

# Sentiment Analysis of User Reviews

## Introduction

This report details the approach taken to analyze the sentiment of user reviews from a given CSV file. The dataset contains user reviews, and the goal was to preprocess the text data, perform sentiment analysis, and generate a summary report of the sentiment distribution.

## Approach

Step 1: Load the Dataset

The dataset was loaded using the Pandas library.

Step 2: Data Cleaning

Data cleaning involved removing any rows with null values in the 'review' column and keeping only the 'id' and 'review' columns.

Step 3: Text Preprocessing

Text preprocessing included converting the text to lowercase and removing punctuation to standardize the text data.

Step 4: Sentiment Analysis

The TextBlob library was used to perform sentiment analysis. Each review was classified as positive, negative, or neutral based on its sentiment polarity.

Step 5: Generate Summary Report

## Challenges Faced

1. **Handling File Path:** Ensuring the file path was correctly interpreted by the script, especially since it contained backslashes which are escape characters in Python strings. Using a raw string (`r""`) helped resolve this.
2. **Text Preprocessing:** Deciding on the appropriate preprocessing steps, such as lowercasing and punctuation removal, to ensure uniformity without losing the sentiment context.
3. **Sentiment Analysis Accuracy:** The basic sentiment analysis using TextBlob may not capture complex sentiments accurately, but it was sufficient for a basic analysis.

## Assumptions Made

1. **Basic Preprocessing:** Assumed that lowercasing and punctuation removal were adequate preprocessing steps for this analysis.
2. **TextBlob for Sentiment Analysis:** Assumed that TextBlob's sentiment polarity scores were sufficient for classifying reviews into positive, negative, and neutral categories

## Conclusion

The process successfully loaded, cleaned, preprocessed, and analyzed the sentiment of user reviews, providing a basic distribution of sentiments. The approach and tools used were effective for a preliminary analysis, although further refinement could improve the accuracy of sentiment classification.

```
In [1]: pip install pandas textblob
```

```
Requirement already satisfied: pandas in c:\users\pooja\anaconda3\lib\site-packages (1.5.3)
Collecting textblob
  Downloading textblob-0.18.0.post0-py3-none-any.whl.metadata (4.5 kB)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\pooja\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\pooja\anaconda3\lib\site-packages (from pandas) (2022.7)
Requirement already satisfied: numpy>=1.21.0 in c:\users\pooja\anaconda3\lib\site-packages (from pandas) (1.24.3)
Requirement already satisfied: nltk>=3.8 in c:\users\pooja\anaconda3\lib\site-packages (from textblob) (3.8.1)
Requirement already satisfied: click in c:\users\pooja\anaconda3\lib\site-packages (from nltk>=3.8->textblob) (8.0.4)
Requirement already satisfied: joblib in c:\users\pooja\anaconda3\lib\site-packages (from nltk>=3.8->textblob) (1.2.0)
Requirement already satisfied: regex>=2021.8.3 in c:\users\pooja\anaconda3\lib\site-packages (from nltk>=3.8->textblob) (2022.7.9)
Requirement already satisfied: tqdm in c:\users\pooja\anaconda3\lib\site-packages (from nltk>=3.8->textblob) (4.65.0)
Requirement already satisfied: six>=1.5 in c:\users\pooja\anaconda3\lib\site-packages (from python-dateutil>=2.8.1->pandas) (1.16.0)
Requirement already satisfied: colorama in c:\users\pooja\anaconda3\lib\site-packages (from click->nltk>=3.8->textblob) (0.4.6)
```

```
import pandas as pd
import string
from textblob import TextBlob

# Step 1: Load the Dataset
file_path = r"C:\Users\pooja\Downloads\user_review.csv"
df = pd.read_csv(file_path)
print("Dataset Loaded Successfully!")

# Step 2: Data Cleaning
df = df.dropna(subset=['review']) # Remove rows where 'review' is null

# Keep only 'id' and 'review' columns
df = df[['id', 'review']]
print("Data Cleaned Successfully!")

# Step 3: Text Preprocessing
def preprocess_text(text):
    text = text.lower() # Convert text to Lowercase
    text = text.translate(str.maketrans('', '', string.punctuation)) # Remove punctuation
    return text

df['review'] = df['review'].apply(preprocess_text)
print("Text Preprocessing Done!")
```

```

# Step 4: Sentiment Analysis
def analyze_sentiment(review):
    analysis = TextBlob(review)
    if analysis.sentiment.polarity > 0:
        return 'positive'
    elif analysis.sentiment.polarity < 0:
        return 'negative'
    else:
        return 'neutral'

df['sentiment'] = df['review'].apply(analyze_sentiment)
print("Sentiment Analysis Completed!")

# Step 5: Generate Summary Report
sentiment_distribution = df['sentiment'].value_counts()
print("Sentiment Distribution Report:")
print(sentiment_distribution)

```

```

Dataset Loaded Successfully!
Data Cleaned Successfully!
Text Preprocessing Done!
Sentiment Analysis Completed!
Sentiment Distribution Report:
positive      32
negative      10
neutral        8

```

---

```

Dataset Loaded Successfully!
Data Cleaned Successfully!
Text Preprocessing Done!
Sentiment Analysis Completed!
Sentiment Distribution Report:
positive      32
negative      10
neutral        8
Name: sentiment, dtype: int64

```

