```
pip install nltk
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.9.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.4.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2024.9.11)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.6)
```

```
pip install wordcloud
```

```
Requirement already satisfied: wordcloud in /usr/local/lib/python3.10/dist-packages (1.9.4)
Requirement already satisfied: numpy>=1.6.1 in /usr/local/lib/python3.10/dist-packages (from wordcloud) (1.26.4)
Requirement already satisfied: pillow in /usr/local/lib/python3.10/dist-packages (from wordcloud) (11.0.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (from wordcloud) (3.8.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (4.55.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (1.4.7)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (24.2)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib->wordcloud
```

```
pip install pandas
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.2.2)
Requirement already satisfied: numpy>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

Double-click (or enter) to edit

```
pip install numpy
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (1.26.4)
```

```
pip install seaborn
```

```
Requirement already satisfied: seaborn in /usr/local/lib/python3.10/dist-packages (0.13.2)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in /usr/local/lib/python3.10/dist-packages (from seaborn) (1.26.4)
Requirement already satisfied: pandas>=1.2 in /usr/local/lib/python3.10/dist-packages (from seaborn) (2.2.2)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in /usr/local/lib/python3.10/dist-packages (from seaborn) (3.8.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (0.12
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (2
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (11
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seabor
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.2->seaborn) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.2->seaborn) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud
from collections import Counter
import nltk
from nltk.corpus import stopwords
def warn(*args, **kwargs):
   pass
import warnings
warnings.warn = warn
warnings.filterwarnings('ignore')
```

```
df = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/-H0BcPr512mhbTDDfkcTJA/comments-with-emotions.csv')
```

```
df
```

	comment	like_count	published_at	author	sentiment	emotions	anger	fear	negative	positive	tru
0	If this fight were real and not just an exhibi...	0	2024-10-26T01:19:27Z	@thesecond4113	Positive	{'anger': 3, 'fear': 3, 'negative': 2, 'positi...	3.0	3.0	2.0	8.0	7
1	Tyson will bust his hide. People forget with h...	0	2024-10-26T01:09:55Z	@bluesslider76	Neutral	{'fear': 2, 'negative': 3, 'anticipation': 1, ...	1.0	2.0	3.0	0.0	0
2	We signing the contract with this one 🗣️🗣️🔥🔥🔥🔥	0	2024-10-26T00:19:29Z	@thechaosmaster9934	Neutral	{}	0.0	0.0	0.0	0.0	0
3	Watching cm punk fight in the ufc is better th...	0	2024-10-26T00:16:10Z	@tokesenari	Positive	{'anger': 2, 'fear': 1, 'negative': 1}	2.0	1.0	1.0	0.0	0
4	Yall forget how much Ring IQ this guy has. AND...	0	2024-10-25T23:53:49Z	@HumbleAstronaut	Positive	{'negative': 1, 'positive': 1, 'anticipation': 1}	0.0	0.0	1.0	1.0	0
...	...	...	...	...	...	...	...	...	...	...	
1151	WTF	0	2024-10-15T13:01:03Z	@ischubbypinkcheekscute8408	Negative	{}	0.0	0.0	0.0	0.0	0
1152	Is this real or a joke?	42	2024-10-15T13:01:00Z	@moondancecoffee	Positive	{'positive': 1, 'trust': 1, 'negative': 1}	0.0	0.0	1.0	1.0	1
1153	Bruh	0	2024-10-15T13:00:56Z	@H20_BadOnMyEnd	Neutral	{}	0.0	0.0	0.0	0.0	0
1154	First comment 😂	0	2024-10-15T13:00:37Z	@josmith1184	Positive	{}	0.0	0.0	0.0	0.0	0
1155	Maa kasam jisne like subscribe na kiya wo exam...	1	2024-10-15T13:00:30Z	@hania.5911	Negative	{'anticipation': 1}	0.0	0.0	0.0	0.0	0

1156 rows × 17 columns

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1156 entries, 0 to 1155
Data columns (total 17 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   comment         1156 non-null   object 
 1   like_count      1156 non-null   int64  
 2   published_at    1156 non-null   object 
 3   author          1156 non-null   object 
 4   sentiment       1156 non-null   object 
 5   emotions        1156 non-null   object 
 6   anger           1156 non-null   float64
 7   fear            1156 non-null   float64
 8   negative        1156 non-null   float64
 9   positive        1156 non-null   float64
 10  trust           1156 non-null   float64
 11  anticipation    1156 non-null   float64
 12  joy             1156 non-null   float64
 13  surprise        1156 non-null   float64
 14  sadness         1156 non-null   float64
```

```
 15  disgust         1156 non-null   float64
 16  positive_count  1156 non-null   int64
dtypes: float64(10), int64(2), object(5)
memory usage: 153.7+ KB
```

```
# Step 1: Remove duplicates
df = df.drop_duplicates()

# Step 2: Remove rows with missing values
df = df.dropna()

# Step 3: Drop unnecessary columns
df = df.drop(columns=['published_at', 'author'])

df
```

	comment	like_count	sentiment	emotions	anger	fear	negative	positive	trust	anticipation	joy	surprise	sadness
0	If this fight were real and not just an exhibi...	0	Positive	{'anger': 3, 'fear': 3, 'negative': 2, 'positi...	3.0	3.0	2.0	8.0	7.0	6.0	5.0	2.0	2.0
1	Tyson will bust his hide. People forget with h...	0	Neutral	{'fear': 2, 'negative': 3, 'anticipation': 1, ...	1.0	2.0	3.0	0.0	0.0	1.0	0.0	0.0	0.0
2	We signing the contract with this one 🗣️🗣️🗣️🔥🔥🔥	0	Neutral	{}	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	Watching cm punk fight in the ufc is better th...	0	Positive	{'anger': 2, 'fear': 1, 'negative': 1}	2.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
4	Yall forget how much Ring IQ this guy has. AND...	0	Positive	{'negative': 1, 'positive': 1, 'anticipation': 1}	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1151	WTF	0	Negative	{}	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1152	Is this real or a joke?	42	Positive	{'positive': 1, 'trust': 1, 'negative': 1}	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0
1153	Bruh	0	Neutral	{}	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1154	First comment 😂	0	Positive	{}	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1155	Maa kasam jisne like subscribe na kiya wo exam...	1	Negative	{'anticipation': 1}	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

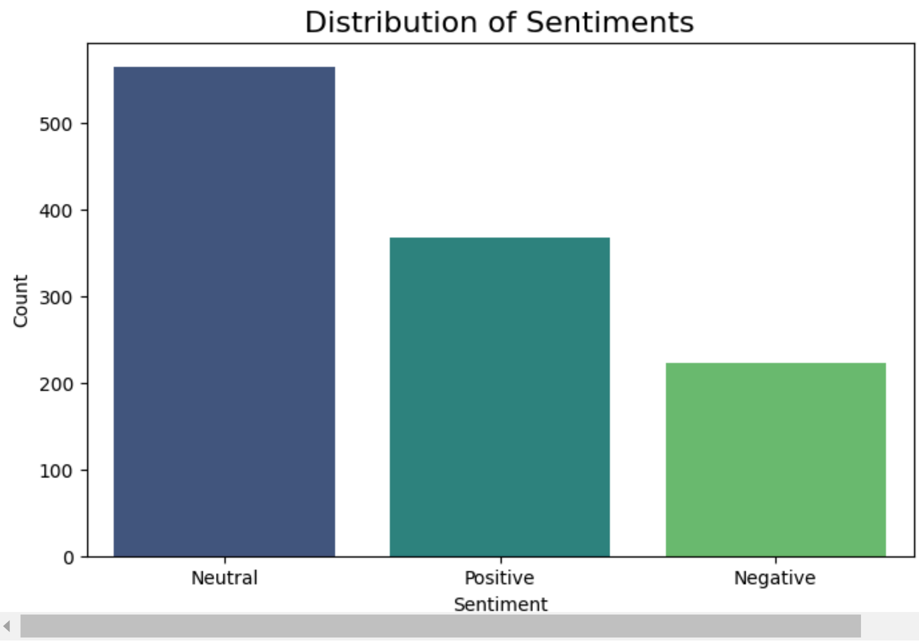
1156 rows × 15 columns

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt

# WordCloud for 'comment' column
text = " ".join(comment for comment in df['comment'].astype(str))
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(text)
```

```
# Plot WordCloud
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title("WordCloud of Comments", fontsize=16)
plt.show()
```

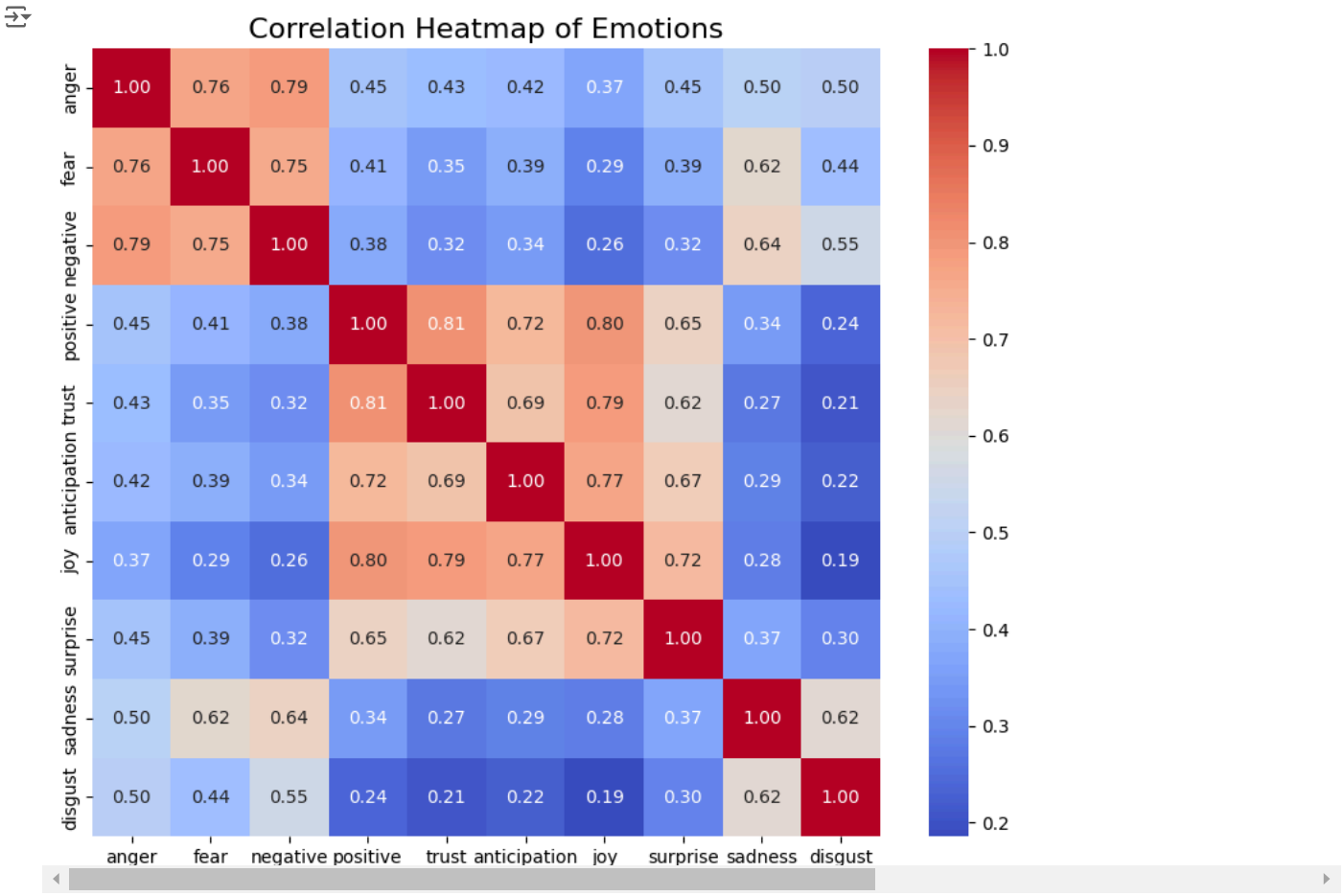

```
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x='sentiment', order=df['sentiment'].value_counts().index, palette='viridis')
plt.title("Distribution of Sentiments", fontsize=16)
plt.xlabel("Sentiment")
plt.ylabel("Count")
plt.show()
```



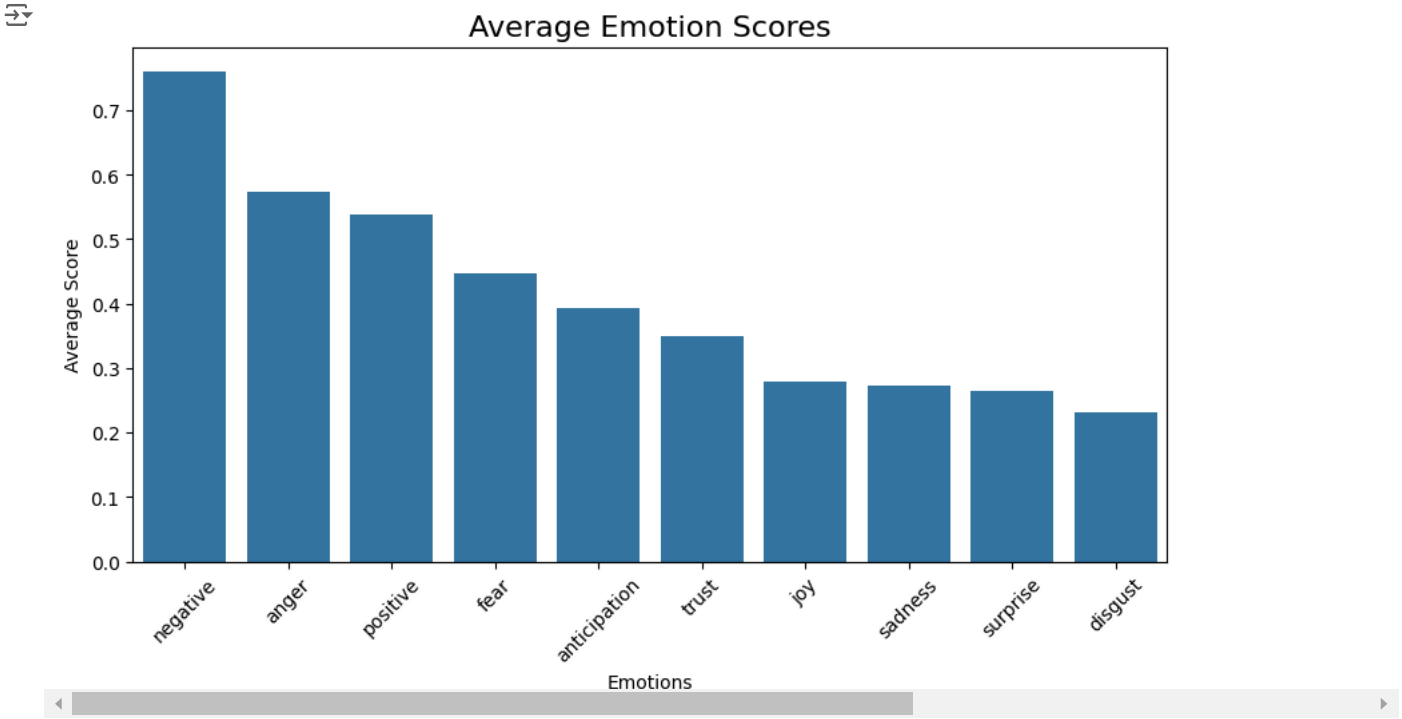

```
import seaborn as sns

emotion_columns = ['anger', 'fear', 'negative', 'positive', 'trust',
                   'anticipation', 'joy', 'surprise', 'sadness', 'disgust']
correlation_matrix = df[emotion_columns].corr()

plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f', square=True)
plt.title("Correlation Heatmap of Emotions", fontsize=16)
plt.show()
```




```
# Barplot of top emotions
mean_emotions = df[emotion_columns].mean().sort_values(ascending=False)
plt.figure(figsize=(10, 5))
sns.barplot(x=mean_emotions.index, y=mean_emotions.values)
plt.title("Average Emotion Scores", fontsize=16)
plt.xlabel("Emotions")
plt.ylabel("Average Score")
plt.xticks(rotation=45)
plt.show()
```

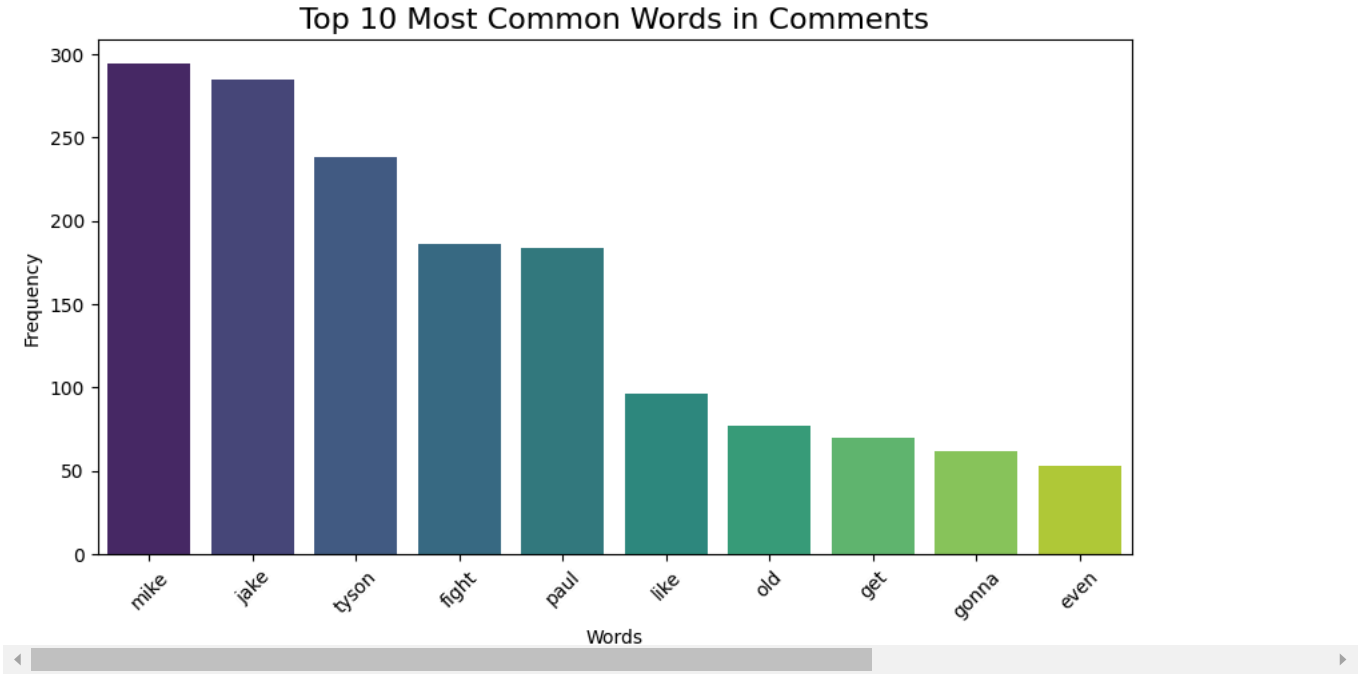



```
# Top 10 most common words in comments
nltk.download('stopwords')
stop_words = set(stopwords.words('english'))
```

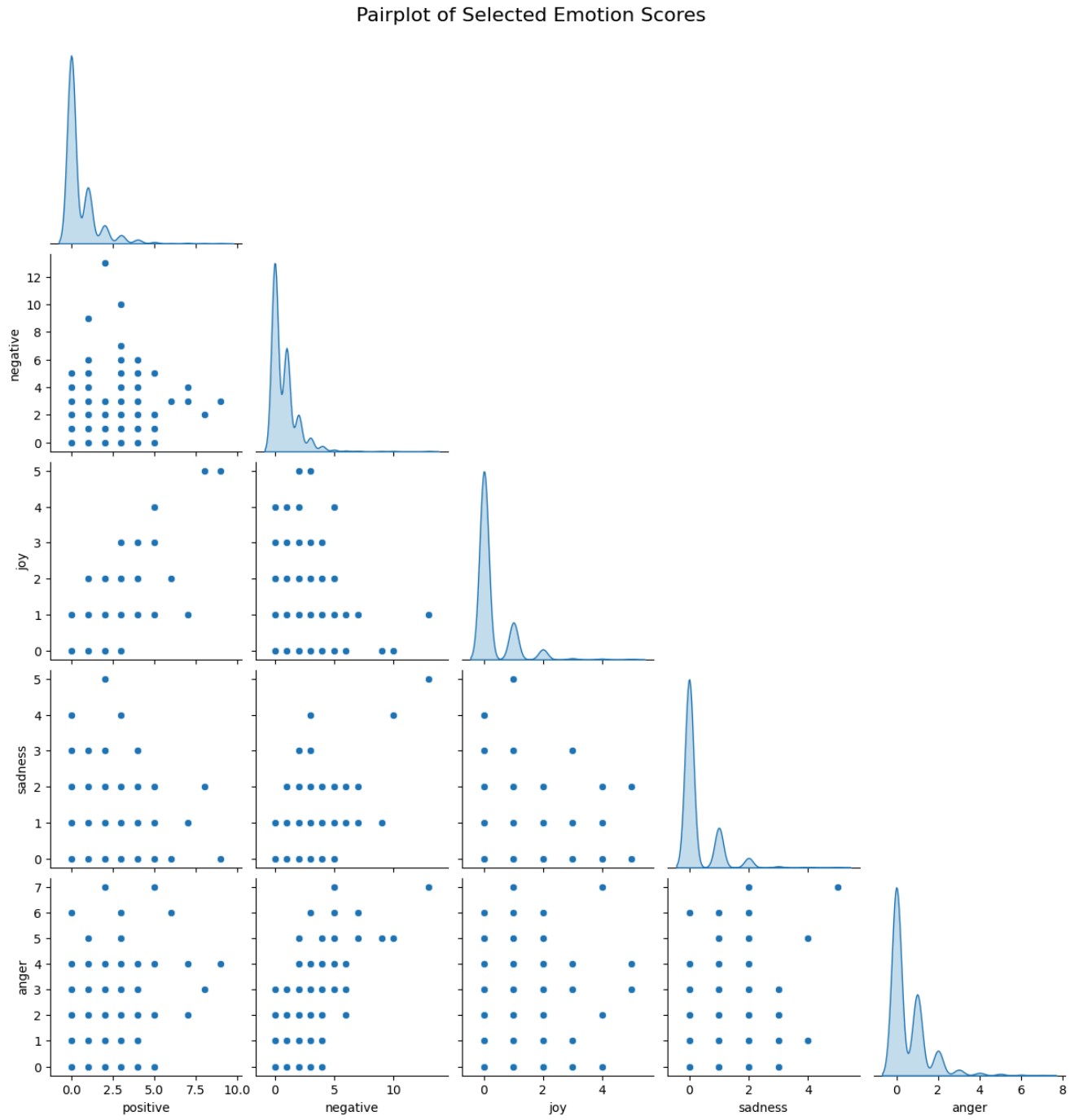
```
all_words = " ".join(df['comment'].astype(str)).split()
filtered_words = [word.lower() for word in all_words if word.isalpha() and word.lower() not in stop_words]
word_counts = Counter(filtered_words)
most_common_words = word_counts.most_common(10)
words, counts = zip(*most_common_words)

plt.figure(figsize=(10, 5))
sns.barplot(x=list(words), y=list(counts), palette='viridis')
plt.title("Top 10 Most Common Words in Comments", fontsize=16)
plt.xlabel("Words")
plt.ylabel("Frequency")
plt.xticks(rotation=45)
plt.show()
```

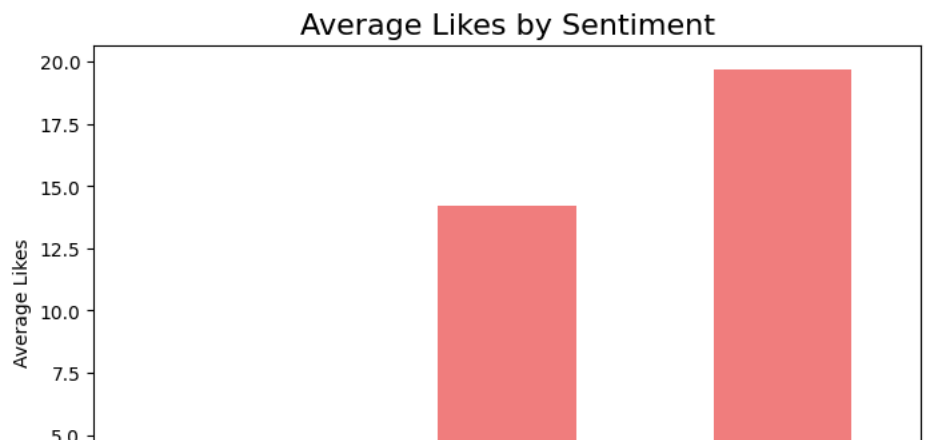
```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```




```
# Pairplot for selected emotion columns
selected_emotions = ['positive', 'negative', 'joy', 'sadness', 'anger']
sns.pairplot(df[selected_emotions], diag_kind='kde', corner=True)
plt.suptitle("Pairplot of Selected Emotion Scores", y=1.02, fontsize=16)
plt.show()
```




```
# Average likes by sentiment
avg_likes = df.groupby('sentiment')['like_count'].mean()
plt.figure(figsize=(8, 5))
avg_likes.plot(kind='bar', color='lightcoral')
plt.title("Average Likes by Sentiment", fontsize=16)
plt.xlabel("Sentiment")
plt.ylabel("Average Likes")
plt.xticks(rotation=45)
plt.show()
```




```
pd.set_option('display.max_colwidth', None)

# Comments with highest positive emotion scores
top_positive_comments = df.sort_values(by='positive', ascending=False).head(5)

# Correct syntax to print specific columns
print(top_positive_comments[['comment', 'positive']])
```

```
     57                                                                I think you past your prime and are foolish enough to think the internet c
     0                                                               If this fight were real and not just an exhibition like Tyson's match against Roy
     440
     121  This fight is entertainment. It has a predetermined outcome.\nFace saving draw for both fighters?\nRedemptive humbling defeat f
     749                                                                                                  Why did Mike accepted this fight? I mean

          positive
     57        9.0
     0         8.0
     440       7.0
     121       7.0
     749       6.0
```

```
top_positive_comments = df.sort_values(by='negative', ascending=False).head(5)

# Correct syntax to print specific columns
print(top_positive_comments[['comment', 'negative']])
```

```
     280  Mr Tyson  no disrespect bro , please , PLEASE withdraw from this fight with the young hungry heavyweight prospect & future Undi
     90                                                                                                                  Even if Mike Tyson los
     949
     668
     601

          negative
     280      13.0
     90       10.0
     949       9.0
     668       7.0
     601       7.0
```

Youtube Sentiment Analysis: Jake Paul vs. Mike Tyson **Trailer bold text**

---

sentiment analysis with Python and data visualization techniques to evaluate YouTube comments on the Jake Paul vs. Mike Tyson trailer. This hands-on project guides you through data processing methods to classify sentiments and extract actionable insights using libraries like pandas, numpy, and nltk.

Setup