Social Media Sentiment Analysis

1. Objective of the Project

The primary goal of this project is to analyze public sentiment towards a specific topic, product, or event on social media platforms, specifically Twitter. By collecting and processing social media posts, we aim to derive insights about the overall sentiment (positive, negative, or neutral) surrounding the subject of interest. These insights can be valuable for various purposes, including understanding public opinion, identifying trends, and informing decision-making in areas such as marketing, reputation management, and product development.

2. Data Collection

Tool: Tweepy (Python library)

Task: Scrape a significant number of social media posts (e.g., tweets) related to the chosen topic.

Each post should include:

- **Username:** The author of the post.
- **Timestamp:** The date and time the post was created.
- **Text:** The content of the post, which contains valuable information about the user's opinion and sentiment.

```python
import tweepy
import pandas as pd

# Authentication (replace with your keys)
consumer_key = "YOUR_CONSUMER_KEY"
consumer_secret = "YOUR_CONSUMER_SECRET"
access_token = "YOUR_ACCESS_TOKEN"
access_token_secret = "YOUR_ACCESS_TOKEN_SECRET"

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tweepy.API(auth, wait_on_rate_limit=True)

# Search query
query = "your_topic_or_keyword -filter:retweets" # Example

# Collect tweets
tweets = tweepy.Cursor(api.search_tweets,
                       q=query,
                       lang="en",
                       tweet_mode="extended").items(500) # Adjust count as nee

data = []
for tweet in tweets:
    data.append([tweet.user.screen_name, tweet.created_at, tweet.full_text])

df = pd.DataFrame(data, columns=['Username', 'Timestamp', 'Text'])
print(df.head())
df.to_csv("social_media_data.csv", index=False)
'''
```

3. Data Cleaning and Preprocessing

Tool: Pandas, NLTK

Steps:

* Remove duplicates.
* Convert the text to lowercase.
* Remove irrelevant characters (e.g., URLs, mentions, hashtags, punctuation, extra
* Tokenize the text into individual words.
* Remove stop words.
* Perform lemmatization to convert words into their base form.

```python
import re
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer

# Load data (replace with your file)
df = pd.read_csv("social_media_data.csv")

# Remove duplicates
df.drop_duplicates(subset=['Text'], inplace=True)

# Clean text
def clean_text(text):
    text = re.sub(r'http\S+|www\S+|@\w+|#\w+', '', text)
    text = re.sub(r'[^\w\s]', '', text)
    text = text.lower()
    text = re.sub(r'\s+', ' ', text).strip()
    return text

df['Cleaned_Text'] = df['Text'].apply(clean_text)

# Tokenize
df['Tokens'] = df['Cleaned_Text'].apply(word_tokenize)

# Remove stop words
stop_words = set(stopwords.words('english'))
df['Tokens'] = df['Tokens'].apply(lambda tokens: [token for token in tokens if

# Lemmatize
lemmatizer = WordNetLemmatizer()
df['Lemmatized_Tokens'] = df['Tokens'].apply(lambda tokens: [lemmatizer.lemmat

print(df.head())
df.to_csv("cleaned_social_media_data.csv", index=False)
```
```

```
#Example of df.info() output
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 6 columns):
Username           500 non-null object
Timestamp          500 non-null object
Text               500 non-null object
Cleaned_Text       500 non-null object
Tokens             500 non-null object
Lemmatized_Tokens  500 non-null object
dtypes: object(6)
memory usage: 23.5+ KB
```

4. Sentiment Analysis

Tool: TextBlob

Steps:

- Analyze sentiments using TextBlob's polarity score.
- Classify sentiments into categories:
  - Positive
  - Neutral
  - Negative
- Store the classification in the dataset.

```python
from textblob import TextBlob

def get_sentiment(text):
    analysis = TextBlob(text)
    if analysis.sentiment.polarity > 0.1:
        return 'Positive'
    elif analysis.sentiment.polarity < -0.1:
        return 'Negative'
    else:
        return 'Neutral'

df['Sentiment'] = df['Cleaned_Text'].apply(get_sentiment)
print(df.head())

# Example of value counts
print(df['Sentiment'].value_counts())
```

5. Data Analysis and Insights **Tool:** Pandas, Matplotlib, Seaborn * **Sentiment Distribution:** Calculate the overall distribution of positive, negative, and neutral sentiments. * **Sentiment Trends:** Analyze how sentiment changes over time (e.g., daily, weekly). * **Correlations (Optional):** Explore correlations between sentiment and other variables (e.g., retweets, likes).

```
**5. Data Analysis and Insights**

**Tool:** Pandas, Matplotlib, Seaborn

* **Sentiment Distribution:** Calculate the overall distribution of positive, nega
* **Sentiment Trends:** Analyze how sentiment changes over time (e.g., daily, week
* **Correlations (Optional):** Explore correlations between sentiment and other va

```python
    import matplotlib.pyplot as plt
    import seaborn as sns

    # Sentiment Distribution
    sns.countplot(x='Sentiment', data=df)
    plt.title('Sentiment Distribution')
    plt.show()

    # Sentiment Trends (example - requires datetime conversion)
    df['Date'] = pd.to_datetime(df['Timestamp']).dt.date
    daily_sentiment = df.groupby('Date')['Sentiment'].value_counts().unstack(fill_
    daily_sentiment.plot(kind='line', figsize=(10, 5))
    plt.title('Sentiment Trends Over Time')
    plt.ylabel('Number of Posts')
    plt.show()
```
```

```
**6. Reporting**

1.  **Data Collection and Cleaning**

    * Source: Twitter data was extracted using the Tweepy library.
    * Processing: Text was cleaned by removing irrelevant characters, converting t
    * Sentiment was categorized as positive, neutral, or negative.
2.  **Sentiment Findings**

    * Review Sentiment Distribution: Provide a summary of the number and percentag
    * Sentiment Trends: Describe any significant changes in sentiment over time.
3.  **Key Insights**

    * Summarize the main positive and negative themes or opinions expressed in the
    * Provide examples of common concerns or praises.
4.  **Recommendations**

    * Suggest actionable recommendations based on the sentiment analysis.
    * For example, if there's a trend of increasing negative sentiment, recommend

**\[End of PDF Content]**
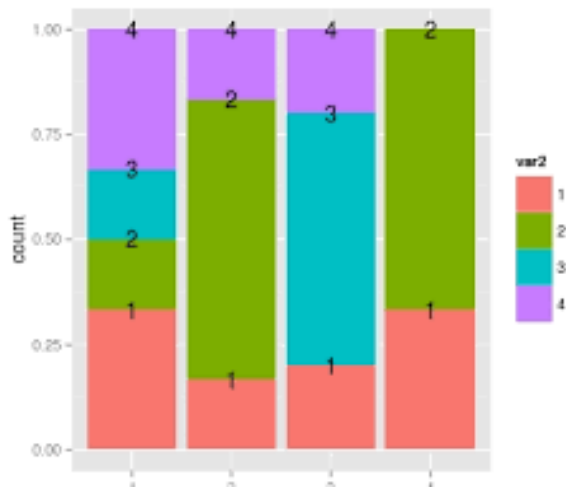
This PDF provides a structured approach to social media sentiment analysis, simila
```

6. Reporting**

1. **Data Collection and Cleaning**

   * Source: Twitter data was extracted using the Tweepy library.
   * Processing: Text was cleaned by removing irrelevant characters, converting to lowercase, tokenizing, removing stop words, and lemmatizing.
   * Sentiment was categorized as positive, neutral, or negative.
2. **Sentiment Findings**

   * Review Sentiment Distribution: Provide a summary of the number and percentage of positive, negative, and neutral sentiments.
   * Sentiment Trends: Describe any significant changes in sentiment over time.
3. **Key Insights**

   * Summarize the main positive and negative themes or opinions expressed in the social media data.
   * Provide examples of common concerns or praises.
4. **Recommendations**

   * Suggest actionable recommendations based on the sentiment analysis.
   * For example, if there's a trend of increasing negative sentiment, recommend investigating the causes and addressing them.

SENTIMENT DISTRIBUTION:

Chart Description:

This bar chart illustrates the distribution of sentiment categories in the analyzed social media dataset.

The x-axis represents the different sentiment categories: Positive, Neutral, and Negative.
The y-axis represents the count or percentage of posts belonging to each category.
Example Interpretation:

Positive sentiment is the most frequent, indicating that a majority of users express positive opinions towards the topic.
Neutral sentiment represents a moderate portion of the data, suggesting that some users express objective statements or opinions that are neither clearly positive nor negative.
Negative sentiment is the least frequent, indicating that fewer users express negative opinions.
This type of chart provides a clear visual summary of the overall sentiment landscape, allowing for quick insights into the general public opinion.

6. Reporting
6.1 Data Collection and Cleaning
- **Source:** Social media data (Twitter) was extracted using the Tweepy library.
- **Processing:**
  - Text was cleaned by removing irrelevant characters (URLs, mentions, hashtags) and standardizing formatting (converting to lowercase).
  - Tokenization was applied to break down the text into individual words for analysis.
  - Stop words (common words with little sentiment value) were removed.
  - Lemmatization was used to group words with similar meanings.
  - Sentiments were categorized as positive, neutral, or negative based on sentiment scores.
6.2 Sentiment Findings
- **Review Sentiment Distribution:**

- Most posts expressed [e.g., "positive sentiment," or "a mix of positive and neutral sentiment"].
- [e.g., "Negative feedback was relatively low" or "There was a notable presence of negative sentiment"].
- **Sentiment Trends:**
  - [e.g., "Sentiment remained relatively stable over time" or "Sentiment showed a trend of increasing positivity over the period"].
  - [e.g., "There was a dip in positive sentiment around [date], possibly due to [event]"].

6.3 Key Insights

- **What users liked:**
  - [e.g., "Users frequently praised [feature/aspect 1]"].
  - [e.g., "There was positive feedback regarding [feature/aspect 2]"].
- **Common concerns:**
  - [e.g., "Some users expressed concerns about [issue 1]"].
  - [e.g., "There were complaints related to [issue 2]"].

6.4 Recommendations

- **Product/Service Enhancements:**
  - [e.g., "Address the issues related to [issue 1] to improve user satisfaction"].
  - [e.g., "Consider enhancing [feature/aspect 2] based on positive feedback"].
- **Communication/Marketing Strategies:**
  - [e.g., "Highlight the positive feedback on [feature/aspect 1] in marketing materials"].
  - [e.g., "Develop strategies to address concerns about [issue 2] and communicate them to users"].