

PHASE 4: Development Part 2

TOPIC: SENTIMENT ANALYSIS FOR MARKETING

Basic Python Libraries

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import string

from wordcloud import WordCloud

import re
```

Natural Language Processing Libraries

```
import nltk

from nltk.corpus import stopwords

from nltk.stem import WordNetLemmatizer
```

Scikit-Learn (Machine Learning Library for Python)

```
from sklearn.feature_extraction.text import CountVectorizer

from sklearn.model_selection import GridSearchCV

from sklearn.ensemble import RandomForestClassifier
```

Evaluation Metrics

```
from sklearn.metrics import
accuracy_score,precision_score,recall_score,confusion_matrix,roc_curve,classification_report

from scikitplot.metrics import plot_confusion_matrix
```

```
df_train = pd.read_csv("train.txt",delimiter=';',names=['text','label'])
```

```
df_val = pd.read_csv("val.txt",delimiter=';',names=['text','label'])
```

```
df = pd.concat([df_train,df_val])
```

```
df.reset_index(inplace=True,drop=True)
```

```
print("Shape of the DataFrame:",df.shape)
```

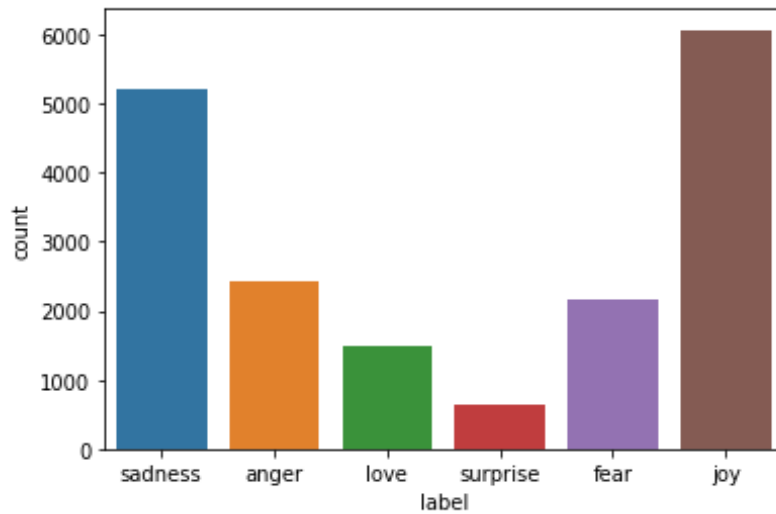
```
df.sample(5)
```

OUTPUT

	text	label
976	i feel kind of alone and helpless in	sadness
17072	i love your style and feel very comfortable wi...	joy
14098	i was little i always had this exciting jitter...	joy
15082	i just want the best for that boy maybe i can ...	anger
8044	i frantically try to get it done and now feel ...	fear

```
sns.countplot(df.label)
```

OUTPUT



```
dist_labels={}
for num,key in enumerate(list(set(df.label))):
    dist_labels[key]=num
print(dist_labels)
```

OUTPUT

```
{'anger': 0, 'fear': 1, 'joy': 2, 'sadness': 3, 'love': 4, 'surprise': 5}
```

```
df['label']=df['label'].map(dist_labels)
df.head()
```

OUTPUT

	text	label
0	i didnt feel humiliated	3
1	i can go from feeling so hopeless to so damned...	3
2	im grabbing a minute to post i feel greedy wrong	0
3	i am ever feeling nostalgic about the fireplac...	4
4	i am feeling grouchy	0

Data Pre-processing

#object of WordNetLemmatizer

lm = WordNetLemmatizer()

def clean_text(text):

corpus = []

text = ''.join([word.lower() for word in text if word not in string.punctuation])

tokens = re.split('\W+', text)

text = [lm.lemmatize(word) for word in tokens if word not in set(stopwords.words('english'))]

corpus.append(' '.join(str(x) for x in text))

return ' '.join(str(x) for x in text)

corpus = df['text'].apply(lambda x:clean_text(x))

Word Cloud

corpus.head()

OUTPUT

0 didnt feel humiliated

