```
Start coding or generate with AI.
```

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

```
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Package punkt_tab is already up-to-date!
True
```

```
# Install NLTK if not already installed
!pip install nltk

# Download necessary NLTK data (stopwords and punkt)
import nltk
nltk.download('stopwords')
nltk.download('punkt')
nltk.download('punkt_tab') # Added to resolve LookupError
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.12/dist-packages (
Requirement already satisfied: click in /usr/local/lib/python3.12/dist-packages
Requirement already satisfied: joblib in /usr/local/lib/python3.12/dist-packages
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.12/dist
Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
True
```

```
display(tweets_df[['text', 'cleaned_text']].head())
```

	text	cleaned_text
0	@VirginAmerica What @dhepburn said.	said
1	@VirginAmerica plus you've added commercials t...	plus added commercials experience tacky
2	@VirginAmerica I didn't today... Must mean I n...	today must mean need take another trip
3	@VirginAmerica it's really aggressive to blast...	really aggressive blast obnoxious entertainmen...
4	@VirginAmerica and it's a really big bad thing...	really big bad thing

And here is the head of the TF-IDF matrix, showing the numerical representation of your cleaned tweets:

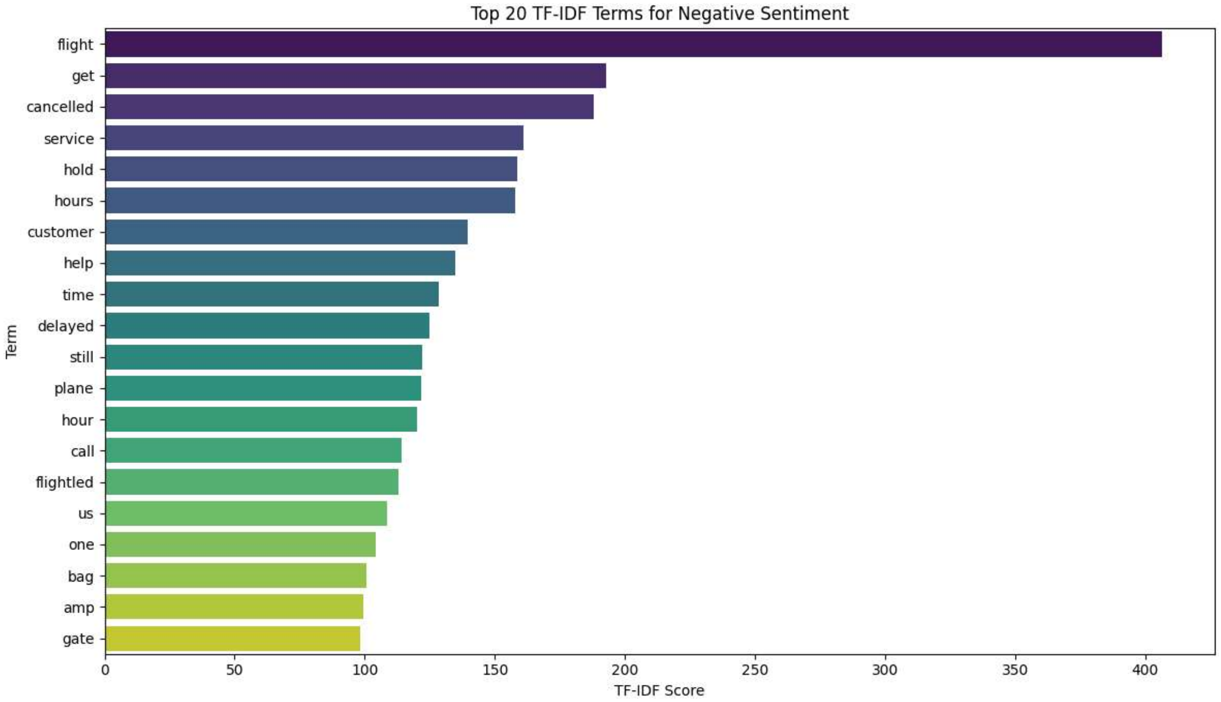
```
display(tfidf_df.head())
```

```
    aa  aadvantage  abc  abilities  ability  able  aboard  abq  abroad  absolute
0  0.0         0.0  0.0        0.0      0.0   0.0     0.0  0.0     0.0       0.0
1  0.0         0.0  0.0        0.0      0.0   0.0     0.0  0.0     0.0       0.0
2  0.0         0.0  0.0        0.0      0.0   0.0     0.0  0.0     0.0       0.0
3  0.0         0.0  0.0        0.0      0.0   0.0     0.0  0.0     0.0       0.0
4  0.0         0.0  0.0        0.0      0.0   0.0     0.0  0.0     0.0       0.0

5 rows × 5000 columns
```

The visualizations focusing on negative sentiment vocabulary (the bar chart of top terms and the word cloud) were already generated and should be visible in the output above this response.

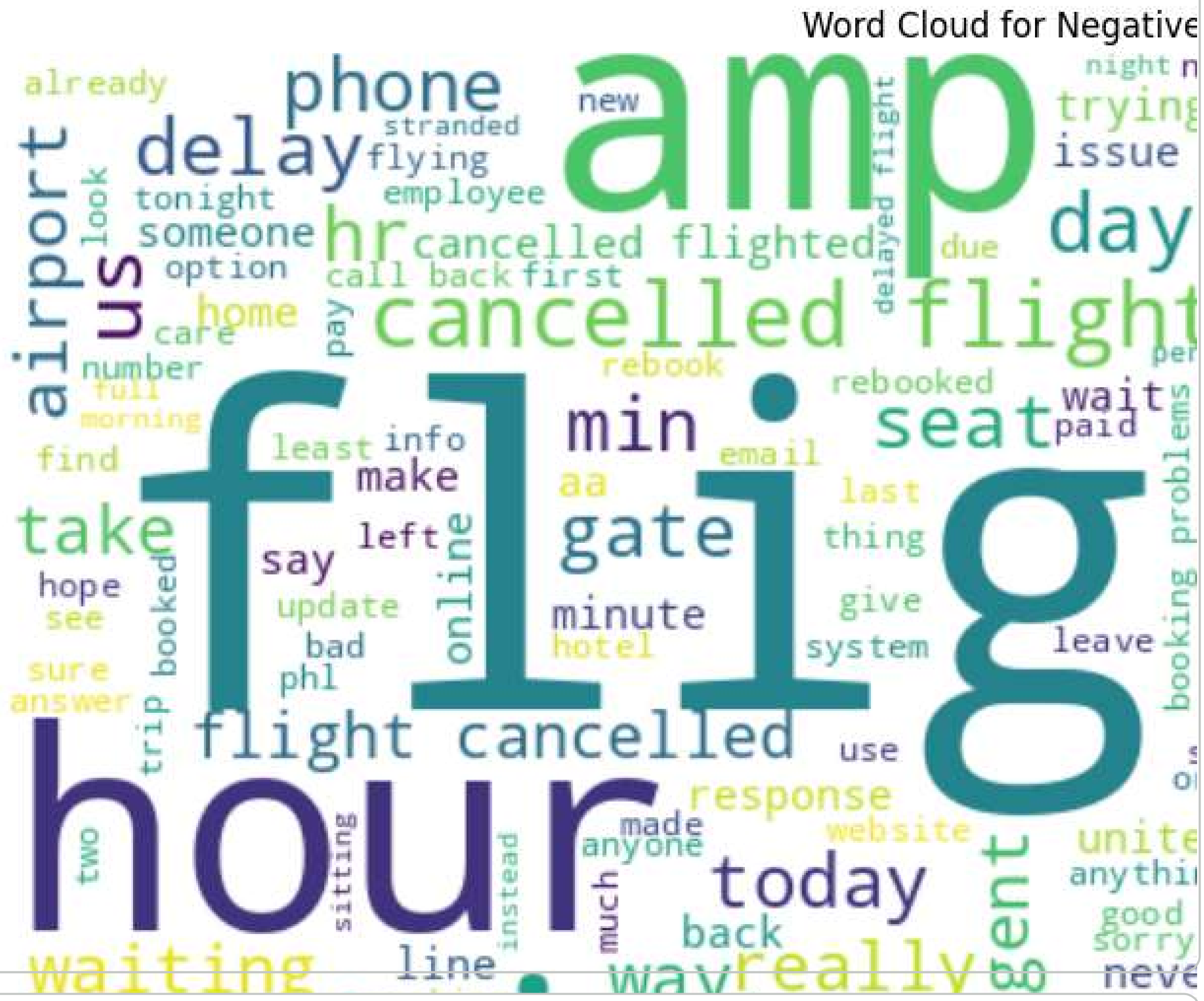
```
import matplotlib.pyplot as plt
import seaborn as sns

# Create a bar chart for the top 20 negative terms
plt.figure(figsize=(12, 7))
sns.barplot(x=top_negative_terms.head(20).values, y=top_negative_terms.head(20)
plt.title('Top 20 TF-IDF Terms for Negative Sentiment')
plt.xlabel('TF-IDF Score')
plt.ylabel('Term')
plt.tight_layout()
plt.show()
```

```
/tmp/ipython-input-87981263.py:6: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v

  sns.barplot(x=top_negative_terms.head(20).values, y=top_negative_terms.head(20
```



```
# Install the wordcloud library if not already installed
!pip install wordcloud

from wordcloud import WordCloud

# Combine all cleaned text from negative tweets into a single string
all_negative_text = ' '.join(tweets_df[tweets_df['airline_sentiment'] == 'n

# Generate a word cloud image
wordcloud = WordCloud(width=800, height=400, background_color='white').gener

# Display the generated image:
plt.figure(figsize=(15, 8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Word Cloud for Negative Sentiment Tweets')
plt.show()
```

```
Requirement already satisfied: wordcloud in /usr/local/lib/python3.12/dist-packa
Requirement already satisfied: numpy>=1.19 in /usr/local/lib/python3.12/dist-pac
Requirement already satisfied: pillow in /usr/local/lib/python3.12/dist-packages
Requirement already satisfied: matplotlib in /usr/local/lib/python3.12/dist-pack
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.12/dis
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.12/dist-pa
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.12/di
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.12/di
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.12/dis
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.12
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packag
```


Word Cloud for Negative

```
# Filter tweets with negative sentiment
negative_tweets_df = tweets_df[tweets_df['airline_sentiment'] == 'negative']

# Get the indices of these negative tweets in the original DataFrame
negative_indices = negative_tweets_df.index

# Select the corresponding rows from the TF-IDF matrix
negative_tfidf_matrix = tfidf_matrix[negative_indices]

# Sum the TF-IDF scores for each term across all negative tweets
sum_tfidf_negative = negative_tfidf_matrix.sum(axis=0)
```

```
# Get the feature names (words) from the TF-IDF vectorizer
feature_names = tfidf_vectorizer.get_feature_names_out()

# Create a pandas Series for better manipulation and sorting
top_negative_terms = pd.Series(sum_tfidf_negative.A1, index=feature_names)

# Sort the terms by their summed TF-IDF scores in descending order
top_negative_terms = top_negative_terms.sort_values(ascending=False)

print("\nTop 20 TF-IDF terms for Negative Sentiment:")
display(top_negative_terms.head(20))
```

```
Top 20 TF-IDF terms for Negative Sentiment:
                    0
   flight  406.330730
      get  192.670594
cancelled  187.994529
  service  161.129885
     hold  158.465816
    hours  157.735342
 customer  139.632482
     help  134.928191
     time  128.395707
  delayed  125.027488
    still  122.206777
    plane  121.572885
     hour  120.181316
     call  114.178105
flightled  113.002818
       us  108.490301
      one  104.091798
      bag  100.530121
      amp   99.555918
     gate   98.149583

dtype: float64
```

```
!unzip /content/Tweets.csv.zip -d /content/
```

```
Archive:  /content/Tweets.csv.zip
  inflating: /content/Tweets.csv
```

```
import pandas as pd

# Assuming the unzipped file is named 'Tweets.csv' in the /content/ director
tweets_df = pd.read_csv('/content/Tweets.csv')

# Display the first 5 rows of the DataFrame to verify the data loading
display(tweets_df.head())
```

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativ
0	570306133677760513	neutral	1.0000	
1	570301130888122368	positive	0.3486	
2	570301083672813571	neutral	0.6837	
3	570301031407624196	negative	1.0000	Ba
4	570300817074462722	negative	1.0000	C