SENTIMENT ANALYSIS USING VADER

In [ ]:
```
import pandas as pd
```

In [ ]:
```
from nltk.sentiment import vader
```

In [ ]:
```
from google.colab import drive
drive.mount('/content/drive')
df = pd.read_table(r"/content/drive/MyDrive/sentiment-topic-final-test.tsv")
```

```
Mounted at /content/drive
```

In [ ]:
```
df.head(10)
```

Out[ ]:

	sentence id	text	sentiment	topic
0	0	It took eight years for Warner Brothers to rec...	negative	movie
1	1	All the New York University students love this...	positive	restaurant
2	2	This Italian place is really trendy but they h...	negative	restaurant
3	3	In conclusion, my review of this book would be...	positive	book
4	4	The story of this movie is focused on Carl Bra...	neutral	movie
5	5	Chris O'Donnell stated that while filming for ...	neutral	movie
6	6	My husband and I moved to Amsterdam 6 years ag...	positive	restaurant
7	7	Dame Maggie Smith performed her role excellent...	positive	movie
8	8	The new movie by Mr. Kruno was shot in New Yor...	neutral	movie
9	9	I always have loved English novels, but I just...	negative	book

In [ ]:
```
df=df.get(["text","sentiment"])
```

In [ ]:
```
df.head()
```

Out[ ]:

	sentence id	text	sentiment	topic
0	0	It took eight years for Warner Brothers to rec...	negative	movie
1	1	All the New York University students love this...	positive	restaurant
2	2	This Italian place is really trendy but they h...	negative	restaurant
3	3	In conclusion, my review of this book would be...	positive	book
4	4	The story of this movie is focused on Carl Bra...	neutral	movie

In [ ]:
```
text=df.get("text")
sentiment=df.get("sentiment")
```

In [ ]:
```
def vader_output_to_label(vader_output):
    result = vader_output['compound']

    if result < 0:
        return 'negative'
    elif result == 0.0:
        return 'neutral'
    elif result > 0.0:
```

```
        return 'positive'

assert vader_output_to_label( {'neg': 0.0, 'neu': 0.0, 'pos': 1.0, 'compound': 0.0}) ==
assert vader_output_to_label( {'neg': 0.0, 'neu': 0.0, 'pos': 1.0, 'compound': 0.01}) ==
assert vader_output_to_label( {'neg': 0.0, 'neu': 0.0, 'pos': 1.0, 'compound': -0.01}) =
```

In [ ]:
```
import spacy
nlp = spacy.load('en_core_web_sm') # 'en_core_web_sm'
```

```
/usr/local/lib/python3.9/dist-packages/torch/cuda/__init__.py:497: UserWarning: Can't in
itialize NVML
  warnings.warn("Can't initialize NVML")
```

In [ ]:
```
def run_vader(textual_unit,
              lemmatize=False,
              parts_of_speech_to_consider=None,
              verbose=0):
    vader_input = []
    document = nlp(textual_unit)



    for i in document.sents:
        for token in i:
            to_add = token.text
            if lemmatize:
                to_add = token.lemma_

                if to_add == '-PRON-':
                    to_add = token.text

            if parts_of_speech_to_consider:
                if token.pos_ in parts_of_speech_to_consider:
                    vader_input.append(to_add)
            else:
                vader_input.append(to_add)

    scores = vader_model.polarity_scores(' '.join(vader_input))

    if verbose >= 1:
        print()
        print('INPUT SENTENCE', sent)
        print('INPUT TO VADER', vader_input)
        print('OUTPUT', scores)

    return scores
```

In [ ]:
```
import nltk
nltk.downloader.download('vader_lexicon')
```

```
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
```

Out[ ]:
```
True
```

In [ ]:
```
from nltk.sentiment.vader import SentimentIntensityAnalyzer
vader_model = SentimentIntensityAnalyzer()
all_vader_output = []
gold = sentiment

to_lemmatize = True
pos = set()

for i in text:
    vader_output = run_vader(i,to_lemmatize)
    vader_label = vader_output_to_label(vader_output)
```

```
    all_vader_output.append(vader_label)
for i in range(len(all_vader_output)):
  print("ID: ",i+1,"\nText: " ,text[i],"\nGold: ",gold[i],"\nOutput: ",all_vader_output[
```

```
ID:  1
Text:  It took eight years for Warner Brothers to recover from the disaster that was thi
s movie.
Gold:  negative
Output:  negative


ID:  2
Text:  All the New York University students love this diner in Soho so it makes for a fu
n young atmosphere.
Gold:  positive
Output:  positive


ID:  3
Text:  This Italian place is really trendy but they have forgotten about the most import
ant part of a restaurant, the food.
Gold:  negative
Output:  positive


ID:  4
Text:  In conclusion, my review of this book would be: I like Jane Austen and understand
why she is famous.
Gold:  positive
Output:  positive


ID:  5
Text:  The story of this movie is focused on Carl Brashear played by Cuba Gooding Jr. wh
o wants to be the first African American deep sea diver in the Navy.
Gold:  neutral
Output:  positive


ID:  6
Text:  Chris O'Donnell stated that while filming for this movie, he felt like he was in
a toy commercial.
Gold:  neutral
Output:  positive


ID:  7
Text:  My husband and I moved to Amsterdam 6 years ago and for as long as we have lived
here, Blauwbrug has been our favorite place to eat!
Gold:  positive
Output:  positive


ID:  8
Text:  Dame Maggie Smith performed her role excellently, as she does in all her movies.
Gold:  positive
Output:  positive


ID:  9
Text:  The new movie by Mr. Kruno was shot in New York, but the story takes place in Los
Angeles.
Gold:  neutral
Output:  negative
```

```
ID:  10
Text:  I always have loved English novels, but I just couldn't get into this one.
Gold:  negative
Output:  positive
```

In [ ]:

```python
import sklearn
from sklearn.metrics import classification_report

report = classification_report(gold,all_vader_output,digits = 3)

print(report)
```

```
              precision    recall  f1-score   support

    negative      0.500     0.333     0.400         3
     neutral      0.000     0.000     0.000         3
    positive      0.500     1.000     0.667         4

    accuracy                          0.500        10
   macro avg      0.333     0.444     0.356        10
weighted avg      0.350     0.500     0.387        10
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/metrics/_classification.py:1344: Undefine
dMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels wit
h no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.9/dist-packages/sklearn/metrics/_classification.py:1344: Undefine
dMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels wit
h no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.9/dist-packages/sklearn/metrics/_classification.py:1344: Undefine
dMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels wit
h no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

In [ ]: