# **PROGRAM :**

import pandas as pd

import nltk

from nltk.sentiment.vader import SentimentIntensityAnalyzer

import matplotlib.pyplot as plt

import seaborn as sns

from wordcloud import WordCloud

from collections import Counter

import math

# Download the VADER lexicon (only need to do this once)

nltk.download('vader\_lexicon')

# Sample marketing text data

data = {

'Text': [

"Amazon Puma",

"Myntra Ajio ",

"Adidas Amazon woodland ",

"Ajio Adidas Ajio Reliance Bruton ",

"Flipkart Puma ",

"Reliance Ajio Puma ",

"Puma Reliance Ajio woodland ",

"woodland Myntra Flipkart ",

]

} ]

}

# Create a DataFrame from the sample data

df = pd.DataFrame(data)

# Create a SentimentIntensityAnalyzer object

sia = SentimentIntensityAnalyzer()

# Calculate sentiment scores for each text in the DataFrame

df['Sentiment Scores'] = df['Text'].apply(lambda x: sia.polarity\_scores(x))

# Extract compound scores and sentiment labels

df['Compound'] = df['Sentiment Scores'].apply(lambda x: x['compound'])

df['Sentiment'] = df['Compound'].apply(lambda x: 'Positive' if x >= 0.05 else ('Negative' if x <= -0.05 else 'Neutral'))

# Calculate word frequencies

words = " ".join(df['Text']).split()

word\_freq = Counter(words)

# Visualize sentiment distribution and word cloud

plt.figure(figsize=(16, 8))

# Sentiment Distribution

plt.subplot(2, 2, 1)

sns.countplot(data=df, x='Sentiment', palette='Set2')

plt.xlabel('Sentiment')

plt.ylabel('Count')

plt.title('Sentiment Distribution')

# Sentiment Distribution Pie Chart

plt.subplot(2, 2, 2)

sentiment\_stats = df['Sentiment'].value\_counts(normalize=True).reset\_index()

sentiment\_stats.columns = ['Sentiment', 'Percentage']

plt.pie(sentiment\_stats['Percentage'], labels=sentiment\_stats['Sentiment'], autopct='%1.1f%%', colors=['lightgreen', 'lightcoral', 'lightblue'])

plt.title('Sentiment Distribution (Percentage)')

# Word Cloud

plt.subplot(2, 2, 3)

wordcloud = WordCloud(width=800, height=400, background\_color='white').generate\_from\_frequencies(word\_freq)

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis('off')

plt.title('Word Cloud - Most Frequent Words')

# Word Frequency Bar Chart (Top 10 words)

plt.subplot(2, 2, 4)

common\_words = word\_freq.most\_common(10)

common\_words\_df = pd.DataFrame(common\_words, columns=['Word', 'Frequency'])

sns.barplot(data=common\_words\_df, x='Frequency', y='Word', palette='Set2')

plt.xlabel('Frequency')

plt.ylabel('Word')

plt.title('Top 10 Most Frequent Words')

plt.tight\_layout()

plt.show()

# Display detailed statistics

print("\nSentiment Statistics:")

print(sentiment\_stats)

# Explore specific examples of each sentiment

positive\_example = df[df['Sentiment'] == 'Positive']['Text'].iloc[0]

negative\_example = df[df['Sentiment'] == 'Negative']['Text'].iloc[0]

neutral\_example = df[df['Sentiment'] == 'Neutral']['Text'].iloc[0]

print("\nExample of Positive Sentiment:")

print(positive\_example)

print("\nExample of Negative Sentiment:")

print(negative\_example)

print("\nExample of Neutral Sentiment:")

print(neutral\_example)