Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

Final Report

CS-179G-001-23F

Part 1: Data Collection Report

1a. Project description. What is the goal of your analysis, and what do you expect to find? Any hypotheses?

Project description:

- Sentiment Analysis of Political Subreddits posts

Motivation:

- Politics are a divisive issue in the United States
- Public opinions toward certain politicized topics and figures
- Understanding Public Opinion
    - We can understand the public opinion on specific political issues or keywords by analyzing sentiment in political subreddits

Hypotheses:

- Reddit posts with the keyword, “Republican” will have a negative sentiment trend
- Reddit posts with keywords related to political issues will have a negative sentiment.

Approach:

- We utilize a Web crawler, specifically PRAW (Python Reddit API Wrapper), to gather data from posts within the subreddit.
- Using the Twitter labeled dataset provided in lab 4 to train a machine learning model for sentiment analysis.
- We will employ a combination of a bar graph, table, and word cloud to effectively visualize our findings.

1b. Describe your data using a table (number of records, length of each record, number of attributes, and so on), a few example records, and a description of the data.

Number of posts: 9729

Number of columns: 7

- Post Title, ID, body, time of post, number of upvotes, number of comments, comments.

Size of Dataset -> 1.01GB

Since our data contains lots of post comments, we have a smaller number of posts (rows).

Screenshot of JSON file:

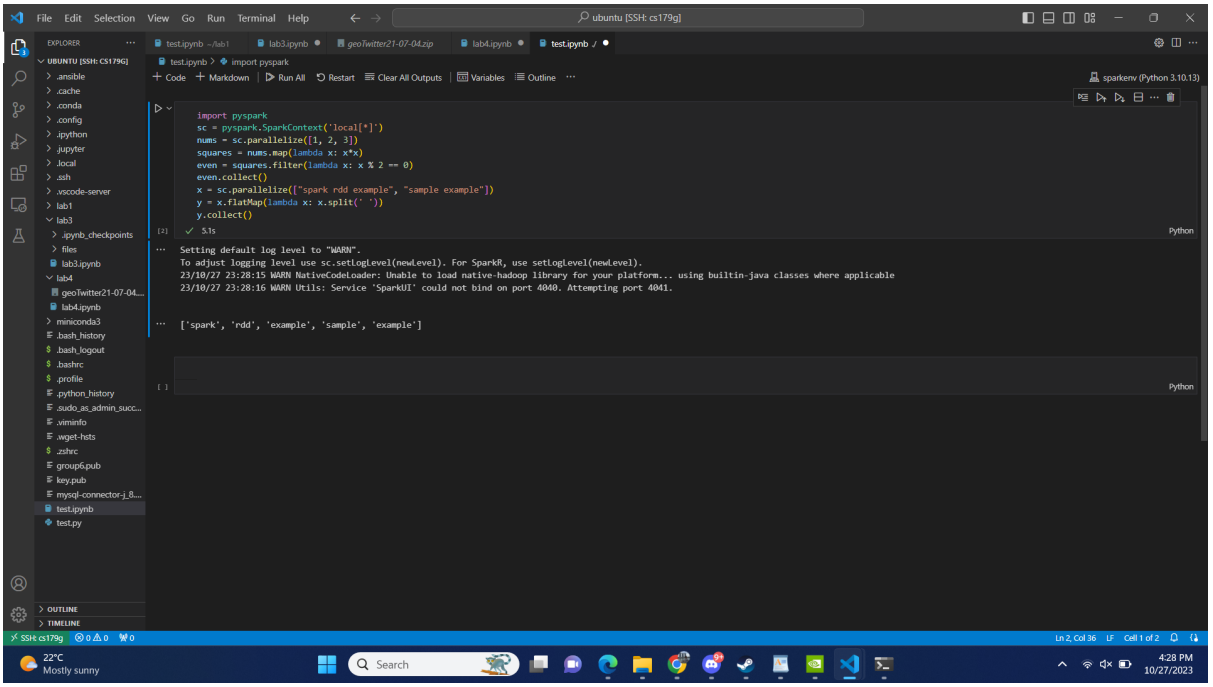

```
[{"postTitle":"Megathread: Joe Biden Projected to Defeat President Donald Trump and Win the 2020 US Presidential
Election","postID":"jptq5n","postBody":"Former Vice President Joseph Biden has secured the 270 electoral votes necessary to defeat President
Donald Trump and become the 46th President of the United States, according to multiple sources.\n\n---\n\n##Submissions that may interest
you\n\nSUBMISSION | DOMAIN\n---|----\n[Biden defeats Trump to win White House, NBC News projects](https://www.nbcnews.com/politics/2020-
election/biden-defeats-trump-win-white-house-nbc-news-projects-n1246912) | nbcnews.com\n[US election 2020: Joe Biden projected to beat Donald
Trump and become next president](http://news.sky.com/story/us-election-2020-joe-biden-projected-to-beat-donald-trump-and-become-next-president-
12123216) | news.sky.com\n[Joe Biden Presidential Election Win Over Trump](https://www.buzzfeednews.com/article/nidhiprakash/biden-won-2020-
election-trump-lost?utm_source=dynamic&utm_campaign=bftwbuzzfeed&ref=bftwbuzzfeed) | buzzfeednews.com\n[Joe Biden to become the 46th president of
the United States, CNN projects](https://www.cnn.com/2020/11/07/politics/joe-biden-wins-us-presidential-election) | cnn.com\n[Election results
live: Biden wins the presidency, defeats Trump](https://www.businessinsider.com/joe-biden-wins-general-election-against-donald-trump-2020-11) |
businessinsider.com\n[Joe Biden to become the 46th president of the United States, CNN projects](https://www.cnn.com/2020/11/07/politics/joe-
biden-wins-us-presidential-election/index.html) | cnn.com\n[Joe Biden is projected to defeat incumbent Donald Trump in the presidential election]
(https://www.cnbc.com/2020/11/07/2020-election-winner-biden-final-count-results.html?
__source=iosappshare%7Ccom.apple.UIKit.activity.PostToTwitter) | cnbc.com\n[CNN projects Joe Biden has won the 2020 Presidential Race]
(https://www.cnn.com/election/2020/results/state/pennsylvania/president?iid=politics_election_crm) | cnn.com\n[Joe Biden is projected to defeat
incumbent Donald Trump in the presidential election](https://www.cnbc.com/amp/2020/11/07/2020-election-winner-biden-final-count-results.html?
__twitter_impression=true) | cnbc.com\n[Democrats Waste No Time Punching Left in the Wake of Biden\u2019s Win]
(https://theintercept.com/2020/11/06/election-biden-democrats-progressives/) | theintercept.com\n[Biden edges closer to win as Pennsylvania focus
intensifies](https://apnews.com/article/Biden-Trump-US-election-2020-results-fd58df73aa677acb74fce2a69adb71f9) | apnews.com\n[How Indigenous
voters swung the 2020 election - In Arizona and Wisconsin, Native turnout \u2014 which often leans liberal \u2014 made the difference in
Biden\u2019s slim but winning margin.](https://www.hcn.org/articles/indigenous-affairs-how-indigenous-voters-swung-the-2020-election) |
hcn.org\n[Joe Biden elected president of the United States](https://apnews.com/article/election-2020-joe-biden-north-america-national-elections-
elections-7200c2d4901d8e47f1302954685a737f) | apnews.com\n[Joe Biden Wins Pennsylvania, Clinching Presidency in Historic Comeback]
(https://www.nbcphiladelphia.com/news/politics/decision-2020/joe-biden-wins-pennsylvania-clinching-presidency-in-historic-comeback/2581293/) |
nbcphiladelphia.com\n[Joe Biden elected president of the United States](https://www.npr.org/2020/11/07/928803493/biden-wins-presidency-according-
to-ap-edging-trump-in-turbulent-race?utm_term=nprnews&utm_medium=social&utm_source=twitter.com&utm_campaign=npr) | npr.org\n[Biden wins]
(https://www.nytimes.com/interactive/2020/11/03/us/elections/results-president.html?action=click&pgtype=Article&state=default&module=styln-
elections-2020&region=TOP_BANNER&context=election_recirc) | nytimes.com\n[America Won](https://www.theatlantic.com/ideas/archive/2020/11/joe-
biden-wins/616960/) | theatlantic.com\n[US election live results: Joe Biden wins, say US media - latest news]
(https://www.telegraph.co.uk/news/2020/11/07/us-election-results-2020-live-joe-biden-donald-trump-president/) | telegraph.co.uk\n[CNN and MSNBC
projects President elect Joe Biden](https://www.msnbc.com/msnbc/amp-video/mmvo95509573627?__twitter_impression=true) | msnbc.com\n[Biden wins:
Democrat who vowed a return to 'normalcy' defeats Trump after cliffhanger election](https://amp.usatoday.com/amp/6168297002) |
```

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

Screenshot of starting dataframe:

```
+-----------+----------+--------+--------------------+-------+-------------+--------------------+
|numComments|numUpvotes|postBody|        postComments| postID|     postTime|           postTitle|
+-----------+----------+--------+--------------------+-------+-------------+--------------------+
|       NULL|      NULL|    NULL|                NULL|   NULL|         NULL|                NULL|
|         70|       317|        |[Hi all,\n\nA rem...|17jcqmj|1.698613842E9|The federal defic...|
|        145|      1036|        |[Hi all,\n\nA rem...|17j3ruj|1.698588734E9|The ‘great wealth...|
|        238|      1190|        |[Hi all,\n\nA rem...|17j0lr4|1.698577439E9|Check Your Email:...|
|         17|       113|        |[Hi all,\n\nA rem...|17jcyv4|1.698614455E9|The ‘great wealth...|
|        142|       375|        |[Hi all,\n\nA rem...|17j3rhj|1.698588704E9|When Idiot Savant...|
|          1|        23|        |[Hi all,\n\nA rem...|17jklhk|1.698637256E9|More Americans fa...|
|         47|       174|        |[Hi all,\n\nA rem...|17j6bdg|1.698596094E9|Welcome to the ag...|
|          7|        80|        |[Hi all,\n\nA rem...|17jb7s8|1.698609639E9|Hold onto your ha...|
|         47|        61|        |[Hi all,\n\nA rem...|17jbdpd|1.698610078E9|Australia’s econo...|
|         46|        24|        |[Hi all,\n\nA rem...|17jh3pk|1.698626181E9|Question-if the f...|
|          1|         5|        |[Hi all,\n\nA rem...|17jituo|1.698631514E9|Spain to be euroz...|
|          1|        17|        |[Hi all,\n\nA rem...|17j83u8|1.698601115E9|Inflation & Unemp...|
|        174|       823|        |[Hi all,\n\nA rem...|17ilmfa|1.698522834E9|To revive Canada’...|
|          1|         2|        |[Hi all,\n\nA rem...|17jknn3|1.698637482E9|Lower PCE Has Mar...|
|          1|         2|        |[Hi all,\n\nA rem...|17jjbel|1.698633045E9|Q3 GDP Beats Expe...|
|        413|       864|        |[Hi all,\n\nA rem...|17ieoqk|1.698502521E9|Never Mind the 1%...|
|        142|       364|        |[Hi all,\n\nA rem...|17ig5r3| 1.69850696E9|Gen X leads in ne...|
|          4|       113|        |[Hi all,\n\nA rem...|17imvg3|1.698526508E9|Canon invents a n...|
|        315|       560|        |[Hi all,\n\nA rem...|17ib6n7|1.698490075E9|Union workers sco...|
+-----------+----------+--------+--------------------+-------+-------------+--------------------+
only showing top 20 rows
```

1c. Run the following code on Spark and show screenshots of the results



Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

Screenshot of Web crawler

```
#submissions = subreddit.top(time_filter="all", limit=None)  # for 'top' posts
submissions = subreddit.top(time_filter="all")                # for 'hot' posts
for items in submissions:
    if len(items.comments.list()) == 0:
        postDict.update({"postComments": "no comments"})

    commentsCollection = []

    for comment in items.comments.list():
        if isinstance(comment, MoreComments):
            continue
        commentsCollection.append(comment.body)

    postDict.update({"postBody": items.selftext})
    postDict.update({"postTitle": items.title})
    postDict.update({"postID": items.id})
    postDict.update({"postTime": items.created_utc})
    postDict.update({"numUpvotes": items.score})
    postDict.update({"numComments": len(items.comments.list())})
    postDict.update({"postComments": commentsCollection})

    # Estimate the size of the current post and update the data size
    post_size = estimate_data_size(postDict)
    current_data_size += post_size

    postCollection.append(postDict.copy())

# Output data into a JSON file
with open("sample.json", "w") as outfile:
    outfile.write('[' + ',\n'.join(json.dumps(i, separators=(',', ':')) for i in postCollection) + ']\n')
```

Part 2:Spark Data Processing and Store in MySQL.

Cleaning Data

- replace NaN or missing value with 0
- removed rows where the post had no text for the post body and post title
- combined post body and post title into a single post text column
- converted the text column to lowercase using the ‘.lower()’ string method
- translated the date column from Unix time to readable date values
- removed unnecessary columns from the main data frame

Sentiment analysis

- Using the Twitter labeled dataset provided in lab 4 to train a machine learning model for sentiment analysis. (accuracy of 75.96% on Twitter data)
- Subsequently, the trained model was utilized to append a predicted post sentiment column to our data frame.

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

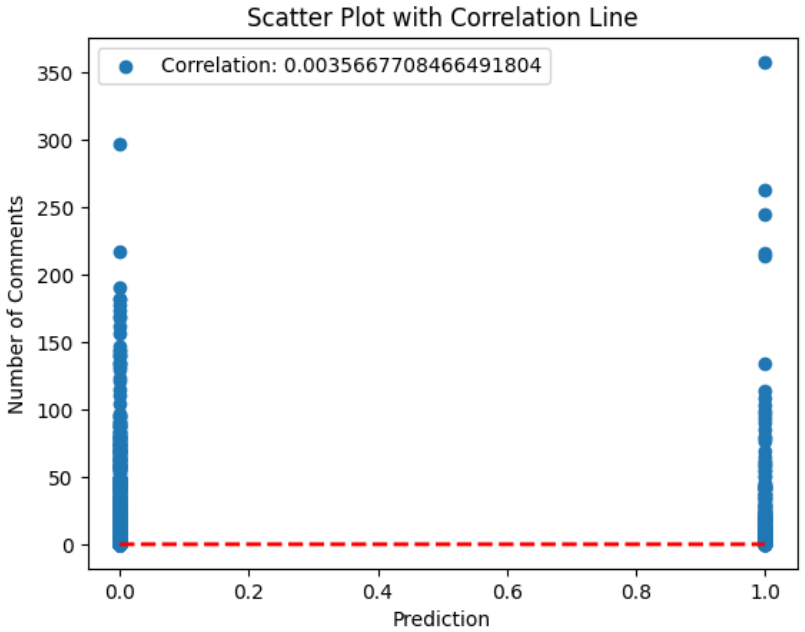
	label	features	rawPrediction	probability	prediction
0	0.0	(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 3.24530063...	[0.6190829283761483, -0.6190829283761483]	[0.6500099462279291, 0.34999005377207093]	0.0
1	0.0	(0.0, 0.0, 3.0071857325997775, 0.0, 0.0, 0.0, ...	[3.0475891618239572, -3.0475891618239572]	[0.9546783295613331, 0.04532167043866686]	0.0
2	0.0	(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[1.2134188518981317, -1.2134188518981317]	[0.7709033164066016, 0.2290966835933984]	0.0
3	0.0	(0.0, 0.0, 0.0, 0.0, 3.1167571186782292, 0.0, ...	[3.2691854781561256, -3.2691854781561256]	[0.9633564292863565, 0.03664357071364355]	0.0
4	0.0	(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[-2.0798257656509858, 2.0798257656509858]	[0.11107316873441093, 0.888926831265589]	1.0

id	numComments	numUpvotes	postBody	postComments	postTitle	date_column	combinedTextColumn	label	features	rawPrediction	probability	prediction
0	60	27	I'm trying to ful...	[Why should the D...	Do you feel like ...	2023-10-28	Do you feel like ...	0.0	(262144,[7],[3.24...	[0.61908292837614...	[0.65000994622792...	0.0
1	1	10	[Netanyahu is a c...	#TBT: Clinton Con...	#TBT: Clinton Con...	2023-10-28	#TBT: Clinton Con...	0.0	(262144,[2,26,30,...	[3.04758916182395...	[0.95467832956133...	0.0
2	14	89	Kind of knew it w...	[Yeah, this is ho...	Maine rampage sho...	2023-10-28	Maine rampage sho...	0.0	(262144,[25,33,10...	[1.21341885189813...	[0.77090331640660...	0.0
3	0	65		[]	Arizona judge rej...	2023-10-27	Arizona judge rej...	0.0	(262144,[4,56,235...	[3.26918547815612...	[0.96335642928635...	0.0
4	9	242		[Because a person...	Ron DeSantis Call...	2023-10-27	Ron DeSantis Call...	0.0	(262144,[12,18,35...	[-2.0798257656509...	[0.11107316873441...	1.0
5	11	120	This rightwing nu...	[This jackass doe...	Speaker Mike John...	2023-10-27	Speaker Mike John...	0.0	(262144,[4,6,13,2...	[4.98251299315464...	[0.99318988588452...	0.0
6	1	27		[Guess what, Geor...	George Santos Ple...	2023-10-27	George Santos Ple...	0.0	(262144,[5,32,61,...	[1.33142291969754...	[0.79107590498427...	0.0
7	0	16		[]	Democratic lieute...	2023-10-27	Democratic lieute...	0.0	(262144,[1,33,48,...	[0.12255799746792...	[0.53060120526390...	0.0
8	5	50		[In order for thi...	Democrats Introdu...	2023-10-27	Democrats Introdu...	0.0	(262144,[7,39,121...	[4.78584244067889...	[0.99172200827458...	0.0
9	40	505		[I'll believe it ...	Republicans Move ...	2023-10-27	Republicans Move ...	0.0	(262144,[20,81,20...	[3.45803849815159...	[0.96946996372713...	0.0
10	2	100		[I am currently r...	Rachel Maddow's '...	2023-10-27	Rachel Maddow's '...	0.0	(262144,[0,10,35,...	[-1.5692636016001...	[0.17232139648908...	1.0
11	14	40		[The guy is a nut...	Speaker Mike John...	2023-10-27	Speaker Mike John...	0.0	(262144,[24,39,40...	[-0.0475061845555...	[0.48812568697675...	1.0
12	19	92		[This article may...	GIFTED ARTICLE. N...	2023-10-27	GIFTED ARTICLE. N...	0.0	(262144,[3,147,98...	[0.19922967358816...	[0.54964332144882...	0.0
13	32	360		[I'm surprised so...	The face of Rober...	2023-10-27	The face of Rober...	0.0	(262144,[15,49,29...	[0.70576805735989...	[0.66946538176609...	0.0
14	0	25		[]	BREAKING: With hi...	2023-10-27	BREAKING: With hi...	0.0	(262144,[21,51,14...	[4.12383697753104...	[0.98407539280389...	0.0
15	19	144		[please put up bi...	This tells you al...	2023-10-27	This tells you al...	0.0	(262144,[8,31,78,...	[-3.5433804186175...	[0.02810281101887...	1.0
16	0	35		[]	FACT SHEET: Biden...	2023-10-27	FACT SHEET: Biden...	0.0	(262144,[66,201,3...	[4.56458714260259...	[0.98969315938197...	0.0
17	1	28		[I think he's sho...	REMINDER: DeSanti...	2023-10-27	REMINDER: DeSanti...	0.0	(262144,[8,31,124...	[0.00479900367542...	[0.50119974861629...	0.0
18	22	95		[Eliminate the el...	TIL: In the last ...	2023-10-27	TIL: In the last ...	0.0	(262144,[157,256,...	[1.10357281776624...	[0.75092894528600...	0.0
19	245	303		[I own a business...	What's your opini...	2023-10-27	What's your opini...	0.0	(262144,[176,317,...	[-1.1582062403602...	[0.23899337340530...	1.0

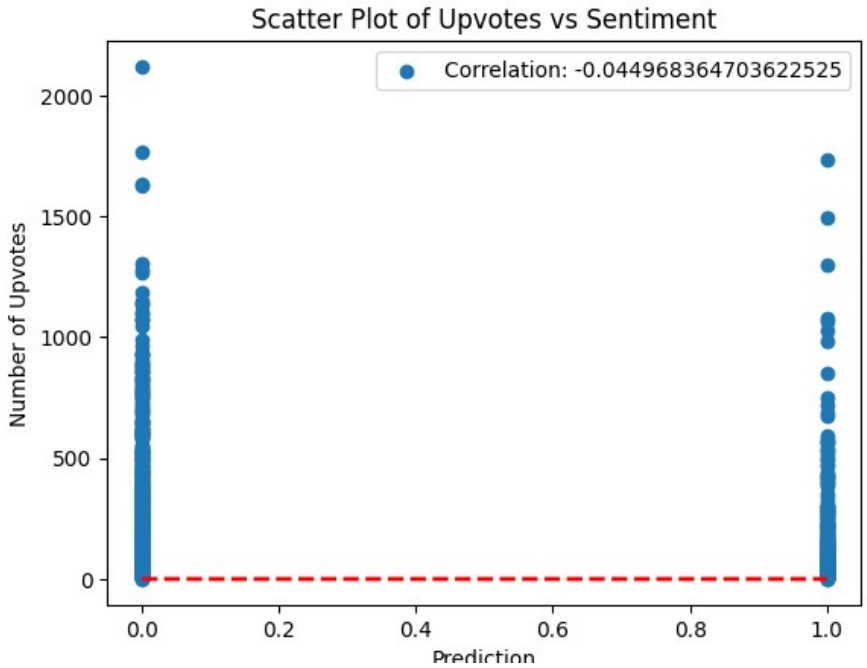
only showing top 20 rows

EDA

- **Correlation analysis between columns and prediction**
  - **numComments and Sentiment**
  - In examining the variables numComments and Sentiment, our analysis led to the determination of a negligible positive correlation between the number of comments on a Reddit submission and the computed sentiment value.



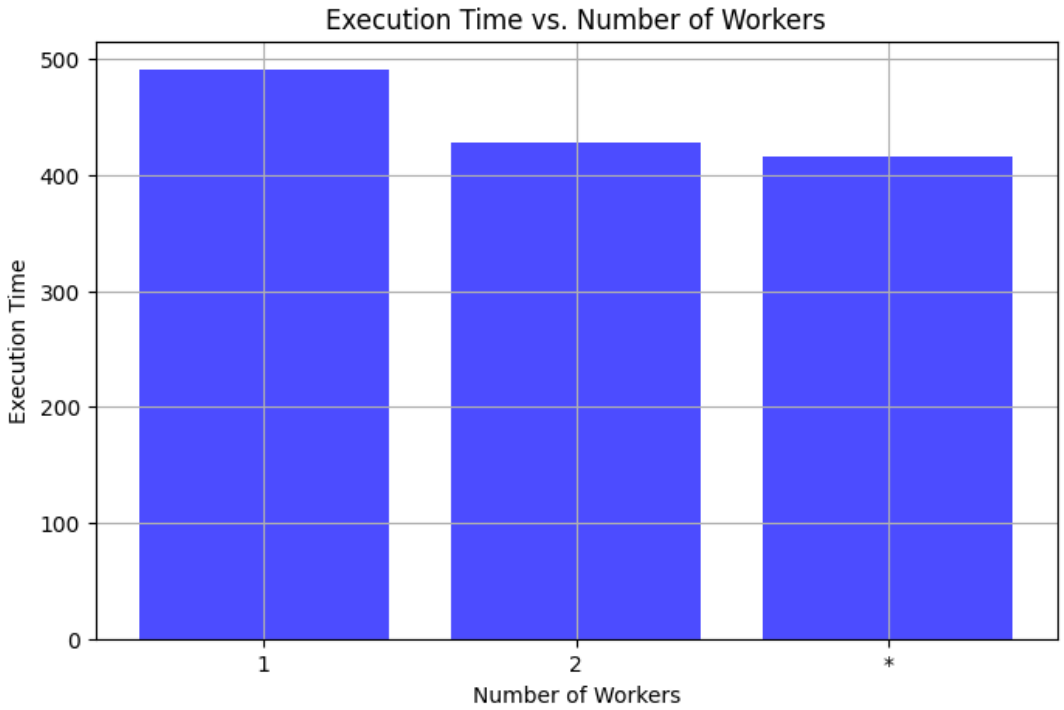
  - **numUpvotes and Sentiment**

- In assessing the variables numUpvotes and Sentiment, our findings indicated a negligible negative correlation between the number of upvotes on a Reddit submission and the computed sentiment value.
- 

- **Evaluation Graphs**
- Different Spark workers with a **fixed size of the dataset(3.8 MB)**

Number of workers	Execution time (s)
1	490.9176
2	428.1602
*	415.5687

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

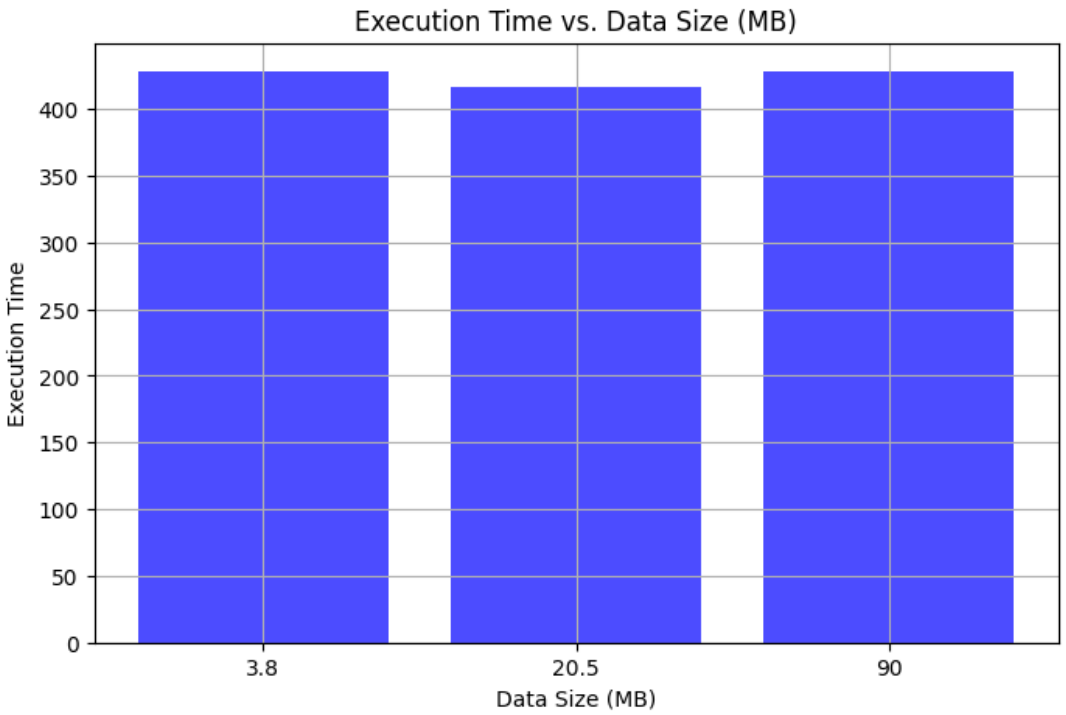


Here we can conclude that an increase in the number of workers results in decreases in execution time.

- Different Dataset sizes and **fixed number of workers(2)**

Data size(MB)	Execution time (s)
3.8	428.1602
20.5	416.9236
90	427.7713

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao



We can conclude that the execution time increases as the dataset size increases with the fixed number of spark workers. However, we don't know why the 3.8MB dataset took longer to execute than the other datasets.

Plan To Implement By Part 3

- Compare the sentiment for posts between 2016 ~ 2019, and 2020 ~ 2023
- Additionally, we will incorporate an analysis that breaks down the distribution of positive and negative sentiment for each keyword, presenting the results through graphs and a word cloud.
- **Store data in the MySQL table**
- **Build Web Interface**
    - Allow searching sentiment by keyword.
    - Allow searching sentiment by year
        - **the outcome**
            - graphs
            - post sentiment over time
            - sentiment distribution of upvote
            - word cloud

Contributions:

We all worked on it together over Discord group calls and screen sharing.

Part 3: Build Web Interface

Instructions To Run Web Interface

- Must be logged into EC2 instance:
  (HostName cs179g-fall-2023-03.cs.ucr.edu)
- Change terminal directory to inside project folder:
  (cd Reddit-Political-Sentiment-Analysis-Web-App)
- Run (streamlit run Home_🏠.py --server.address localhost --server.port 8502)
- Connect to local port 8052

Requirements

- Retrieve the data from MySQL and integrate it into the web framework.
  - To meet the requirements, store the data in a MySQL table and proceed with the planned implementation outlined in Part 2. Additionally, execute the tasks specified in Part 3.

Design

- Used Streamlit, a Python framework to build the front-end and connect with the MySQL database
- A dedicated framework to construct a back-end was not necessary since we used Streamlit

Implementation

- **Store the data in a MySQL table**
  - Accessing the data directly using Streamlit

Storing Data in MySQL

```
db_connection = mysql.connector.connect(user="group6", password="1234")
db_cursor = db_connection.cursor()
db_cursor.execute("USE cs179g;")
db_cursor.execute("CREATE TABLE IF NOT EXISTS party_sentiment_counts(positive_count INT, negative_count INT, \
                   partyLabel TINYTEXT);")
sentiment_counts = sentiment_counts.toPandas()

temp = list(sentiment_counts.itertuples(index=False, name=None))
df_string = ",".join(["(" + ",".join([str(w) for w in wt]) + ")" for wt in temp])

print(df_string)
db_cursor.execute("INSERT INTO party_sentiment_counts(partyLabel, positive_count, \
                    negative_count) VALUES " + df_string + ";")
```

  -

Updated version

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

Storing Data in MySQL

```
db_connection = mysql.connector.connect(user="group6", password="1234")
db_cursor = db_connection.cursor()
db_cursor.execute("USE cs179g;")
db_cursor.execute("CREATE TABLE IF NOT EXISTS master_table(keyword TINYTEXT, negative INT, \
                   positive INT, twenty_fifteen INT, twenty_sixteen INT, twenty_seventeen INT, twenty_eighteen INT, twenty_nineteen INT, twenty_twenty INT, twenty_twenty_one INT, twenty_twenty_two INT, twenty_twenty_three INT, \
                  zero INT, one_hundred INT, five_hundred INT, fifteen_hundred INT, total INT);")
#filtered_df = filtered_df.toPandas()

temp = list(filtered_df.itertuples(index=False, name=None))
#df_string = ",".join(["(" + ",".join([str(w) for w in wt]) + ")" for wt in temp])
df_string = ",".join(["('" + "','".join([str(w) for w in wt]) + "')" for wt in temp])


print(df_string)
db_cursor.execute("INSERT INTO master_table(keyword, negative, \
                    positive, twenty_fifteen, twenty_sixteen, twenty_seventeen, twenty_eighteen, twenty_nineteen, twenty_twenty, twenty_twenty_one, twenty_twenty_two, twenty_twenty_three, \
                  zero, one_hundred, five_hundred, fifteen_hundred, total) VALUES " + df_string + ";")
```

- **More EDA**
    - Sentiment analysis by keyword (e.g. Trump, Biden, COVID, Ukraine)
    - Sentiment analysis by party-bias (e.g. Republican, Democrat, Neutral)
    - Distribution analysis (e.g. party-bias across upvote ranges, keywords across time)
    - Post Sentiment over Time
        - 2015~2019.
        - 2020~2023.
    - World cloud
        - Republicans
        - Democrat
        - Neutral
        - Positive posts
        - Negative posts
        - 2016 - 2019
        - 2020 - 2023
    - **Our graphs and world clouds will be provided in how we addressed the feedback section**.

Evaluation(revised version)

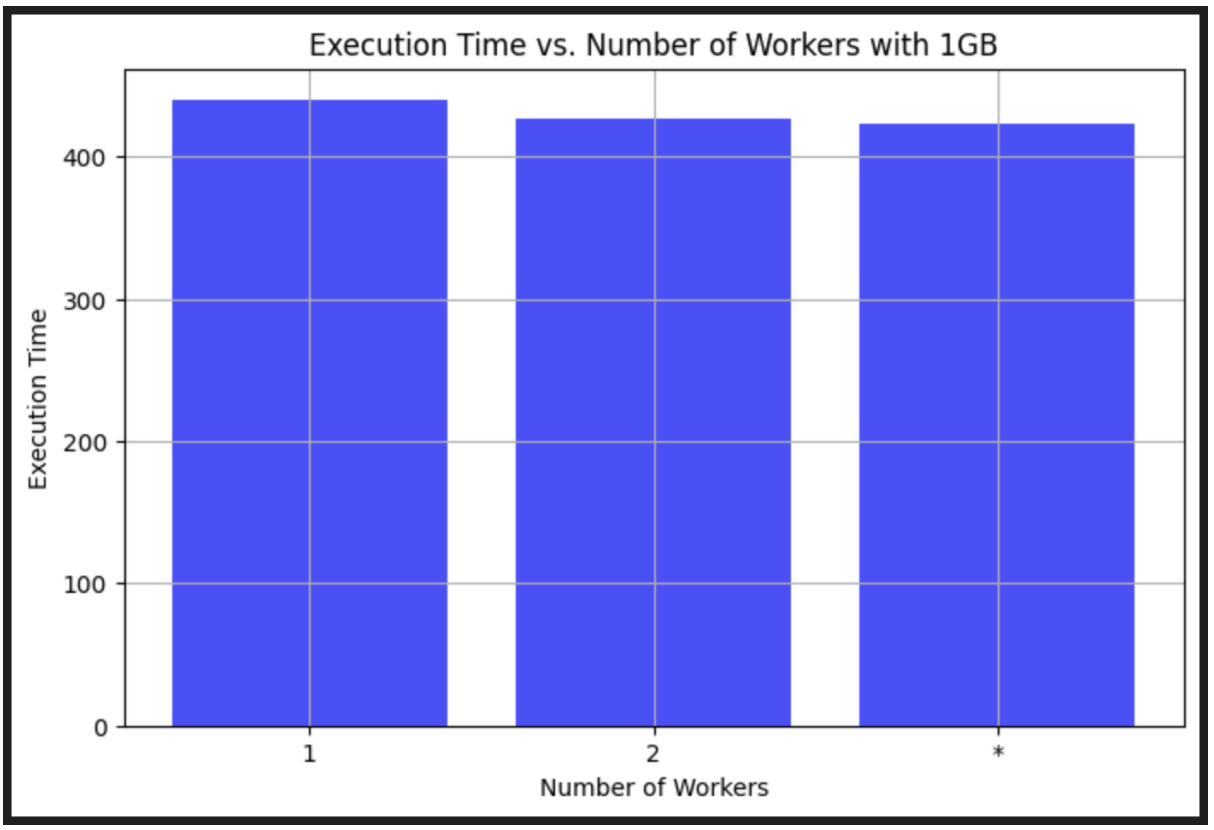
of Workers vs size for Execution Time

	1	2	*
250 mb	425.44761896133423	418.94419264793396	415.43906807899475
500 mb	434.52948212623596	424.8018355369568	419.3991768360138
750 mb	436.7343327999115	426.10309624671936	421.09590339660645
1GB	439.901873588562	427.2693681716919	423.69473218917847

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

- Different Spark workers with a **fixed size of the dataset(1GB)**

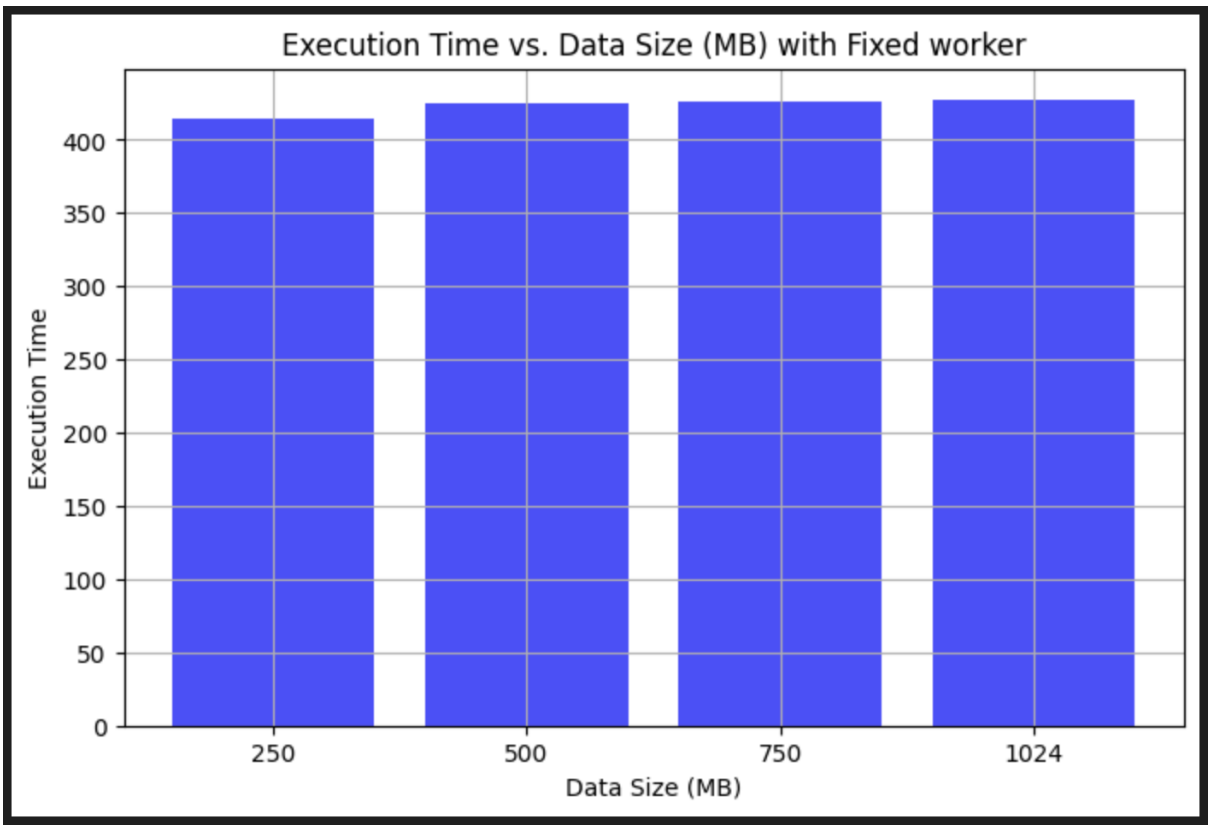
Number of workers	Execution time (s)
1	490.9176
2	428.1602
*	415.5687



- Different Dataset sizes and **fixed number of workers(2)**

Data size(MB)	Execution time (s)
250	414.9442
500	424.8018
750	426.1031
1024	427.2694

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao




```
%pip install nltk
%pip install spark-nlp
%pip install --upgrade spark-nlp
%pip install pandas
%pip install mysql-connector-python
%pip install pyspark
%pip install matplotlib
%pip install plotly
%pip install seaborn
```

```
import pandas as pd
import mysql.connector
from pyspark.sql import SparkSession
import time

spark = SparkSession.builder.config("spark.jars", "/usr/share/java/mysql-connector-j-8.0.31.jar") \
    .master("local[*]").appName("Final_Project").getOrCreate()

# Record start time
start_time = time.time()

original = spark.read.json("./reddit-data.json")
df = original

df.show()
```

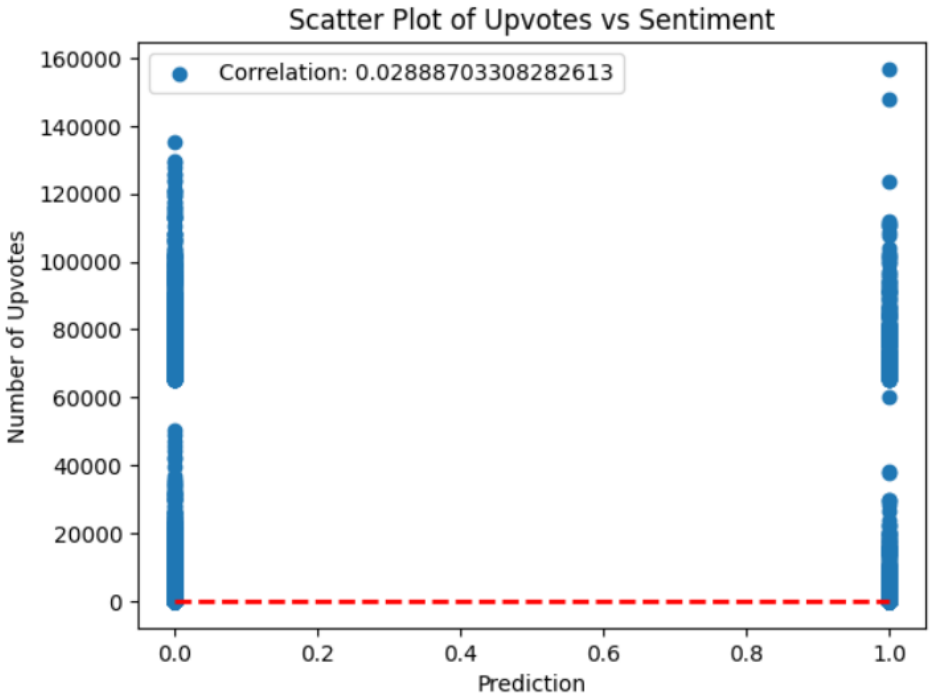
```
:  # Record end time
   end_time = time.time()

   # Calculate and print the execution time
   execution_time = end_time - start_time
   print(f"Execution Time: {execution_time} seconds")

   Execution Time: 415.43906807899475 seconds
```

The feedback we received from Part 1 and how we addressed them

- What is the total size of the dataset?
    - Total size of the dataset = 1.01 GB

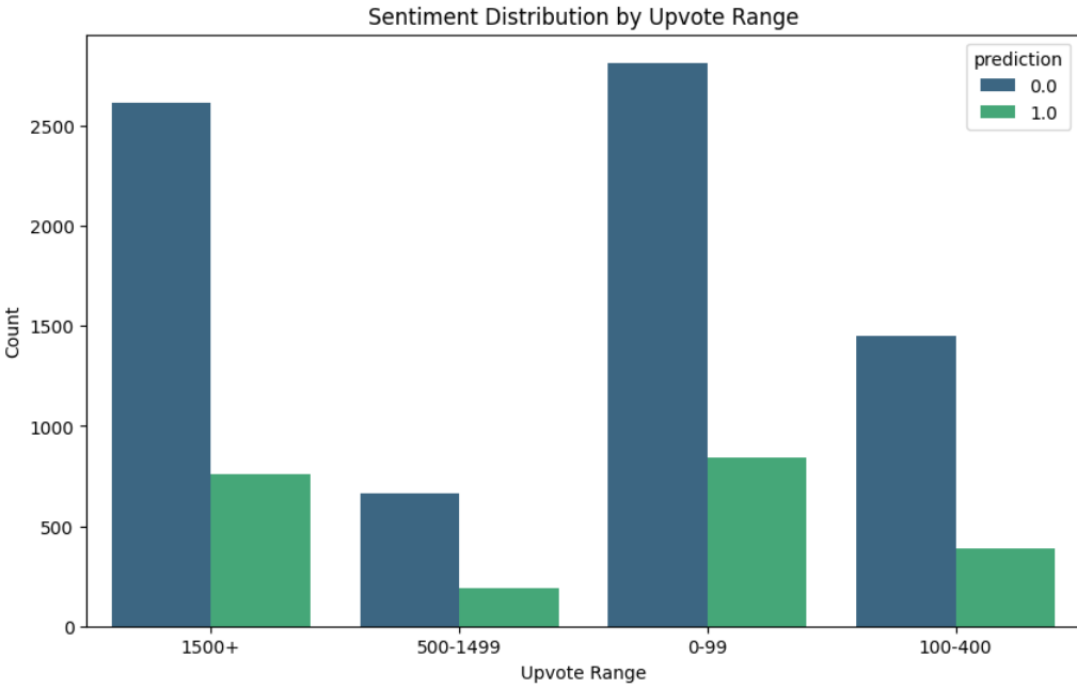
The feedback you received from Part 2 and how you addressed them

1. **Modify the correlation graph between the number of upvotes for posts and sentiments**
    a. **Before Modification**

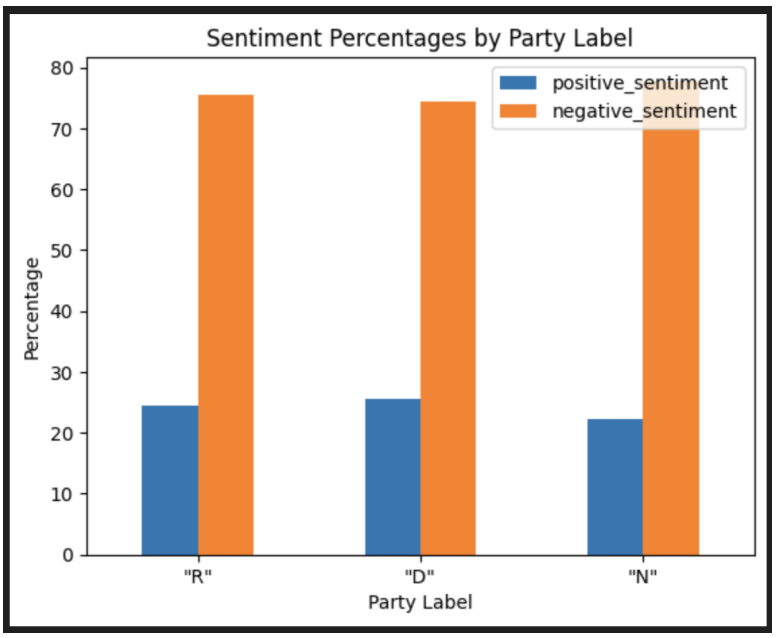


    b. **How we address this:**
        i. Using a bar graph
        ii. Replace x-axis as the upvote range
        iii. Replace y-axis as the number of posts counted in the range
        iv. Blue bar represents the negative sentiment

v. Green bar represents the positive sentiment Negative sentiment overwhelming the political trend



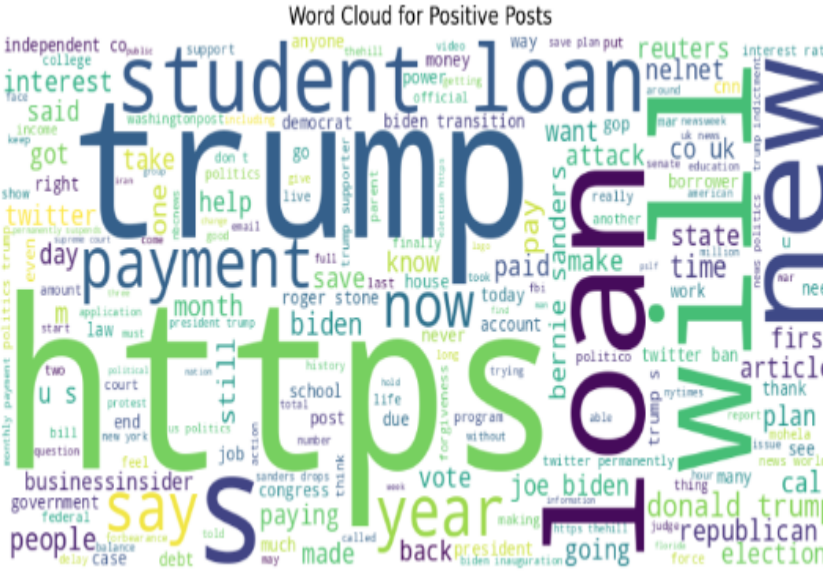
2. **Sentiment Analysis based on keywords.**
   a. **How we address this:**
      i. Identify keywords (Republicans, Democrats, Neutral)
      ii. Using a bar graph
      iii. Using Word Cloud
      iv.



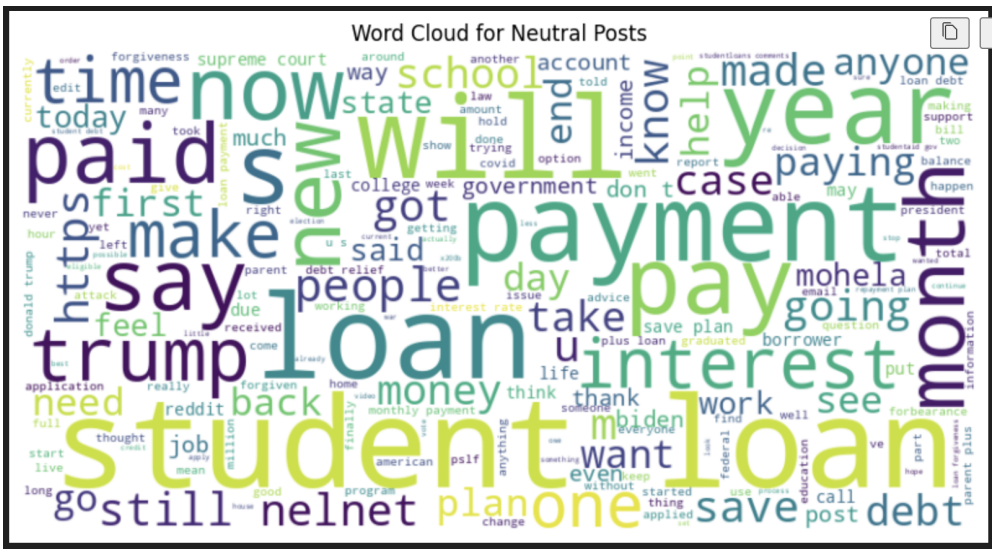
      v. Negative sentiment overwhelming the political trend

3. **Information about what words are most common in negative and positive posts**

a. **How we address this:**
   i. Divide the sentiment into three different groups
   ii. Positive, Negative, Neutral
   iii.

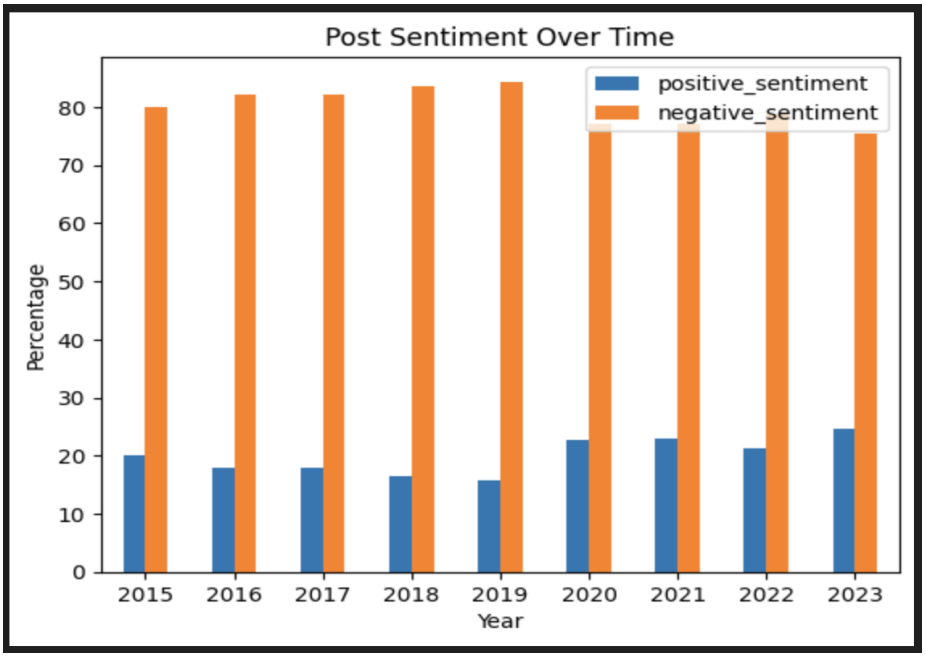


   iv.

v. 

vi. Student loans, relief funds, and payments are common

4. **Observing the trending topics related to politics during specific political periods, such as the 2016 or 2020 elections.**
    a. **How we address this:**
        i. Analysis of the sentiment based on every year
        ii. Rearrange the time for the 2016 and 2020 election
        iii. Using Word Cloud

iv. 

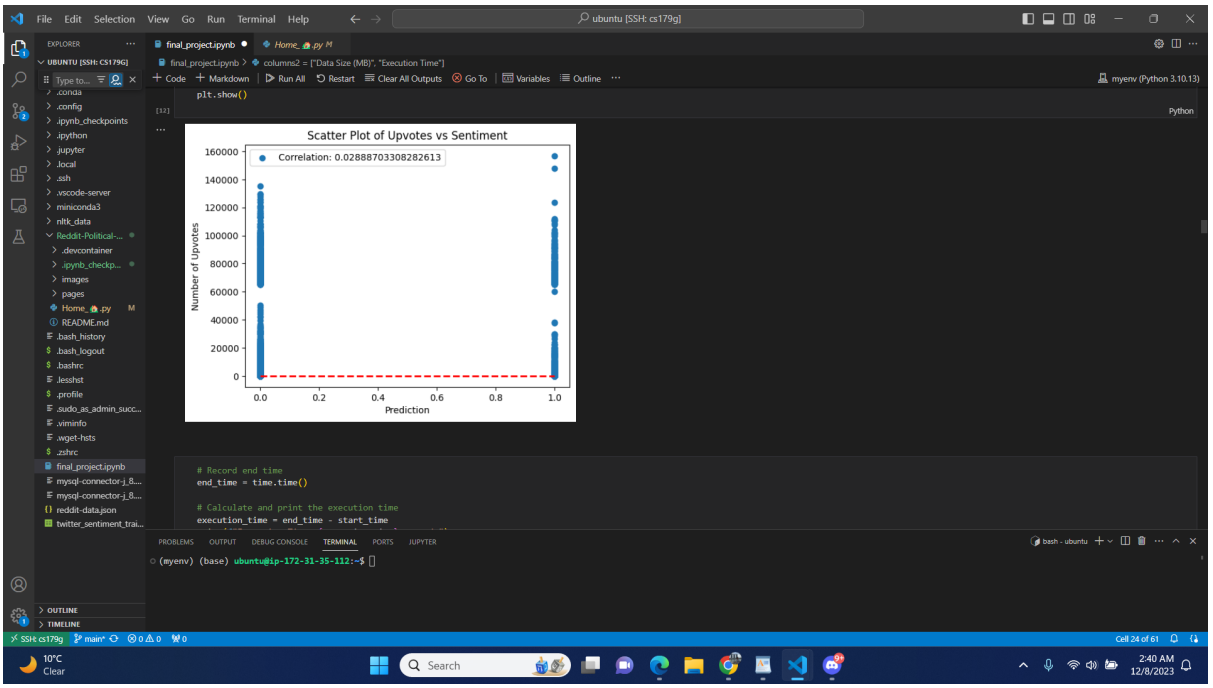
v.

vi.

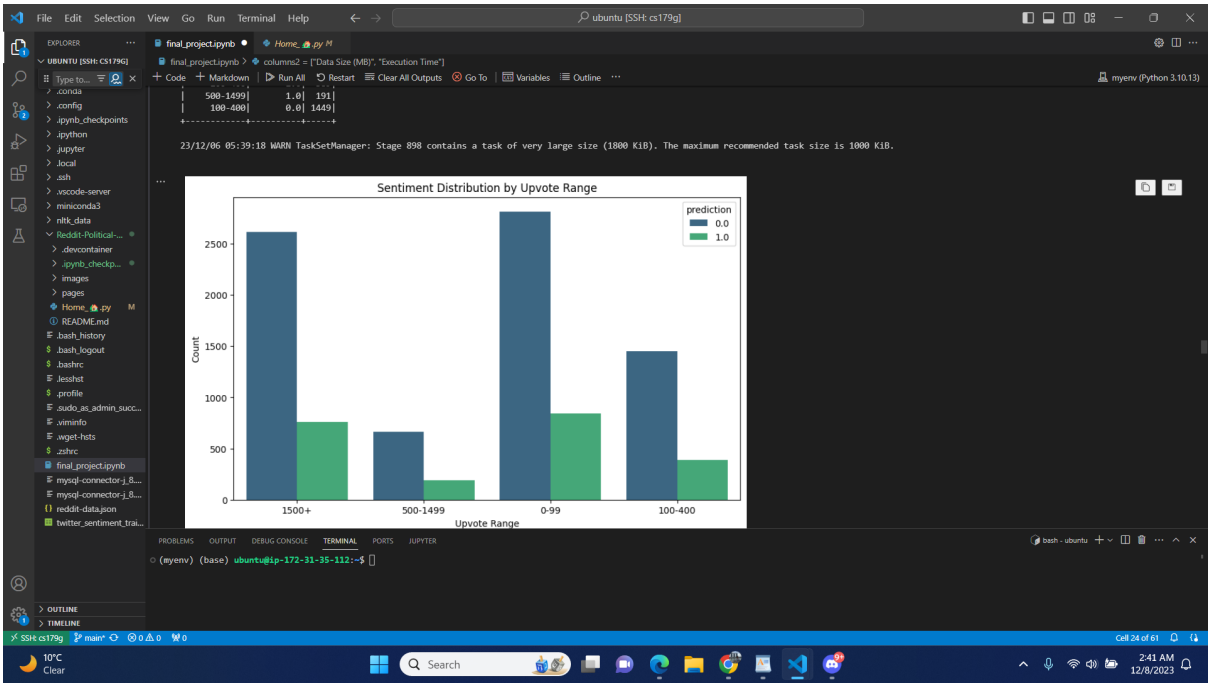
vii. no major fluctuation

viii. The negative sentiment of Reddit posts seems to remain fairly consistent regardless of time and major political events

1. Modify the correlation graph between the number of upvotes for posts and sentiments



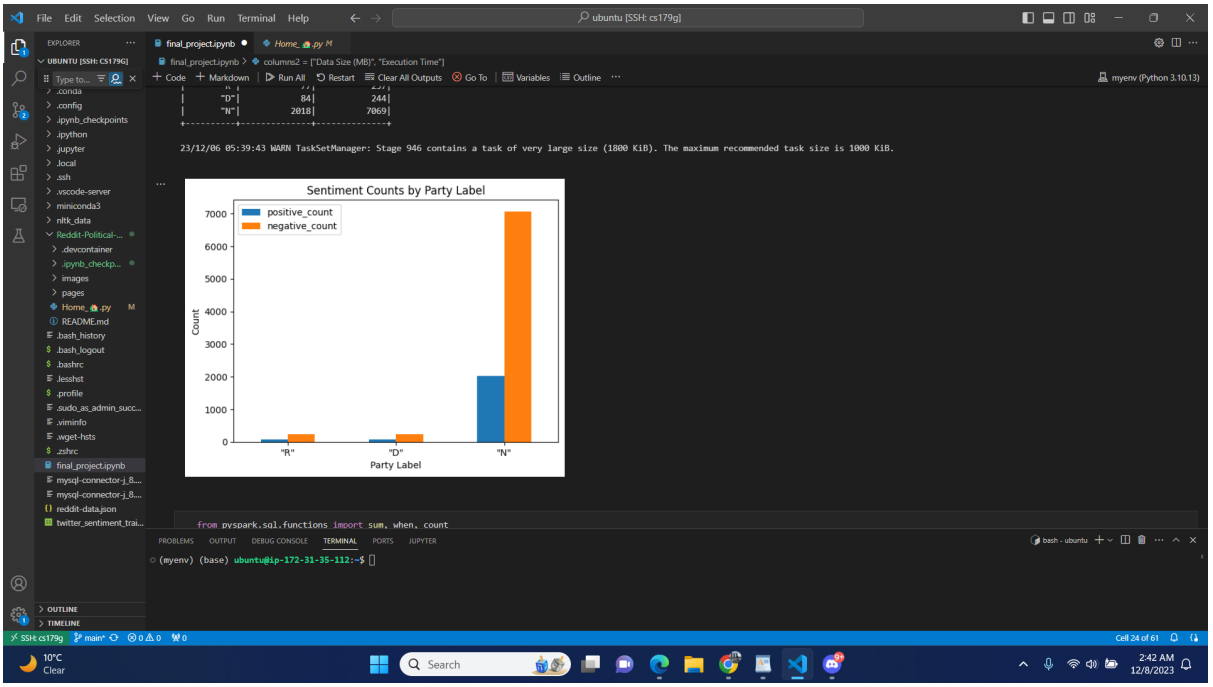
How we address this: Instead of making a scatter-points graph with barely any visible information, we switch to using a bar graph which would show the number of negative and positive posts between the upvotes range(0-99,100-400,...).



2. Categorize data into keywords such as Republican and Democrat

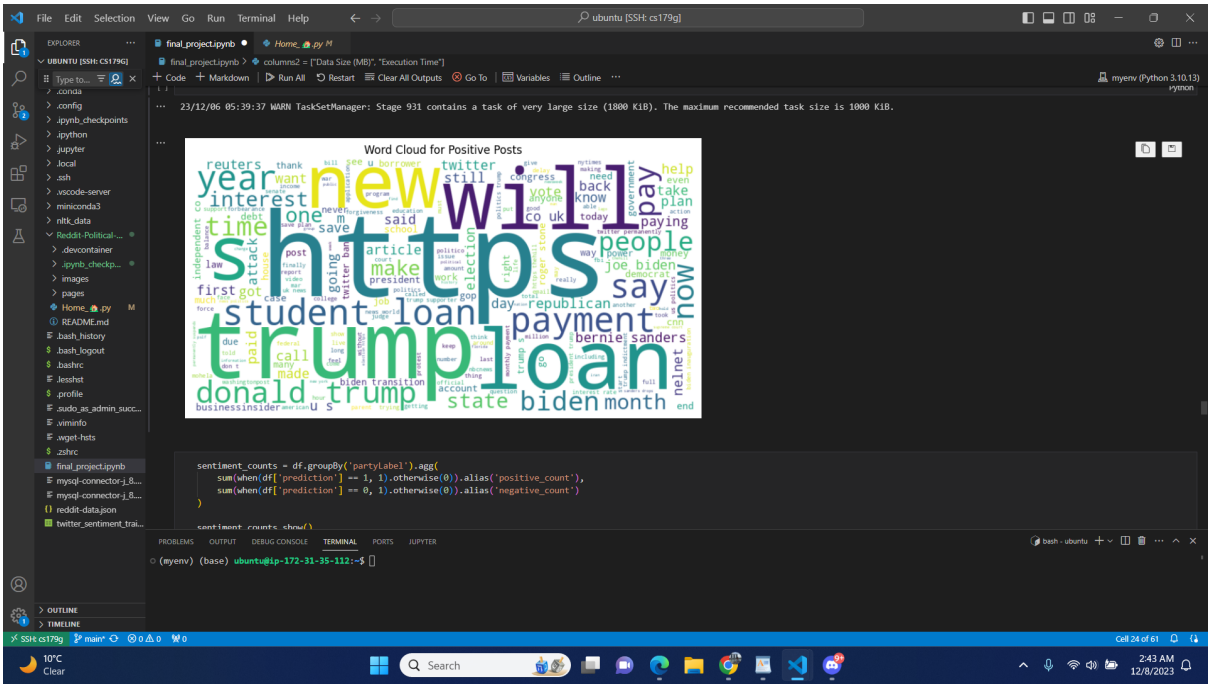
How we address this: We identify the keywords such as Republican and Democrat and find those keywords in the body of each post and classify them respectively. Moreover, we also make a bar graph that would display the number of negative and positive posts based on these categories.

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao



3. Information about what words are most common in negative and positive posts

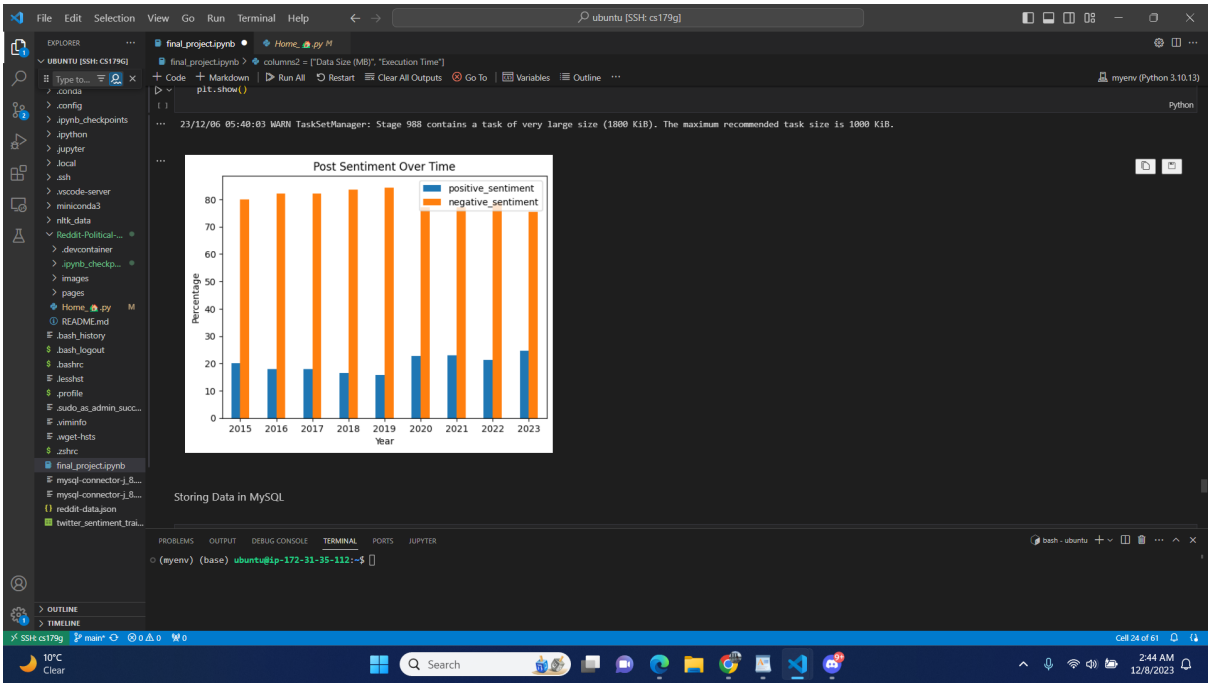
How we address this: We use WordCloud to identify which words generally appear in posts classified as negative or positive.



4. Seeing politics trend during specific political periods such as the 2016, or 2020 election.

How we address this:

We were able to take the summation of posts that existed within every year existent in our dataset and get the positive and negative sentiment counts for each. We then used bar graphs to create a clear visualization of the distribution.



How are you storing data to MySQL (how many tables, what are the columns of the tables)?

We connect to the MySQL database filled with data processed with Pyspark using a MySQL connector. We transform the cleaned and processed Pyspark dataframes into Pandas dataframes and then create the MySQL table based on this Panda Dataframe using the standard “CREATE TABLE and INSERT INTO” command. We create a table with 18 columns shown below and a table with 2 columns. These columns contain our keywords and the preprocessed calculations we need to generate our visualizations based on the user’s input.

```
db_connection = mysql.connector.connect(user="group6", password="1234")
db_cursor = db_connection.cursor()
db_cursor.execute("USE cs179g;")
db_cursor.execute("CREATE TABLE IF NOT EXISTS master_table(keyword TINYTEXT, negative INT, \
                   positive INT, twenty_fifteen INT, twenty_sixteen INT, twenty_seventeen INT, twenty_eighteen INT, twenty_nineteen INT, twenty_twenty INT, twenty_twenty_one INT, twenty_twenty_two IN
                  zero INT, one_hundred INT, five_hundred INT, fifteen_hundred INT, total INT);")
filtered_df = filtered_df.toPandas()

temp = list(filtered_df.itertuples(index=False, name=None))
#df_string = ",".join(["(" + ",".join([str(w) for w in wt]) + ")" for wt in temp])
df_string = ",".join(["('" + "', '".join([str(w) for w in wt]) + "')" for wt in temp])


print(df_string)
db_cursor.execute("INSERT INTO master_table(keyword, negative, \
                   positive, twenty_fifteen, twenty_sixteen, twenty_seventeen, twenty_eighteen, twenty_nineteen, twenty_twenty, twenty_twenty_one, twenty_twenty_two, twenty_twenty_three, \
                  zero, one_hundred, five_hundred, fifteen_hundred, total) VALUES " + df_string + ";")
```

Front-end technology details, what framework, library etc.

For the front-end of our web application, we used the Streamlit framework. This is an intuitive Python framework that has built-in UI components, MySQL and Pandas compatibility, and eliminates the need for developers to interact with HTML and CSS files.

These are the libraries that we use for our project:

- **matplotlib.pyplot**
- **Pandas**
- **seaborn**
- **plotly**
- **datetime**
- **WordCloud**
- **Streamlit**

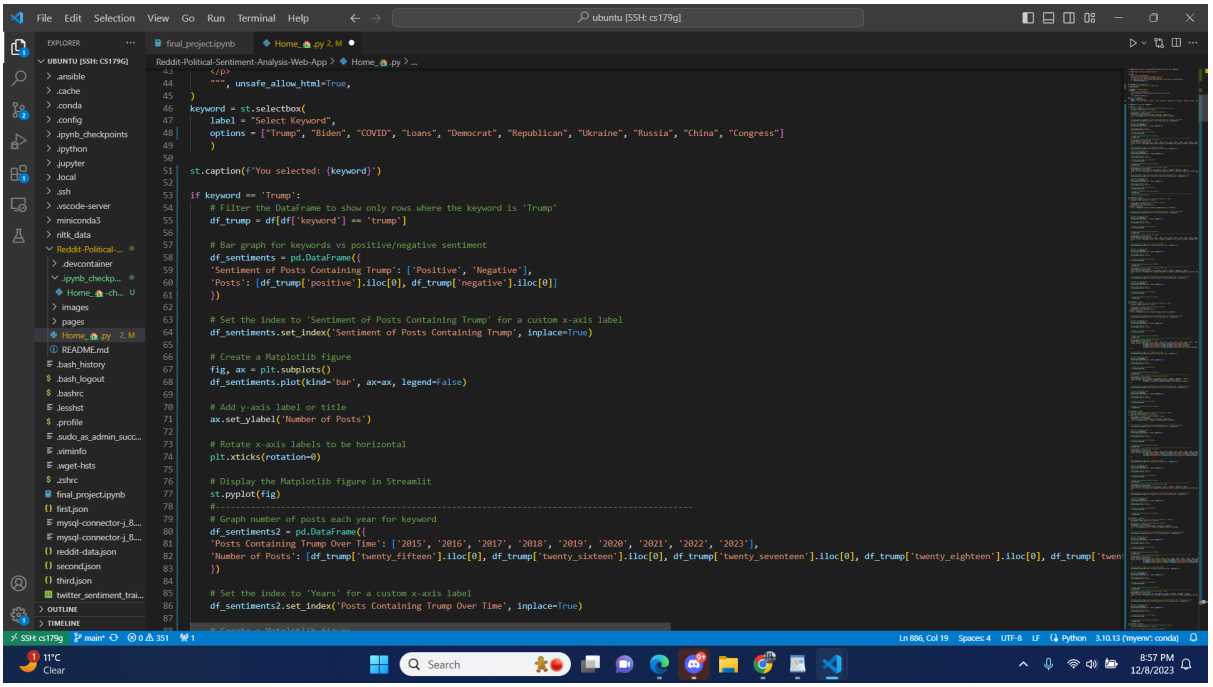
Most of these libraries are essential for our data visualization and web interface.

What happens when users submit a query to your website (through buttons or search bar)? How is that query processed and a result is returned?

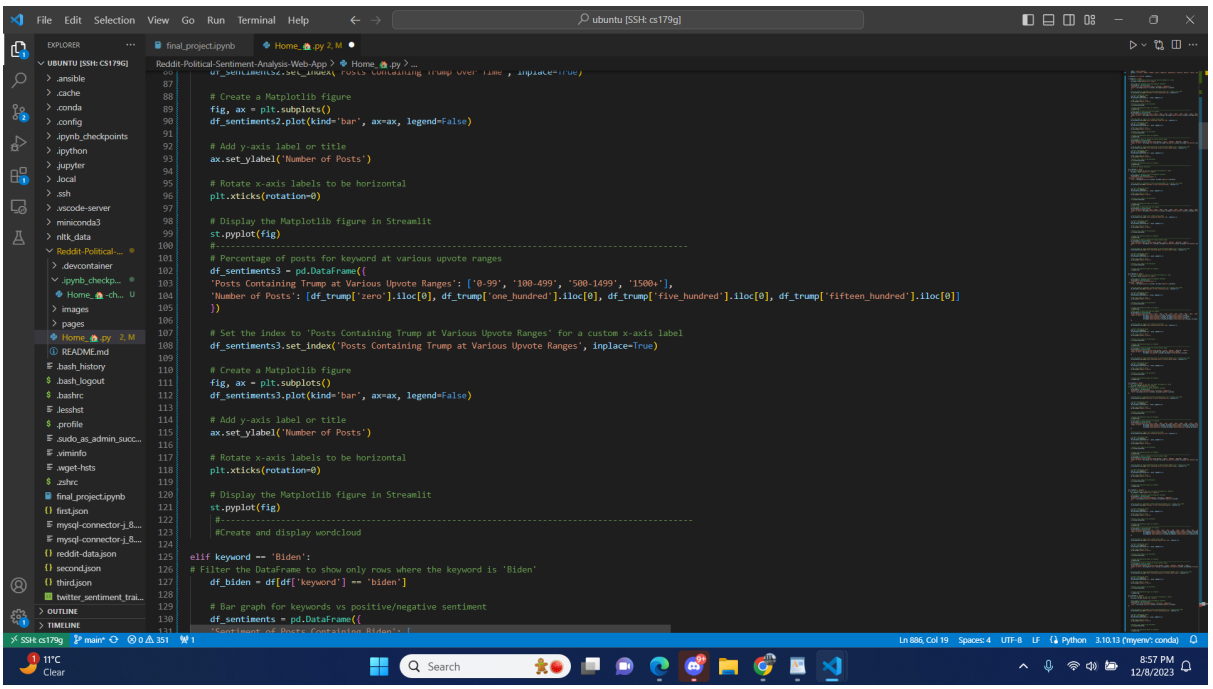
There is a list of keywords that can be selected from a dropdown menu on our web application. Once a user selects a keyword, all visualizations related to the specific keyword will be generated in real-time. We do this by hosting our web application project on the same EC2 instance as our MySQL tables, such that we can run a SQL query in our web application code. For the keyword selected, we will pull individual rows from our tables that correspond to that keyword.

For example, this is what happens when the keyword ‘Trump’ is chosen and the code associated with it:

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao



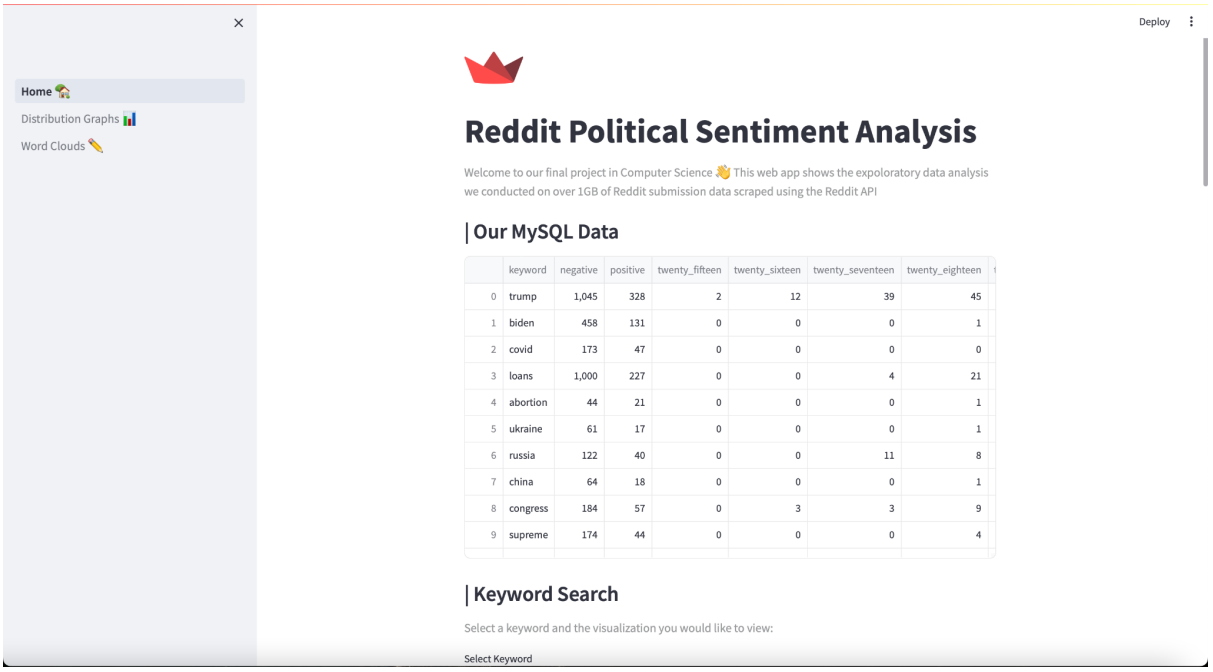
```
    """, unsafe_allow_html=True,
)
keyword = st.selectbox(
    label = "Select Keyword",
    options = ["Trump", "Biden", "COVID", "Loans", "Democrat", "Republican", "Ukraine", "Russia", "China", "Congress"]
    )

st.caption(f'You selected: {keyword}')

if keyword == 'Trump':
    # Filter the DataFrame to show only rows where the keyword is 'Trump'
    df_trump = df[df['keyword'] == 'trump']

    # Bar graph for keywords vs positive/negative sentiment
    df_sentiments = pd.DataFrame({
    'Sentiment of Posts Containing Trump': ['Positive', 'Negative'],
    'Posts': [df_trump['positive'].iloc[0], df_trump['negative'].iloc[0]]
    })

    # Set the index to 'Sentiment of Posts Containing Trump' for a custom x-axis label
    df_sentiments.set_index('Sentiment of Posts Containing Trump', inplace=True)

    # Create a Matplotlib figure
    fig, ax = plt.subplots()
    df_sentiments.plot(kind='bar', ax=ax, legend=False)

    # Add y-axis label or title
    ax.set_ylabel('Number of Posts')

    # Rotate x-axis labels to be horizontal
    plt.xticks(rotation=0)

    # Display the Matplotlib figure in Streamlit
    st.pyplot(fig)
    #-------------------------------------------------------------------------------------------
    # Graph number of posts each year for keyword
    df_sentiments2 = pd.DataFrame({
    'Posts Containing Trump Over Time': ['2015', '2016', '2017', '2018', '2019', '2020', '2021', '2022', '2023'],
    'Number of Posts': [df_trump['twenty_fifteen'].iloc[0], df_trump['twenty_sixteen'].iloc[0], df_trump['twenty_seventeen'].iloc[0], df_trump['twenty_eighteen'].iloc[0], df_trump['twen
    })

    # Set the index to 'Years' for a custom x-axis label
    df_sentiments2.set_index('Posts Containing Trump Over Time', inplace=True)
```



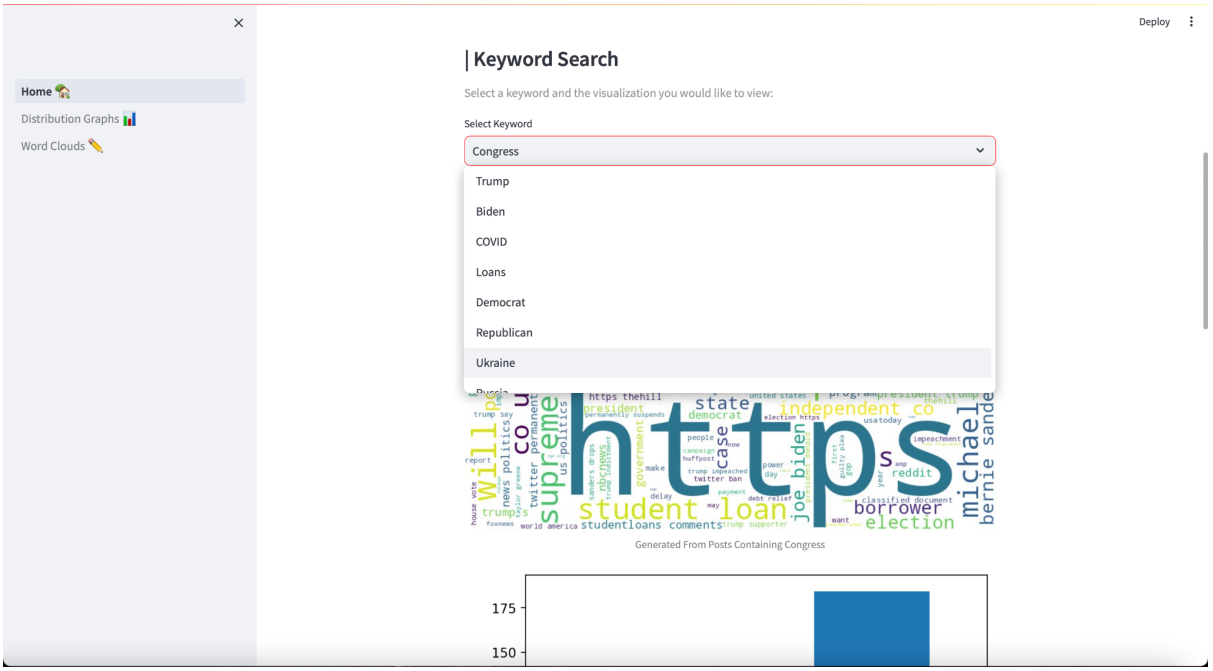
```
    # Create a Matplotlib figure
    fig, ax = plt.subplots()
    df_sentiments2.plot(kind='bar', ax=ax, legend=False)

    # Add y-axis label or title
    ax.set_ylabel('Number of Posts')

    # Rotate x-axis labels to be horizontal
    plt.xticks(rotation=0)

    # Display the Matplotlib figure in Streamlit
    st.pyplot(fig)
    #-------------------------------------------------------------------------------------------
    # Percentage of posts for keyword at various upvote ranges
    df_sentiments3 = pd.DataFrame({
    'Posts Containing Trump at Various Upvote Ranges': ['0-99', '100-499', '500-1499', '1500+'],
    'Number of Posts': [df_trump['zero'].iloc[0], df_trump['one_hundred'].iloc[0], df_trump['five_hundred'].iloc[0], df_trump['fifteen_hundred'].iloc[0]]
    })

    # Set the index to 'Posts Containing Trump at Various Upvote Ranges' for a custom x-axis label
    df_sentiments3.set_index('Posts Containing Trump at Various Upvote Ranges', inplace=True)

    # Create a Matplotlib figure
    fig, ax = plt.subplots()
    df_sentiments3.plot(kind='bar', ax=ax, legend=False)

    # Add y-axis label or title
    ax.set_ylabel('Number of Posts')

    # Rotate x-axis labels to be horizontal
    plt.xticks(rotation=0)

    # Display the Matplotlib figure in Streamlit
    st.pyplot(fig)
    #-------------------------------------------------------------------------------------------
    #Create and display wordcloud

elif keyword == 'Biden':
# Filter the DataFrame to show only rows where the keyword is 'Biden'
    df_biden = df[df['keyword'] == 'biden']

    # Bar graph for keywords vs positive/negative sentiment
    df_sentiments = pd.DataFrame({
```

Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao

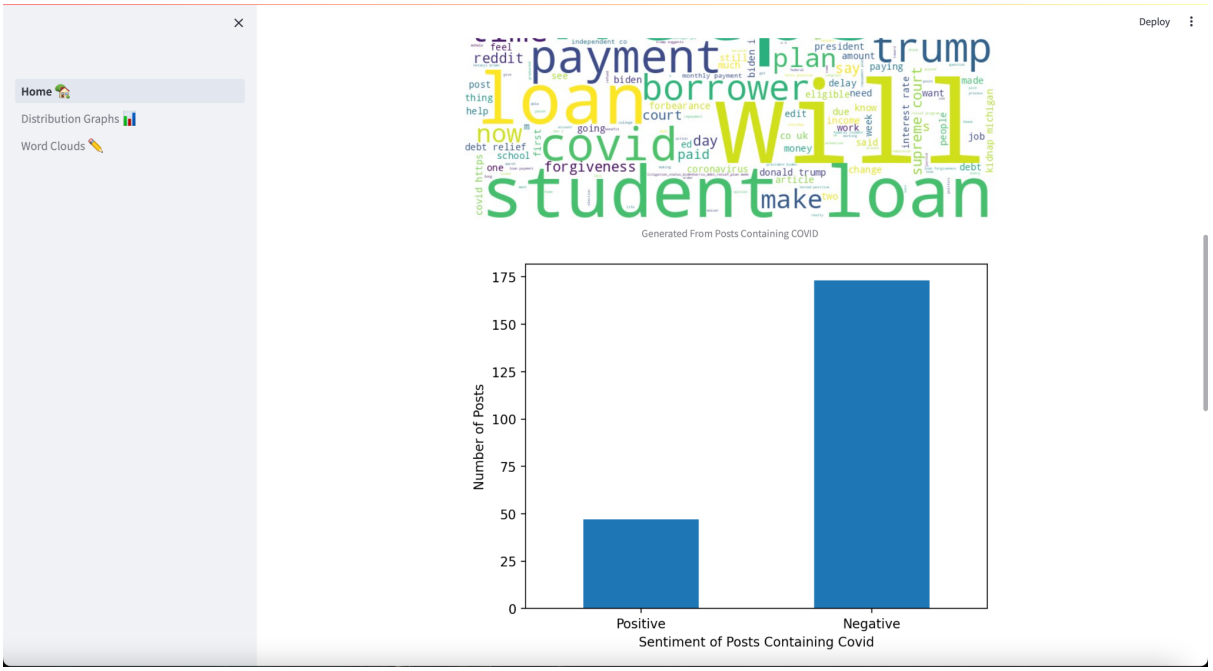
Screenshots of the running system

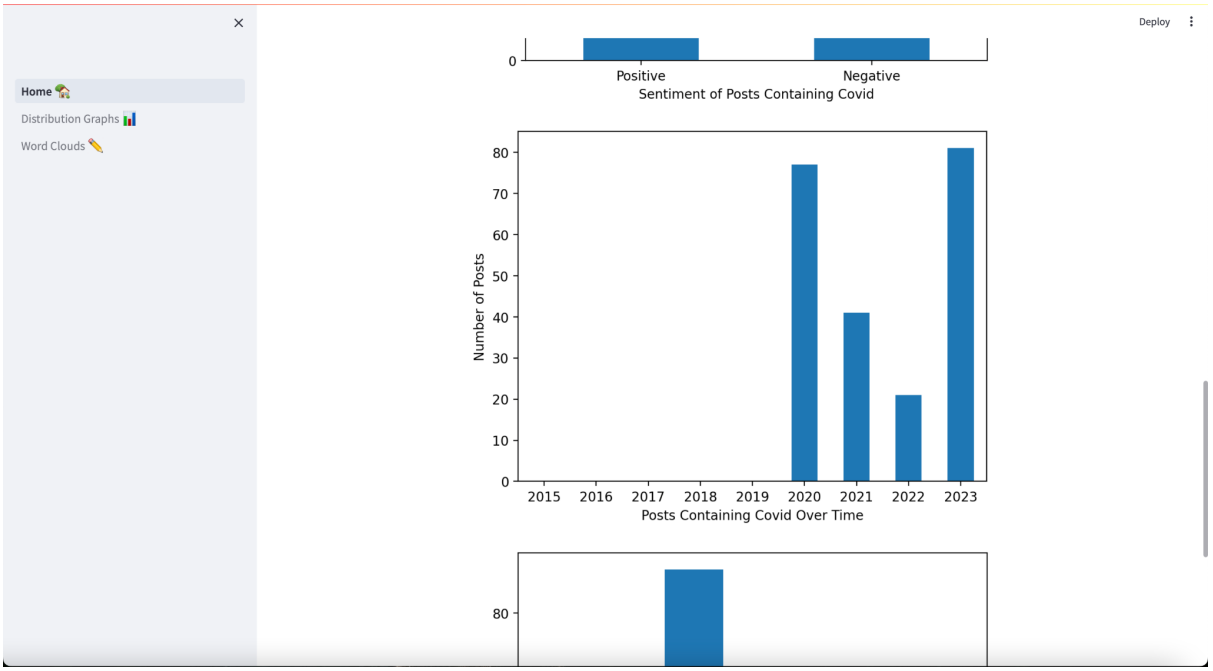


	keyword	negative	positive	twenty_fifteen	twenty_sixteen	twenty_seventeen	twenty_eighteen
0	trump	1,045	328	2	12	39	45
1	biden	458	131	0	0	0	1
2	covid	173	47	0	0	0	0
3	loans	1,000	227	0	0	4	21
4	abortion	44	21	0	0	0	1
5	ukraine	61	17	0	0	0	1
6	russia	122	40	0	0	11	8
7	china	64	18	0	0	0	1
8	congress	184	57	0	3	3	9
9	supreme	174	44	0	0	0	4

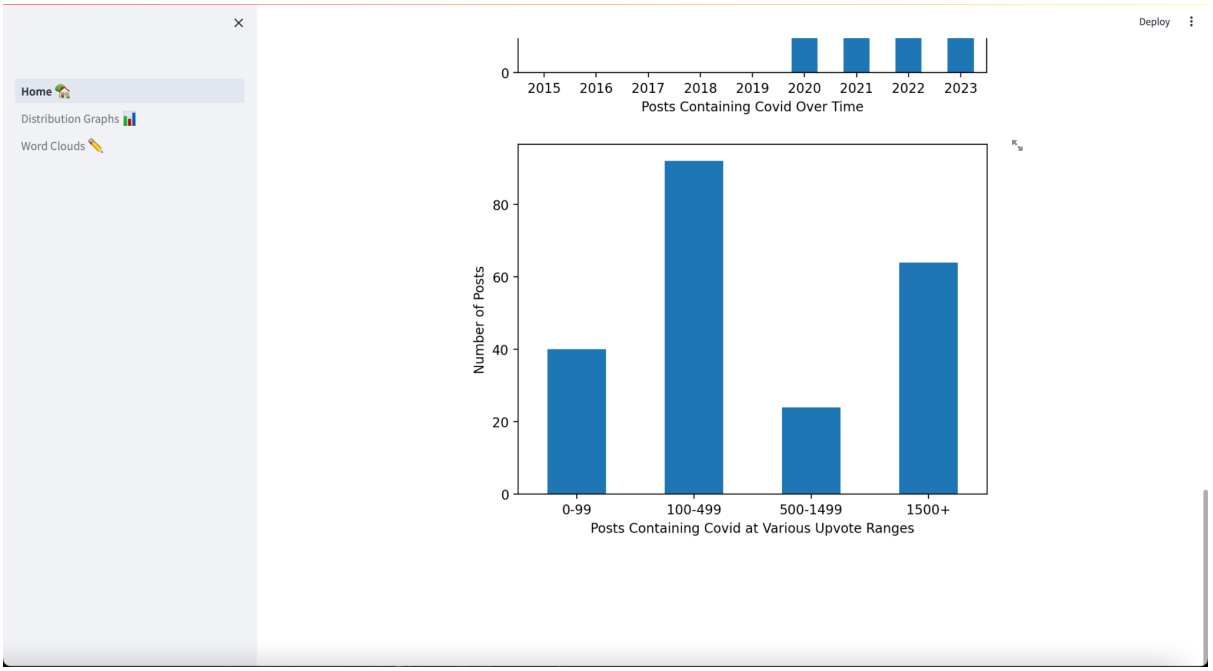


Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao





Group 6: Daniel Birouty, Jinseok Lee, Kendrew Christanto, Toan Bao



Contributions: We all worked together on discord over screen share.