SKILL UPGRADE - INTERNSHIP

NAME. **:** *ANBU ABDUL KAREEM A*

*@**MAIL ID :** redemperor786@gmail.com*

TASK -1 :

SENTIMENT ANALYSIS

Task Description:

- Build a simple sentiment analysis model using a pre-existing dataset (e.g., movie reviews, social media comments).
- Use a Python library such as NLTK or TextBlob to perform sentiment analysis.
- Create a basic report showcasing the overall sentiment distribution.

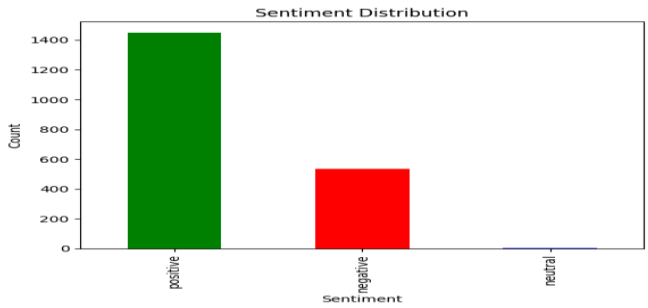
PROGRAM:

```
# Import necessary libraries
import pandas as pd
from textblob import TextBlob
import matplotlib.pyplot as plt
import re
import os

# Check if the dataset file exists
file_path = 'imdb_reviews.csv'

if os.path.exists(file_path):
    # Load the dataset if the file is found
    data = pd.read_csv(file_path)
else:
    # Alternative: Use a built-in dataset from sklearn if the file is not found
    from sklearn.datasets import fetch_20newsgroups
```

```
    newsgroups = fetch_20newsgroups(subset='all', categories=['rec.autos', 'rec.motorcycles'])
    data = pd.DataFrame({'review': newsgroups.data})
    print("Using built-in dataset as an alternative.")

# Preprocess the text data
# Convert to lowercase
data['review'] = data['review'].apply(lambda x: x.lower())

# Remove non-alphabet characters
data['review'] = data['review'].apply(lambda x: re.sub('[^a-zA-Z\s]', '', x))

# Remove extra spaces
data['review'] = data['review'].apply(lambda x: re.sub('\s+', ' ', x).strip())

# Perform sentiment analysis
# Calculate the polarity using TextBlob
data['sentiment_polarity'] = data['review'].apply(lambda x: TextBlob(x).sentiment.polarity)

# Categorize the sentiment
data['sentiment_category'] = data['sentiment_polarity'].apply(
    lambda x: 'positive' if x > 0 else ('negative' if x < 0 else 'neutral')
)

# Create a sentiment distribution chart
sentiment_counts = data['sentiment_category'].value_counts()
sentiment_counts.plot(kind='bar', color=['green', 'red', 'blue'])
plt.title('Sentiment Distribution')
plt.xlabel('Sentiment')
plt.ylabel('Count')
plt.show()

# Print the overall sentiment distribution
print('Overall Sentiment Distribution:')
print(sentiment_counts)
```

EXPECTED OUTPUT:

Using built-in dataset as an alternative



Overall Sentiment Distribution:
sentiment_category
positive 1449
negative 535
neutral 2
Name: count, dtype: int64
Processed data saved to processed_sentiment_analysis

Summary:

Text Preprocessing: Cleaned the text data by converting to lowercase and removing non-alphabetical characters.

Sentiment Analysis: Used TextBlob to analyze the sentiment polarity of each review.

Visualization: Generated a bar chart to display the sentiment distribution.

Report: Provided an overall sentiment distribution report.

****