In [1]:
```
import numpy as np
import pandas as pd
```

In [3]:
```
df = pd.read_csv("news.csv")
df.head()
```

Out[3]:

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

In [6]:
```
labels = df.label
labels.head()
```

Out[6]:
```
0    FAKE
1    FAKE
2    REAL
3    FAKE
4    REAL
Name: label, dtype: object
```

In [10]:
```
labels.shape
```

Out[10]:
```
(6335,)
```

In [11]:
```
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
```

In [12]:
```
x_train,x_test,y_train,y_test=train_test_split(df["text"],labels,test_size=0.2,rando
```

In [13]:
```
tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)
tfidf_train=tfidf_vectorizer.fit_transform(x_train)
tfidf_test=tfidf_vectorizer.transform(x_test)
```

In [14]:
```
pac=PassiveAggressiveClassifier(max_iter=50)
pac.fit(tfidf_train,y_train)
```

Out[14]:
```
PassiveAggressiveClassifier(max_iter=50)
```

In [15]:
```
y_pred=pac.predict(tfidf_test)
score=accuracy_score(y_test,y_pred)
print(f'Accuracy: {round(score*100,2)}%')
```

```
Accuracy: 92.74%
```

In [16]:
```
confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])
```

Out[16]:
```
array([[590,  48],
       [ 44, 585]], dtype=int64)
```

In [ ]:
```

```