SCHOOL OF COMPUTER SCIENCE AND ENGINEERING (SCOPE)

NOVEMBER 2020

CSE4022 – Natural Language Processing

FINAL PROJECT REPORT

INTERACTIVE DASHBOARD FOR TWITTER SENTIMENT ANALYSIS OF US AIRLINES

SUBMITTED BY

17BCE2383 TEK RAJ AWASTHI

17BCE2380 ALOK SINHA

17BCE2393 BIBEK SINGH

UNDER THE GUIDANCE OF

PROF. SHARMILA BANU K

VIT UNIVERSITY, VELLORE

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

TABLE OF CONTENT

1.ABSTRACT

Sending, posting, or sharing negative, harmful, false, or mean content about someone is found predominant in social medias. The sentiment based on the airlines are predicted with the categorical data based on the classification of responses. The tweet might be positive, negative and neutral. It specially focuses the sentiment based on the lexical analyser where it analyses the words and sentences and classify with respect to word cloud and generate the responses accordingly. The main moto is to bright the classification on the visualization chart to predict the classification in a real image. The Natural Processing has come up with the analysis part of the sentiment of review made by users for the future response with very good interactive dashboard. The essential and basic thought of the project is that, realizing how individuals feel certain statuses can be utilized for classification.

2.INTRODUCTION

The Sentiment analysis is the process of determine whether a piece of writing based on the different level of classification. The Sentiment analysis helps data analysts within large enterprises gauge public opinion, conduct nuanced market research, monitor brand and product reputation, and understand customer experiences. In essence, it is the process of determining the emotional tone behind a series of words, used to gain an understanding of the the attitudes, opinions and emotions expressed within an online mention. This project helps to gain the sentiment based on the lexicon where it parses and identify the level of classification. This includes the twitter *US Airline Sentiment Data* and interactive dashboard with silent features of sidebar including the option to choose the bar graph and pie chart and develop the interactive sentiment of different airlines based on their sentiment. And even to reallocate the user’s location from where the sentiment is enhanced and made. The categorical view of the sentiment gives the fully functionality demo for the user to view and make real image of the current happening scenario. The plot location of data views the tweets made by the user to have effective more decision on the Airlines and its facility. This helps user to select the view and make decision of travelling with the effective cost and time.

3. PROBLEM STATEMENT

Information management tool that visually track, analyses and displays the metrics and key points and insights on data are very important in business, finance. And sentiment analysis without interactive dashboard visualisation of sentiment in any business, corporation is yet incomplete for companies to understand and solve the problems of customers. Interactive data visualisation of sentiment analysis of any context depicts the clear ratio, percentage of positive feedback, negative feedback, and neutral feedback of customers, which helps businesses, companies to be more awareness and progressive toward solving customer's dissatisfaction and gain profit in coming days.

So we are going to develop an web based interactive dashboard which shows the different insights on twitter sentiment analysis of different US Airlines, according to the user's/ airlines administrator's choice.

4. CLASSIFICATION LEVEL

The pattern analyser will be classified into three categories based on the different forms of feedback that are-

- Positive Response
- Negative Response
- Neutral Response

These three conditions are going to be examined to get the feedback for the different comments or status made by users. With the appropriate tools and technique, we are going to see the action of different techniques used by the machine learning algorithm

5. LITERATURE SURVEY

(Mohammed Nazrul Islam Arif, Sarwar Hussain Paplu, Prof. Dr. Karsten Berns, 2019) recognize the emotions present in the communication or the emotions of the involved users to make those interactions more humanly. This study summarizes different approaches that can be used to analyze the emotion from the text and recognize the emotion between Text-Based Communications. Approaches like finding subjective feeling through introspection, Word categorization, Kinetic Typography to convey emotion, Meta-analysis on mood induction, Emotion expression through text-based communication, Sentimental analysis are discussed.

(Osama Mohammad Rababah, and Nour Alokaily, 2019) present novel system that offers personalized user experiences and solves the semantic-pragmatic gap. Having a system for forecasting sentiments might allow extracting opinions from the internet and predicting online user's favorites, which could determine valuable for commercial or marketing research. The data used belongs to the tagged corpus positive and negative processed movie reviews.

(Oyebode, O., & Orji, R. ,2019) aim to identifying public sentiments towards two popular candidates with the aim of determining their chances of being elected based on social media into the highest position of authority in Nigeria. Using lexicon-based and supervised machine learning (ML) techniques with the aim of detecting their sentiment polarity, the operation was performed sentiment analysis on election related posts from Naira land. The methods are implemented based on lexicon-based classifiers and others ML-based classifier which leads to analysis of sentiment based on positive and negative response.

Isah, H., Trundle, P., & Neagu, D. ,2014) reports the work that was held progress with contributions including: the development of a framework for gathering and analyzing the views and experiences of datas using machine learning, text mining and sentiment analysis. This application of the proposed framework on social for brand analysis, and the description of how to develop a product safety lexicon and training data for modelling a machine learning classifier was found. The result signifies the usefulness of text mining and sentiment analysis on social media data while the use of machine learning classifiers for predicting the sentiment orientation to monitor brand or product sentiment trends in order to act in the event of sudden or significant rise in negative sentiments.

(V Kharde, 2016) utilize different component extraction strategy. They utilized the structure where the pre-processor is applied to the raw sentences which make it more fitting to understand. Further, the different machine learning strategies prepare the dataset with include vectors and afterward the semantic examination offers an enormous arrangement of equivalents and comparability which gives the extremity of the substance. They give an overview and near investigation of existing procedures for conclusion mining including AI and dictionary-based methodologies, along with cross area and cross-lingual strategies and some assessment measurements

(F Del Vigna12, 2017) presented the principal of hate speech classifier for Italian writings. Thinking about a binary grouping, the classifier accomplished outcomes similar with those got in generally researched supposition examination for Italian language. Empowered by such encouraging result, they leave for future work the refinement of the classifier results while thinking about differentiation among hate levels and among different types of hate speech. They are developing the explanation process, both to build the corpus size and to gather more comments for a single comment. They are trying new explanation techniques, assessing the between annotator understanding or approving the explanation on the various degrees of hate speech.

6. COMPARATIVE STUDY

Reference No	Method involved	Advantage	Disadvantage	Future Scope
1	Tweet extracted directly from Twitter API, then cleaning and discovery of data performed.	Several algorithms to enhance the accuracy of classifying tweets as positive, negative and neutral.	Previously labelled data do not exist at first using lexicon-based algorithm.	An algorithm that can automatically classify tweets would be an interesting area of research.
2	Information extraction, information visualization, machine learning based and lexicon-based approaches.	Analysis on micro-blog, especially Twitter, can provide *supportive* information for producers in smartphone industry to make some decision about their next generation product.	The result from the MSAS may not yield direct feedback for the product feature itself.	More research on algorithm will be performed to improve the performance and accuracy of the system.
3	Multiple deep learning architectures to learn semantic word embeddings to handle this complexity	Investigated the application of deep neural network architectures for the task of hate speech detection and found them to significantly outperform the existing methods	Require much more data than traditional machine learning algorithms, as in at least thousands if not millions of labelled samples.	They plan to explore the importance of the user network features for the task.
4	Lexicon analysis to find or calculate the polarity and Machine Learning involves formation of models from labeled training dataset Introspection, Word categorization, Kinetic Typography to convey emotion, Meta-analysis on mood	widely used on big data to gather public critics in order to assess internaut’s satisfaction of a subject (services, products, events, topics or different persons) in different domains	It cannot understand emoticons. Second, they used only Twitter data. Third, cannot be accessed large data for this algorithm Topic-detection system	Propose a more efficient and global model that can work on larger volumes of Data. This particular method of

5	induction, Emotion expression through text-based communication.	Generate relevant emotional expression and verbal replies as the robot's reaction based on its personality and they improve the user engagement in conversation with the robot in real-time.	contributed more to their engagement than the sentiment-analysis system, though the differences between their statistical means is too high.	analysis seems to be more appropriate for finding the emotional expression in Text-Based communication.
6	10-fold cross validation and calculated corresponding precision, recall and F-measure.	Author information can be leveraged to improve the detection of misbehaviour in online social networks.	The limited size of the dataset.	consider the pragmatics of conversations between authors of same gender versus opposite gender. Moreover, it is worthwhile to compare different classification approaches and analyse their performances.

7.DATA SET

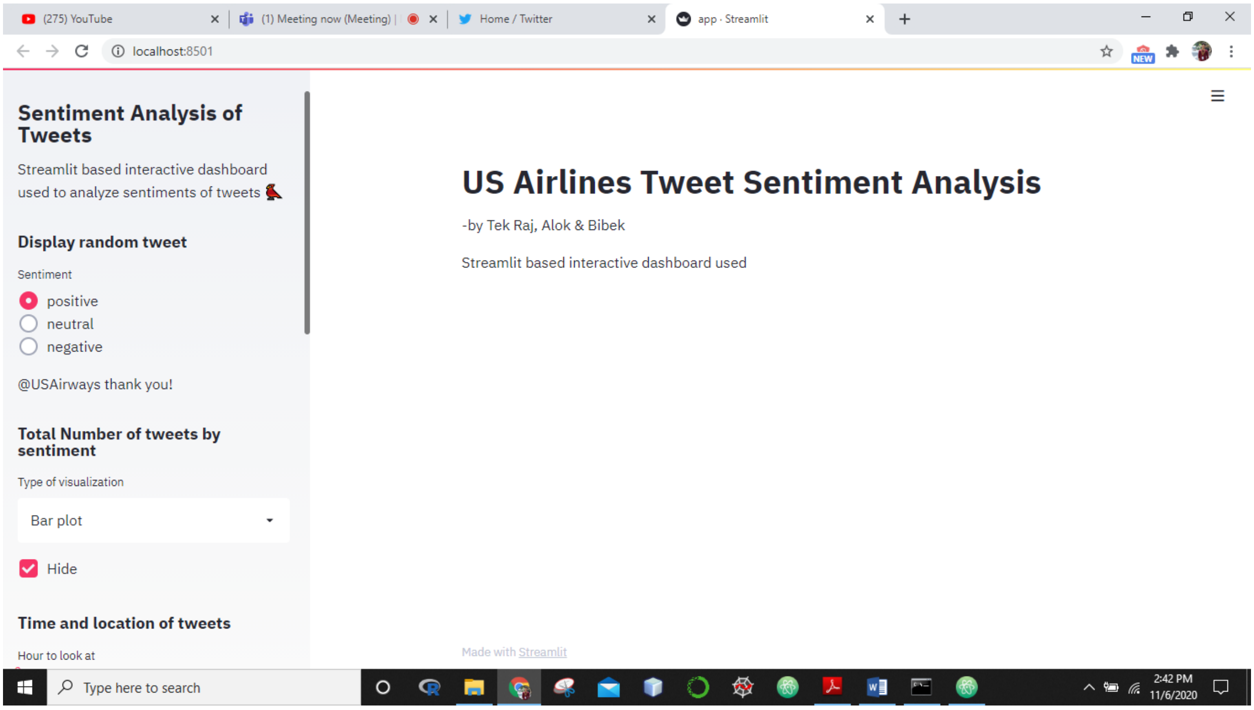
Source: *https://www.kaggle.com/crowdflower/twitter-airline-sentiment*

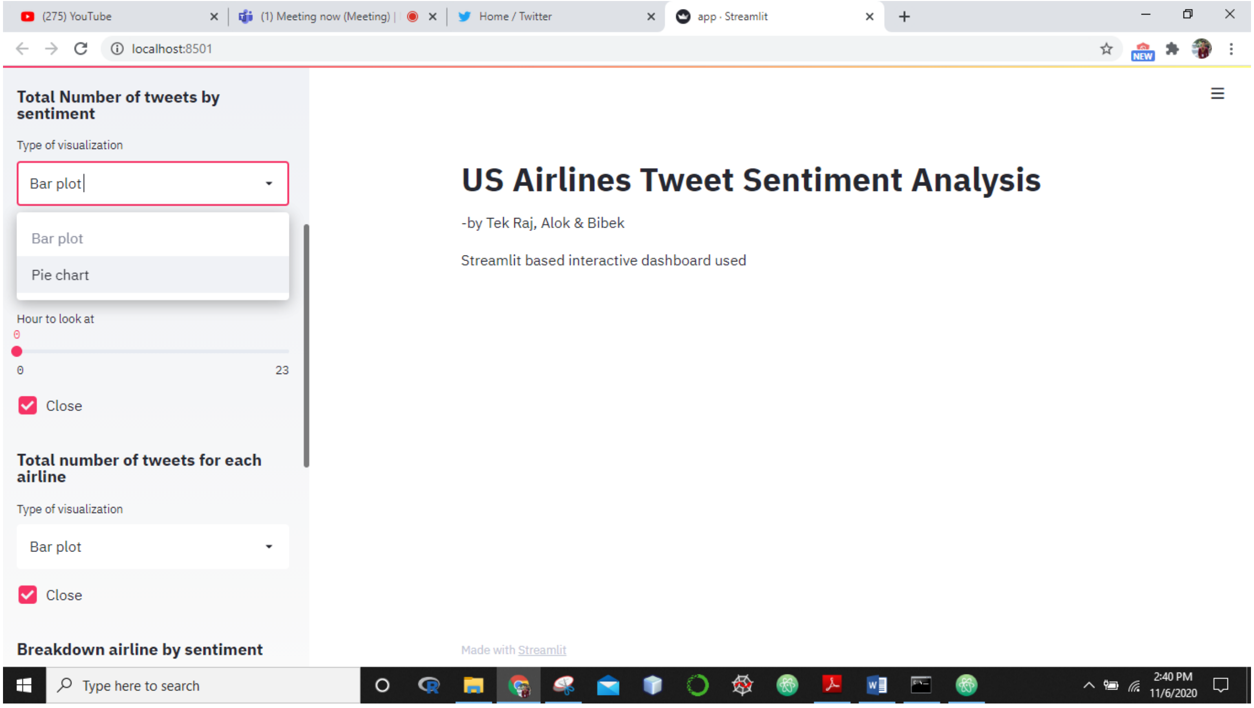
8. METHODOLOGY && RESULTS

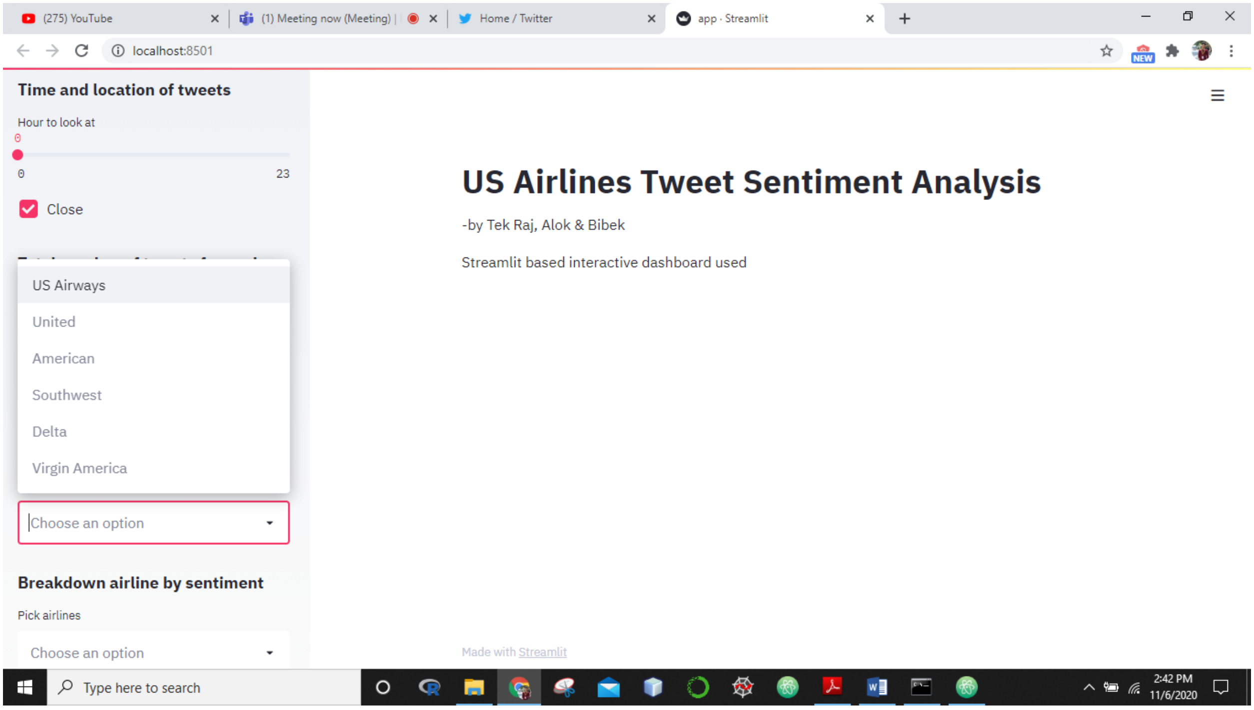
Libraries/Frameworks Used:

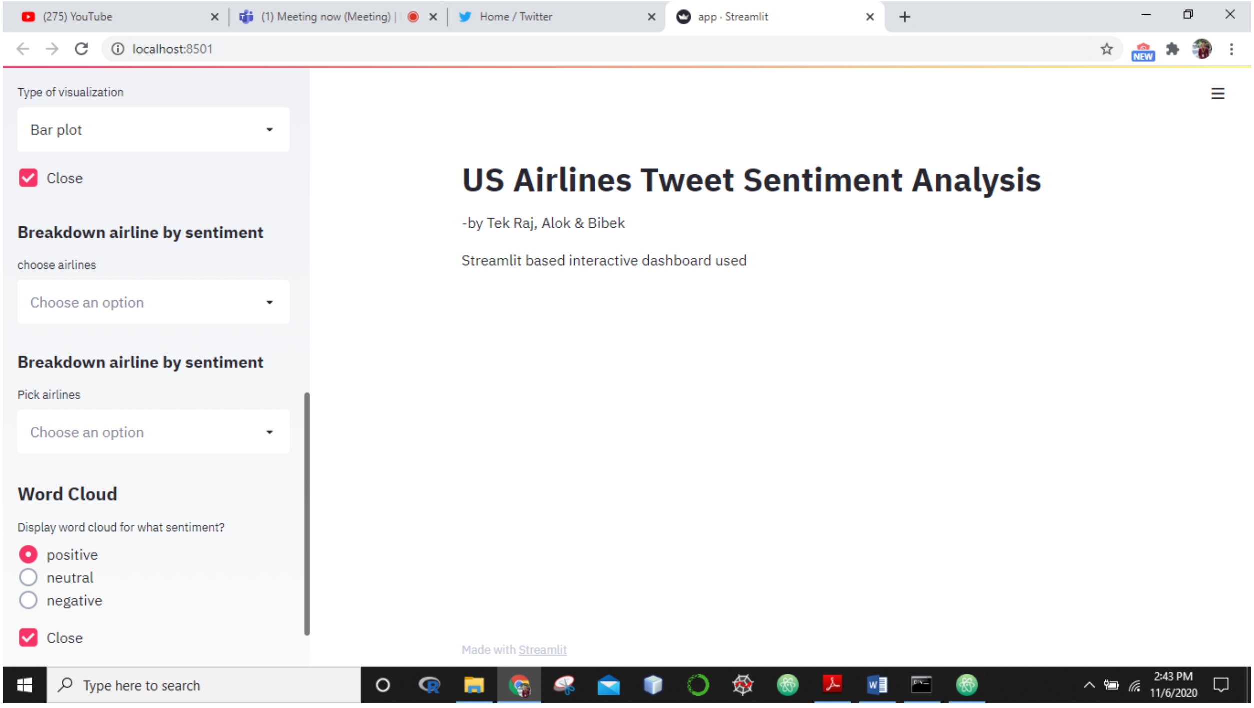
1. **Streamlit-** An open-source Python library for creating and sharing custom web apps for machine learning and data science purpose. It is most used powerful library to build and deploy the web apps for leveraging the power of data and machine learning and visualize the results in the form of interactive dashboards.
2. **Pandas** – Pandas is python-based software written library used for data manipulation and analysis. Here, we have used pandas for reading dataset, creating data frames.
3. **Matplotlib** – Matplotlib is a comprehensive python library for creating static and interactive dashboards, Here, we have plotted Bar graph and Pi-charts for visualization of tweets based on sentiment type and in the hierarchy of airplane names to show the total number/percentage of positive, negative, and neutral tweets.
4. **Wordcloud-** Word cloud is a python library for visual representation of texts. Word cloud is very important for quickly understanding the most prominent terms, user feedbacks in the short word type. Here, we have visualized the word cloud for positive, negative and neutral tweet sentiment.

GUI of Interactive Dashboard API:



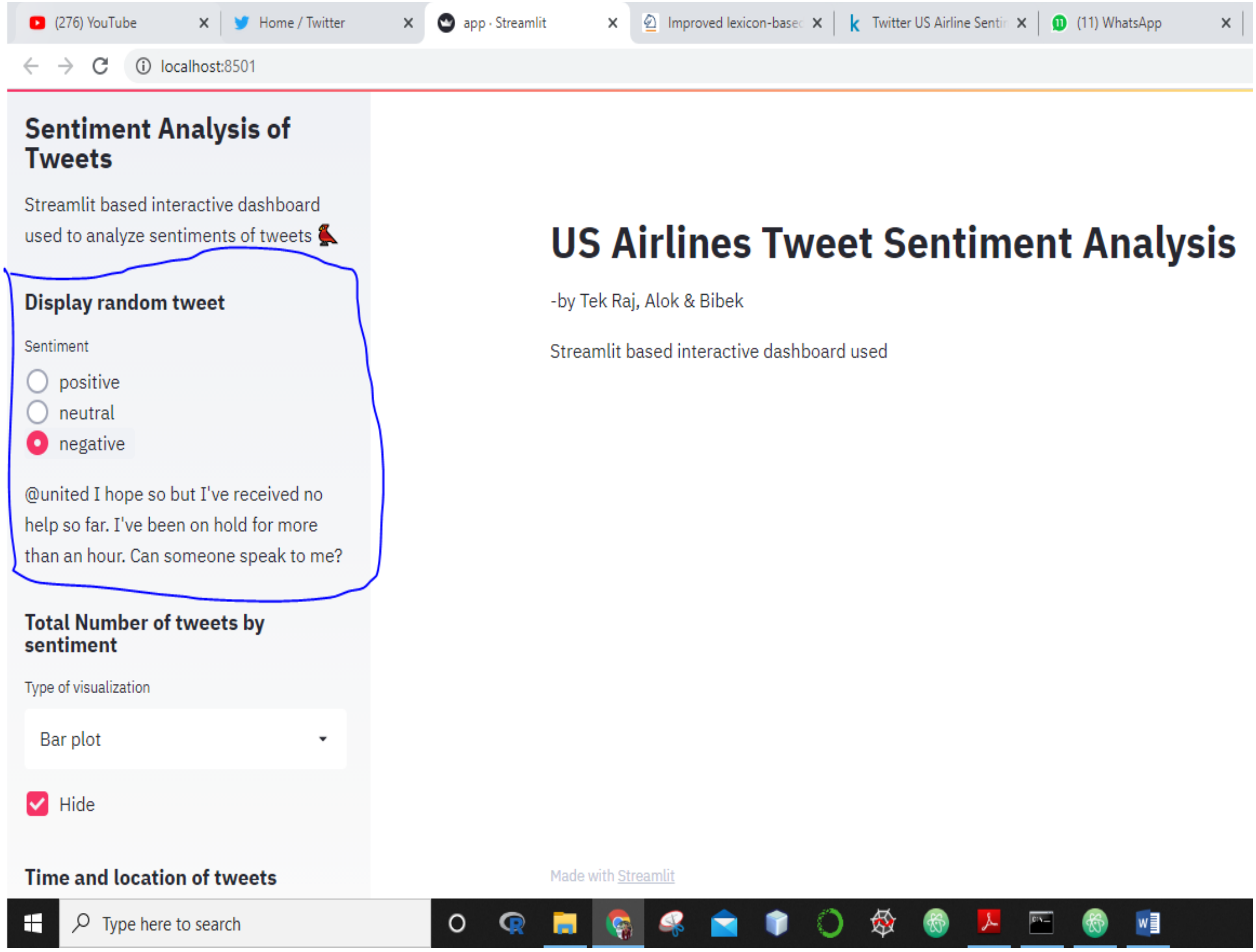

(275) YouTube
(1) Meeting now (Meeting) |
Home / Twitter
app · Streamlit
localhost:8501
Total Number of tweets by sentiment
Type of visualization
Bar plot
Bar plot
Pie chart
Hour to look at
0
23
Close
Total number of tweets for each airline
Type of visualization
Bar plot
Close
Breakdown airline by sentiment
US Airlines Tweet Sentiment Analysis
-by Tek Raj, Alok & Bibek
Streamlit based interactive dashboard used
Made with Streamlit
Type here to search
2:40 PM
11/6/2020


(275) YouTube
(1) Meeting now (Meeting) |
Home / Twitter
app · Streamlit
localhost:8501
Time and location of tweets
Hour to look at
0
23
Close
US Airways
United
American
Southwest
Delta
Virgin America
Choose an option
Breakdown airline by sentiment
Pick airlines
Choose an option
US Airlines Tweet Sentiment Analysis
-by Tek Raj, Alok & Bibek
Streamlit based interactive dashboard used
Made with Streamlit
Type here to search
2:42 PM
11/6/2020



Displaying Random Tweet form dataset according to user's choice.

In this project, we have employed Lexicon based Sentiment Analysis which recognizes the sentiment of tweet by counting number of positive, negative and neutral words. And then displaying any random tweet according to our choice.

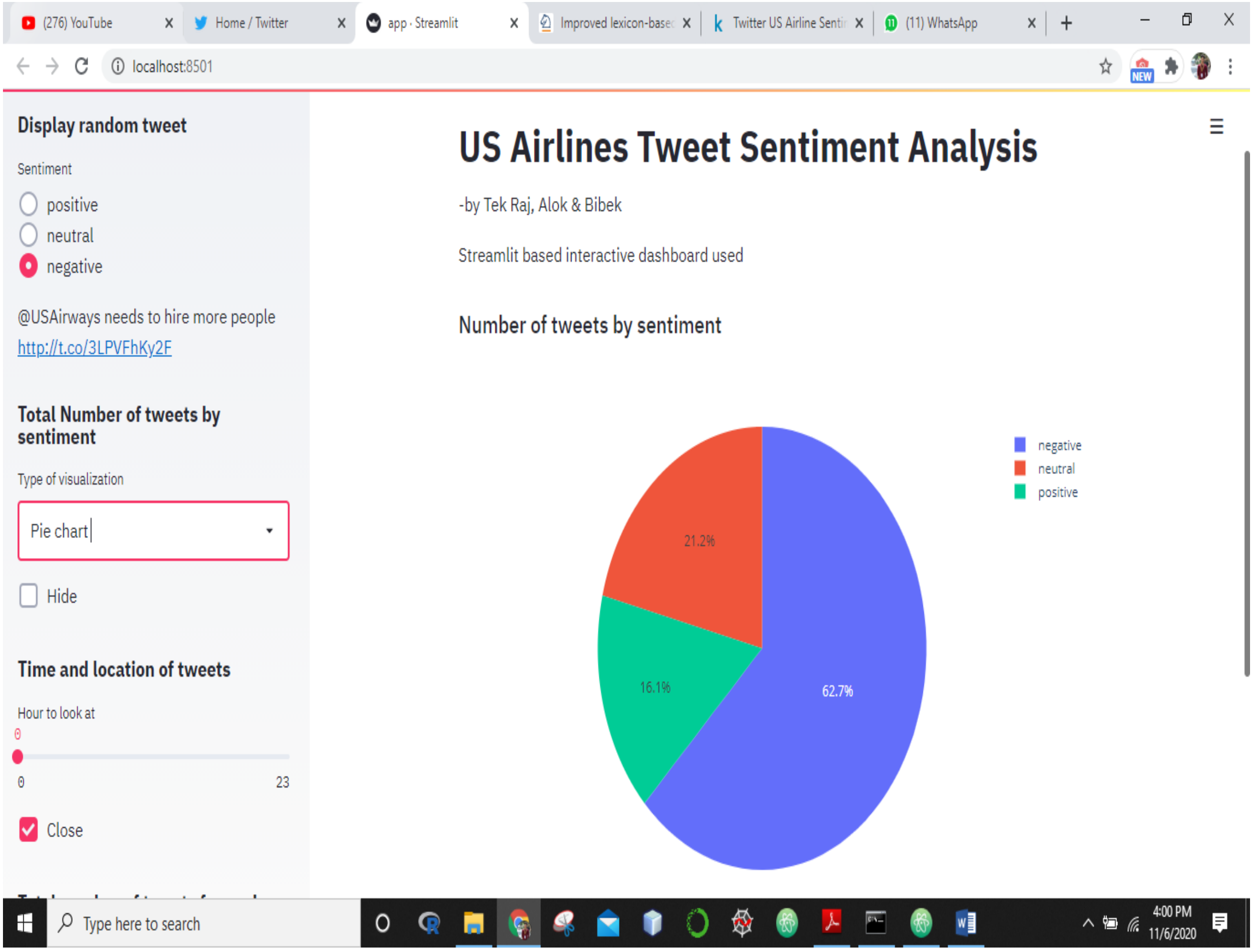


Visualization of total number of Tweets by their sentiment in Bar plot

The bar plot visualized below shows that 9000+ US Airlines tweets are of negative sentiment, 3000 of neutral and around 2500 tweets of positive sentiment. This dashboard depicts that customers or passengers are not at all satisfied by the service provided by US Airlines.

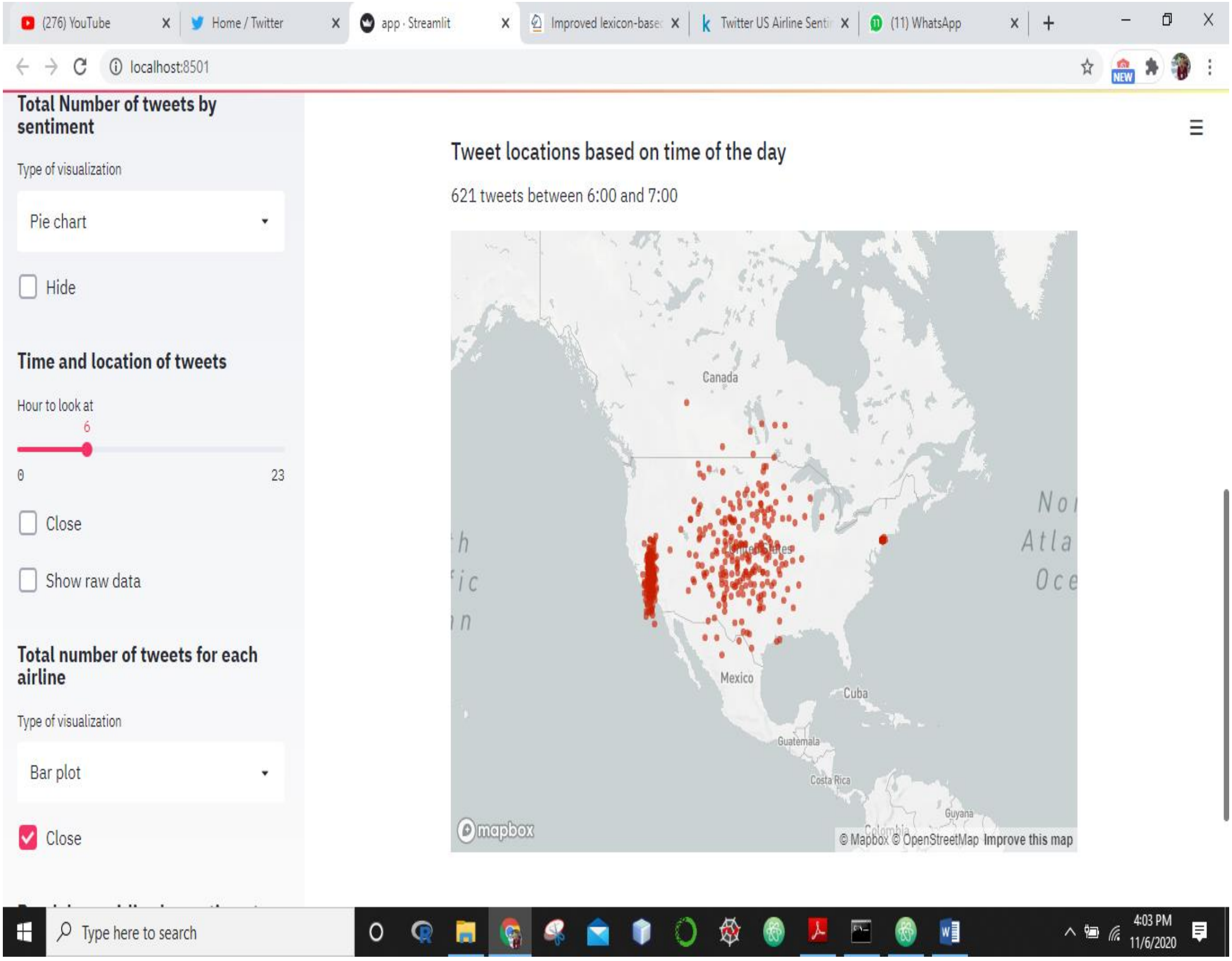
Visualization of total number of Tweets by their sentiment in Pie-chart

The pie-chart visualized shows that 62.7% of US Airlines tweet are of negative sentiment, 21.2% of neutral and 16.1% of positive sentiment. This dashboard depicts that customers or passengers are not at all satisfied by the service provided by US Airlines.



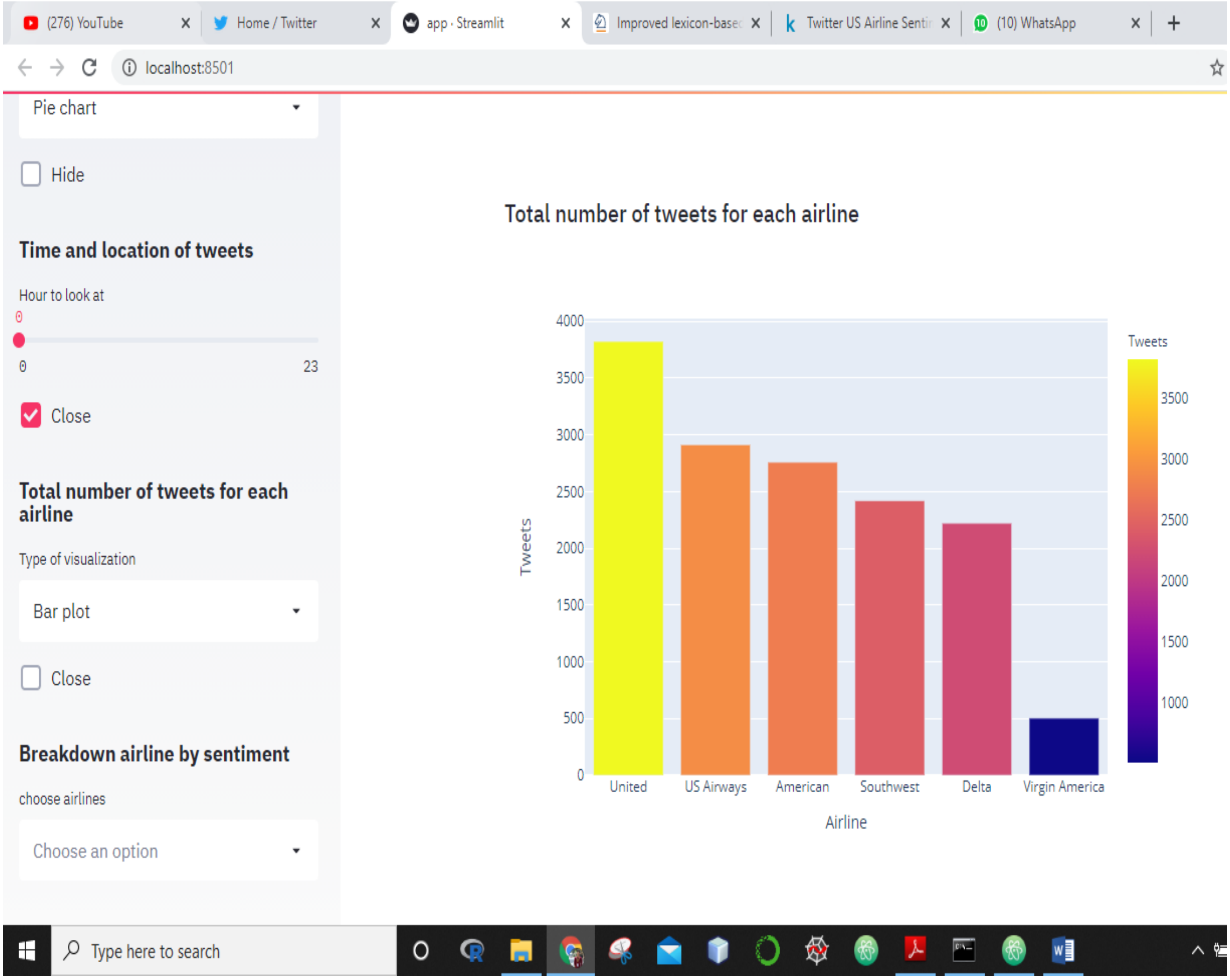
Tweets based on Time and Location of Tweets

Map based visualization of Tweets based on the when and from where the tweets are posted. This helps the Airlines company improve their services on travels on time basis or places basis, ratgetting the customer's feedback.



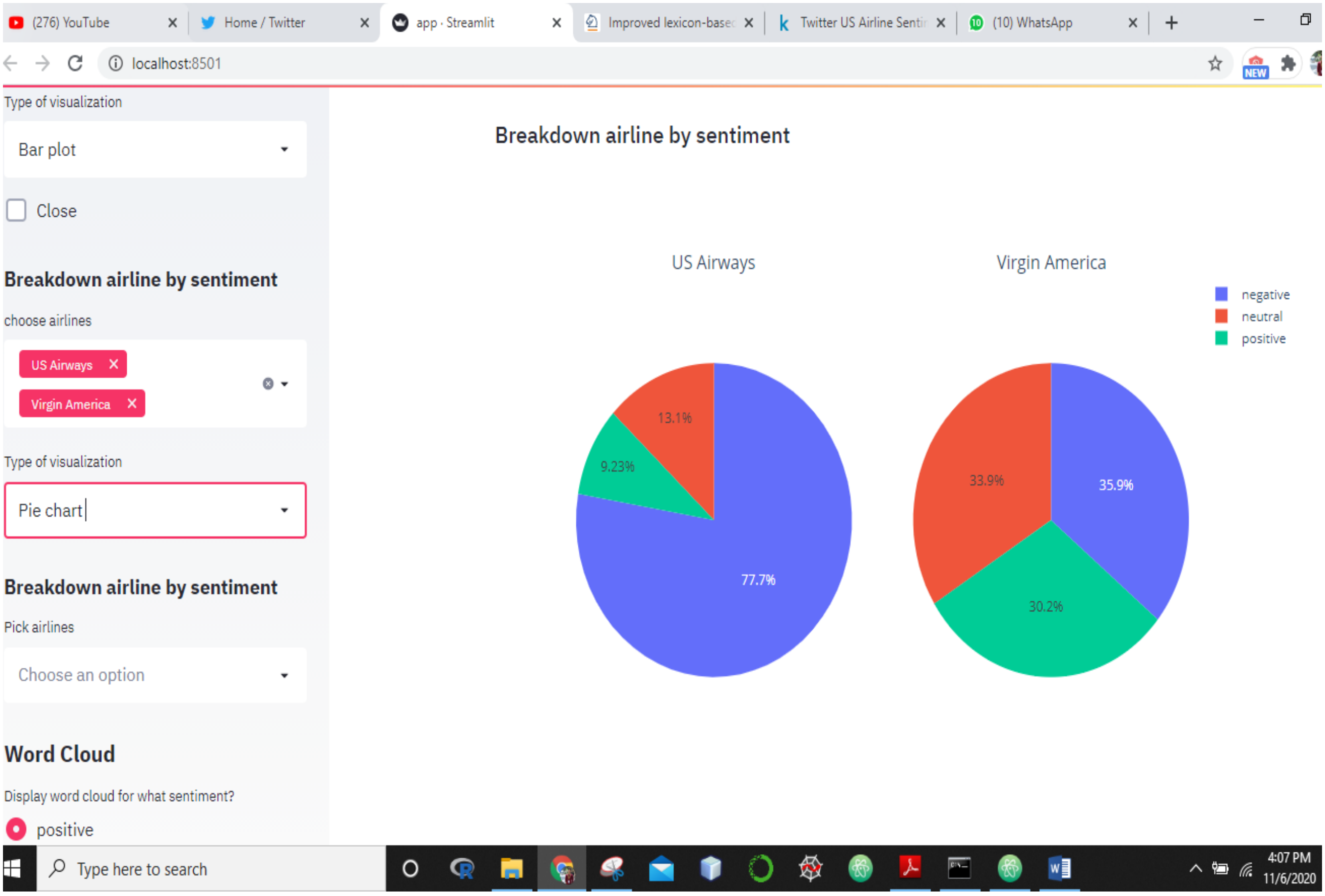
(276) YouTube
Home / Twitter
app · Streamlit
Improved lexicon-base
Twitter US Airline Senti
(11) WhatsApp
localhost:8501
Total Number of tweets by sentiment
Type of visualization
Pie chart
Hide
Time and location of tweets
Hour to look at
6
0
23
Close
Show raw data
Total number of tweets for each airline
Type of visualization
Bar plot
Close
Tweet locations based on time of the day
621 tweets between 6:00 and 7:00
Canada
Mexico
Cuba
Guatemala
Costa Rica
Guyana
mapbox
© Mapbox © OpenStreetMap Improve this map
Type here to search
4:03 PM
11/6/2020

Total number of tweets for each airline



Breakdown Airlines by sentiment

Here, users can choose the name of Airlines and then see the interactive dashboards of all airline's tweet sentiment. We have visualized the pie-chart dashboard of US Airways and Virginia America based on tweet sentiment.

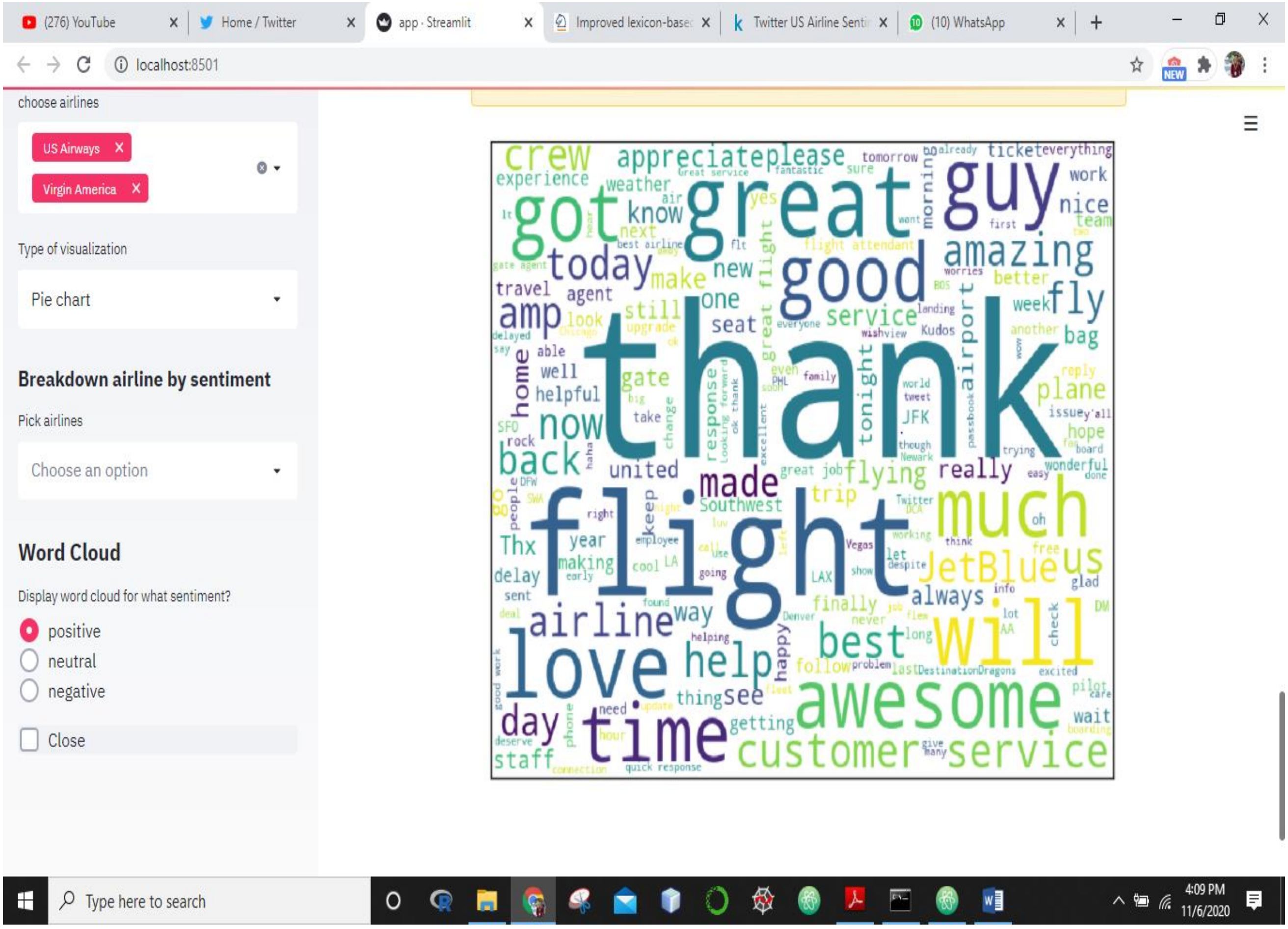


Word Cloud of Tweets

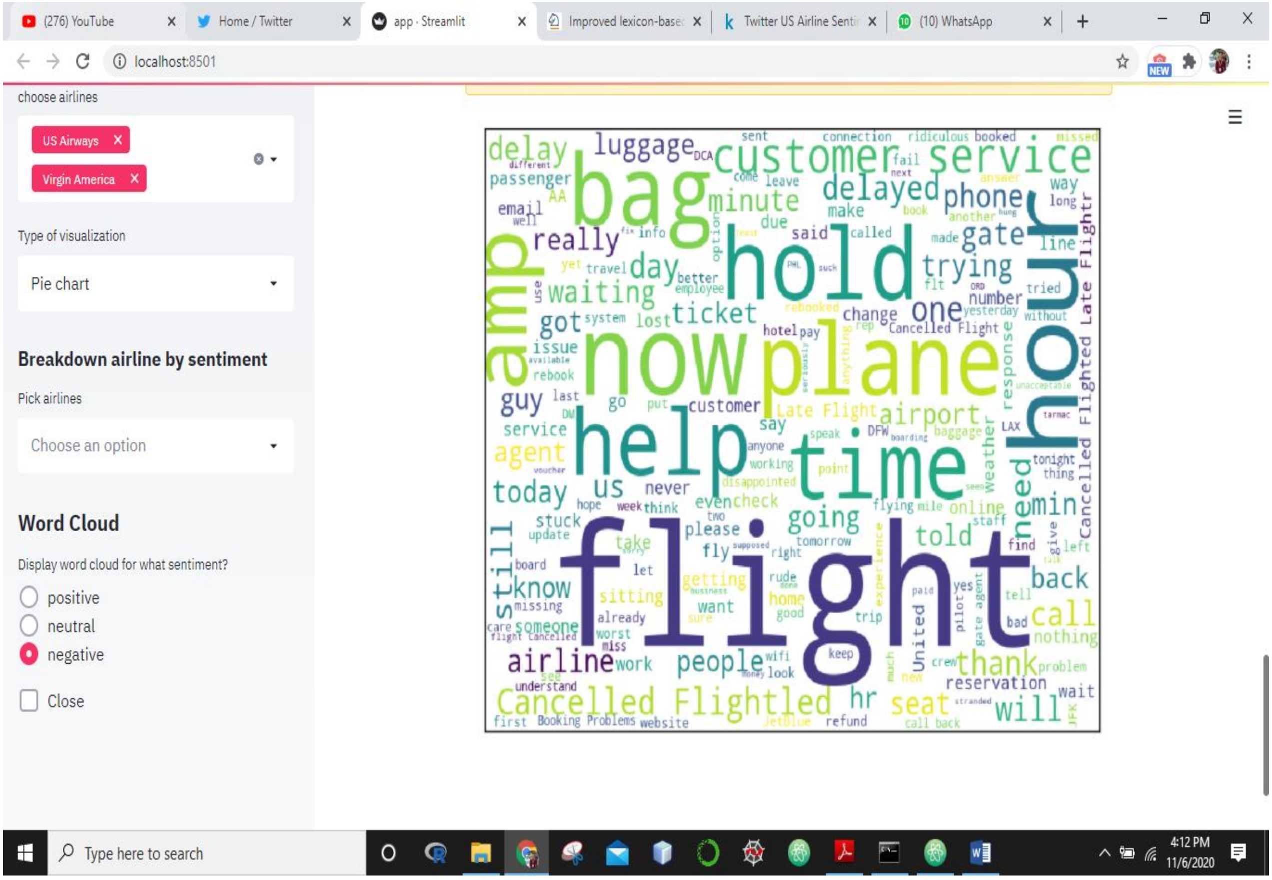
We have done Text pre-processing or text mining to remove the unnecessary words in tweet which doesn't carry any meaning in the tweet for sentiment classification like http or links, @, #, RT. So, these words are removed using Stop word in-built function and then Word Cloud based on Tweet Sentiment is displayed for each sentiment type.

After doing Text pre-processing, stop words removal, we have visualized the Word Cloud of all tweet sentiments. Word cloud contains the cloud of words in the tweet dataset, based on the type of sentiment.

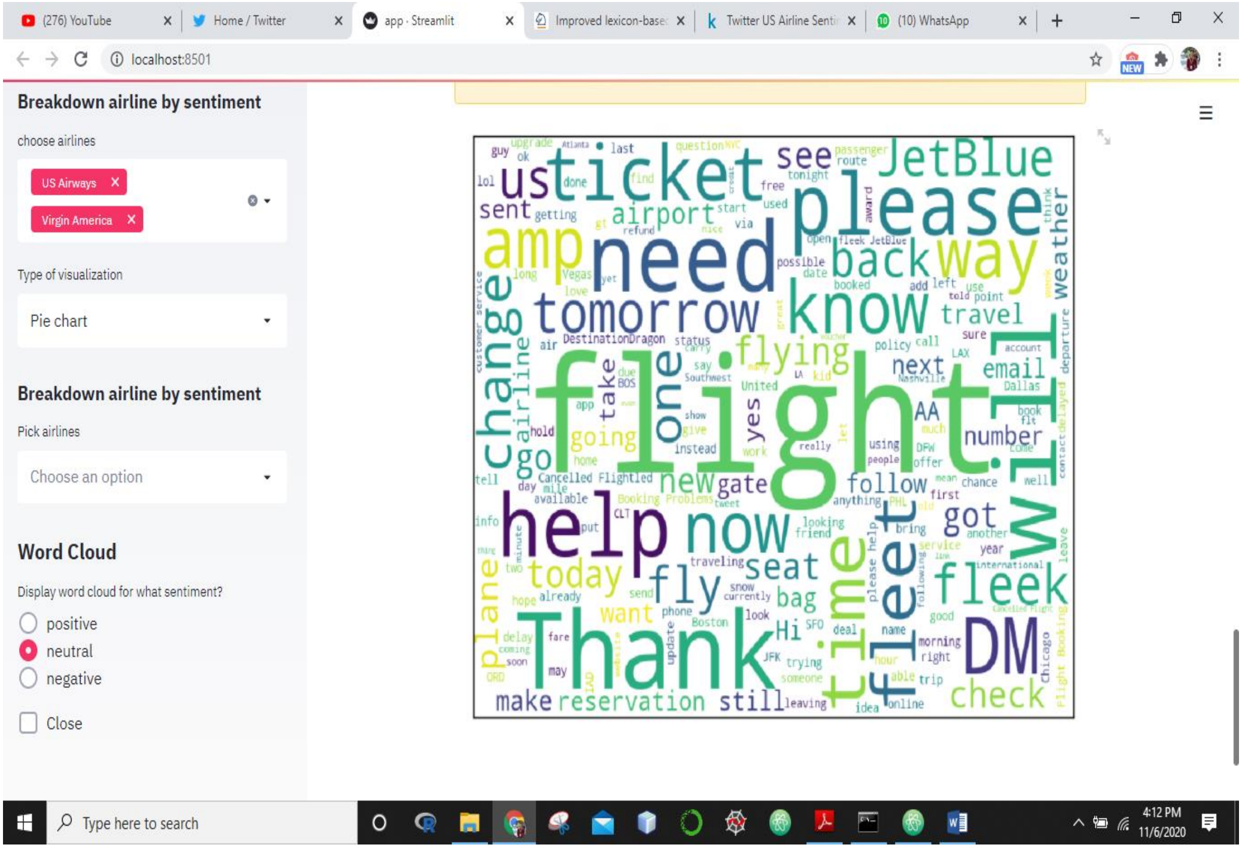
Word Cloud for Positive tweets



Word Cloud for Negative tweets



Word Cloud for Neutral tweets



Codes:

app.py — E:\Jupyter Notebook — Atom

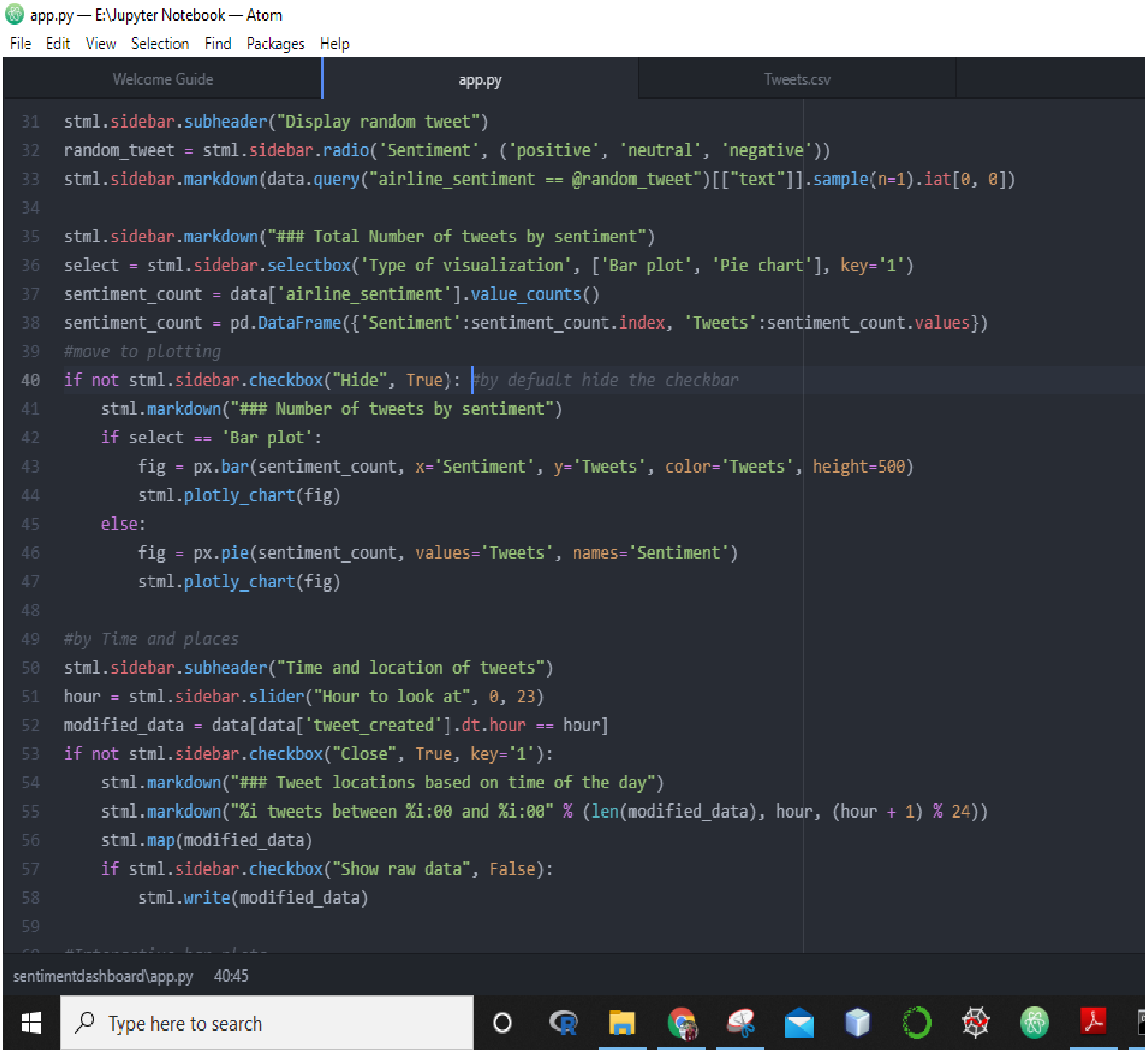
File Edit View Selection Find Packages Help

Welcome Guide | app.py | Tweets.csv

```python
import streamlit as stml
import pandas as pd
import numpy as np
import plotly.express as px
from plotly.subplots import make_subplots
import plotly.graph_objects as go
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt


DATA_URL = (
    "E:\Jupyter Notebook\sentimentdashboard\Tweets.csv"
)

stml.title("US Airlines Tweet Sentiment Analysis")
stml.markdown("-by Tek Raj, Alok & Bibek")
stml.sidebar.title("Sentiment Analysis of Tweets")
stml.markdown("Streamlit based interactive dashboard used ")

stml.sidebar.markdown("Streamlit based interactive dashboard used "
            "to analyze sentiments of tweets 🐦")

@stml.cache(persist=True)
def load_data():
    data = pd.read_csv(DATA_URL)
    data['tweet_created'] = pd.to_datetime(data['tweet_created'])
    return data

data = load_data()
```

sentimentdashboard\app.py 49:19

Type here to search




```python
stml.sidebar.subheader("Display random tweet")
random_tweet = stml.sidebar.radio('Sentiment', ('positive', 'neutral', 'negative'))
stml.sidebar.markdown(data.query("airline_sentiment == @random_tweet")[["text"]].sample(n=1).iat[0, 0])

stml.sidebar.markdown("### Total Number of tweets by sentiment")
select = stml.sidebar.selectbox('Type of visualization', ['Bar plot', 'Pie chart'], key='1')
sentiment_count = data['airline_sentiment'].value_counts()
sentiment_count = pd.DataFrame({'Sentiment':sentiment_count.index, 'Tweets':sentiment_count.values})
#move to plotting
if not stml.sidebar.checkbox("Hide", True): #by defualt hide the checkbar
    stml.markdown("### Number of tweets by sentiment")
    if select == 'Bar plot':
        fig = px.bar(sentiment_count, x='Sentiment', y='Tweets', color='Tweets', height=500)
        stml.plotly_chart(fig)
    else:
        fig = px.pie(sentiment_count, values='Tweets', names='Sentiment')
        stml.plotly_chart(fig)

#by Time and places
stml.sidebar.subheader("Time and location of tweets")
hour = stml.sidebar.slider("Hour to look at", 0, 23)
modified_data = data[data['tweet_created'].dt.hour == hour]
if not stml.sidebar.checkbox("Close", True, key='1'):
    stml.markdown("### Tweet locations based on time of the day")
    stml.markdown("%i tweets between %i:00 and %i:00" % (len(modified_data), hour, (hour + 1) % 24))
    stml.map(modified_data)
    if stml.sidebar.checkbox("Show raw data", False):
        stml.write(modified_data)
```

app.py — E:\Jupyter Notebook — Atom

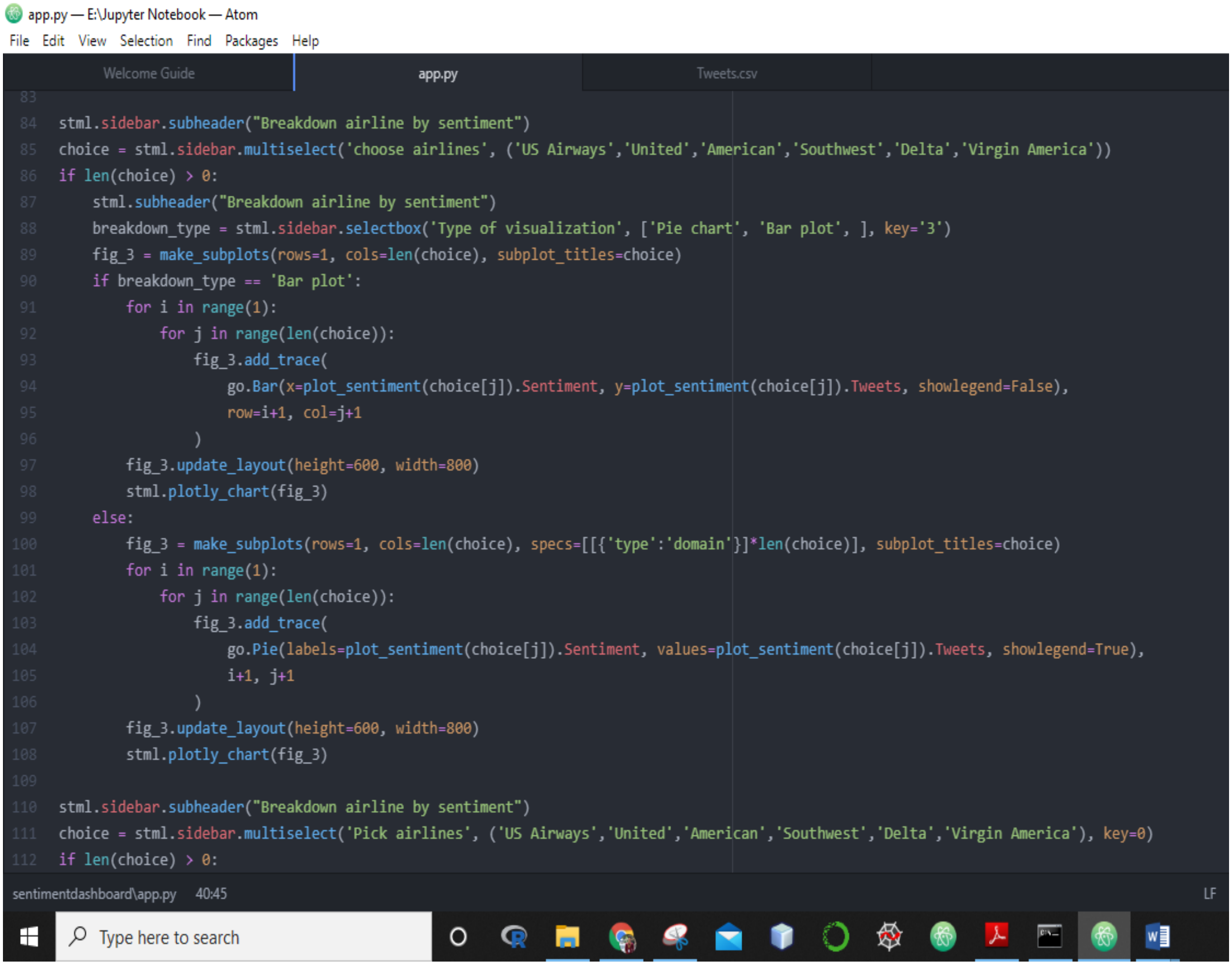
File Edit View Selection Find Packages Help

Welcome Guide | app.py | Tweets.csv

```python
        stml.write(modified_data)

#Interactive bar plots
stml.sidebar.subheader("Total number of tweets for each airline")
each_airline = stml.sidebar.selectbox('Type of visualization', ['Bar plot', 'Pie chart'], key='2')
airline_sentiment_count = data.groupby('airline')['airline_sentiment'].count().sort_values(ascending=False)
airline_sentiment_count = pd.DataFrame({'Airline':airline_sentiment_count.index, 'Tweets':airline_sentiment_count.values.flatten()})
if not stml.sidebar.checkbox("Close", True, key='2'):
    if each_airline == 'Bar plot':
        stml.subheader("Total number of tweets for each airline")
        fig_1 = px.bar(airline_sentiment_count, x='Airline', y='Tweets', color='Tweets', height=500)
        stml.plotly_chart(fig_1)
    if each_airline == 'Pie chart':
        stml.subheader("Total number of tweets for each airline")
        fig_2 = px.pie(airline_sentiment_count, values='Tweets', names='Airline')
        stml.plotly_chart(fig_2)

#No. of Tweets by sentiment of Each Airline
@stml.cache(persist=True)
def plot_sentiment(airline):
    df = data[data['airline']==airline]
    count = df['airline_sentiment'].value_counts()
    count = pd.DataFrame({'Sentiment':count.index, 'Tweets':count.values.flatten()})
    return count


stml.sidebar.subheader("Breakdown airline by sentiment")
choice = stml.sidebar.multiselect('choose airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'))
if len(choice) > 0:
    stml.subheader("Breakdown airline by sentiment")
```

sentimentdashboard\app.py 40:45 LF

Type here to search



app.py — E:\Jupyter Notebook — Atom

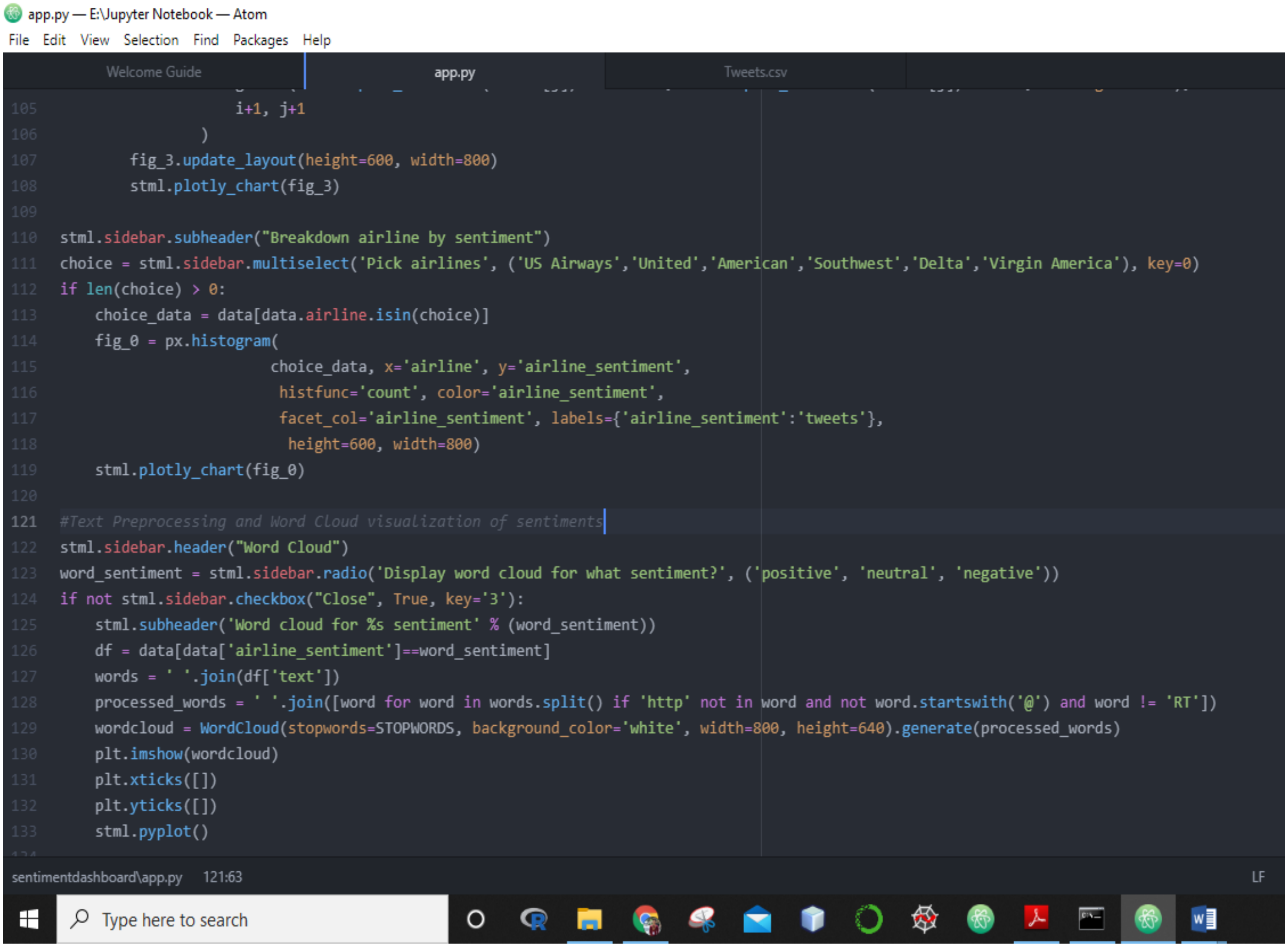
File Edit View Selection Find Packages Help

Welcome Guide | app.py | Tweets.csv

```python
stml.sidebar.subheader("Breakdown airline by sentiment")
choice = stml.sidebar.multiselect('choose airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'))
if len(choice) > 0:
    stml.subheader("Breakdown airline by sentiment")
    breakdown_type = stml.sidebar.selectbox('Type of visualization', ['Pie chart', 'Bar plot', ], key='3')
    fig_3 = make_subplots(rows=1, cols=len(choice), subplot_titles=choice)
    if breakdown_type == 'Bar plot':
        for i in range(1):
            for j in range(len(choice)):
                fig_3.add_trace(
                    go.Bar(x=plot_sentiment(choice[j]).Sentiment, y=plot_sentiment(choice[j]).Tweets, showlegend=False),
                    row=i+1, col=j+1
                )
        fig_3.update_layout(height=600, width=800)
        stml.plotly_chart(fig_3)
    else:
        fig_3 = make_subplots(rows=1, cols=len(choice), specs=[[{'type':'domain'}]*len(choice)], subplot_titles=choice)
        for i in range(1):
            for j in range(len(choice)):
                fig_3.add_trace(
                    go.Pie(labels=plot_sentiment(choice[j]).Sentiment, values=plot_sentiment(choice[j]).Tweets, showlegend=True),
                    i+1, j+1
                )
        fig_3.update_layout(height=600, width=800)
        stml.plotly_chart(fig_3)

stml.sidebar.subheader("Breakdown airline by sentiment")
choice = stml.sidebar.multiselect('Pick airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'), key=0)
if len(choice) > 0:
```

sentimentdashboard\app.py 40:45 LF

Type here to search




```python
                    i+1, j+1
                )
        fig_3.update_layout(height=600, width=800)
        stml.plotly_chart(fig_3)

stml.sidebar.subheader("Breakdown airline by sentiment")
choice = stml.sidebar.multiselect('Pick airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'), key=0)
if len(choice) > 0:
    choice_data = data[data.airline.isin(choice)]
    fig_0 = px.histogram(
                        choice_data, x='airline', y='airline_sentiment',
                         histfunc='count', color='airline_sentiment',
                         facet_col='airline_sentiment', labels={'airline_sentiment':'tweets'},
                          height=600, width=800)
    stml.plotly_chart(fig_0)

#Text Preprocessing and Word Cloud visualization of sentiments
stml.sidebar.header("Word Cloud")
word_sentiment = stml.sidebar.radio('Display word cloud for what sentiment?', ('positive', 'neutral', 'negative'))
if not stml.sidebar.checkbox("Close", True, key='3'):
    stml.subheader('Word cloud for %s sentiment' % (word_sentiment))
    df = data[data['airline_sentiment']==word_sentiment]
    words = ' '.join(df['text'])
    processed_words = ' '.join([word for word in words.split() if 'http' not in word and not word.startswith('@') and word != 'RT'])
    wordcloud = WordCloud(stopwords=STOPWORDS, background_color='white', width=800, height=640).generate(processed_words)
    plt.imshow(wordcloud)
    plt.xticks([])
    plt.yticks([])
    stml.pyplot()
```

```
import streamlit as stml
import pandas as pd
import numpy as np
import plotly.express as px
from plotly.subplots import make_subplots
import plotly.graph_objects as go
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt



DATA_URL = (
    "E:\Jupyter Notebook\sentimentdashboard\Tweets.csv"
)

stml.title("US Airlines Tweet Sentiment Analysis")
stml.markdown("-by Tek Raj, Alok & Bibek")
stml.sidebar.title("Sentiment Analysis of Tweets")
stml.markdown("Streamlit based interactive dashboard used ")

stml.sidebar.markdown("Streamlit based interactive dashboard used "
            "to analyze sentiments of tweets 🐦 ")

@stml.cache(persist=True)
def load_data():
    data = pd.read_csv(DATA_URL)
    data['tweet_created'] = pd.to_datetime(data['tweet_created'])
    return data

data = load_data()
```

```
stml.sidebar.subheader("Display random tweet")

random_tweet = stml.sidebar.radio('Sentiment', ('positive', 'neutral', 'negative'))

stml.sidebar.markdown(data.query("airline_sentiment ==
@random_tweet")[["text"]].sample(n=1).iat[0, 0])


stml.sidebar.markdown("### Total Number of tweets by sentiment")

select = stml.sidebar.selectbox('Type of visualization', ['Bar plot', 'Pie chart'], key='1')

sentiment_count = data['airline_sentiment'].value_counts()

sentiment_count = pd.DataFrame({'Sentiment':sentiment_count.index,
'Tweets':sentiment_count.values})

#move to plotting

if not stml.sidebar.checkbox("Hide", True): #by defualt hide the checkbar

   stml.markdown("### Number of tweets by sentiment")

   if select == 'Bar plot':

      fig = px.bar(sentiment_count, x='Sentiment', y='Tweets', color='Tweets', height=500)

      stml.plotly_chart(fig)

   else:

      fig = px.pie(sentiment_count, values='Tweets', names='Sentiment')

      stml.plotly_chart(fig)


#by Time and places

stml.sidebar.subheader("Time and location of tweets")

hour = stml.sidebar.slider("Hour to look at", 0, 23)

modified_data = data[data['tweet_created'].dt.hour == hour]

if not stml.sidebar.checkbox("Close", True, key='1'):

   stml.markdown("### Tweet locations based on time of the day")

   stml.markdown("%i tweets between %i:00 and %i:00" % (len(modified_data), hour, (hour
+ 1) % 24))

   stml.map(modified_data)
```

```
    if stml.sidebar.checkbox("Show raw data", False):
        stml.write(modified_data)


#Interactive bar plots
stml.sidebar.subheader("Total number of tweets for each airline")
each_airline = stml.sidebar.selectbox('Type of visualization', ['Bar plot', 'Pie chart'], key='2')
airline_sentiment_count =
data.groupby('airline')['airline_sentiment'].count().sort_values(ascending=False)
airline_sentiment_count = pd.DataFrame({'Airline':airline_sentiment_count.index,
'Tweets':airline_sentiment_count.values.flatten()})
if not stml.sidebar.checkbox("Close", True, key='2'):
    if each_airline == 'Bar plot':
        stml.subheader("Total number of tweets for each airline")
        fig_1 = px.bar(airline_sentiment_count, x='Airline', y='Tweets', color='Tweets',
height=500)
        stml.plotly_chart(fig_1)
    if each_airline == 'Pie chart':
        stml.subheader("Total number of tweets for each airline")
        fig_2 = px.pie(airline_sentiment_count, values='Tweets', names='Airline')
        stml.plotly_chart(fig_2)


#No. of Tweets by sentiment of Each Airline
@stml.cache(persist=True)
def plot_sentiment(airline):
    df = data[data['airline']==airline]
    count = df['airline_sentiment'].value_counts()
    count = pd.DataFrame({'Sentiment':count.index, 'Tweets':count.values.flatten()})
    return count
```

```
stml.sidebar.subheader("Breakdown airline by sentiment")

choice = stml.sidebar.multiselect('choose airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'))

if len(choice) > 0:

    stml.subheader("Breakdown airline by sentiment")

    breakdown_type = stml.sidebar.selectbox('Type of visualization', ['Pie chart', 'Bar plot', ], key='3')

    fig_3 = make_subplots(rows=1, cols=len(choice), subplot_titles=choice)

    if breakdown_type == 'Bar plot':

        for i in range(1):

            for j in range(len(choice)):

                fig_3.add_trace(

                    go.Bar(x=plot_sentiment(choice[j]).Sentiment, y=plot_sentiment(choice[j]).Tweets, showlegend=False),

                    row=i+1, col=j+1

                )

        fig_3.update_layout(height=600, width=800)

        stml.plotly_chart(fig_3)

    else:

        fig_3 = make_subplots(rows=1, cols=len(choice), specs=[[{'type':'domain'}]*len(choice)], subplot_titles=choice)

        for i in range(1):

            for j in range(len(choice)):

                fig_3.add_trace(

                    go.Pie(labels=plot_sentiment(choice[j]).Sentiment, values=plot_sentiment(choice[j]).Tweets, showlegend=True),

                    i+1, j+1

                )

        fig_3.update_layout(height=600, width=800)

        stml.plotly_chart(fig_3)
```

```
stml.sidebar.subheader("Breakdown airline by sentiment")
choice = stml.sidebar.multiselect('Pick airlines', ('US Airways','United','American','Southwest','Delta','Virgin America'), key=0)
if len(choice) > 0:
    choice_data = data[data.airline.isin(choice)]
    fig_0 = px.histogram(
                        choice_data, x='airline', y='airline_sentiment',
                         histfunc='count', color='airline_sentiment',
                         facet_col='airline_sentiment', labels={'airline_sentiment':'tweets'},
                          height=600, width=800)
    stml.plotly_chart(fig_0)


#Text Preprocessing and Word Cloud visualization of sentiments
stml.sidebar.header("Word Cloud")
word_sentiment = stml.sidebar.radio('Display word cloud for what sentiment?', ('positive', 'neutral', 'negative'))
if not stml.sidebar.checkbox("Close", True, key='3'):
    stml.subheader('Word cloud for %s sentiment' % (word_sentiment))
    df = data[data['airline_sentiment']==word_sentiment]
    words = ' '.join(df['text'])
    processed_words = ' '.join([word for word in words.split() if 'http' not in word and not word.startswith('@') and word != 'RT'])
    wordcloud = WordCloud(stopwords=STOPWORDS, background_color='white', width=800, height=640).generate(processed_words)
    plt.imshow(wordcloud)
    plt.xticks([])
    plt.yticks([])
    stml.pyplot()
```

9. CONCLUSION

This project help to understand about the people sentiment towards the airlines based on the different classification levels. It gives the overview of the airline which will tell the user it is associated with the sentiment is good or not. If the review is good then people will prefer and will likely to be more chance of boarding through the positive airlines. This is achieved with the attractive dashboard with different visualization where it gives the proper view and analysis of the sentiment. The main purpose of this project is to view the sentiment based on the airlines and take action accordingly. Not only user but also to the airlines will help to improve the facilities and infrastructure.

10.REFERNCES

[1]. Arif, M. N. I., Paplu, S. H., & Berns, K. Understanding Emotional Expressions in Text-Based Communication.

[2]. Rababah, O. M., & Alokaily, N. ANovel MACHINE LEARNING SYSTEM FOR SENTIMENT ANALYSIS AND EXTRACTION.

[3]. Oyebode, O., & Orji, R. (2019, October). Social media and sentiment analysis: The nigeria presidential election 2019. In *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (pp. 0140-0146). IEEE.

[4]. Isah, H., Trundle, P., & Neagu, D. (2014, September). Social media analysis for product safety using text mining and sentiment analysis. In *2014 14th UK workshop on computational intelligence (UKCI)* (pp. 1-7). IEEE.

[5]. Kharde, V., & Sonawane, P. (2016). Sentiment analysis of twitter data: a survey of techniques. *arXiv preprint arXiv:1601.06971*.

[6]. Del Vigna12, F., Cimino23, A., Dell'Orletta, F., Petrocchi, M., & Tesconi, M. (2017). Hate me, hate me not: Hate speech detection on facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17)* (pp. 86-95).