In [14]:
```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report,confusion_matrix
```

In [15]:
```
import re
import nltk
nltk.download('stopwords')
```

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

Out[15]:
```
True
```

In [16]:
```
from nltk.util import pr
stemmer = nltk.SnowballStemmer("english")
from nltk.corpus import stopwords
import string
stopword = set(stopwords.words("english"))
```

In [17]:
```
df = pd.read_csv("C:\\Users\\panis\\OneDrive\\Documents\\twitter_data_updated.csv")
print(df.head())
```

```
   count  hate_speech  offensive_language  neither  class  \
0      3            0                   0        3      2
1      3            0                   3        0      1
2      3            0                   3        0      1
3      3            0                   2        1      1
4      6            0                   6        0      1

                                               tweet
0  !!! RT @mayasolovely: As a woman you shouldn't...
1  !!!!! RT @mleew17: boy dats cold...tyga dwn ba...
2  !!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby...
3  !!!!!!!!! RT @C_G_Anderson: @viva_based she lo...
4  !!!!!!!!!!!! RT @ShenikaRoberts: The shit you...
```

In [18]:
```
df['labels'] = df['class'].map({0:"HATE SPEECH DETECTED", 1:"OFFENSIVE LANGUAGE DETECTED", 2:"NEGATIVE", 3:"No hate and offensive speech"})
print(df.head())
```

```
   count  hate_speech  offensive_language  neither  class  \
0      3            0                   0        3      2
1      3            0                   3        0      1
2      3            0                   3        0      1
3      3            0                   2        1      1
4      6            0                   6        0      1

                                               tweet  \
0  !!! RT @mayasolovely: As a woman you shouldn't...
1  !!!!! RT @mleew17: boy dats cold...tyga dwn ba...
2  !!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby...
3  !!!!!!!!! RT @C_G_Anderson: @viva_based she lo...
4  !!!!!!!!!!!! RT @ShenikaRoberts: The shit you...

                       labels
0                    NEGATIVE
1  OFFENSIVE LANGUAGE DETECTED
2  OFFENSIVE LANGUAGE DETECTED
3  OFFENSIVE LANGUAGE DETECTED
4  OFFENSIVE LANGUAGE DETECTED
```

In [19]:
```
df= df[['tweet','labels']]
print(df.head())
```

```
                                               tweet  \
0  !!! RT @mayasolovely: As a woman you shouldn't...
1  !!!!! RT @mleew17: boy dats cold...tyga dwn ba...
2  !!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby...
3  !!!!!!!!! RT @C_G_Anderson: @viva_based she lo...
4  !!!!!!!!!!!! RT @ShenikaRoberts: The shit you...

                       labels
0                    NEGATIVE
1  OFFENSIVE LANGUAGE DETECTED
2  OFFENSIVE LANGUAGE DETECTED
3  OFFENSIVE LANGUAGE DETECTED
4  OFFENSIVE LANGUAGE DETECTED
```

In [20]:
```
def clean(text):
    text= str(text).lower()
    text = re.sub('\[.*?\]','',text)
    text = re.sub('https?://\S+|www\.\S+','',text)
    text = re.sub('<.*?>+','',text)
    text =re.sub('[%s]' % re.escape(string.punctuation),'',text)
    text = re.sub('\n','',text)
    text = re.sub('\w*\d\w*','',text)
    text =[word for word in text.split(' ')if word not in stopword]
    text = " ".join(text)
    return text
df["tweet"]= df["tweet"].apply(clean)
print(df.head())
```

```
                                               tweet  \
0   rt mayasolovely woman shouldnt complain clean...
1   rt  boy dats coldtyga dwn bad cuffin dat hoe ...
2   rt urkindofbrand dawg rt  ever fuck bitch sta...
3            rt cganderson vivabased look like tranny
4   rt shenikaroberts shit hear might true might ...

                       labels
0                    NEGATIVE
1  OFFENSIVE LANGUAGE DETECTED
2  OFFENSIVE LANGUAGE DETECTED
3  OFFENSIVE LANGUAGE DETECTED
4  OFFENSIVE LANGUAGE DETECTED
```

In [21]:
```
x = np.array(df["tweet"])
y = np.array(df["labels"])
cv = CountVectorizer()
x = cv.fit_transform(x)

X_train,X_test,y_train,y_test = train_test_split(x,y, test_size=0.2, random_state=42)
model = DecisionTreeClassifier()
model.fit(X_train,y_train)
```

Out[21]:
```
▾ DecisionTreeClassifier

DecisionTreeClassifier()
```

In [22]:
```
y_pred=model.predict(X_test)
```

In [23]:
```
cm=confusion_matrix(y_test,y_pred)
cm
```

Out[23]:
```
array([[  80,   31,  179],
       [  19,  701,  115],
       [ 141,  148, 3543]], dtype=int64)
```

In [24]:
```
print(classification_report(y_test,y_pred))
```

```
                            precision    recall  f1-score   support

       HATE SPEECH DETECTED       0.33      0.28      0.30       290
                   NEGATIVE       0.80      0.84      0.82       835
OFFENSIVE LANGUAGE DETECTED       0.92      0.92      0.92      3832

                   accuracy                           0.87      4957
                  macro avg       0.68      0.68      0.68      4957
               weighted avg       0.87      0.87      0.87      4957
```

In [ ]:

In [ ]: