These are the steps I performed to perform this task:

1. First I extracted all the data column wise and stored in different variables.
2. Polarity which is in the form of string I applied categorical encoding to use it as dummy variable(0 and 1).
3. List all text file in variable arr.
4. Taking single file at a time read the file and store all content into a variable.
5. Tokenize the whole text(A tokenizer that divides a string into substrings by splitting on the specified string according to french language).
6. Removed all the commas and other symbols from the text using regex (re.sub) .
7. Generate a list of tokens from the text.
8. Filter all stop words and numeric terms from it.
9. Created a nltk text from those tokens.
10. Created a stemmer object in the FrenchStemmer class.
11. Stem whole text (Stemmers remove morphological affixes from words, leaving only the word stem).
12. Count all type of sentiments from text.
13. Find Max of those.

**Read me:::: To run the file store it at ‘C://Users//kiit1//Documents//Python//ABI Health//’ .**

*import pandas as pd*

*import numpy as np*

*import os*

*import nltk*

*from nltk.tokenize import word\_tokenize*

*from nltk.corpus import stopwords*

*import re*

*from nltk.stem.snowball import FrenchStemmer*

* Imported all libraries needed during implementation.

*dataset = pd.read\_csv('FEEL1.csv')*

*dataset = dataset.drop(['id'], axis=1)*

*X = dataset.iloc[:, 1].values*

*Y = dataset.iloc[:, 2].values*

*Z = dataset.iloc[:, 3].values*

*P = dataset.iloc[:, 4].values*

*Q = dataset.iloc[:, 5].values*

*R = dataset.iloc[:, 6].values*

*S = dataset.iloc[:, 7].values*

* **Importing the Data set.**

*from sklearn.preprocessing import LabelEncoder*

*labelencoder\_X = LabelEncoder()*

*X = labelencoder\_X.fit\_transform(X)*

* Encoding categorical data

*arr = os.listdir(path)*

* List all text file

*c = f.read()*

* Read the file

*tokens = nltk.word\_tokenize(c,language='french')*

*no\_commas = re.sub(r'[.|,|\']',' ', c)*

*tokens = nltk.word\_tokenize(no\_commas)*

*text=nltk.Text(tokens)*

*words=[w.lower() for w in text]*

* Tokenize the whole text

*stopword\_list = stopwords.words('french')*

*filtered\_words = []*

*for word in words:*

*if word not in stopword\_list and word.isalpha() and len(word) > 1:*

*filtered\_words.append(word)*

*filtered\_words.sort()*

* Filter all stop words and numeric terms from it.

*stemmed\_words = []*

*stemmer = FrenchStemmer()*

*for word in filtered\_words:*

*stemmed\_word=stemmer.stem(word)*

*stemmed\_words.append(stemmed\_word)*

*stemmed\_words.sort()*

* Stem whole text

*arr = os.listdir(path)*

*print('Total Words Positive Negative Joy Fear Sadness Anger Surprise Disgust')*

*for i in arr:*

*path1 = 'C:\\Users\\kiit1\\Documents\\Python\\ABI Health\\ftragedy\\'+i*

*with open(path1, encoding="utf-8") as f:*

*c = f.read()*

*tokens = nltk.word\_tokenize(c,language='french')*

*no\_commas = re.sub(r'[.|,|\']',' ', c)*

*tokens = nltk.word\_tokenize(no\_commas)*

*text=nltk.Text(tokens)*

*stopword\_list = stopwords.words('french')*

*words=[w.lower() for w in text]*

*filtered\_words = []*

*for word in words:*

*if word not in stopword\_list and word.isalpha() and len(word) > 1:*

*filtered\_words.append(word)*

*filtered\_words.sort()*

*#print(filtered\_words)*

*stemmed\_words = []*

*stemmer = FrenchStemmer()*

*for word in filtered\_words:*

*stemmed\_word=stemmer.stem(word)*

*stemmed\_words.append(stemmed\_word)*

*stemmed\_words.sort()*

*#print(stemmed\_words)*

*total\_words = len(c)*

*#print(total\_row)*

*polarity\_pos = 0*

*polarity\_neg = 0*

*joy1 = 0*

*fear1 = 0*

*sadness1 = 0*

*anger1 = 0*

*surprise1 = 0*

*disgust1 = 0*

*n1=0*

*n2=0*

*n3=0*

*n4=0*

*n5=0*

*n6=0*

*n7=0*

*for j in stemmed\_words:*

*for k in dataset['word']:*

*if j == k:*

*df1=X[n1]*

*df2=Y[n2]*

*df3=Z[n3]*

*df4=P[n4]*

*df5=Q[n5]*

*df6=R[n6]*

*df7=S[n7]*

*if df1:*

*polarity\_pos = polarity\_pos+1*

*else:*

*polarity\_neg = polarity\_neg+1*

*if df2:*

*joy1 = joy1+1*

*if df3:*

*fear1 = fear1+1*

*if df4:*

*sadness1 = sadness1+1*

*if df5:*

*anger1 = anger1+1*

*if df6:*

*surprise1 = surprise1+1*

*if df7:*

*disgust1 = disgust1+1*

*n1= n1+1*

*n2 = n2+1*

*n3= n3+1*

*n4 = n4+1*

*n5 = n5+1*

*n6 = n6+1*

*n7= n7+1*

*print("\n",n8,"\t",total\_words,"\t",polarity\_pos,"\t",polarity\_neg,"\t ",joy1,"\t",fear1 ,"\t",sadness1,"\t ",anger1,"\t ",surprise1 ,"\t ",disgust1)*

*f.close()*

* Count all type of sentiments from text.

Output:

