from textblob) (3.9.1)
Requirement already satisfied: regex>=2021.8.3 in c:\users\nihar\anaconda3\lib\site-packages (from nltk>=3.9->textblob) (2022.7.9)
Requirement already satisfied: joblib in c:\users\nihar\anaconda3\lib\site-packages (from nltk>=3.9->textblob) (1.1.0)
Requirement already satisfied: click in c:\users\nihar\anaconda3\lib\site-packages (from nltk>=3.9->textblob) (8.0.4)
Requirement already satisfied: tqdm in c:\users\nihar\anaconda3\lib\site-packages (from nltk>=3.9->textblob) (4.64.1)
Requirement already satisfied: colorama in c:\users\nihar\anaconda3\lib\site-packages (from click->nltk>=3.9->textblob) (0.4.6)

In [17]:
```python
from nltk.stem import WordNetLemmatizer
from textblob import Word
data['content']=data['content'].apply(lambda x:"".join([Word(word).lemm
data['content'].head()
```

Out[17]:
```
0    darrelllucushousedemaidedidntevenseecomeyslett...
1    danieljflynnflynnhillaryclintonbigwomancampusb...
2    consortiumnewscomwhytruthmightgetfiredwhytruth...
3    jessicapurkiss15civiliankilledsingleuairstrike...
4    howardportnoyiranianwomanjailedfictionalunpubl...
Name: content, dtype: object
```

In [18]:
```python
X = data[['content']]
y = data['label']
```

In [19]:
```python
from sklearn.model_selection import train_test_split
```

In [20]:
```python
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3, rando
```

In [21]:
```python
print(X_train.shape)
print(y_train.shape)
print(X_test.shape)
print(y_test.shape)
```

```
(14560, 1)
(14560,)
(6240, 1)
(6240,)
```

In [22]:
```python
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
In [23]:   1  tfidf_vect = TfidfVectorizer(analyzer='word', token_pattern=r'\w{1,}',
           2  tfidf_vect.fit(data['content'])
           3
           4  xtrain_tfidf = tfidf_vect.transform(X_train['content'])
           5  xtest_tfidf = tfidf_vect.transform(X_test['content'])
```

```
In [24]:   1  from sklearn.linear_model import PassiveAggressiveClassifier
           2  from sklearn import metrics
           3
           4  pclf = PassiveAggressiveClassifier()
           5  pclf.fit(xtrain_tfidf, y_train)
           6  predictions = pclf.predict(xtest_tfidf)
           7  print(metrics.classification_report(y_test, predictions))
```

```
              precision    recall  f1-score   support

           0       0.50      1.00      0.67      3116
           1       1.00      0.01      0.03      3124

    accuracy                           0.51      6240
   macro avg       0.75      0.51      0.35      6240
weighted avg       0.75      0.51      0.35      6240
```

```
In [25]:   1  print(metrics.confusion_matrix(y_test,predictions))
```

```
[[3116    0]
 [3082   42]]
```

```python
In [26]:   1  from sklearn.neural_network import MLPClassifier
           2  mlpclf = MLPClassifier(hidden_layer_sizes=(256,64,16),
           3                         activation='relu',
           4                         solver='adam')
           5  mlpclf.fit(xtrain_tfidf, y_train)
           6  predictions = mlpclf.predict(xtest_tfidf)
           7  print(metrics.classification_report(y_test, predictions))
```

```
              precision    recall  f1-score   support

           0       0.00      0.00      0.00      3116
           1       0.50      1.00      0.67      3124

    accuracy                           0.50      6240
   macro avg       0.25      0.50      0.33      6240
weighted avg       0.25      0.50      0.33      6240
```

```
C:\Users\nihar\Anaconda3\lib\site-packages\sklearn\metrics\_classificatio
n.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined a
nd being set to 0.0 in labels with no predicted samples. Use `zero_divisio
n` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\nihar\Anaconda3\lib\site-packages\sklearn\metrics\_classificatio
n.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined a
nd being set to 0.0 in labels with no predicted samples. Use `zero_divisio
n` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\nihar\Anaconda3\lib\site-packages\sklearn\metrics\_classificatio
n.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined a
nd being set to 0.0 in labels with no predicted samples. Use `zero_divisio
n` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

```python
In [47]:   1  print(metrics.confusion_matrix(y_test,predictions))
```

```
[[   0 3116]
 [   0 3124]]
```

```python
In [48]:   1  import pickle
           2  pickle.dump(mlpclf,open("fakenews1.pkl","wb"))
```

```python
In [49]:   1  def fake_news(news):
           2      input_data=[news]
           3      vectorized_input_data=tfidf_vect.transform(input_data)
           4      prediction=pclf.predict(vectorized_input_data)
           5      print(prediction)
```

```python
In [54]:   1  fake_news('u.s. Secretary of state john F.Kerry said monday that he wil
```

```
[0]
```

In [52]:    1  fake_news("""President Barack Obama has been campaigning hard for the w

[0]

In [ ]:    1