Experiment 11: Opinion analysis

Opinion mining, also known as sentiment analysis, involves determining the sentiment (positive, negative, or neutral) expressed in a piece of text. There are various techniques and tools available for opinion mining, ranging from rule-based methods to machine learning approaches. In this program, demonstrate a simple implementation of opinion mining using the VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment analysis tool, which is available in the NLTK (Natural Language Toolkit) library for Python.

First, you need to install the NLTK library if you haven't already. You can install it using pip:

pip install nltk

Next, you can use the following Python code to perform opinion mining using the VADER sentiment analysis tool:

```
import nltk

from nltk.sentiment.vader import SentimentIntensityAnalyzer

# Download VADER lexicon for sentiment analysis

nltk.download('vader_lexicon')

# Create a SentimentIntensityAnalyzer object

sid = SentimentIntensityAnalyzer()

# Sample texts for opinion mining

texts = [

    "I love this product! It's amazing.",

    "The service was terrible. I'm very disappointed.",

    "This movie is neither good nor bad.",

    "I don't have any strong feelings about this issue."

]

# Perform sentiment analysis on each text

for text in texts:

    # Analyze the sentiment of the text

    sentiment_scores = sid.polarity_scores(text)

        # Determine the sentiment label based on the compound score

    if sentiment_scores['compound'] >= 0.05:
```

```
        sentiment = 'Positive'
    elif sentiment_scores['compound'] <= -0.05:
        sentiment = 'Negative'
    else:
        sentiment = 'Neutral'
        # Print the text and its sentiment
    print(f"Text: '{text}'")
    print(f"Sentiment: {sentiment} (Compound Score: {sentiment_scores['compound']})")
    print("-" * 30)
```

In this program, the SentimentIntensityAnalyzer class from the NLTK library's nltk.sentiment.vader module is used to perform sentiment analysis on the given texts. The polarity_scores method returns a dictionary containing positive, negative, neutral, and compound scores. The compound score represents the overall sentiment of the text. You can adjust the threshold values (0.05 and -0.05 in this case) to classify the sentiment as positive, negative, or neutral based on the compound score.

Make sure you have NLTK installed and the VADER lexicon downloaded to run this code successfully. You can modify the texts list with your own input texts to analyze different opinions.