```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from wordcloud import WordCloud
import re



import nltk
nltk.download('punkt_tab')
```

```
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
True
```

```
df = pd.read_csv('/content/sentimentdataset.csv')

df['Timestamp'] = pd.to_datetime(df['Timestamp'])

print(df.columns)

df = df.drop(columns=['Unnamed: 0'])

nltk.download('punkt')
nltk.download('stopwords')

nltk.download('punkt_tab')

stop_words = set(stopwords.words('english'))

def preprocess_text(text):
    text = re.sub(r'http\S+|www\S+|https\S+', '', text)
    text = re.sub(r'[^A-Za-z0-9\s]+', '', text)
    tokens = word_tokenize(text.lower())
    tokens = [word for word in tokens if word not in stop_words]
    return ' '.join(tokens)

df['Cleaned_Text'] = df['Text'].apply(preprocess_text)
```

```
Index(['Unnamed: 0.1', 'Unnamed: 0', 'Text', 'Sentiment', 'Timestamp', 'User',
       'Platform', 'Hashtags', 'Retweets', 'Likes', 'Country', 'Year', 'Month',
       'Day', 'Hour'],
      dtype='object')
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Package punkt_tab is already up-to-date!
```

```
nltk.download('vader_lexicon')
sid = SentimentIntensityAnalyzer()

df['Sentiment_Scores'] = df['Cleaned_Text'].apply(lambda x: sid.polarity_scores(x))
df['Compound'] = df['Sentiment_Scores'].apply(lambda x: x['compound'])

def classify_sentiment(compound_score):
    if compound_score >= 0.05:
        return 'Positive'
    elif compound_score <= -0.05:
        return 'Negative'
    else:
        return 'Neutral'

df['Sentiment'] = df['Compound'].apply(classify_sentiment)
```

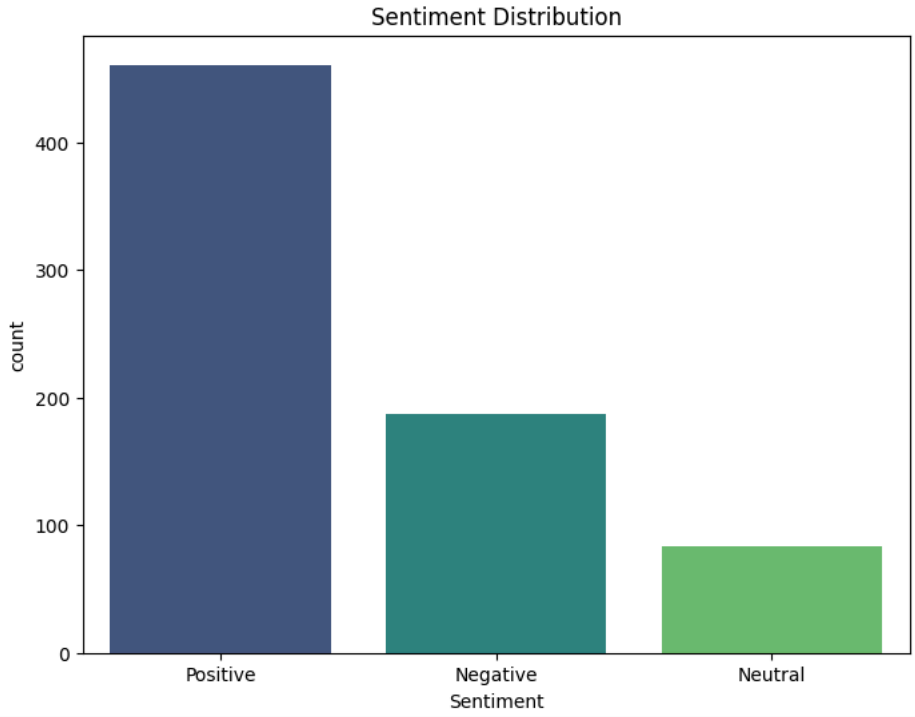
```
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
```

```
plt.figure(figsize=(8, 6))
sns.countplot(x='Sentiment', data=df, palette='viridis')
plt.title('Sentiment Distribution')
plt.show()
```

```
<ipython-input-10-468704df2c73>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

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




```
positive_text = ' '.join(df[df['Sentiment'] == 'Positive']['Cleaned_Text'])
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(positive_text)

plt.figure(figsize=(10, 8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('Word Cloud for Positive Sentiments')
plt.axis('off')
plt.show()
```


```
df = pd.read_csv('/content/sentimentdataset.csv')


df['Timestamp'] = pd.to_datetime(df['Timestamp'])

print(df.columns)

df = df.drop(columns=['Unnamed: 0']) # Assuming 'Unnamed: 0' is the column to drop
import nltk
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('punkt_tab')
```

```
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords

stop_words = set(stopwords.words('english'))

def preprocess_text(text):
    text = re.sub(r'http\S+|www\S+|https\S+', '', text)
    text = re.sub(r'[^A-Za-z0-9\s]+', '', text)
    tokens = word_tokenize(text.lower())
    tokens = [word for word in tokens if word not in stop_words]
    return ' '.join(tokens)

df['Cleaned_Text'] = df['Text'].apply(preprocess_text)
df.set_index('Timestamp', inplace=True)
df_resampled = df.resample('D').agg({'Sentiment': lambda x: x.mode()[0] if not x.empty else np.nan})

plt.figure(figsize=(12, 6))
df_resampled['Sentiment'].value_counts().plot(kind='line', marker='o')
plt.title('Sentiment Trend Over Time')
plt.ylabel('Frequency')
plt.show()
```

```
Index(['Unnamed: 0.1', 'Unnamed: 0', 'Text', 'Sentiment', 'Timestamp', 'User',
       'Platform', 'Hashtags', 'Retweets', 'Likes', 'Country', 'Year', 'Month',
       'Day', 'Hour'],
      dtype='object')
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Package punkt_tab is already up-to-date!
```