COVID-19 TWITTER ANALYSIS

COURSE: BIG DATA PROGRAMMING

TEAM 4

FINAL REPORT

Vidyullatha Lakshmi Kaza- 8

Aparna Manda- 11

Lohitha Yenugu- 19

Introduction:

Data analysis on tweets pertaining to COVID19. The entire world is shutdown due to the virus and we wanted to know the public opinion on this situation. So, we collected their opinion through tweets. Collected real-time tweets talking about the corona virus with keywords- COVID19, Corona and performed analysis using big data technologies- Map Reduce, Hive, Cassandra and Map Reduce sentimental analysis.

Background:

Analyzed twitter data- json structure to extract useful attributes

<https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/intro-to-tweet-json>

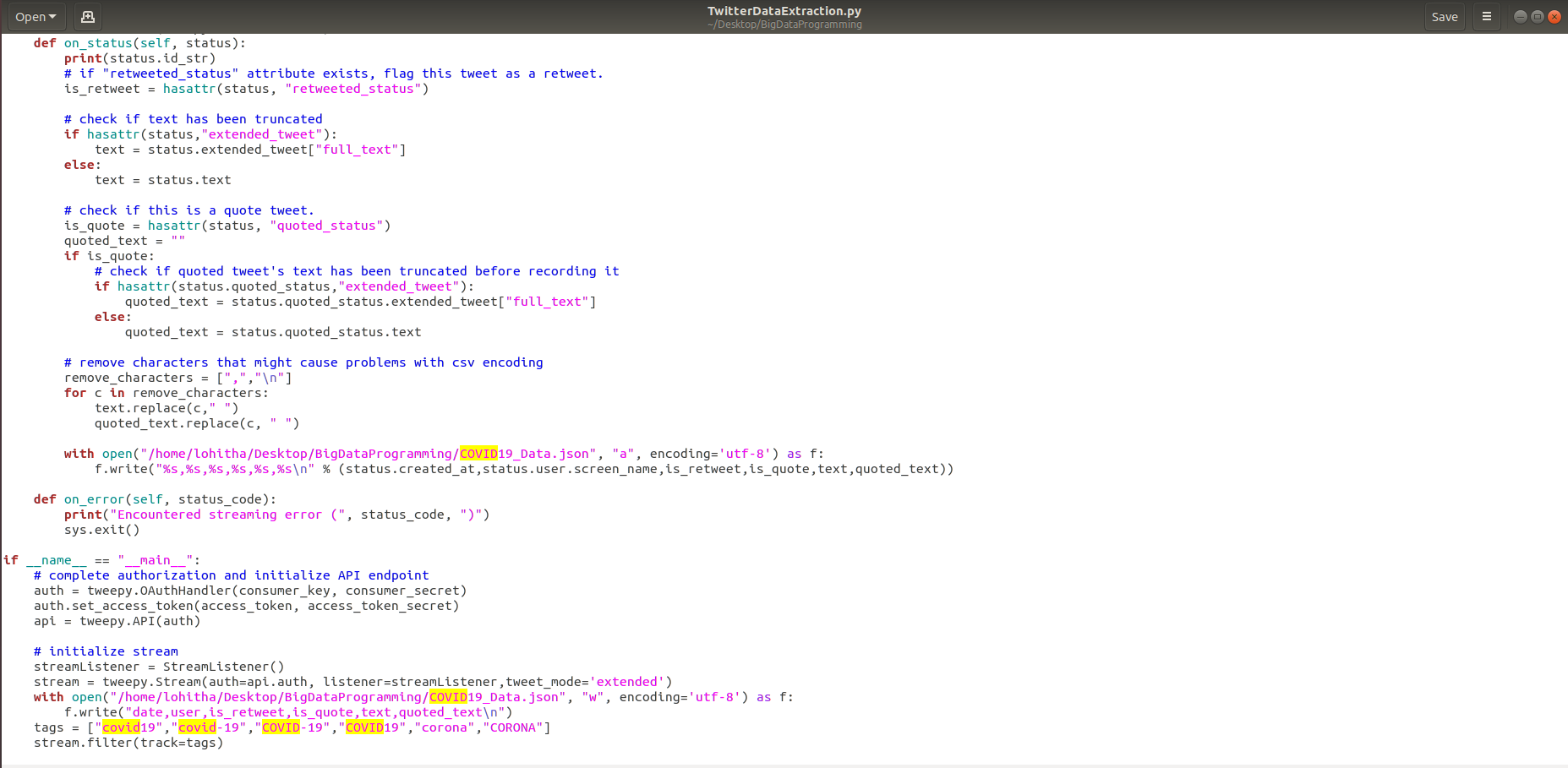
Twitter data preprocessing to remove characters like spaces, new lines and commas that might cause issues during csv encoding.

Features developed:

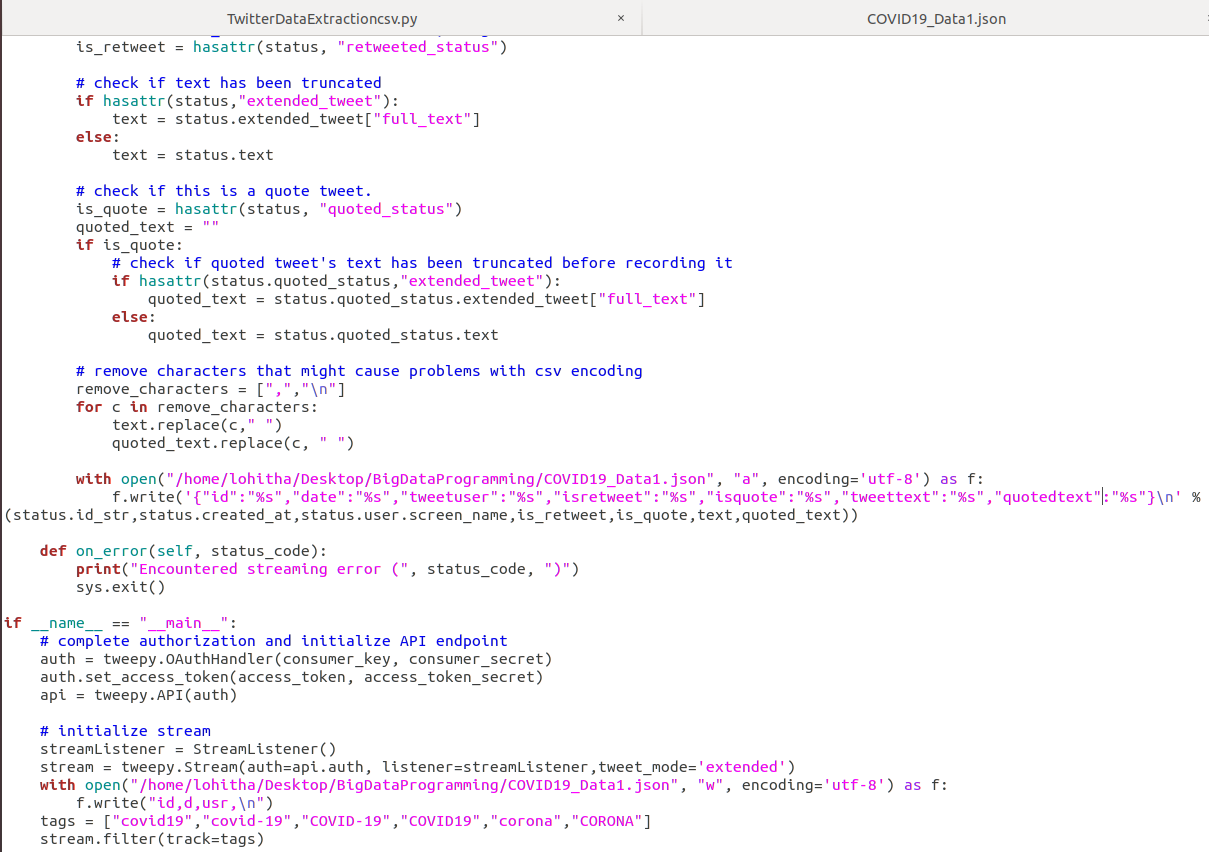
1. Twitter data collection on COVID-19
2. Map Reduce to count tweets by each user
3. Data Analysis of tweets using Hive
4. Sentiment Analysis of tweets using Map Reduce
5. Data Analysis of tweets using Cassandra
6. Twitter Data Analysis using Spark SQL

Data Collection:

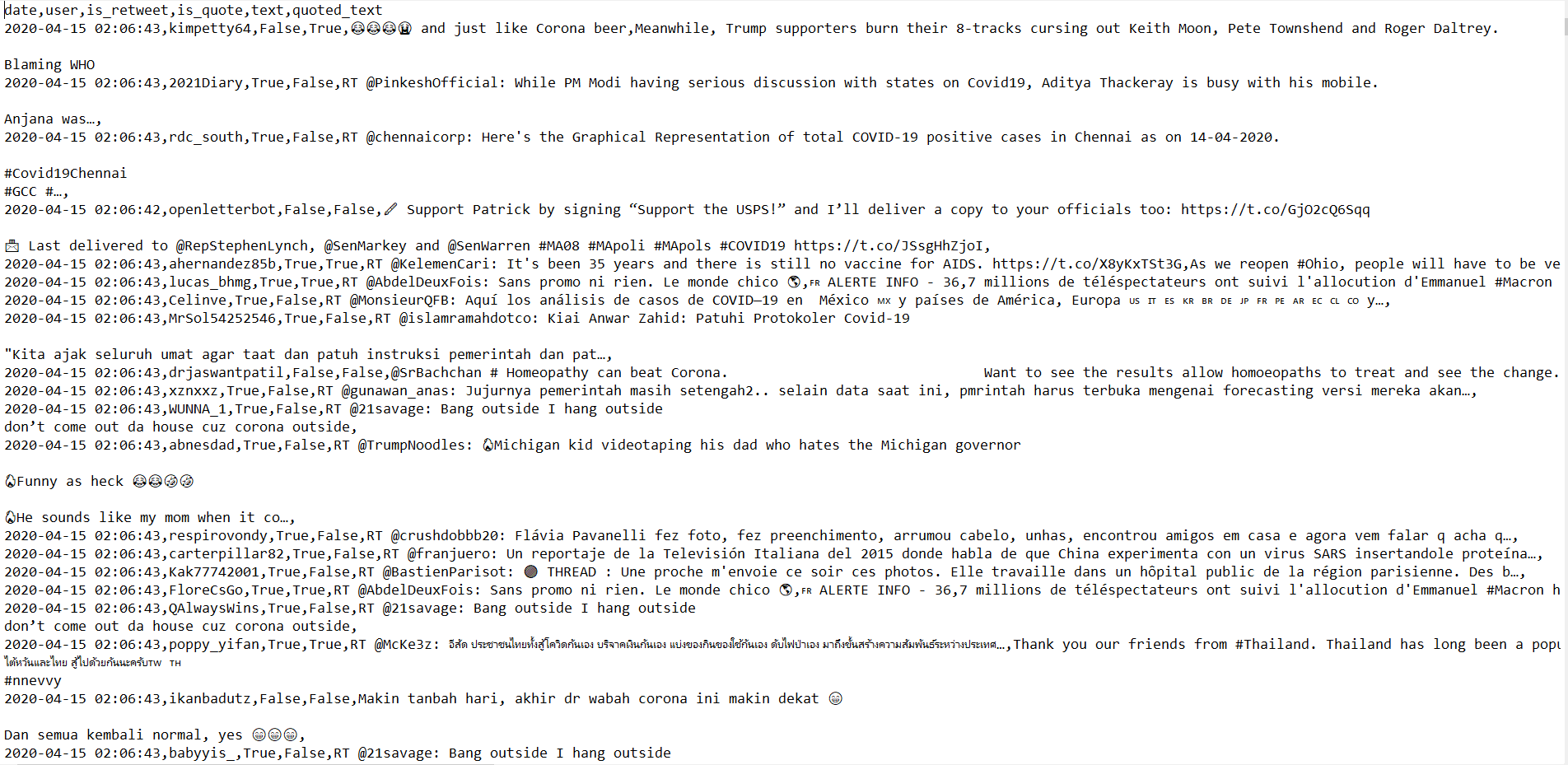
Python Code to extract live stream twitter data as CSV:



Python Code to extract live stream twitter data as JSON:



Collected Tweets:



Dataset:

Collected real-time tweets using twitter streaming api- tweepy. Extracted features - tweet\_date, tweet\_user, tweet\_text, is\_retweet, is\_quote, quoted\_text.

tweet\_date – timestamp

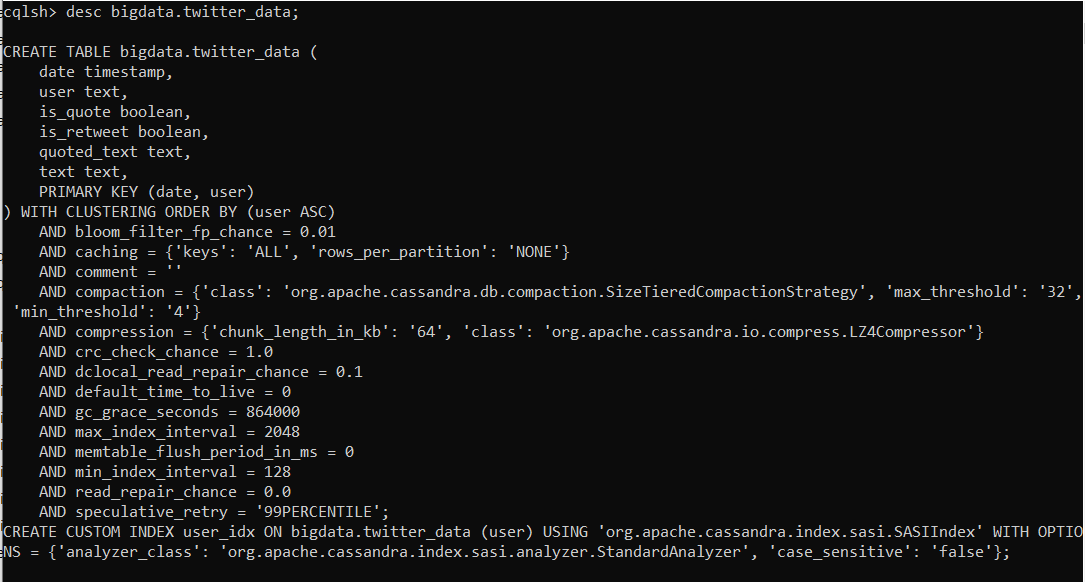
tweet\_user – text

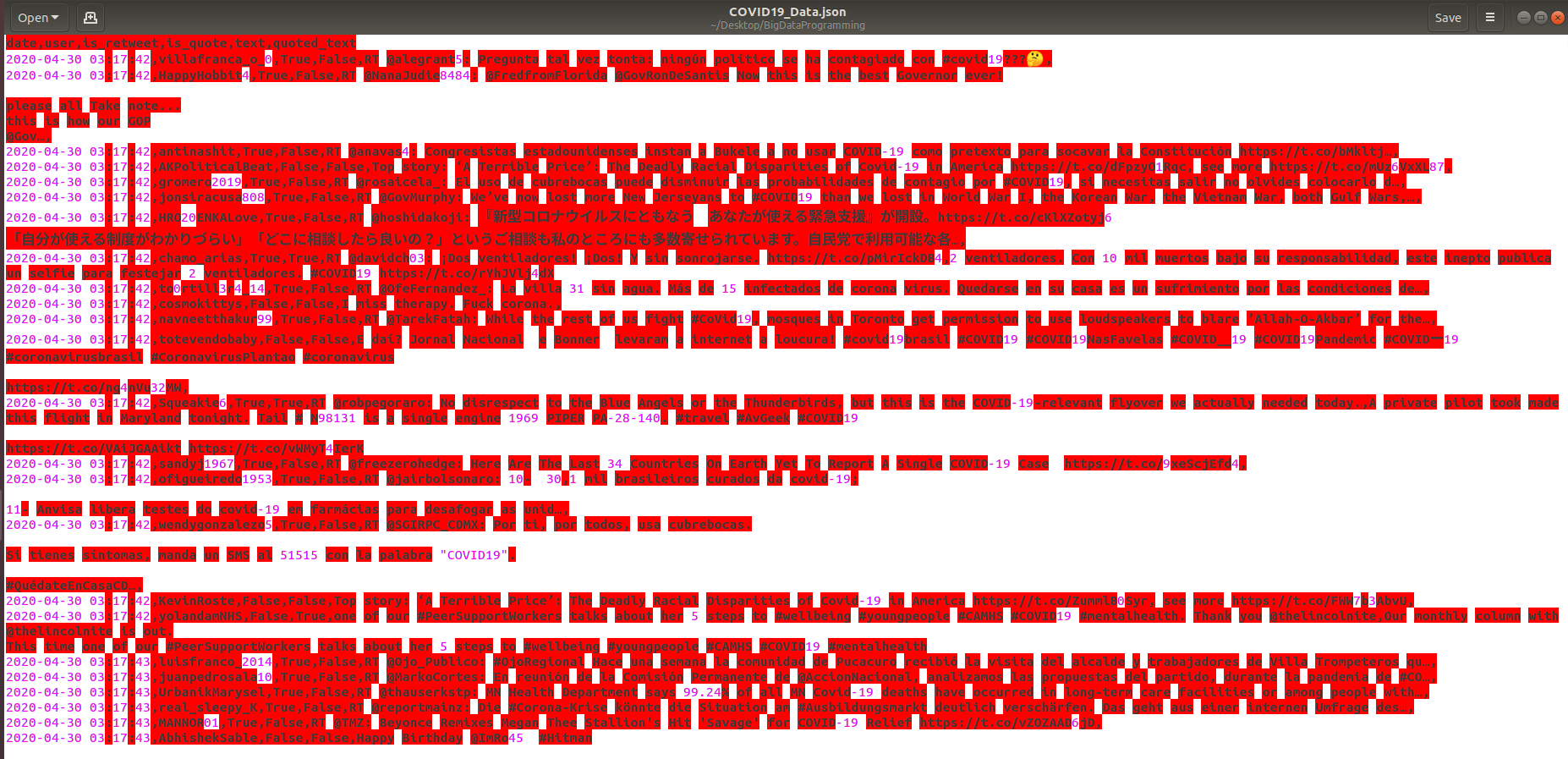
tweet\_text – text

is\_retweet – boolean

is\_quote – boolean

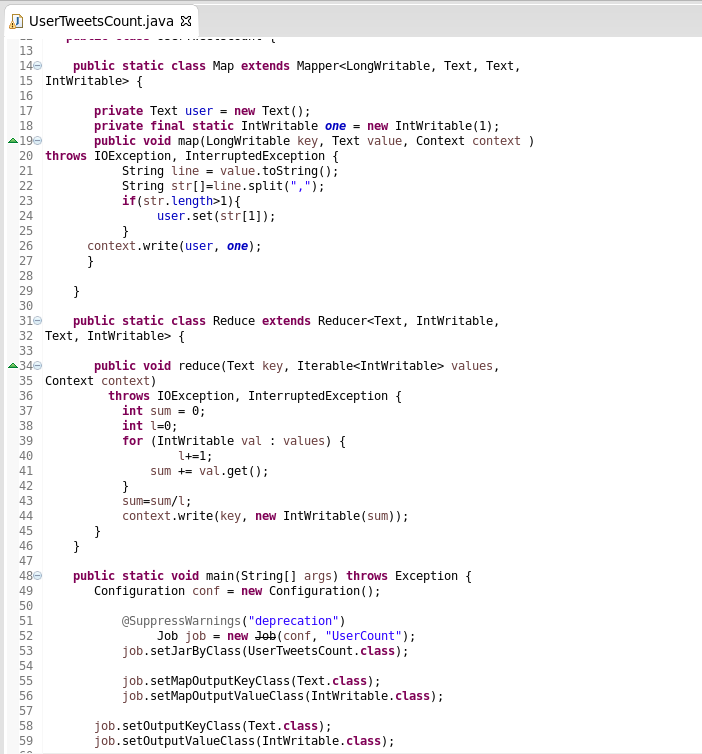
quoted\_text – text



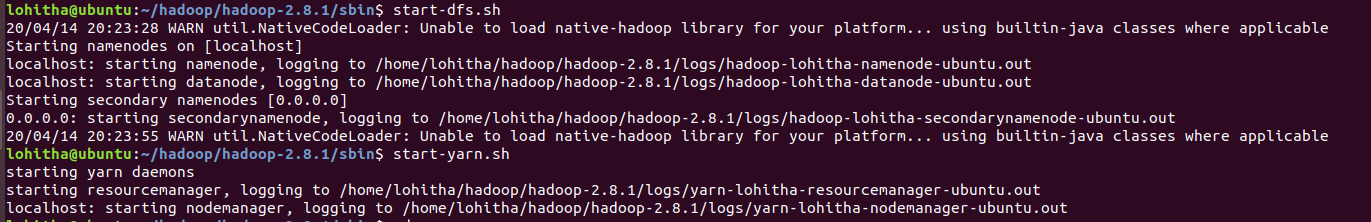


Analysis of data/Implementation/Results:

**Use Case 1:** Map Reduce to count tweets by each user



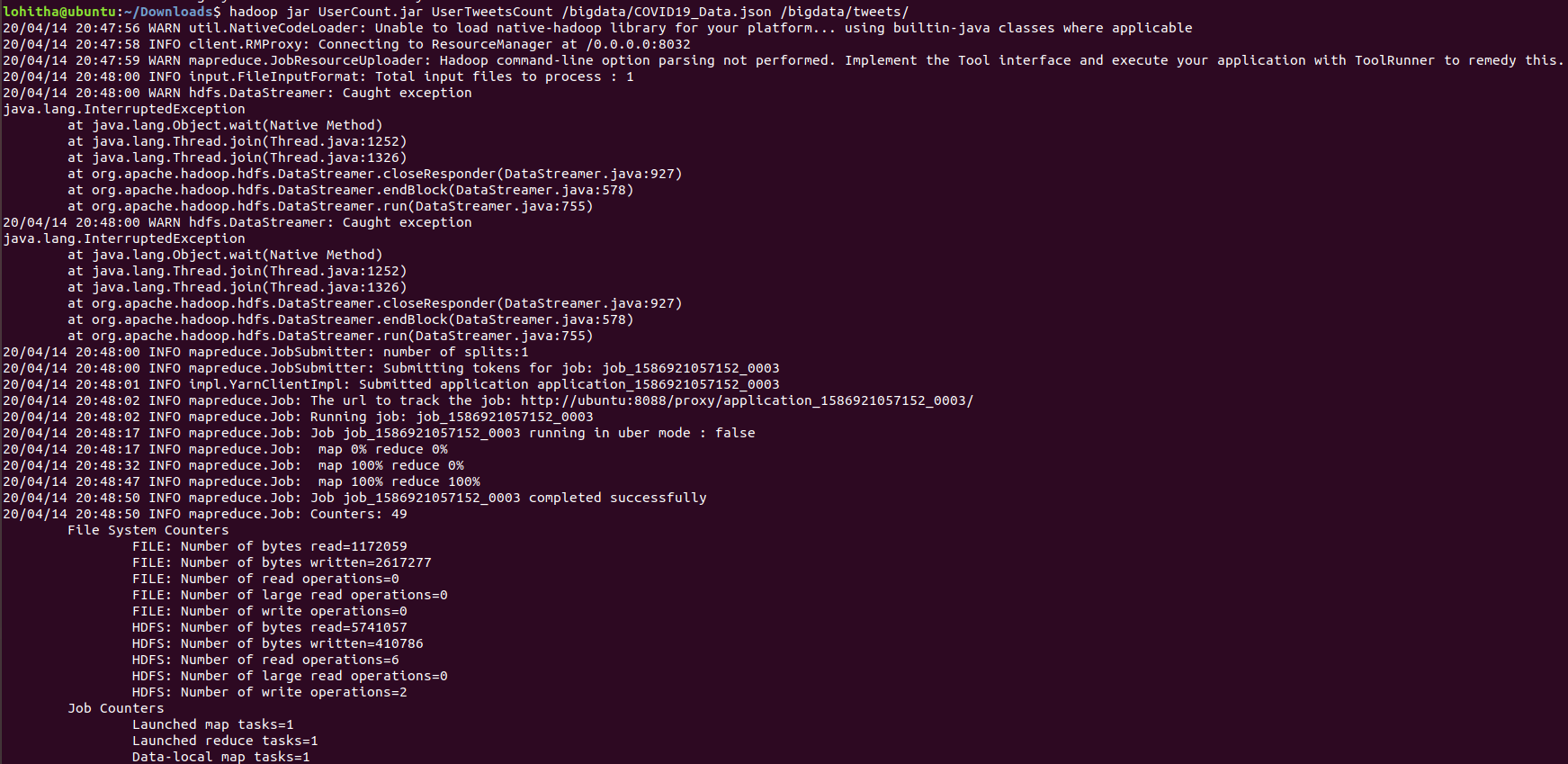
Starting HDFS namenode and datanodes:

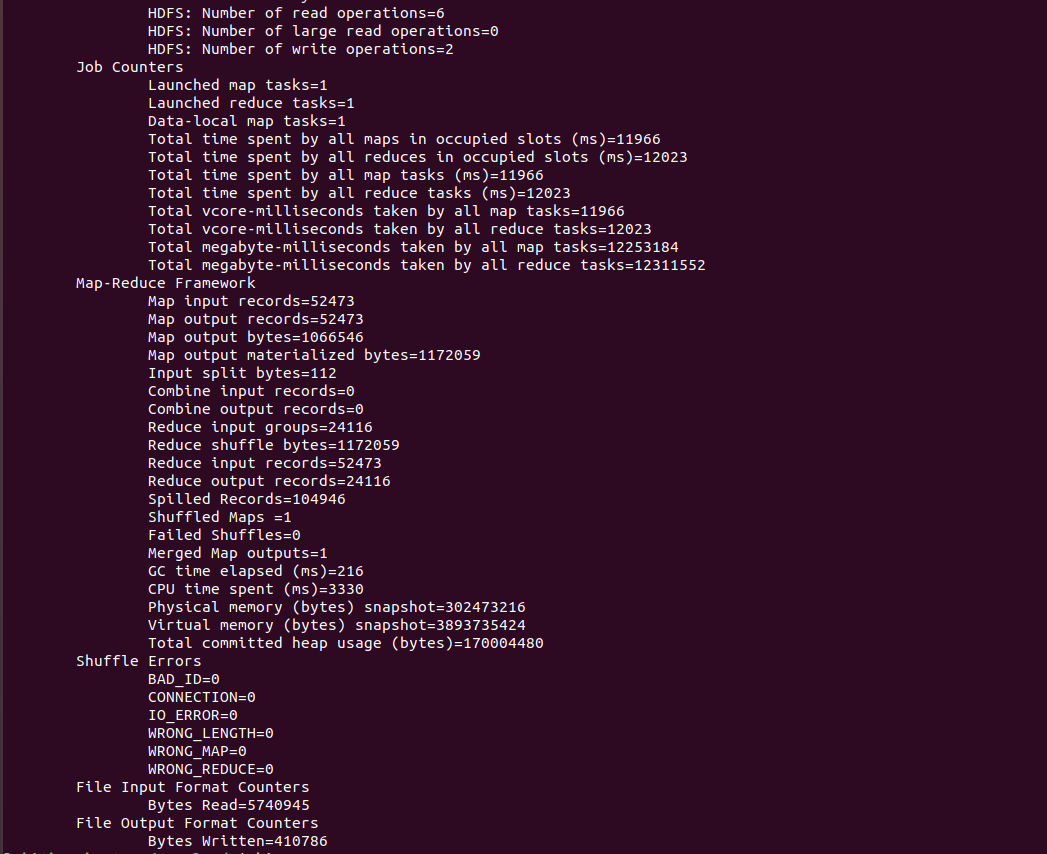


Loading input data from local to HDFS:

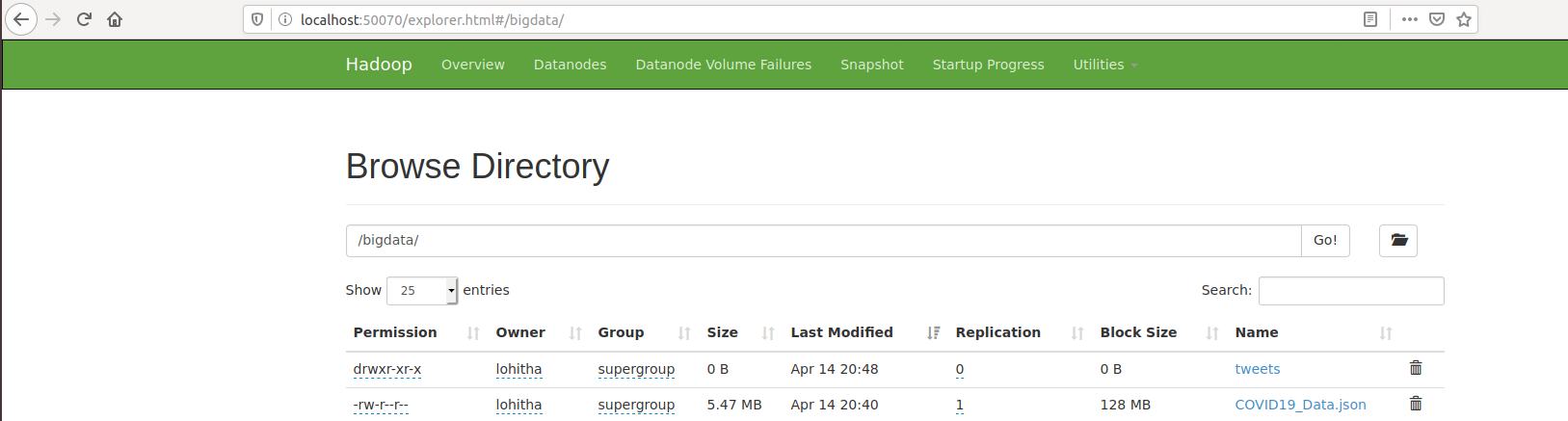


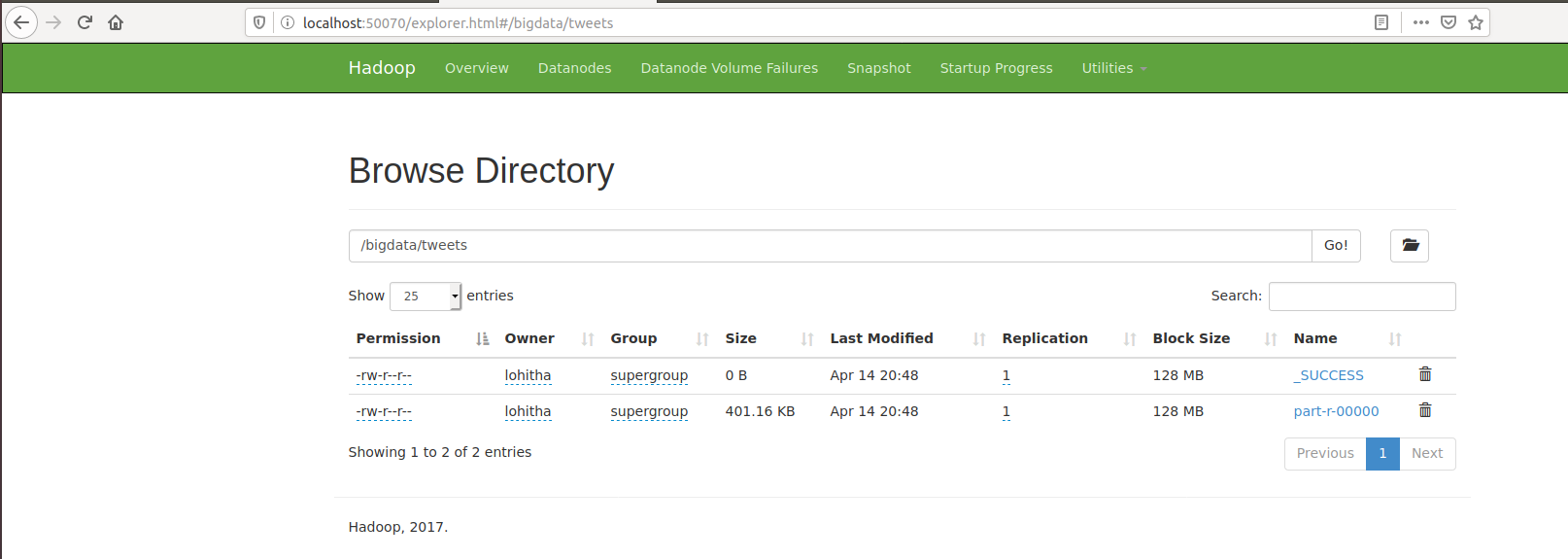
Running MapReduce:



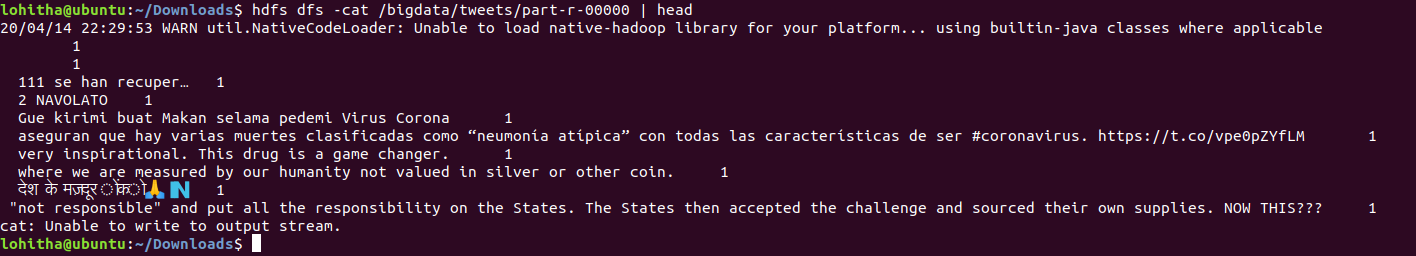


HDFS File Structure:



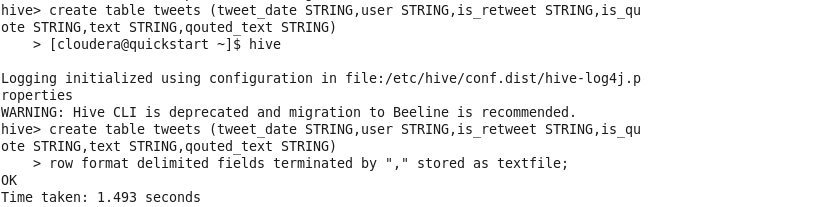


Output File:

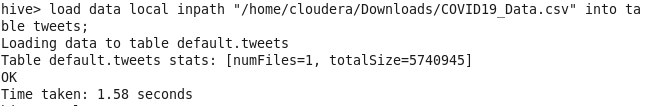


**Use Case 2:** Data analysis of tweets using Hive

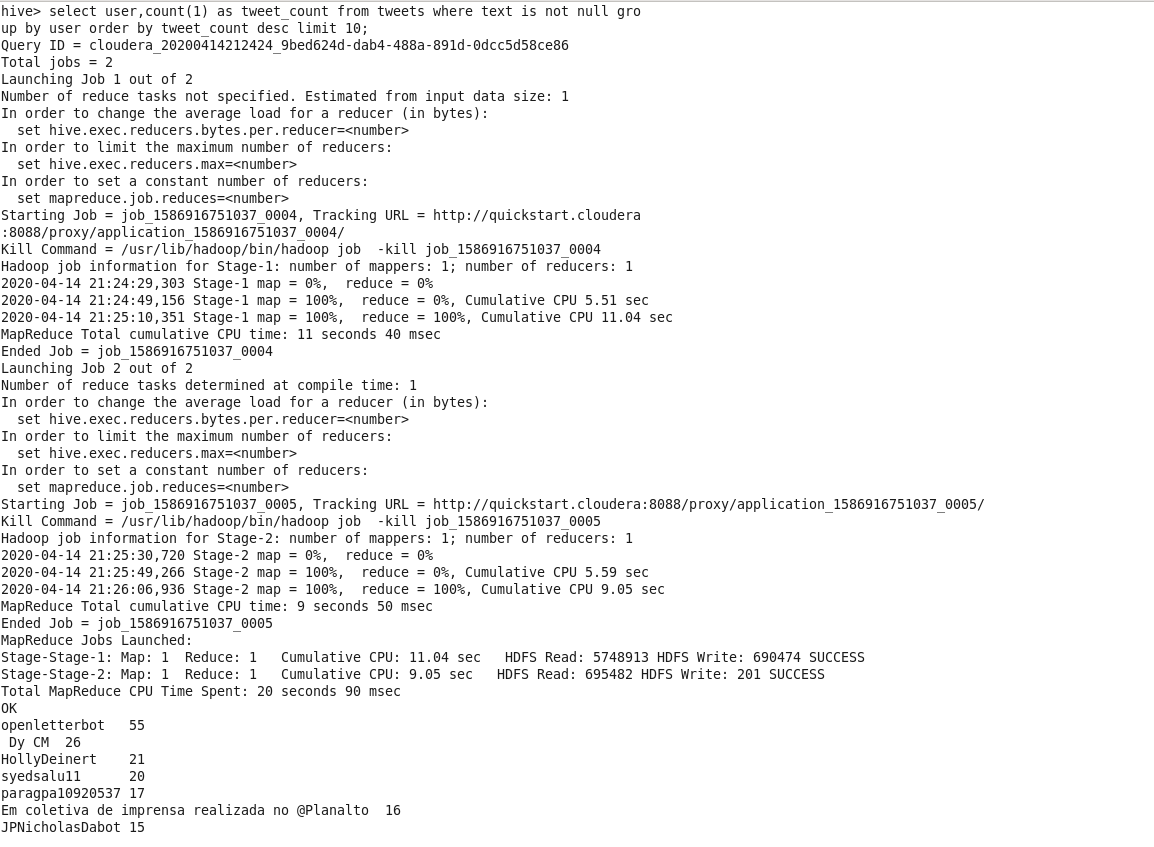
Create Tweets Table:

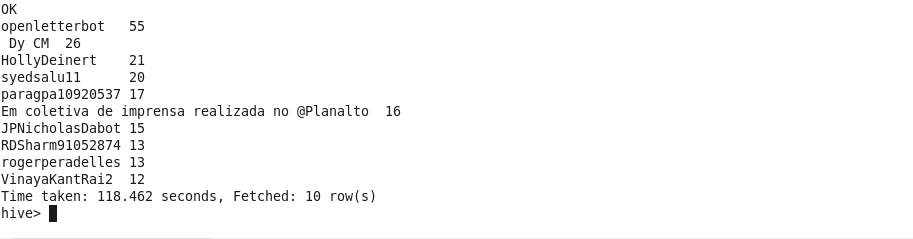


Load Twitter Data into Tweets table:

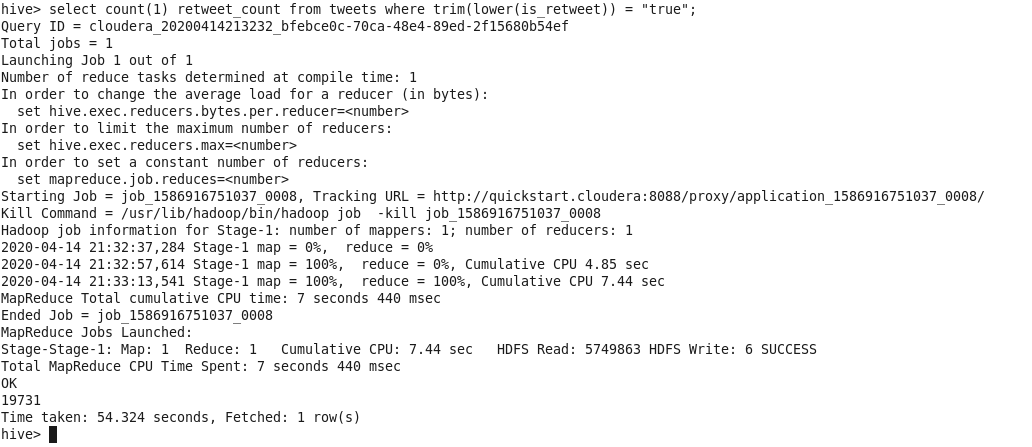


Query 1: Fetch top users with more number of tweets

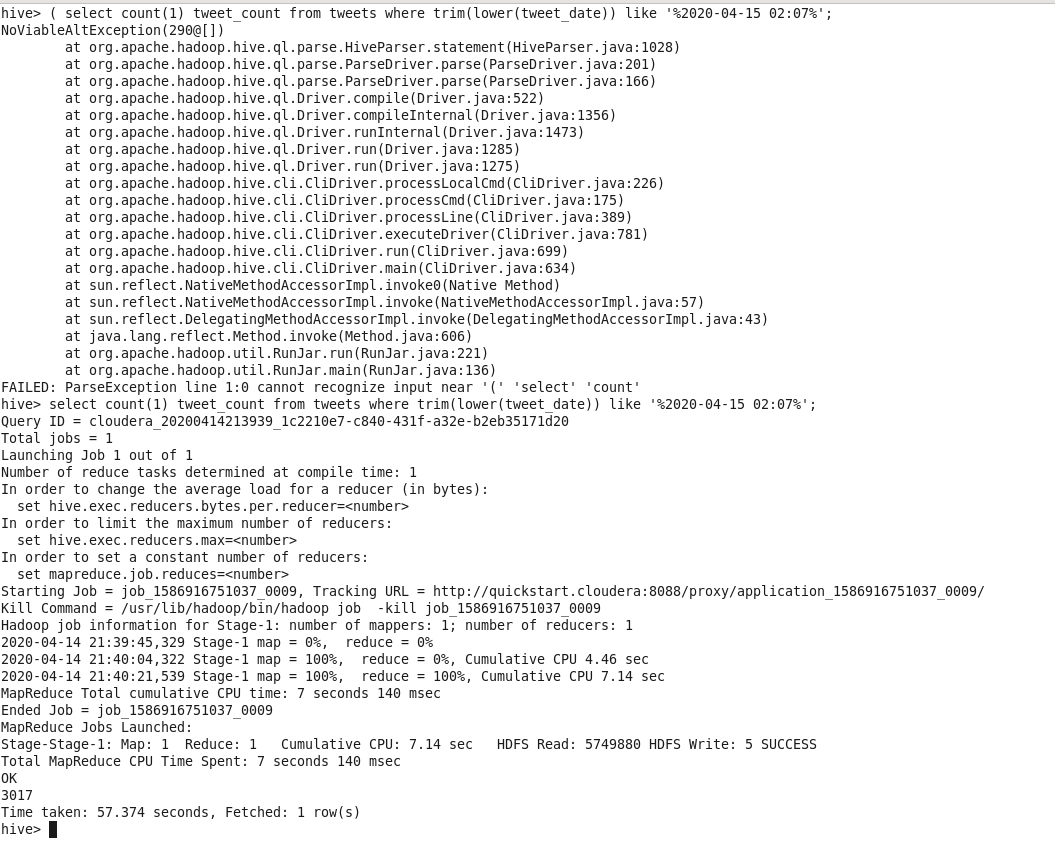




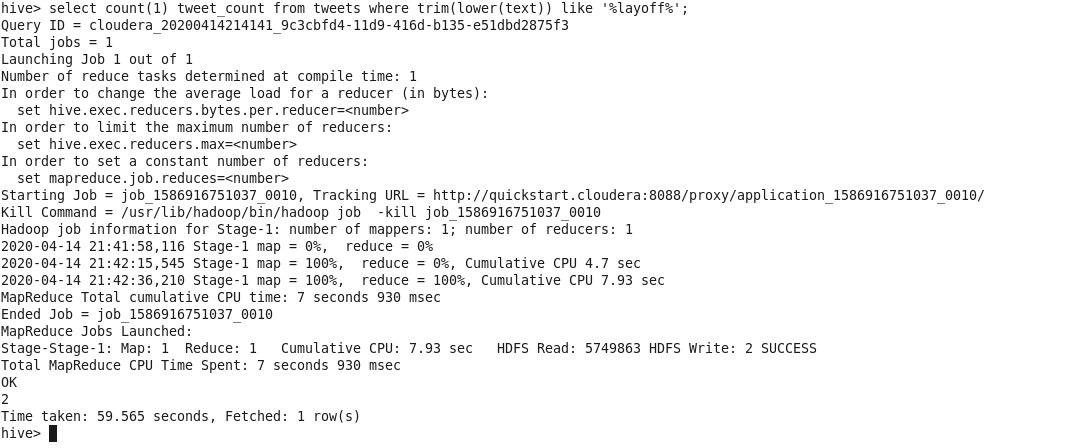
Query 2: Retweet count



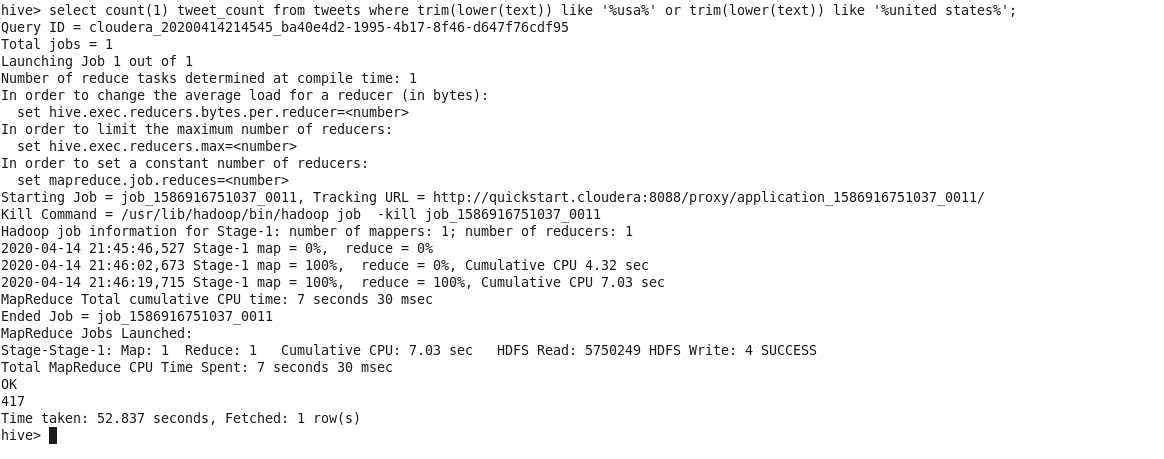
Query 3: Tweets per minute



Query 4: Tweets on Layoffs



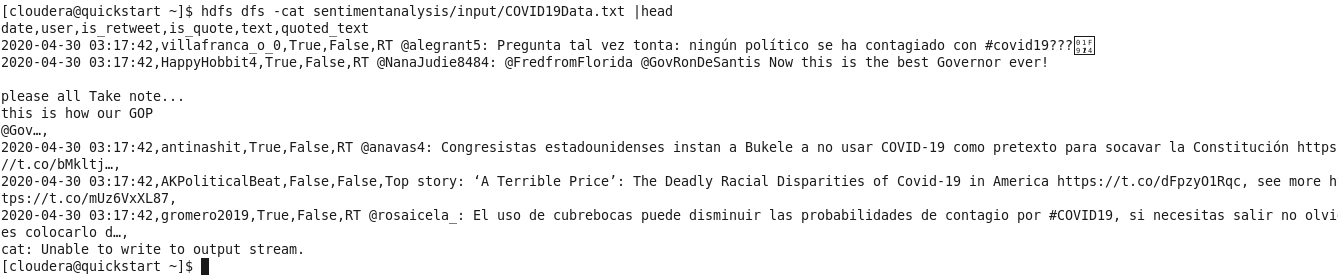
Query 5: Tweets on USA



**Use Case 3:** Twitter data sentimental analysis using Map Reduce

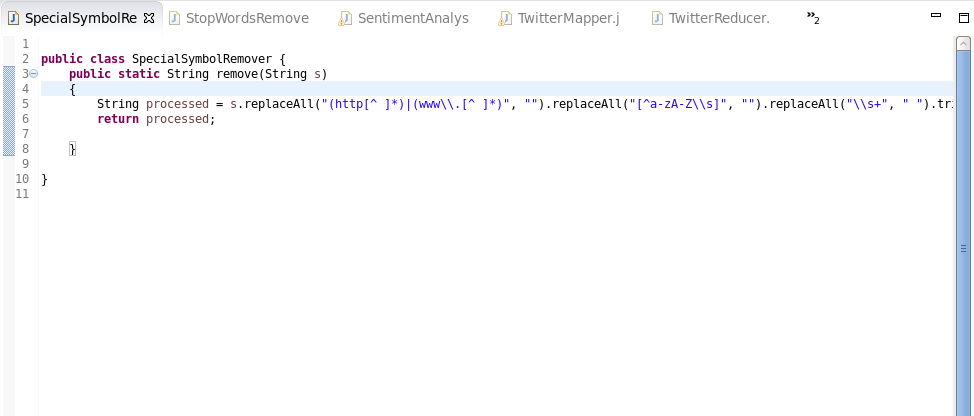
Analyzed the sensitivity of tweets. Divided tweets into 3 categories- positive, negative and neutral and displayed count for each category.

Data: Loaded input file to HDFS and displayed

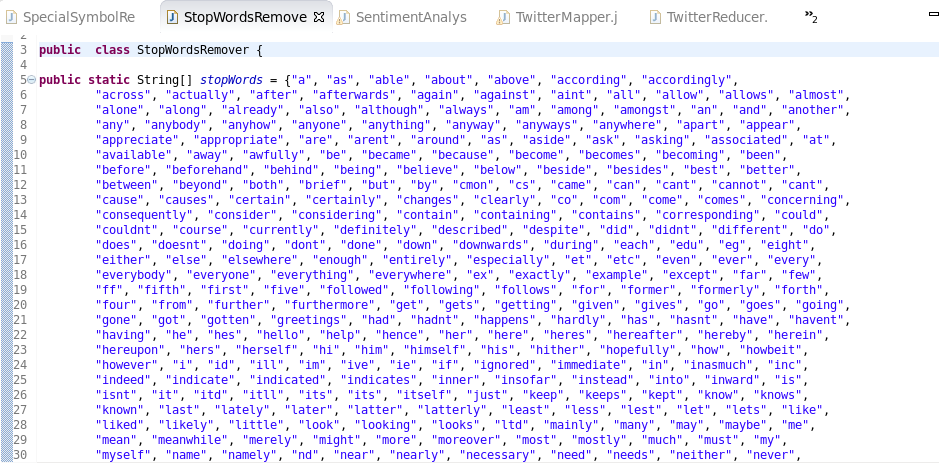


Code:

Created class- SpecialSymbolRemover for data preprocessing to remove urls in the tweets as they do not contribute to sensitivity of tweets.



Created class- StopWordsRemover for data preprocessing to remove stop words from data. Stop words like- a, to, pronouns are removed as they do not define tweet sensitivity.

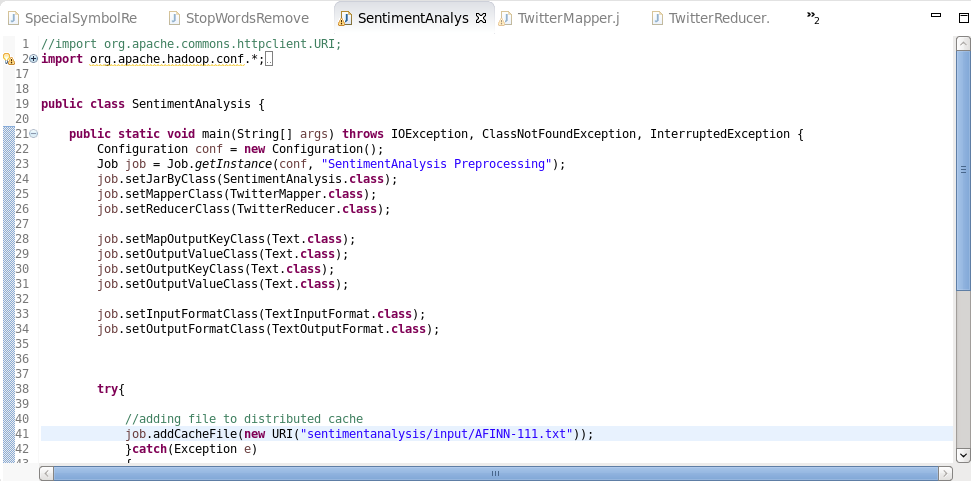






Main class- SentimentAnalysis

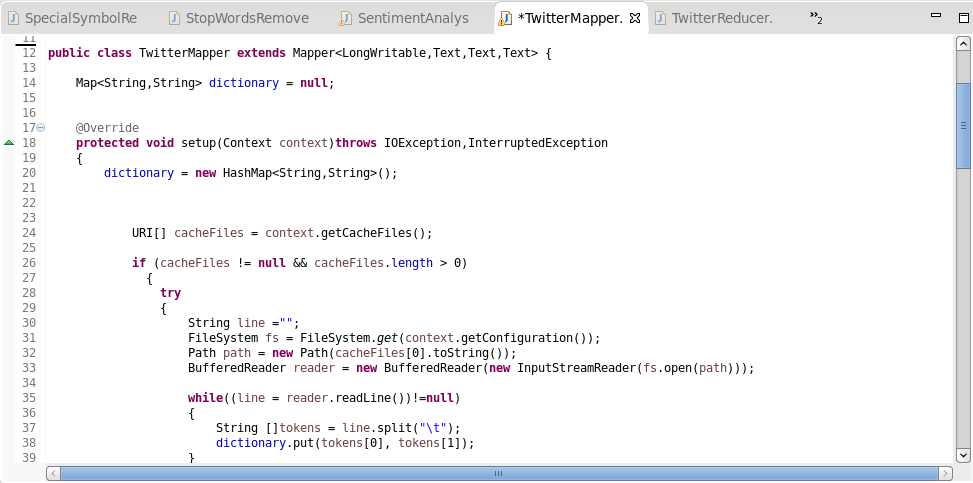
Created map, reduce jobs and added afinn-111 file to cache. Afinn-111 consists of sentiment values for various words describing emotions. Words describing emotions like anger, hatred have negative sentiment values. Words describing emotions like happiness, excitement have positive sentiment values. Words that describe neutral emotions hold sentiment value as ZERO.

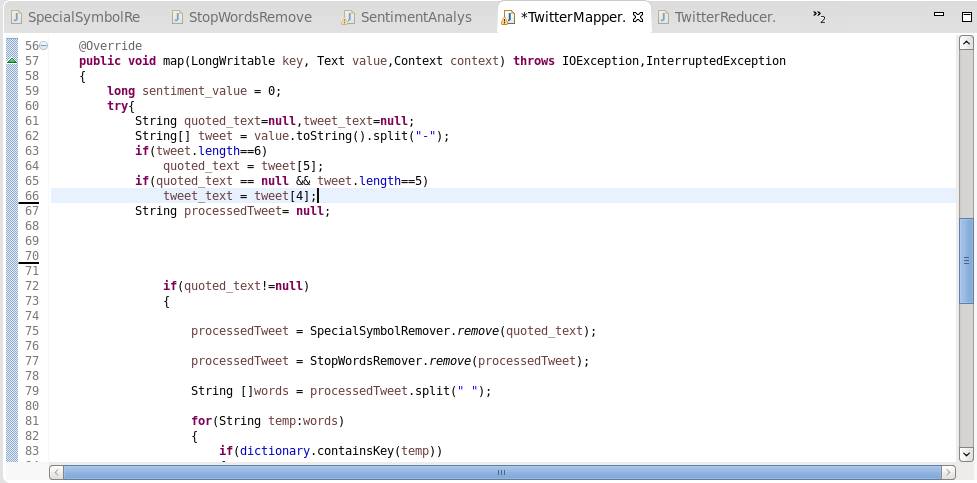




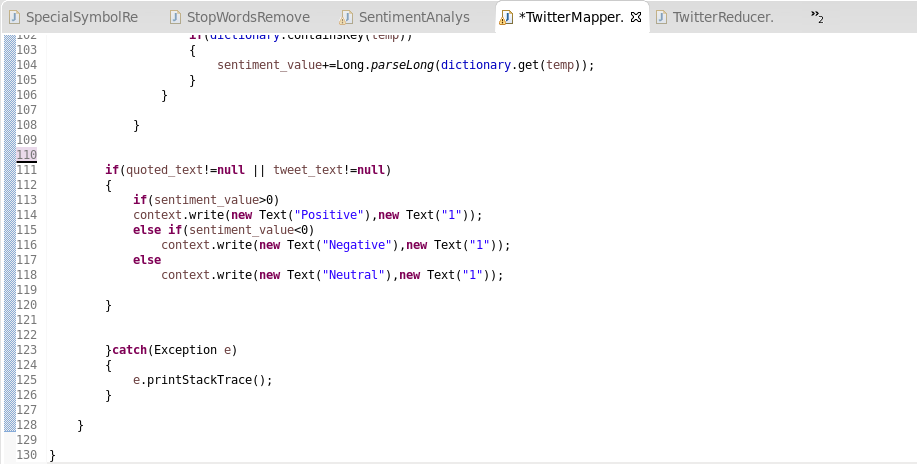
Mapper class- TwitterMapper

In the mapper class, loaded file from the cache and created map out of it. In the map function, for each tweet sentiment is calculated by summing each word sentiment. The output of map function is (sentiment\_category, 1). Example: (positive, 1)



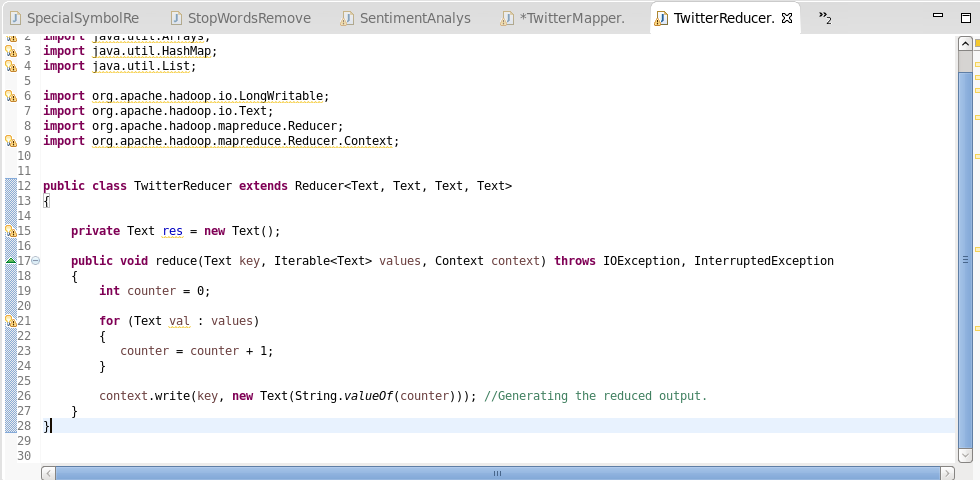


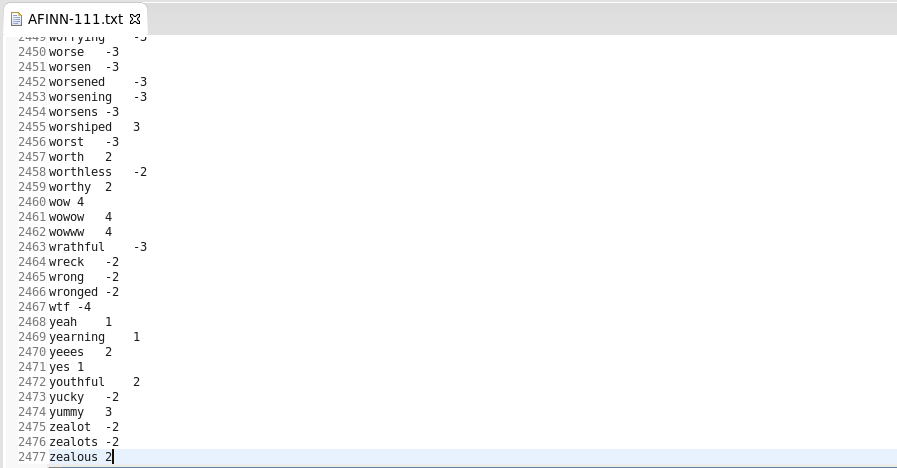




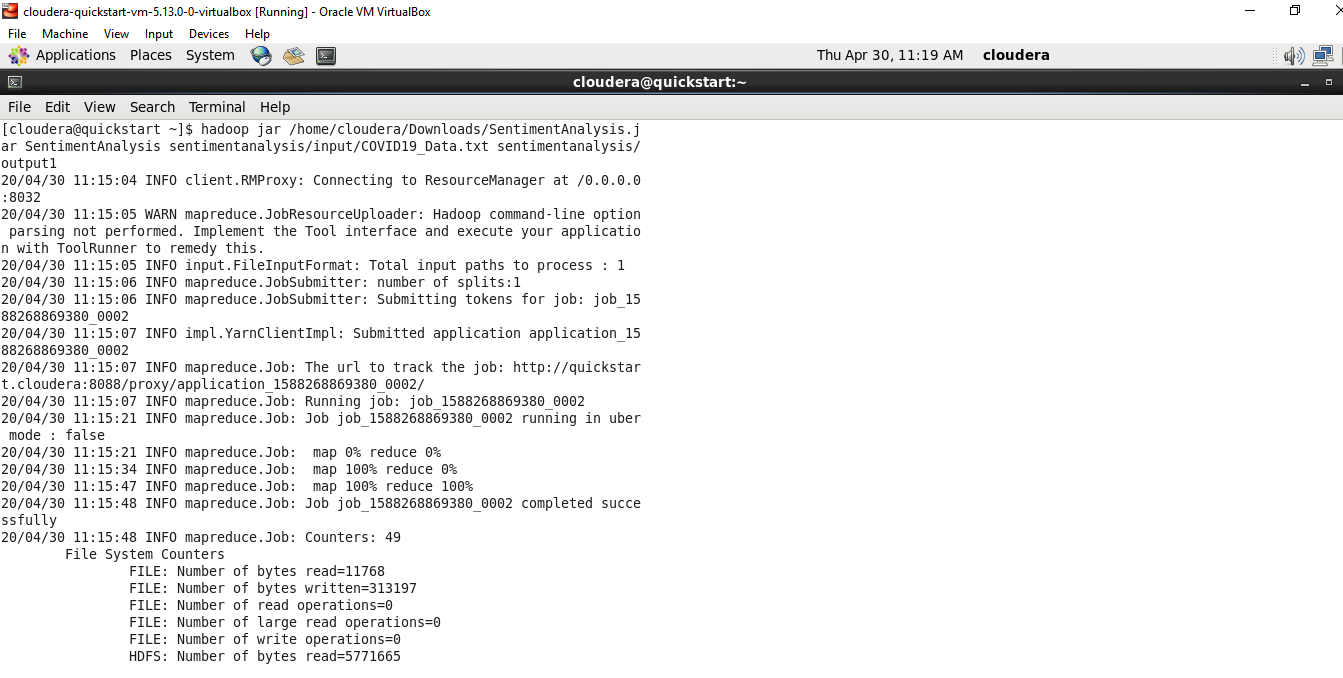
Reducer class- TwitterReducer

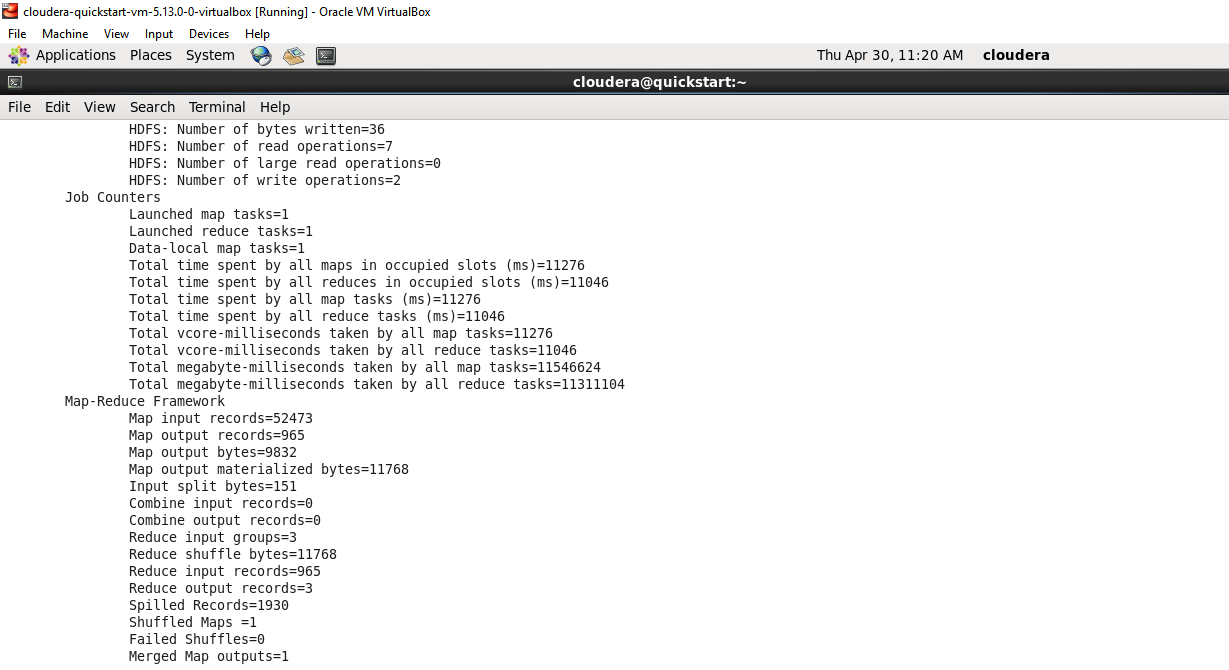
In the reducer class, each tweet category is summed for the final tweet count for each category.

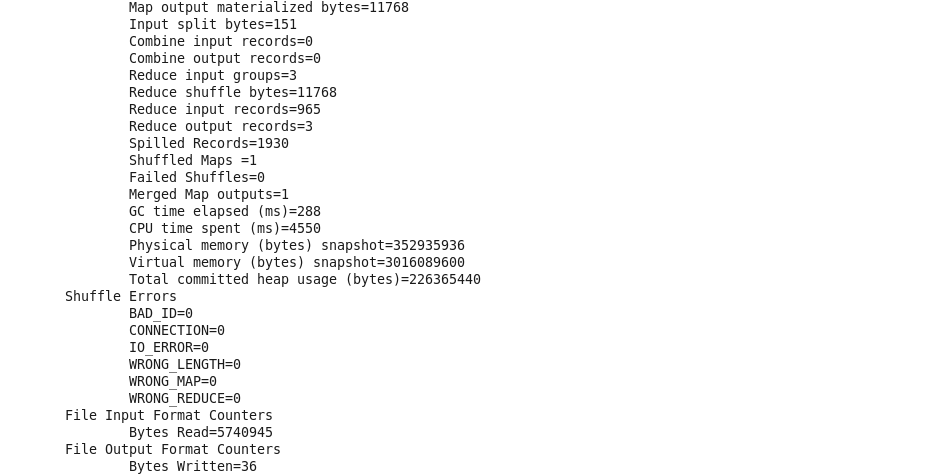


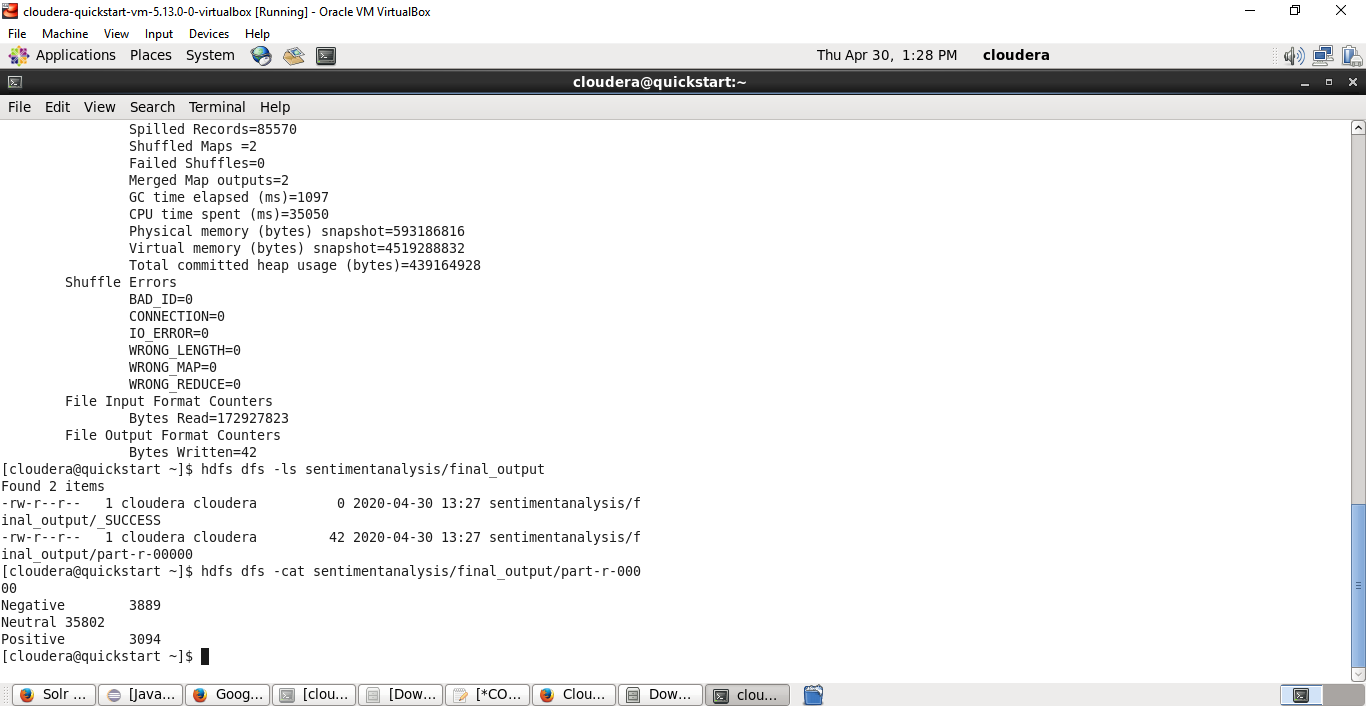


Output:









**Use Case 4:** Twitter data sentimental analysis using Cassandra

Useful analysis on tweets are performed using complex queries and user defined functions(UDFs).

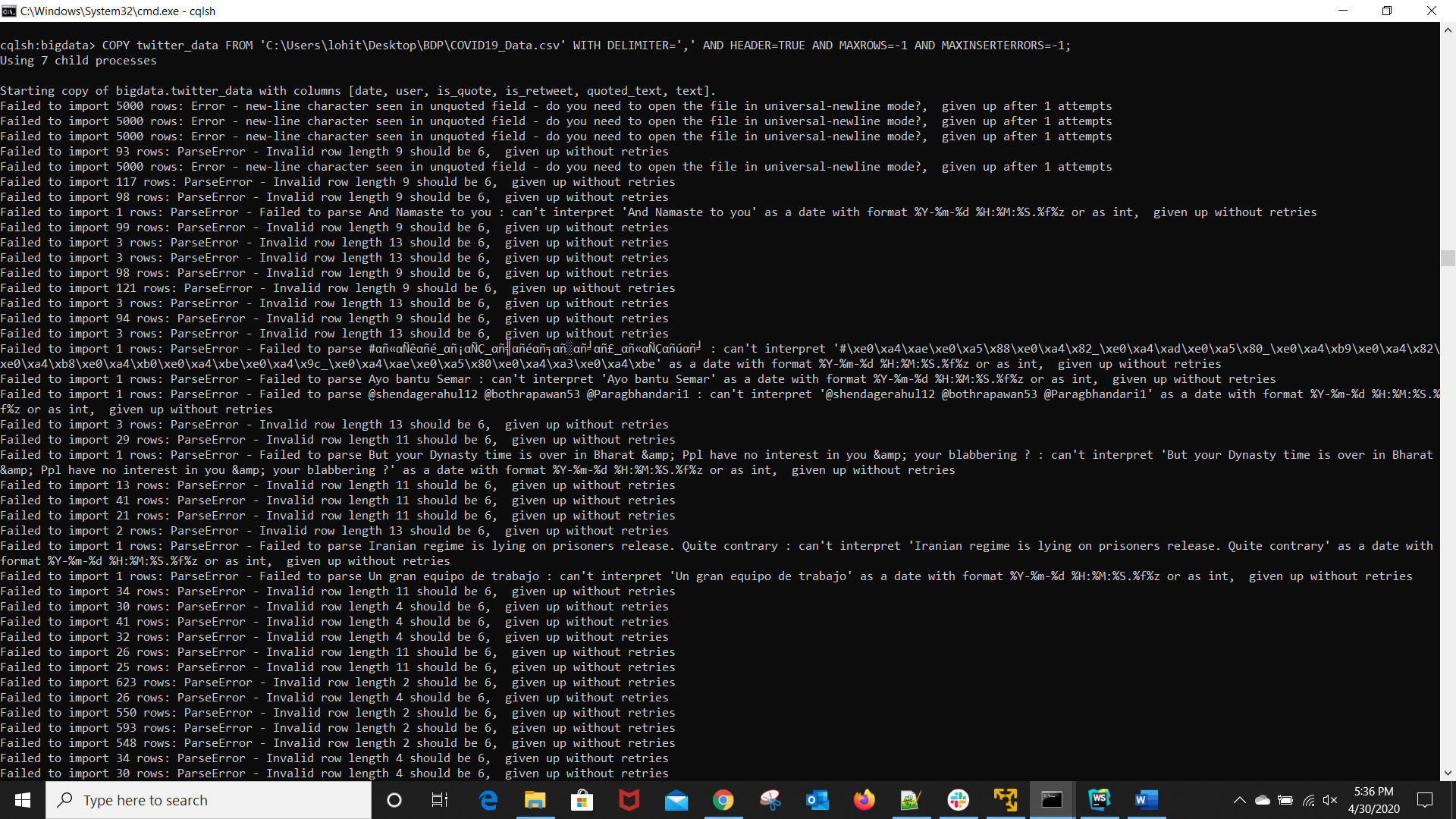
Create Table:

Twitter\_data table is created with 6 columns- date, user, is\_retweet, is\_quote, text, quoted\_text. Primary key is a composite key with columns (date,user).



Load Twitter data:

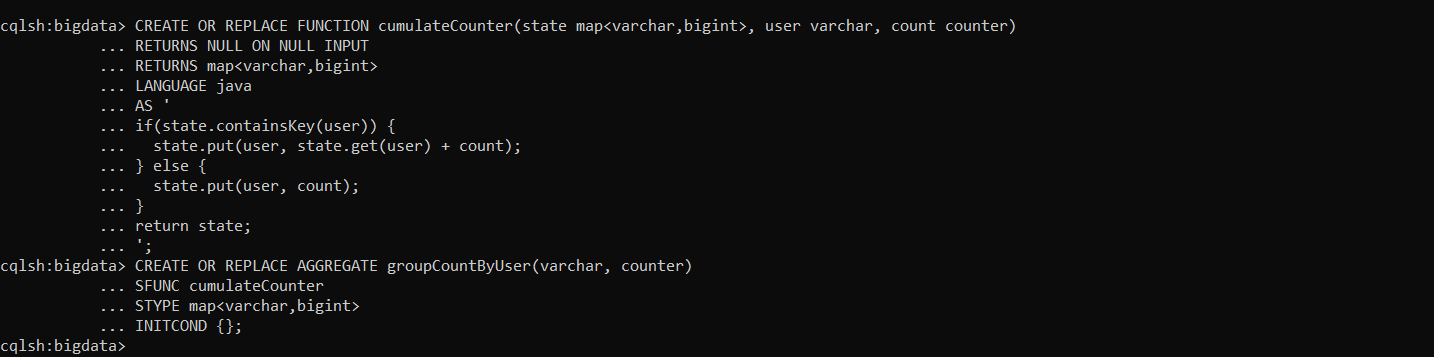
COVID19 csv file is loaded into twitter\_data table with more than 2 lakh records.

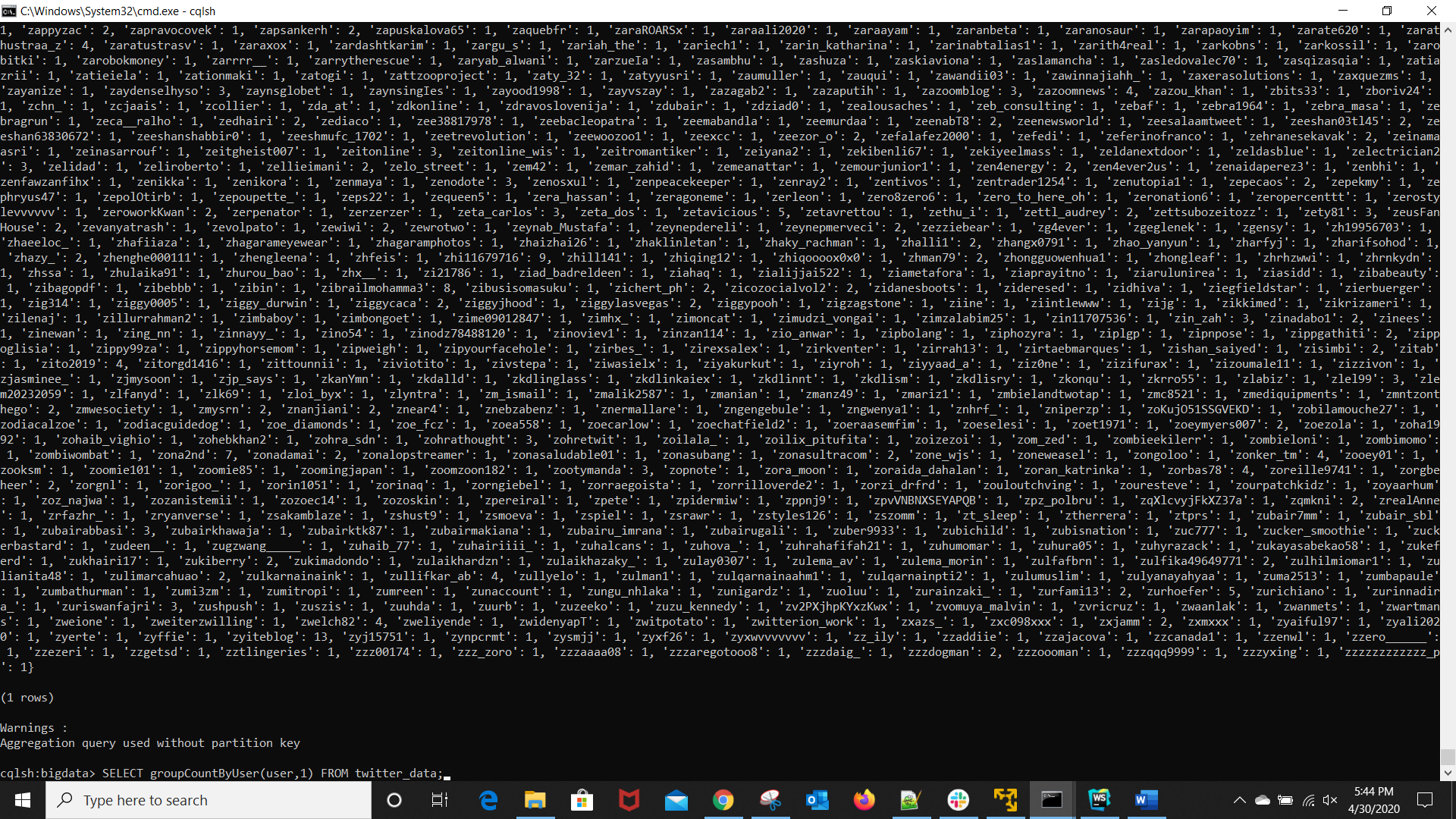




Analysis 1: Display the users and their tweet count

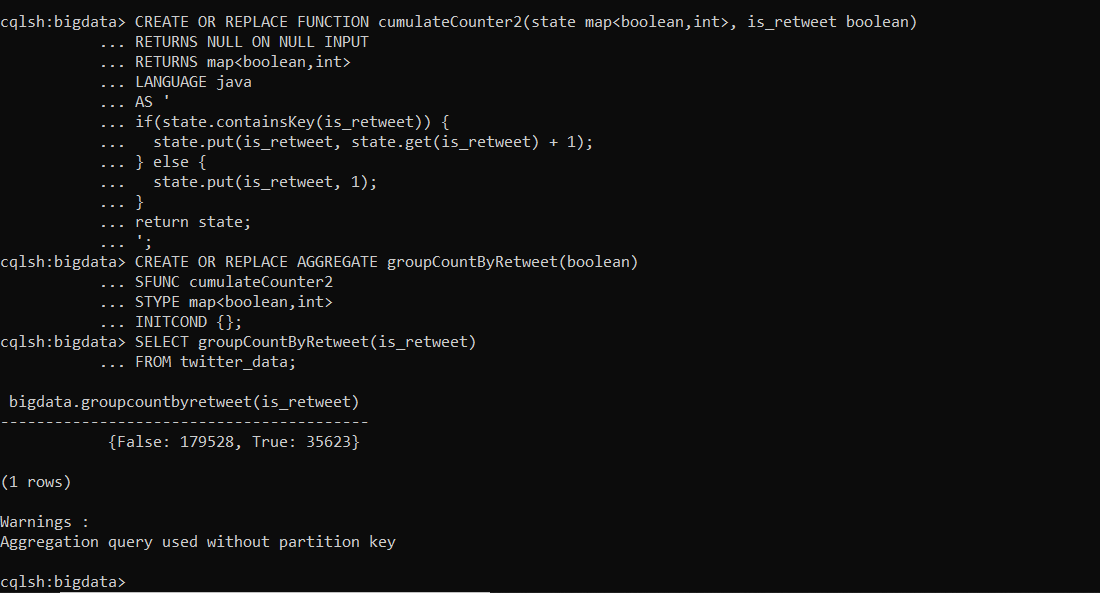
Cassandra does not support GROUPBY clause. To perform group by on the user column to count the tweets by each user, created UDFs- cumulateCounter and groupCountByUser. CumulateCounter maps each user to 1 if it is a new user else increments the existing count.





Analysis 2: Count the number of retweets

To perform GROUPBY aggregation on is\_retweet column, created UDFs- cumulateCounter2 and groupCountByRetweet. CumulateCounter2 aggregates retweets by their Boolean value- false and true.

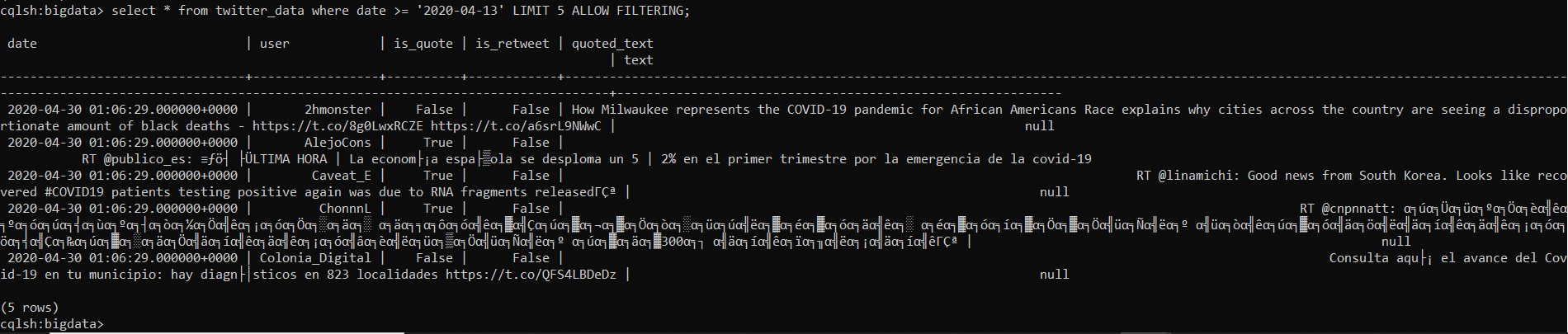


Analysis 3: Display users and their tweets whose username starts with character ‘L’

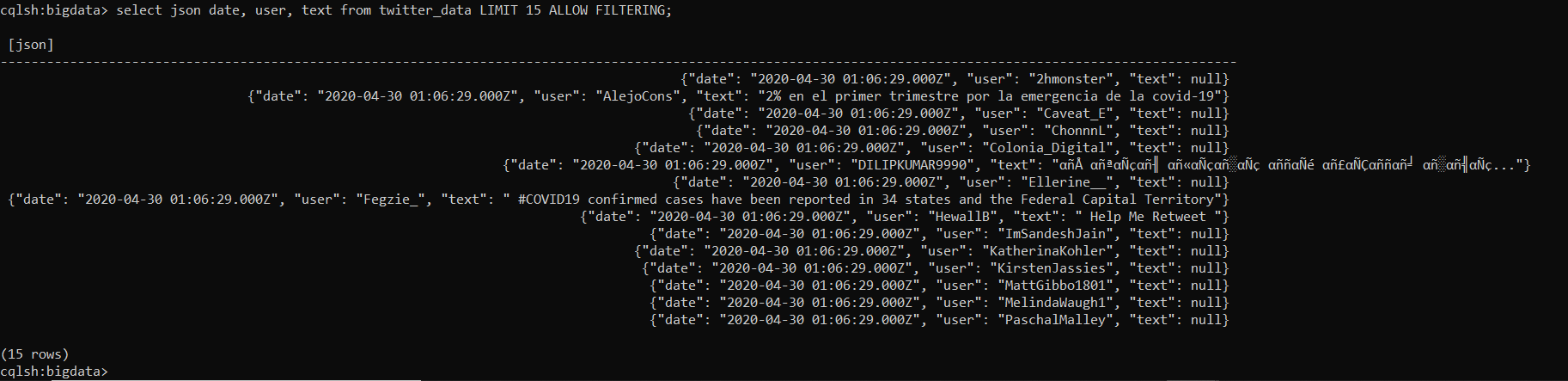
Cassandra directly does not support LIKE operator. To display user tweets whose username starts with ‘L’, SASI index is created on user column.



Analysis 4: Display tweets that are tweeted after date- '2020-04-13'

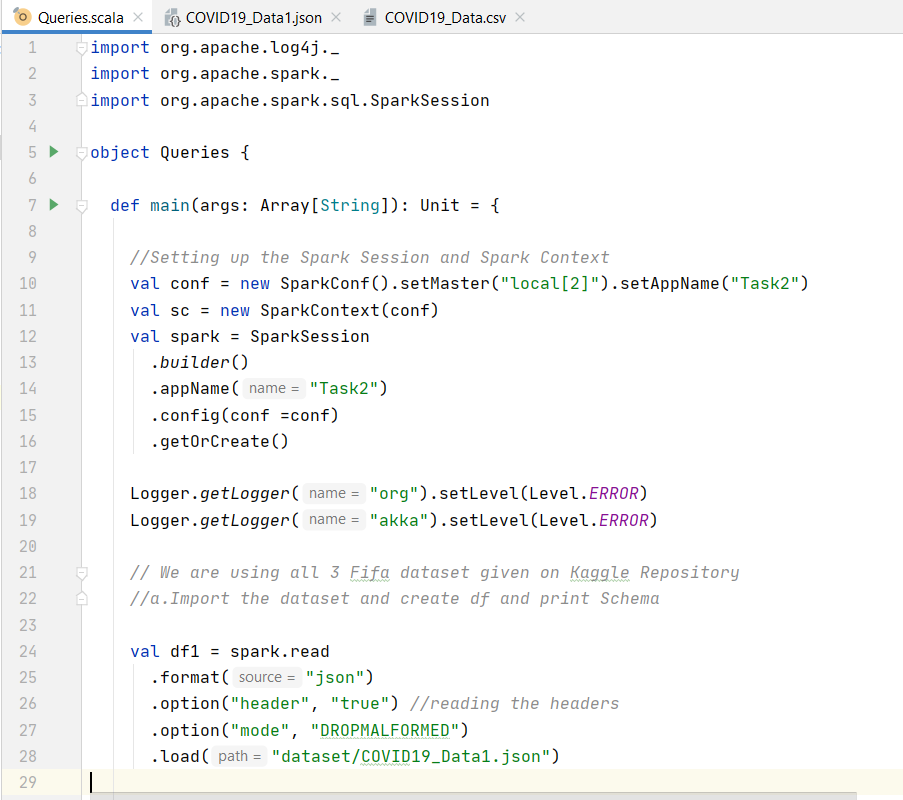


Query 5: Display table data in json format

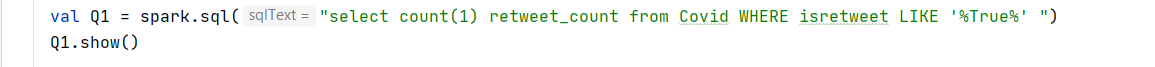


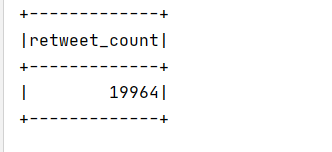
**Use Case 5:** Twitter data sentimental analysis using Spark SQL

Load json data into SparkContext

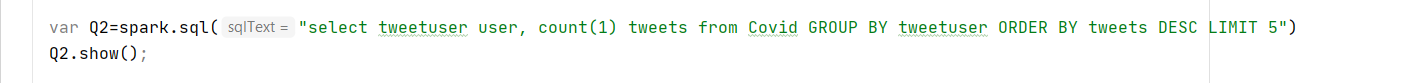


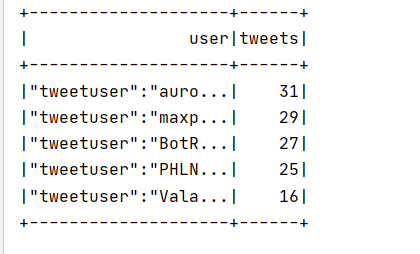
Analysis 1: Display the count of the tweets that are retweeted



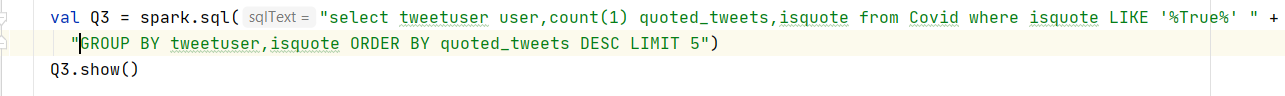


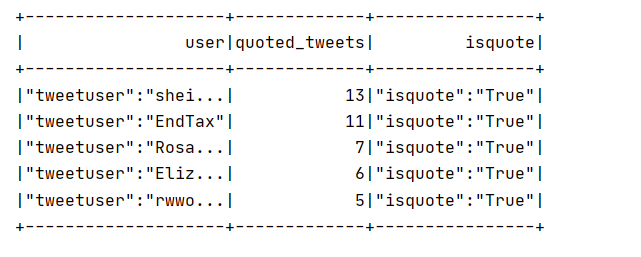
Analysis 2: Display the top 5 users with highest number of tweets along with count



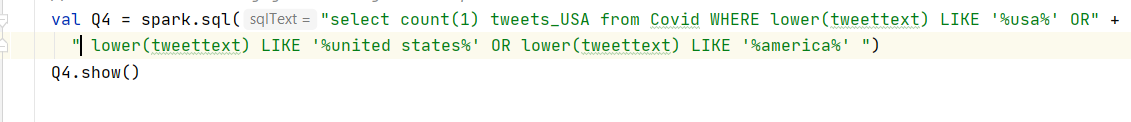


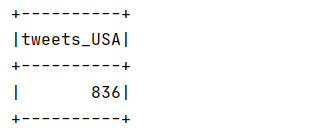
Analysis 3: Display the top 5 users with highest number of quoted tweets along with count





Analysis 4: Display the number of tweets taking about ‘United States’





Work Completed (100%):

* Collected Data
* Analyzed data using map reduce and hive
* Analyzed data using Cassandra
* Performed sentiment analysis using map reduce
* Analyzed data using Spark SQL

Responsibility:

Vidyullatha Lakshmi Kaza- 34%

Aparna Manda- 33%

Lohitha Yenugu- 33%

References:

1. <https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/intro-to-tweet-json>
2. <https://cassandra.apache.org/doc/latest/cql/dml.html>
3. <https://stackoverflow.com/questions/17342176/max-distinct-and-group-by-in-cassandra>
4. <https://docs.datastax.com/en/cql-oss/3.3/cql/cql_reference/cqlCreateAggregate.html>