

Twitter-AIRLINE-analysis[Om.Mule78].ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text

Share Gemini OM

import pandas as pd
import numpy as np
from nltk import word_tokenize,sent_tokenize
import os
from google.colab import files
uploaded = files.upload()
data = pd.read_csv('Tweets.csv')
data.head()

Choose Files

Tweets.csv

Tweets.csv(text/csv) · 3421431 bytes, last modified: 4/27/2025 · 100% done

Saving Tweets.csv to Tweets (1).csv


1 to 5 of 5 entries

Filter

?

index	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	airline_sentiment_gold	name	negativereason_gold
0	570306133677760513	neutral	1.0	NaN	NaN	Virgin America	NaN	cairdin	NaN
1	570301130888122368	positive	0.3486	NaN	0.0	Virgin America	NaN	jnardino	NaN
2	570301083672813571	neutral	0.6837	NaN	NaN	Virgin America	NaN	yvonnalynn	NaN

Connected to Python 3 Google Compute Engine backend

 Twitter-AIRLINE-analysis[Om.Mule78].ipynb ☆ ☁

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

✓ RAM
Disk

Share Gemini OM

1.Finding the total number of tweets?

```
[25] total_tweets = len(data)
      print("The total Number of Tweets are:",total_tweets)
```

The total Number of Tweets are: 14648

2.Identify the number of tweets per airline.


```
tweets_per_airline = data['airline'].value_counts()
print(tweets_per_airline)
```

airline
United 3822
US Airways 2913
American 2759
Southwest 2428
Delta 2222
Virgin America 504
Name: count, dtype: int64

3.Find the percentage of each sentiment (positive, neutral, negative)

```
[5] sentiment_percent = data['airline_sentiment'].value_counts(normalize=True) * 100
     print(sentiment_percent)
```

✓ Connected to Python 3 Google Compute Engine backend

 Twitter-AIRLINE-analysis[Om.Mule78].ipynb ☆ ☁
File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

RAM
Disk

3. Find the percentage of each sentiment (positive, neutral, negative)

```
[5] sentiment_percent = data['airline_sentiment'].value_counts(normalize=True) * 100
print(sentiment_percent)
```

```
airline_sentiment
negative    62.691257
neutral    21.168033
positive    16.140710
Name: proportion, dtype: float64
```

4. Identify the airline with the most negative tweets.

bold text


```
neg_tweets = data[data['airline_sentiment'] == 'negative']
most_negative_airline = neg_tweets['airline'].value_counts().idxmax()
print("The Most negative tweets are for:", most_negative_airline, "Airlines")
```

```
The Most negative tweets are for: United Airlines
```

5. Find the mean confidence level for positive tweets.

```
[22] positive_confidence = data[data['airline_sentiment'] == 'positive']['airline_sentiment_confidence'].mean()
print(f"The Mean confidence level obtained : {positive_confidence}")
```

Connected to Python 3 Google Compute Engine backend

 Twitter-AIRLINE-analysis[Om.Mule78].ipynb ☆ ☁
File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

✓ RAM
Disk

5. Find the mean confidence level for positive tweets.

```
[22] positive_confidence = data[data['airline sentiment'] == 'positive']['airline_sentiment_confidence'].mean()
      print(f"The Mean confidence level obtained : {positive_confidence}")
```

→ The Mean confidence level obtained : 0.872038933559035

6. Find the top 5 most common negative reasons.

```
top_neg_reasons = data['negativereason'].value_counts().head(5)
print(top_neg_reasons)
```




→ negativereason
Customer Service Issue 2910
Late Flight 1665
Can't Tell 1198
Cancelled Flight 847
Lost Luggage 724
Name: count, dtype: int64

7. Calculate the average negative reason confidence.

```
[28] avg_neg_reason_confidence = data['negativereason_confidence'].mean()
      print("The average negative reason confidence obtained:", avg_neg_reason_confidence)
```



→ The average negative reason confidence obtained: 0.6382982797947159

✓ Connected to Python 3 Google Compute Engine backend

 Twitter-AIRLINE-analysis[Om.Mule78].ipynb  

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

 RAM  Disk

8.Find tweets with missing negative reasons.

```
[30] missing_neg_reasons = data['negativereason'].isna().sum()
      print("The Number of missing tweets with missing reasons:",missing_neg_reasons)
```

The Number of missing tweets with missing reasons: 5462

9.Find the earliest and latest tweet created dates.

```
data['tweet_created'] = pd.to_datetime(data['tweet_created'])
earliest = data['tweet_created'].min()
latest = data['tweet_created'].max()
print("The time of the earliest tweet:", earliest)
print("The time of the latest tweet:", latest)
```

The time of the earliest tweet: 2015-02-16 23:36:05-08:00
The time of the latest tweet: 2015-02-24 11:53:37-08:00

10.Find out the busiest hour when tweets were posted.

```
[36] data['hour'] = data['tweet_created'].dt.hour
      busiest_hour = data['hour'].value_counts().idxmax()
      print("The busiest hour was:",busiest_hour,"o'clock")
```

The busiest hour was: 9 o'clock

Connected to Python 3 Google Compute Engine backend

Twitter-AIRLINE-analysis[Om.Mule78].ipynb

File Edit View Insert Runtime Tools Help

Share Gemini OM

Commands + Code + Text

RAM Disk

11.Group tweets by airline and find the average sentiment confidence.

```
avg_sentiment_confidence = data.groupby('airline')['airline_sentiment_confidence'].mean()
print(avg_sentiment_confidence)
```

airline	
American	0.917352
Delta	0.869878
Southwest	0.886516
US Airways	0.921578
United	0.900878
Virgin America	0.876086

Name: airline_sentiment_confidence, dtype: float64

+ Code + Text

12.calculate standard deviation of sentiment confidence.

```
[39] std_sentiment_confidence = np.std(data['airline_sentiment_confidence'])
print("The standard deviation of sentiment confidence is obtained :",std_sentiment_confidence)
```

```
The standard deviation of sentiment confidence is obtained : 0.1628243978712151
```

13.Find the user timezone with the maximum tweets.

```
[40] most_common_timezone = data['user_timezone'].value_counts().idxmax()
print(most_common_timezone)
```

```
Eastern Time (US & Canada)
```

Connected to Python 3 Google Compute Engine backend

Twitter-AIRLINE-analysis[Om.Mule78].ipynb

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

Share Gemini OM

RAM Disk

14. Find the proportion of tweets with no timezone information.

```
no_timezone = data['user_timezone'].isnull().mean() * 100
print(no_timezone)
```

```
32.92349726775956
```

+ Code + Text

15. Identify airlines that have only positive tweets.

```
[45] airlines_with_only_positive = data.groupby('airline')['airline_sentiment'].unique()
only_positive_airlines = airlines_with_only_positive[airlines_with_only_positive.apply(lambda x: set(x) == {'positive'})]
print(only_positive_airlines)
print("series shows no element that means there is no any airline winth ony positive tweets.")
```

```
Series([], Name: airline sentiment, dtype: object)
series shows no element that means there is no any airline winth ony positive tweets
```

16. Find the tweet(s) with the highest sentiment confidence.

```
[46] highest_confidence_tweets = data[data['airline_sentiment_confidence'] == data['airline_sentiment_confidence'].max()]
print(highest_confidence_tweets[['text', 'airline_sentiment_confidence']])
```

```
text \
0      @VirginAmerica What @dhepburn said.
3      @VirginAmerica it's really aggressive to blast...
4      @VirginAmerica and it's a really big bad thing...
5      @VirginAmerica conisderly would pay $20 a flight
```

Connected to Python 3 Google Compute Engine backend

Twitter-AIRLINE-analysis[Om.Mule78].ipynb

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

Share Gemini OM

RAM Disk

16.Find the tweet(s) with the highest sentiment confidence.

```
[46] highest_confidence_tweets = data[data['airline_sentiment_confidence'] == data['airline_sentiment_confidence'].max()]
print(highest_confidence_tweets[['text', 'airline_sentiment_confidence']])
```

	text	airline_sentiment_confidence
0	@VirginAmerica What @dhepburn said.	1.0
3	@VirginAmerica it's really aggressive to blast...	1.0
4	@VirginAmerica and it's a really big bad thing...	1.0
5	@VirginAmerica seriously would pay \$30 a fligh...	1.0
9	@VirginAmerica it was amazing, and arrived an ...	1.0
...
14631	@AmericanAir thx for nothing on getting us out...	1.0
14633	@AmericanAir my flight was Cancelled Flightled...	1.0
14636	@AmericanAir leaving over 20 minutes Late Flig...	1.0
14637	@AmericanAir Please bring American Airlines to...	1.0
14638	@AmericanAir you have my money, you change my ...	1.0

[10445 rows x 2 columns]

Connected to Python 3 Google Compute Engine backend

Twitter-AIRLINE-analysis[Om.Mule78].ipynb

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

Share Gemini OM

RAM Disk

17. Create a pivot table showing sentiment counts per airline.

```
[47] sentiment_pivot = pd.pivot_table(data, index='airline', columns='airline_sentiment', aggfunc='size', fill_value=0)
      print(sentiment_pivot)
```

airline_sentiment	negative	neutral	positive
airline			
American	1908	463	336
Delta	955	723	544
Southwest	1186	664	578
US Airways	2263	381	269
United	2633	697	492
Virgin America	181	171	152

18. Check if there's a correlation between sentiment confidence and negative reason confidence.


```
[50] correlation = data['airline_sentiment_confidence'].corr(data['negative_reason_confidence'])
      print(correlation)
```

0.6858789654178273

19. Find how many tweets mention "help" or "support".

```
[48] help_tweets = data['text'].str.contains('help|support', case=False, na=False).sum()
      print(help_tweets)
```

Connected to Python 3 Google Compute Engine backend

 Twitter-AIRLINE-analysis[Om.Mule78].ipynb ☆ ☁

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text

✓ RAM
Disk

Share Gemini OM

19. Find how many tweets mention "help" or "support".

```
[48] help_tweets = data['text'].str.contains('help|support', case=False, na=False).sum()
      print(help_tweets)
```

1103

20. Get the average length of tweets per airline.

```
data['tweet_length'] = data['text'].str.len()
avg_tweet_length = data.groupby('airline')['tweet_length'].mean()
print(avg_tweet_length)
```

```
airline
American      108.630301
Delta          92.501800
Southwest     103.212810
US Airways    109.261586
United        103.817373
Virgin America 98.930556
Name: tweet_length, dtype: float64
```

Connected to Python 3 Google Compute Engine backend