In [ ]:
```
# What is Fake News?

# A type of yellow journalism, fake news encapsulates pieces of news that may be hoaxes and is generally spread through
# social media and other online media.
# This is often done to further or impose certain ideas and is often achieved with political agendas.
# Such news items may contain false and/or exaggerated claims, and may end up being viralized by algorithms,
# and users may end up in a filter bubble.
```

In [ ]:
```
# What is a TfidfVectorizer?

# TF (Term Frequency): The number of times a word appears in a document is its Term Frequency.
# A higher value means a term appears more often than others, and so,
# the document is a good match when the term is part of the search terms.

# IDF (Inverse Document Frequency): Words that occur many times a document, but also occur many times in many others,
# may be irrelevant. IDF is a measure of how significant a term is in the entire corpus.

# The TfidfVectorizer converts a collection of raw documents into a matrix of TF-IDF features.
```

In [ ]:
```
# What is a PassiveAggressiveClassifier?

# Passive Aggressive algorithms are online learning algorithms.
# Such an algorithm remains passive for a correct classification outcome,
# and turns aggressive in the event of a miscalculation, updating and adjusting.
# Unlike most other algorithms, it does not converge. Its purpose is to make updates that correct the loss,
# causing very little change in the norm of the weight vector.
```

In [ ]:
```
# About Detecting Fake News with Python

# This advanced python project of detecting fake news deals with fake and real news. Using sklearn,
# we build a TfidfVectorizer on our dataset. Then, we initialize a PassiveAggressive Classifier and fit the model.
# In the end, the accuracy score and the confusion matrix tell us how well our model fares.
```

In [ ]:
```
#Importing Libs
```

In [4]:
```
import numpy as np
import pandas as pd
import itertools
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 [ ]:

```
#Importing data
```

In [6]:

```
df=pd.read_csv(r'C:\Users\SAMAD\Downloads\Datasets\news.csv')
df.shape
df.head()
```

Out[6]:

	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 [ ]:

```
#Getting labels.
```

In [7]:

```
labels=df.label
labels.head()
```

Out[7]:

```
0    FAKE
1    FAKE
2    REAL
3    FAKE
4    REAL
Name: label, dtype: object
```

In [ ]:

```
#Spliting the data into training and testing sets.
```

In [8]:

```
x_train, x_test, y_train, y_test = train_test_split(df['text'], labels, test_size=0.2, random_state=7)
```

In [10]:

```
#Initializing TfidfVectorizer
tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)

#Fitting and training the dataset.
```

```
tfidf_train=tfidf_vectorizer.fit_transform(x_train)
tfidf_test=tfidf_vectorizer.transform(x_test)
```

In [12]:

```
#Initializing a PassiveAggressiveClassifier
pac=PassiveAggressiveClassifier(max_iter=50)
pac.fit(tfidf_train,y_train)

#Predicting and calculating the accuracy
y_pred=pac.predict(tfidf_test)
score=accuracy_score(y_test,y_pred)
print(f'Accuracy: {round(score*100,2)}%')
```

```
Accuracy: 92.98%
```

In [13]:

```
#Building confusion matrix
confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])
```

Out[13]:

```
array([[592,  46],
       [ 43, 586]], dtype=int64)
```

In [ ]:

```
# With this model, we get result that we have 589 true positives, 587 true negatives, 42 false positives, and 49 false negatives
```