The Superior University

Name: Fatima Nadeem	Roll No:076	Course: PAI Lab
Semester:4th	Section:4B	Department:SE
Submitted To: Sir Rasikh	Total Marks: 10	Date:

Lab Task 09

NLP TASK SentimentIntensityAnalyzer

The **SentimentIntensityAnalyzer** is a tool used in natural language processing (NLP) to determine the emotional tone behind a body of text. It is part of the VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment analysis tool, which is particularly effective for analyzing social media texts and other informal communication. The SentimentIntensityAnalyzer works by assigning sentiment scores—positive, negative, neutral, and a compound score—to a given input. The compound score, a normalized value between -1 (most negative) and +1 (most positive), summarizes the overall sentiment of the text. This task is crucial in applications such as customer feedback analysis, opinion mining, and social media monitoring, where understanding public sentiment can drive business and policy decisions.

CODE:

```
# import SentimentIntensityAnalyzer class from vaderSentiment.vaderSentiment module.
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer


# Function to print sentiments of the sentence.
def sentiment_scores(sentence):


    # Create a SentimentIntensityAnalyzer object.
    sid_obj = SentimentIntensityAnalyzer()


    # polarity_scores method of SentimentIntensityAnalyzer object gives a sentiment
dictionary.
    # which contains pos, neg, neu, and compound scores.
    sentiment_dict = sid_obj.polarity_scores(sentence)


    print("Overall sentiment dictionary is : ", sentiment_dict)
    print("Sentence was rated as ", sentiment_dict['neg']*100, "% Negative")
    print("Sentence was rated as ", sentiment_dict['neu']*100, "% Neutral")
    print("Sentence was rated as ", sentiment_dict['pos']*100, "% Positive")


    print("Sentence Overall Rated As", end=" ")


    # Decide sentiment as positive, negative, or neutral
    if sentiment_dict['compound'] >= 0.05 :
        print("Positive")
    elif sentiment_dict['compound'] <= -0.05 :
        print("Negative")
    else :
        print("Neutral")
```

```
# Driver code to test the function

if __name__ == "__main__" :


   print("\n1st Statement:")

   sentence = "Geeks For Geeks is the best portal for computer science engineering
students."

   sentiment_scores(sentence)


   print("\n2nd Statement:")

   sentence = "Study is going on as usual."

   sentiment_scores(sentence)


   print("\n3rd Statement:")

   sentence = "Today isa good day."

   sentiment_scores(sentence)
```

```
1st Statement:
Overall sentiment dictionary is :  {'neg': 0.175, 'neu': 0.562, 'pos': 0.263, 'compound': 0.5267}
Sentence was rated as  17.5 % Negative
Sentence was rated as  56.2 % Neutral
Sentence was rated as  26.3 % Positive
Sentence Overall Rated As Positive

2nd Statement:
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}
Sentence was rated as  0.0 % Negative
Sentence was rated as  100.0 % Neutral
Sentence was rated as  0.0 % Positive
Sentence Overall Rated As Neutral

3rd Statement:
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 0.508, 'pos': 0.492, 'compound': 0.4404}
Sentence was rated as  0.0 % Negative
Sentence was rated as  50.8 % Neutral
Sentence was rated as  49.2 % Positive
Sentence Overall Rated As Positive
```