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:

Open ▾ TwitterDataExtraction.py
~/Desktop/BigDataProgramming

```
    def on_status(self, status):
        print(status.id_str)
        # if "retweeted_status" attribute exists, flag this tweet as a retweet.
        is_retweet = hasattr(status, "retweeted_status")

        # check if text has been truncated
        if hasattr(status,"extended_tweet"):
            text = status.extended_tweet["full_text"]
        else:
            text = status.text

        # check if this is a quote tweet.
        is_quote = hasattr(status, "quoted_status")
        quoted_text = ""
        if is_quote:
            # check if quoted tweet's text has been truncated before recording it
            if hasattr(status.quoted_status,"extended_tweet"):
                quoted_text = status.quoted_status.extended_tweet["full_text"]
            else:
                quoted_text = status.quoted_status.text

        # remove characters that might cause problems with csv encoding
        remove_characters = [",","\n"]
        for c in remove_characters:
            text.replace(c," ")
            quoted_text.replace(c, " ")

        with open("/home/lohitha/Desktop/BigDataProgramming/COVID19_Data.json", "a", encoding='utf-8') as f:
            f.write("%s,%s,%s,%s,%s,%s\n" % (status.created_at,status.user.screen_name,is_retweet,is_quote,text,quoted_text))

    def on_error(self, status_code):
        print("Encountered streaming error (", status_code, ")")
        sys.exit()

if __name__ == "__main__":
    # complete authorization and initialize API endpoint
    auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
    auth.set_access_token(access_token, access_token_secret)
    api = tweepy.API(auth)

    # initialize stream
    streamListener = StreamListener()
    stream = tweepy.Stream(auth=api.auth, listener=streamListener,tweet_mode='extended')
    with open("/home/lohitha/Desktop/BigDataProgramming/COVID19_Data.json", "w", encoding='utf-8') as f:
        f.write("date,user,is_retweet,is_quote,text,quoted_text\n")
    tags = ["covid19","covid-19","COVID-19","COVID19","corona","CORONA"]
    stream.filter(track=tags)
```

Python Code to extract live stream twitter data as JSON:

TwitterDataExtractioncsv.py × COVID19_Data1.json

```python
        is_retweet = hasattr(status, "retweeted_status")

        # check if text has been truncated
        if hasattr(status,"extended_tweet"):
            text = status.extended_tweet["full_text"]
        else:
            text = status.text

        # check if this is a quote tweet.
        is_quote = hasattr(status, "quoted_status")
        quoted_text = ""
        if is_quote:
            # check if quoted tweet's text has been truncated before recording it
            if hasattr(status.quoted_status,"extended_tweet"):
                quoted_text = status.quoted_status.extended_tweet["full_text"]
            else:
                quoted_text = status.quoted_status.text

        # remove characters that might cause problems with csv encoding
        remove_characters = [",","\n"]
        for c in remove_characters:
            text.replace(c," ")
            quoted_text.replace(c, " ")

        with open("/home/lohitha/Desktop/BigDataProgramming/COVID19_Data1.json", "a", encoding='utf-8') as f:
            f.write('{"id":"%s","date":"%s","tweetuser":"%s","isretweet":"%s","isquote":"%s","tweettext":"%s","quotedtext":"%s"}\n' %
(status.id_str,status.created_at,status.user.screen_name,is_retweet,is_quote,text,quoted_text))

    def on_error(self, status_code):
        print("Encountered streaming error (", status_code, ")")
        sys.exit()

if __name__ == "__main__":
    # complete authorization and initialize API endpoint
    auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
    auth.set_access_token(access_token, access_token_secret)
    api = tweepy.API(auth)

    # initialize stream
    streamListener = StreamListener()
    stream = tweepy.Stream(auth=api.auth, listener=streamListener,tweet_mode='extended')
    with open("/home/lohitha/Desktop/BigDataProgramming/COVID19_Data1.json", "w", encoding='utf-8') as f:
        f.write("id,d,usr,\n")
    tags = ["covid19","covid-19","COVID-19","COVID19","corona","CORONA"]
    stream.filter(track=tags)
```

Collected Tweets:

```
date,user,is_retweet,is_quote,text,quoted_text
2020-04-15 02:06:43,kimpetty64,False,True,😂😂😂😱 and just like Corona beer,Meanwhile, Trump supporters burn their 8-tracks cursing out Keith Moon, Pete Townshend and Roger Daltrey.

Blaming WHO
2020-04-15 02:06:43,2021Diary,True,False,RT @PinkeshOfficial: While PM Modi having serious discussion with states on Covid19, Aditya Thackeray is busy with his mobile.

Anjana was…,
2020-04-15 02:06:43,rdc_south,True,False,RT @chennaicorp: Here's the Graphical Representation of total COVID-19 positive cases in Chennai as on 14-04-2020.

#Covid19Chennai
#GCC #…,
2020-04-15 02:06:42,openletterbot,False,False,🖊 Support Patrick by signing “Support the USPS!” and I’ll deliver a copy to your officials too: https://t.co/GjO2cQ6Sqq

📤 Last delivered to @RepStephenLynch, @SenMarkey and @SenWarren #MA08 #MApoli #MApols #COVID19 https://t.co/JSsgHhZjoI,
2020-04-15 02:06:43,ahernandez85b,True,True,RT @KelemenCari: It's been 35 years and there is still no vaccine for AIDS. https://t.co/X8yKxTSt3G,As we reopen #Ohio, people will have to be ve
2020-04-15 02:06:43,lucas_bhmg,True,True,RT @AbdelDeuxFois: Sans promo ni rien. Le monde chico 🌎,FR ALERTE INFO - 36,7 millions de téléspectateurs ont suivi l'allocution d'Emmanuel #Macron
2020-04-15 02:06:43,Celinve,True,False,RT @MonsieurQFB: Aquí los análisis de casos de COVID–19 en  México MX y países de América, Europa US IT ES KR BR DE JP FR PE AR EC CL CO y…,
2020-04-15 02:06:43,MrSol54252546,True,False,RT @islamramahdotco: Kiai Anwar Zahid: Patuhi Protokoler Covid-19

"Kita ajak seluruh umat agar taat dan patuh instruksi pemerintah dan pat…,
2020-04-15 02:06:43,drjaswantpatil,False,False,@SrBachchan # Homeopathy can beat Corona.                    Want to see the results allow homoeopaths to treat and see the change.
2020-04-15 02:06:43,xznxxz,True,False,RT @gunawan_anas: Jujurnya pemerintah masih setengah2.. selain data saat ini, pmrintah harus terbuka mengenai forecasting versi mereka akan…,
2020-04-15 02:06:43,WUNNA_1,True,False,RT @21savage: Bang outside I hang outside
don’t come out da house cuz corona outside,
2020-04-15 02:06:43,abnesdad,True,False,RT @TrumpNoodles: 🔥Michigan kid videotaping his dad who hates the Michigan governor

🔥Funny as heck 😂😂😂😂

🔥He sounds like my mom when it co…,
2020-04-15 02:06:43,respirovondy,True,False,RT @crushdobbb20: Flávia Pavanelli fez foto, fez preenchimento, arrumou cabelo, unhas, encontrou amigos em casa e agora vem falar q acha q…,
2020-04-15 02:06:43,carterpillar82,True,False,RT @franjuero: Un reportaje de la Televisión Italiana del 2015 donde habla de que China experimenta con un virus SARS insertandole proteína…,
2020-04-15 02:06:43,Kak77742001,True,False,RT @BastienParisot: ⚫ THREAD : Une proche m'envoie ce soir ces photos. Elle travaille dans un hôpital public de la région parisienne. Des b…,
2020-04-15 02:06:43,FloreCsGo,True,True,RT @AbdelDeuxFois: Sans promo ni rien. Le monde chico 🌎,FR ALERTE INFO - 36,7 millions de téléspectateurs ont suivi l'allocution d'Emmanuel #Macron h
2020-04-15 02:06:43,QAlwaysWins,True,False,RT @21savage: Bang outside I hang outside
don’t come out da house cuz corona outside,
2020-04-15 02:06:43,poppy_yifan,True,True,RT @McKe3z: อีสัด ประชาชนไทยทั้งสู้โควิดกันเอง บริจาคเงินกันเอง แบ่งของกินของใช้กันเอง ดับไฟป่าเอง มาถึงขั้นสร้างความสัมพันธ์ระหว่างประเทศ…,Thank you our friends from #Thailand. Thailand has long been a popu
ไต้หวันและไทย สู้ไปด้วยกันนะครับTW TH
#nnevvy
2020-04-15 02:06:43,ikanbadutz,False,False,Makin tanbah hari, akhir dr wabah corona ini makin dekat 😁

Dan semua kembali normal, yes 😁😁😁,
2020-04-15 02:06:43,babyyis_,True,False,RT @21savage: Bang outside I hang outside
```

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

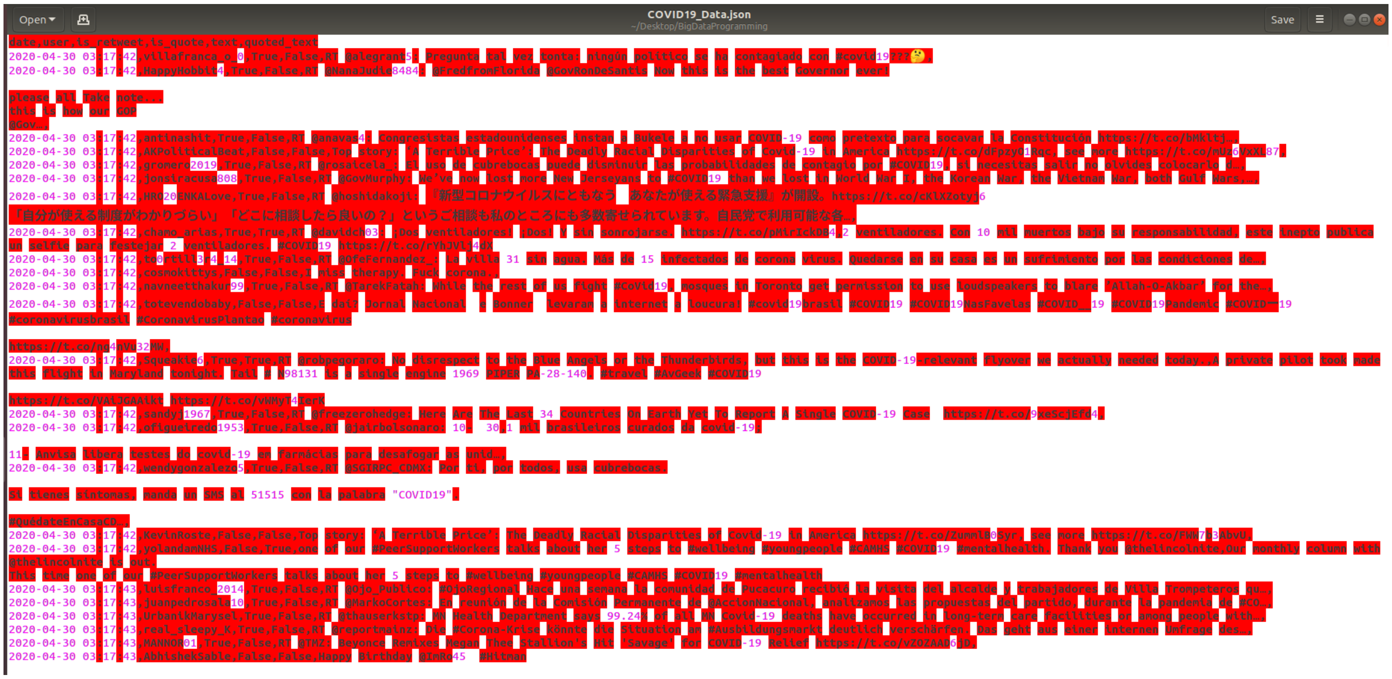
tweet_text – text

is_retweet – boolean

is_quote – boolean

quoted_text – text

```
cqlsh> desc bigdata.twitter_data;

CREATE TABLE bigdata.twitter_data (
    date timestamp,
    user text,
    is_quote boolean,
    is_retweet boolean,
    quoted_text text,
    text text,
    PRIMARY KEY (date, user)
) WITH CLUSTERING ORDER BY (user ASC)
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32',
 'min_threshold': '4'}
    AND compression = {'chunk_length_in_kb': '64', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
    AND crc_check_chance = 1.0
    AND dclocal_read_repair_chance = 0.1
    AND default_time_to_live = 0
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair_chance = 0.0
    AND speculative_retry = '99PERCENTILE';
CREATE CUSTOM INDEX user_idx ON bigdata.twitter_data (user) USING 'org.apache.cassandra.index.sasi.SASIIndex' WITH OPTIO
NS = {'analyzer_class': 'org.apache.cassandra.index.sasi.analyzer.StandardAnalyzer', 'case_sensitive': 'false'};
```


Open
COVID19_Data.json
~/Desktop/BigDataProgramming
Save
date,user,is_retweet,is_quote,text,quoted_text
2020-04-30 03:17:42,villafranca_o_0,True,False,RT @alegrant5: Pregunta tal vez tonta: ningún político se ha contagiado con #covid19???🤔,
2020-04-30 03:17:42,HappyHobbit4,True,False,RT @NanaJudie8484: @FredfromFlorida @GovRonDeSantis Now this is the best Governor ever!
please all Take note...
this is how our GOP
@Gov…,
2020-04-30 03:17:42,antinashit,True,False,RT @anavas4: Congresistas estadounidenses instan a Bukele a no usar COVID-19 como pretexto para socavar la Constitución https://t.co/bMkltj…,
2020-04-30 03:17:42,AKPoliticalBeat,False,False,Top story: ‘A Terrible Price’: The Deadly Racial Disparities of Covid-19 in America https://t.co/dFpzy01Rqc, see more https://t.co/mUz6VxXL87,
2020-04-30 03:17:42,gromero2019,True,False,RT @rosaicela_: El uso de cubrebocas puede disminuir las probabilidades de contagio por #COVID19, si necesitas salir no olvides colocarlo d…,
2020-04-30 03:17:42,jonsiracusa808,True,False,RT @GovMurphy: We’ve now lost more New Jerseyans to #COVID19 than we lost in World War I, the Korean War, the Vietnam War, both Gulf Wars,…,
2020-04-30 03:17:42,HRO20ENKALove,True,False,RT @hoshidakoji: 『新型コロナウイルスにともなう あなたが使える緊急支援』が開設。https://t.co/cKlXZotyj6
「自分が使える制度がわかりづらい」「どこに相談したら良いの？」というご相談も私のところにも多数寄せられています。自民党で利用可能な各…,
2020-04-30 03:17:42,chamo_arias,True,True,RT @davidch03: ¡Dos ventiladores! ¡Dos! Y sin sonrojarse. https://t.co/pMirIckDB4,2 ventiladores. Con 10 mil muertos bajo su responsabilidad, este inepto publica un selfie para festejar 2 ventiladores. #COVID19 https://t.co/rYhJVlj4dX
2020-04-30 03:17:42,to0rtill3r4_14,True,False,RT @OfeFernandez_: La villa 31 sin agua. Más de 15 infectados de corona virus. Quedarse en su casa es un sufrimiento por las condiciones de…,
2020-04-30 03:17:42,cosmokittys,False,False,I miss therapy. Fuck corona.,
2020-04-30 03:17:42,navneetthakur99,True,False,RT @TarekFatah: While the rest of us fight #CoVid19, mosques in Toronto get permission to use loudspeakers to blare ’Allah-O-Akbar’ for the…,
2020-04-30 03:17:42,totevendobaby,False,False,E daí? Jornal Nacional e Bonner levaram a internet a loucura! #covid19brasil #COVID19 #COVID19NasFavelas #COVID__19 #COVID19Pandemic #COVID━19
#coronavirusbrasil #CoronavirusPlantao #coronavirus
https://t.co/ng4nVu32MW,
2020-04-30 03:17:42,Squeakie6,True,True,RT @robpegoraro: No disrespect to the Blue Angels or the Thunderbirds, but this is the COVID-19-relevant flyover we actually needed today.,A private pilot took made this flight in Maryland tonight. Tail # N98131 is a single engine 1969 PIPER PA-28-140. #travel #AvGeek #COVID19
https://t.co/VAiJGAAikt https://t.co/vWMyT4IerK
2020-04-30 03:17:42,sandyj1967,True,False,RT @freezerohedge: Here Are The Last 34 Countries On Earth Yet To Report A Single COVID-19 Case https://t.co/9xeScjEfd4,
2020-04-30 03:17:42,ofigueiredo1953,True,False,RT @jairbolsonaro: 10- 30,1 mil brasileiros curados da covid-19;
11- Anvisa libera testes do covid-19 em farmácias para desafogar as unid…,
2020-04-30 03:17:42,wendygonzalezo5,True,False,RT @SGIRPC_CDMX: Por ti, por todos, usa cubrebocas.
Si tienes síntomas, manda un SMS al 51515 con la palabra "COVID19".
#QuédateEnCasaCD…,
2020-04-30 03:17:42,KevinRoste,False,False,Top story: ‘A Terrible Price’: The Deadly Racial Disparities of Covid-19 in America https://t.co/ZummlB0Syr, see more https://t.co/FWW7b3AbvU,
2020-04-30 03:17:42,yolandamNHS,False,True,one of our #PeerSupportWorkers talks about her 5 steps to #wellbeing #youngpeople #CAMHS #COVID19 #mentalhealth. Thank you @thelincolnite,Our monthly column with @thelincolnite is out.
This time one of our #PeerSupportWorkers talks about her 5 steps to #wellbeing #youngpeople #CAMHS #COVID19 #mentalhealth
2020-04-30 03:17:43,luisfranco_2014,True,False,RT @Ojo_Publico: #OjoRegional Hace una semana la comunidad de Pucacuro recibió la visita del alcalde y trabajadores de Villa Trompeteros qu…,
2020-04-30 03:17:43,juanpedrosala10,True,False,RT @MarkoCortes: En reunión de la Comisión Permanente de @AccionNacional, analizamos las propuestas del partido, durante la pandemia de #CO…,
2020-04-30 03:17:43,UrbanikMarysel,True,False,RT @thauserkstp: MN Health Department says 99.24% of all MN Covid-19 deaths have occurred in long-term care facilities or among people with…,
2020-04-30 03:17:43,real_sleepy_K,True,False,RT @reportmainz: Die #Corona-Krise könnte die Situation am #Ausbildungsmarkt deutlich verschärfen. Das geht aus einer internen Umfrage des…,
2020-04-30 03:17:43,MANNOR01,True,False,RT @TMZ: Beyonce Remixes Megan Thee Stallion's Hit 'Savage' for COVID-19 Relief https://t.co/vZOZAAD6jD,
2020-04-30 03:17:43,AbhishekSable,False,False,Happy Birthday @ImRo45 #Hitman

Analysis of data/Implementation/Results:

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

UserTweetsCount.java

```
13
14    public static class Map extends Mapper<LongWritable, Text, Text,
15 IntWritable> {
16
17        private Text user = new Text();
18        private final static IntWritable one = new IntWritable(1);
19        public void map(LongWritable key, Text value, Context context )
20 throws IOException, InterruptedException {
21            String line = value.toString();
22            String str[]=line.split(",");
23            if(str.length>1){
24                 user.set(str[1]);
25            }
26       context.write(user, one);
27       }
28
29    }
30
31    public static class Reduce extends Reducer<Text, IntWritable,
32 Text, IntWritable> {
33
34        public void reduce(Text key, Iterable<IntWritable> values,
35 Context context)
36          throws IOException, InterruptedException {
37            int sum = 0;
38            int l=0;
39            for (IntWritable val : values) {
40                    l+=1;
41                sum += val.get();
42            }
43            sum=sum/l;
44            context.write(key, new IntWritable(sum));
45        }
46    }
47
48    public static void main(String[] args) throws Exception {
49       Configuration conf = new Configuration();
50
51            @SuppressWarnings("deprecation")
52                 Job job = new Job(conf, "UserCount");
53            job.setJarByClass(UserTweetsCount.class);
54
55            job.setMapOutputKeyClass(Text.class);
56            job.setMapOutputValueClass(IntWritable.class);
57
58        job.setOutputKeyClass(Text.class);
59        job.setOutputValueClass(IntWritable.class);
```

Starting HDFS namenode and datanodes:

```
lohitha@ubuntu:~/hadoop/hadoop-2.8.1/sbin$ start-dfs.sh
20/04/14 20:23:28 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to /home/lohitha/hadoop/hadoop-2.8.1/logs/hadoop-lohitha-namenode-ubuntu.out
localhost: starting datanode, logging to /home/lohitha/hadoop/hadoop-2.8.1/logs/hadoop-lohitha-datanode-ubuntu.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /home/lohitha/hadoop/hadoop-2.8.1/logs/hadoop-lohitha-secondarynamenode-ubuntu.out
20/04/14 20:23:55 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
lohitha@ubuntu:~/hadoop/hadoop-2.8.1/sbin$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/lohitha/hadoop/hadoop-2.8.1/logs/yarn-lohitha-resourcemanager-ubuntu.out
localhost: starting nodemanager, logging to /home/lohitha/hadoop/hadoop-2.8.1/logs/yarn-lohitha-nodemanager-ubuntu.out
```

Loading input data from local to HDFS:

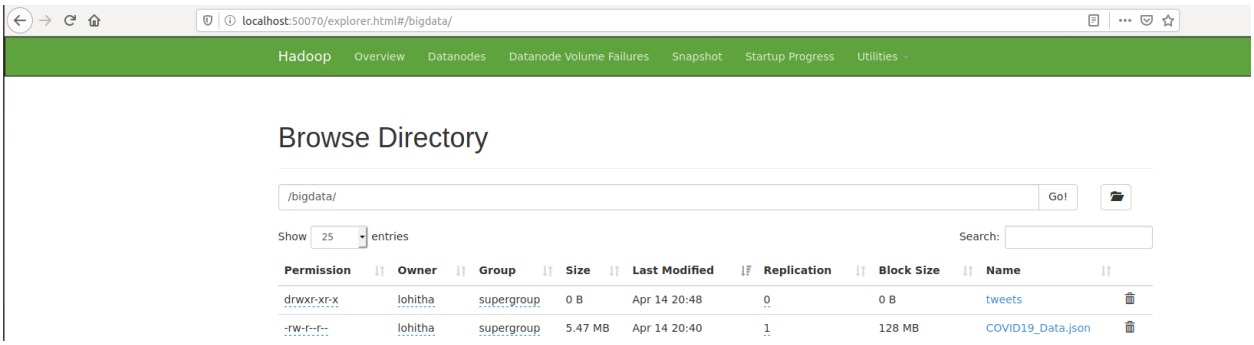
```
lohitha@ubuntu:~/Desktop/BigDataProgramming$ hdfs dfs -put COVID19_Data.json /bigdata/
20/04/14 20:40:03 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

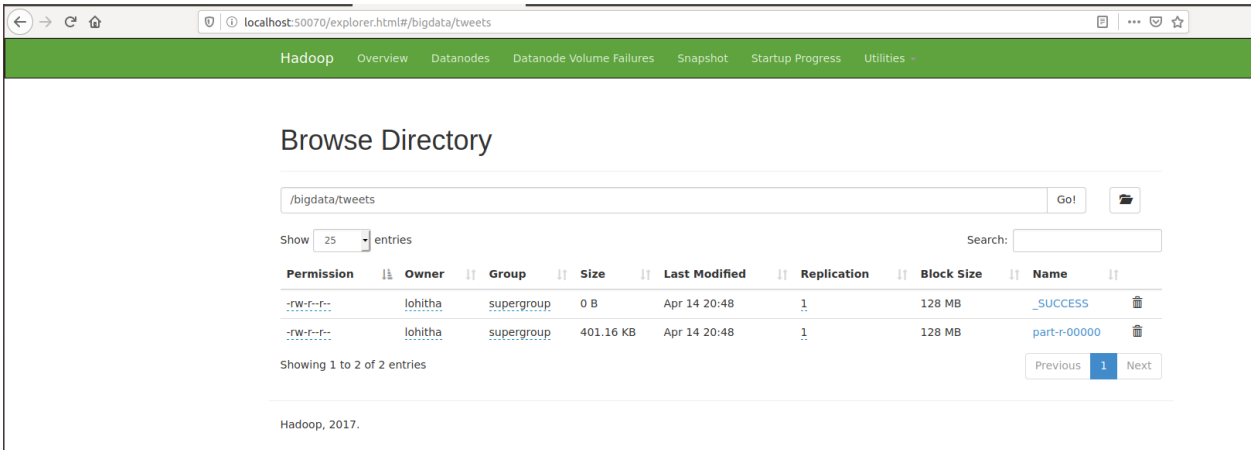
Running MapReduce:

```
lohitha@ubuntu:~/Downloads$ hadoop jar UserCount.jar UserTweetsCount /bigdata/COVID19_Data.json /bigdata/tweets/
20/04/14 20:47:56 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
20/04/14 20:47:58 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
20/04/14 20:47:59 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
20/04/14 20:48:00 INFO input.FileInputFormat: Total input files to process : 1
20/04/14 20:48:00 WARN hdfs.DataStreamer: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1252)
        at java.lang.Thread.join(Thread.java:1326)
        at org.apache.hadoop.hdfs.DataStreamer.closeResponder(DataStreamer.java:927)
        at org.apache.hadoop.hdfs.DataStreamer.endBlock(DataStreamer.java:578)
        at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:755)
20/04/14 20:48:00 WARN hdfs.DataStreamer: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1252)
        at java.lang.Thread.join(Thread.java:1326)
        at org.apache.hadoop.hdfs.DataStreamer.closeResponder(DataStreamer.java:927)
        at org.apache.hadoop.hdfs.DataStreamer.endBlock(DataStreamer.java:578)
        at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:755)
20/04/14 20:48:00 INFO mapreduce.JobSubmitter: number of splits:1
20/04/14 20:48:00 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1586921057152_0003
20/04/14 20:48:01 INFO impl.YarnClientImpl: Submitted application application_1586921057152_0003
20/04/14 20:48:02 INFO mapreduce.Job: The url to track the job: http://ubuntu:8088/proxy/application_1586921057152_0003/
20/04/14 20:48:02 INFO mapreduce.Job: Running job: job_1586921057152_0003
20/04/14 20:48:17 INFO mapreduce.Job: Job job_1586921057152_0003 running in uber mode : false
20/04/14 20:48:17 INFO mapreduce.Job:  map 0% reduce 0%
20/04/14 20:48:32 INFO mapreduce.Job:  map 100% reduce 0%
20/04/14 20:48:47 INFO mapreduce.Job:  map 100% reduce 100%
20/04/14 20:48:50 INFO mapreduce.Job: Job job_1586921057152_0003 completed successfully
20/04/14 20:48:50 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=1172059
                FILE: Number of bytes written=2617277
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=5741057
                HDFS: Number of bytes written=410786
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
```

```
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=11966
                Total time spent by all reduces in occupied slots (ms)=12023
                Total time spent by all map tasks (ms)=11966
                Total time spent by all reduce tasks (ms)=12023
                Total vcore-milliseconds taken by all map tasks=11966
                Total vcore-milliseconds taken by all reduce tasks=12023
                Total megabyte-milliseconds taken by all map tasks=12253184
                Total megabyte-milliseconds taken by all reduce tasks=12311552
        Map-Reduce Framework
                Map input records=52473
                Map output records=52473
                Map output bytes=1066546
                Map output materialized bytes=1172059
                Input split bytes=112
                Combine input records=0
                Combine output records=0
                Reduce input groups=24116
                Reduce shuffle bytes=1172059
                Reduce input records=52473
                Reduce output records=24116
                Spilled Records=104946
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=216
                CPU time spent (ms)=3330
                Physical memory (bytes) snapshot=302473216
                Virtual memory (bytes) snapshot=3893735424
                Total committed heap usage (bytes)=170004480
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=5740945
        File Output Format Counters
                Bytes Written=410786
```

HDFS File Structure:





Output File:

```
lohitha@ubuntu:~/Downloads$ hdfs dfs -cat /bigdata/tweets/part-r-00000 | head
20/04/14 22:29:53 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
        1
        1
  111 se han recuper…   1
  2 NAVOLATO    1
  Gue kirimi buat Makan selama pedemi Virus Corona      1
  aseguran que hay varias muertes clasificadas como “neumonia atipica” con todas las caracteristicas de ser #coronavirus. https://t.co/vpe0pZYfLM       1
  very inspirational. This drug is a game changer.      1
  where we are measured by our humanity not valued in silver or other coin.     1
  देश के मजदूर ◌ंक◌ो🙏N    1
 "not responsible" and put all the responsibility on the States. The States then accepted the challenge and sourced their own supplies. NOW THIS???     1
cat: Unable to write to output stream.
lohitha@ubuntu:~/Downloads$
```

Use Case 2: Data analysis of tweets using Hive

Create Tweets Table:

```
hive> create table tweets (tweet_date STRING,user STRING,is_retweet STRING,is_qu
ote STRING,text STRING,qouted_text STRING)
    > [cloudera@quickstart ~]$ hive

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.p
roperties
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive> create table tweets (tweet_date STRING,user STRING,is_retweet STRING,is_qu
ote STRING,text STRING,qouted_text STRING)
    > row format delimited fields terminated by "," stored as textfile;
OK
Time taken: 1.493 seconds
```

Load Twitter Data into Tweets table:

```
hive> load data local inpath "/home/cloudera/Downloads/COVID19_Data.csv" into ta
ble tweets;
Loading data to table default.tweets
Table default.tweets stats: [numFiles=1, totalSize=5740945]
OK
Time taken: 1.58 seconds
```

Query 1: Fetch top users with more number of tweets

```
hive> select user,count(1) as tweet_count from tweets where text is not null gro
up by user order by tweet_count desc limit 10;
Query ID = cloudera_20200414212424_9bed624d-dab4-488a-891d-0dcc5d58ce86
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0004, Tracking URL = http://quickstart.cloudera
:8088/proxy/application_1586916751037_0004/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0004
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-04-14 21:24:29,303 Stage-1 map = 0%,  reduce = 0%
2020-04-14 21:24:49,156 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 5.51 sec
2020-04-14 21:25:10,351 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 11.04 sec
MapReduce Total cumulative CPU time: 11 seconds 40 msec
Ended Job = job_1586916751037_0004
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0005, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1586916751037_0005/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0005
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2020-04-14 21:25:30,720 Stage-2 map = 0%,  reduce = 0%
2020-04-14 21:25:49,266 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 5.59 sec
2020-04-14 21:26:06,936 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 9.05 sec
MapReduce Total cumulative CPU time: 9 seconds 50 msec
Ended Job = job_1586916751037_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 11.04 sec   HDFS Read: 5748913 HDFS Write: 690474 SUCCESS
Stage-Stage-2: Map: 1  Reduce: 1   Cumulative CPU: 9.05 sec   HDFS Read: 695482 HDFS Write: 201 SUCCESS
Total MapReduce CPU Time Spent: 20 seconds 90 msec
OK
openletterbot   55
 Dy CM  26
HollyDeinert    21
syedsalu11      20
paragpa10920537 17
Em coletiva de imprensa realizada no @Planalto  16
JPNicholasDabot 15
```

```
OK
openletterbot   55
 Dy CM  26
HollyDeinert    21
syedsalu11      20
paragpa10920537 17
Em coletiva de imprensa realizada no @Planalto  16
JPNicholasDabot 15
RDSharm91052874 13
rogerperadelles 13
VinayaKantRai2  12
Time taken: 118.462 seconds, Fetched: 10 row(s)
hive>
```

Query 2: Retweet count

```
hive> select count(1) retweet_count from tweets where trim(lower(is_retweet)) = "true";
Query ID = cloudera_20200414213232_bfebce0c-70ca-48e4-89ed-2f15680b54ef
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0008, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1586916751037_0008/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0008
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-04-14 21:32:37,284 Stage-1 map = 0%,  reduce = 0%
2020-04-14 21:32:57,614 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 4.85 sec
2020-04-14 21:33:13,541 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.44 sec
MapReduce Total cumulative CPU time: 7 seconds 440 msec
Ended Job = job_1586916751037_0008
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.44 sec   HDFS Read: 5749863 HDFS Write: 6 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 440 msec
OK
19731
Time taken: 54.324 seconds, Fetched: 1 row(s)
hive>
```

Query 3: Tweets per minute

```
hive> ( select count(1) tweet_count from tweets where trim(lower(tweet_date)) like '%2020-04-15 02:07%';
NoViableAltException(290@[])
        at org.apache.hadoop.hive.ql.parse.HiveParser.statement(HiveParser.java:1028)
        at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:201)
        at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:166)
        at org.apache.hadoop.hive.ql.Driver.compile(Driver.java:522)
        at org.apache.hadoop.hive.ql.Driver.compileInternal(Driver.java:1356)
        at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1473)
        at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1285)
        at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1275)
        at org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:226)
        at org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:175)
        at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:389)
        at org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:781)
        at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:699)
        at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:634)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
FAILED: ParseException line 1:0 cannot recognize input near '(' 'select' 'count'
hive> select count(1) tweet_count from tweets where trim(lower(tweet_date)) like '%2020-04-15 02:07%';
Query ID = cloudera_20200414213939_1c2210e7-c840-431f-a32e-b2eb35171d20
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0009, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1586916751037_0009/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0009
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-04-14 21:39:45,329 Stage-1 map = 0%,  reduce = 0%
2020-04-14 21:40:04,322 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 4.46 sec
2020-04-14 21:40:21,539 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.14 sec
MapReduce Total cumulative CPU time: 7 seconds 140 msec
Ended Job = job_1586916751037_0009
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.14 sec   HDFS Read: 5749880 HDFS Write: 5 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 140 msec
OK
3017
Time taken: 57.374 seconds, Fetched: 1 row(s)
hive>
```

Query 4: Tweets on Layoffs

```
hive> select count(1) tweet_count from tweets where trim(lower(text)) like '%layoff%';
Query ID = cloudera_20200414214141_9c3cbfd4-11d9-416d-b135-e51dbd2875f3
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0010, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1586916751037_0010/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0010
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-04-14 21:41:58,116 Stage-1 map = 0%,  reduce = 0%
2020-04-14 21:42:15,545 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 4.7 sec
2020-04-14 21:42:36,210 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.93 sec
MapReduce Total cumulative CPU time: 7 seconds 930 msec
Ended Job = job_1586916751037_0010
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.93 sec   HDFS Read: 5749863 HDFS Write: 2 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 930 msec
OK
2
Time taken: 59.565 seconds, Fetched: 1 row(s)
hive>
```

Query 5: Tweets on USA

```
hive> select count(1) tweet_count from tweets where trim(lower(text)) like '%usa%' or trim(lower(text)) like '%united states%';
Query ID = cloudera_20200414214545_ba40e4d2-1995-4b17-8f46-d647f76cdf95
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1586916751037_0011, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1586916751037_0011/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1586916751037_0011
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-04-14 21:45:46,527 Stage-1 map = 0%,  reduce = 0%
2020-04-14 21:46:02,673 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 4.32 sec
2020-04-14 21:46:19,715 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.03 sec
MapReduce Total cumulative CPU time: 7 seconds 30 msec
Ended Job = job_1586916751037_0011
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.03 sec   HDFS Read: 5750249 HDFS Write: 4 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 30 msec
OK
417
Time taken: 52.837 seconds, Fetched: 1 row(s)
hive>
```

Use Case 3: Twitter data sentimental analysis using Map Reduce

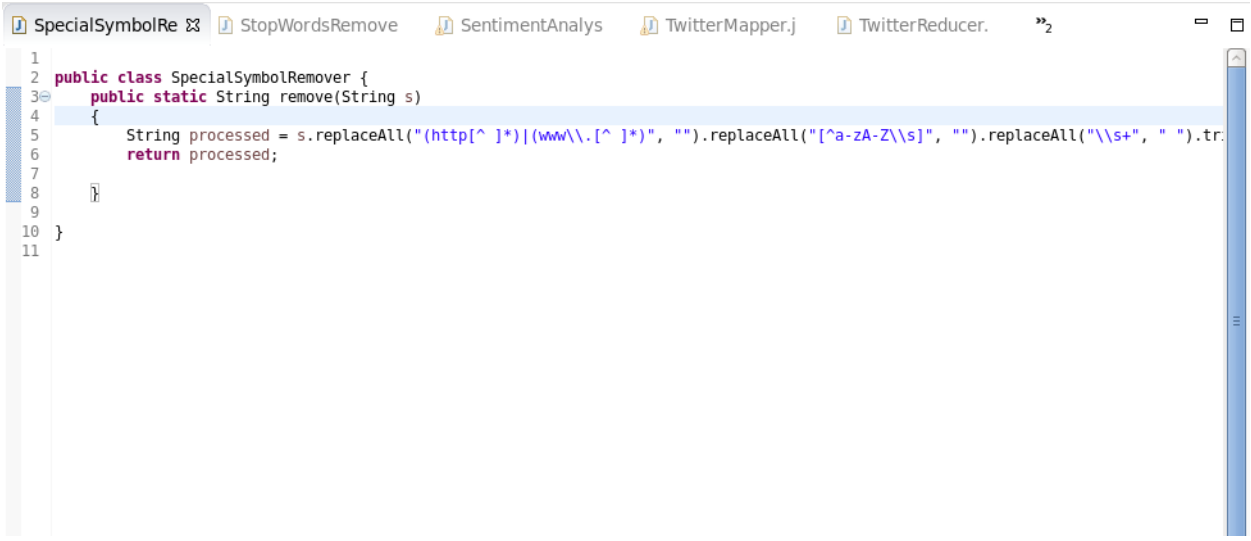
Analyzed the sensitivity of tweets. Divided tweets into 3 categories- positive, negative and neutral and displayed count for each category.

Data: Loaded input file to HDFS and displayed

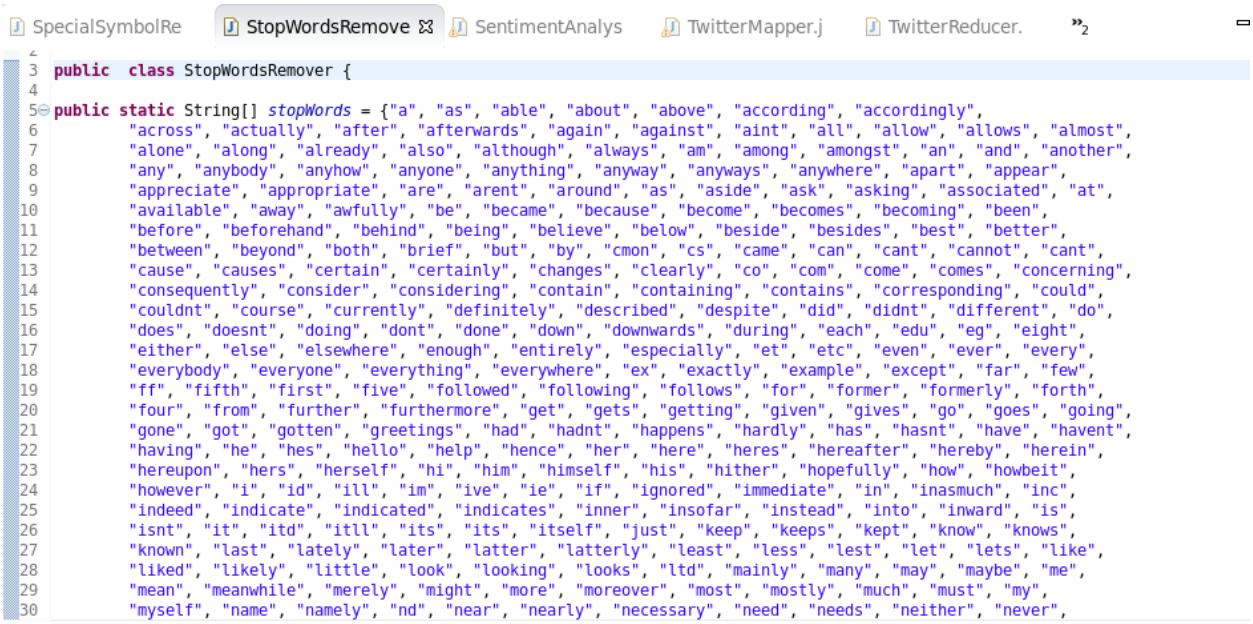
```
[cloudera@quickstart ~]$ hdfs dfs -cat sentimentanalysis/input/COVID19Data.txt |head
date,user,is_retweet,is_quote,text,quoted_text
2020-04-30 03:17:42,villafranca_o_0,True,False,RT @alegrant5: Pregunta tal vez tonta: ningún político se ha contagiado con #covid19???
2020-04-30 03:17:42,HappyHobbit4,True,False,RT @NanaJudie8484: @FredfromFlorida @GovRonDeSantis Now this is the best Governor ever!

please all Take note...
this is how our GOP
@Gov…,
2020-04-30 03:17:42,antinashit,True,False,RT @anavas4: Congresistas estadounidenses instan a Bukele a no usar COVID-19 como pretexto para socavar la Constitución https
//t.co/bMkltj…,
2020-04-30 03:17:42,AKPoliticalBeat,False,False,Top story: 'A Terrible Price': The Deadly Racial Disparities of Covid-19 in America https://t.co/dFpzyO1Rqc, see more h
ttps://t.co/mUz6VxXL87,
2020-04-30 03:17:42,gromero2019,True,False,RT @rosaicela_: El uso de cubrebocas puede disminuir las probabilidades de contagio por #COVID19, si necesitas salir no olvi
es colocarlo d…,
cat: Unable to write to output stream.
[cloudera@quickstart ~]$
```

Code:

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



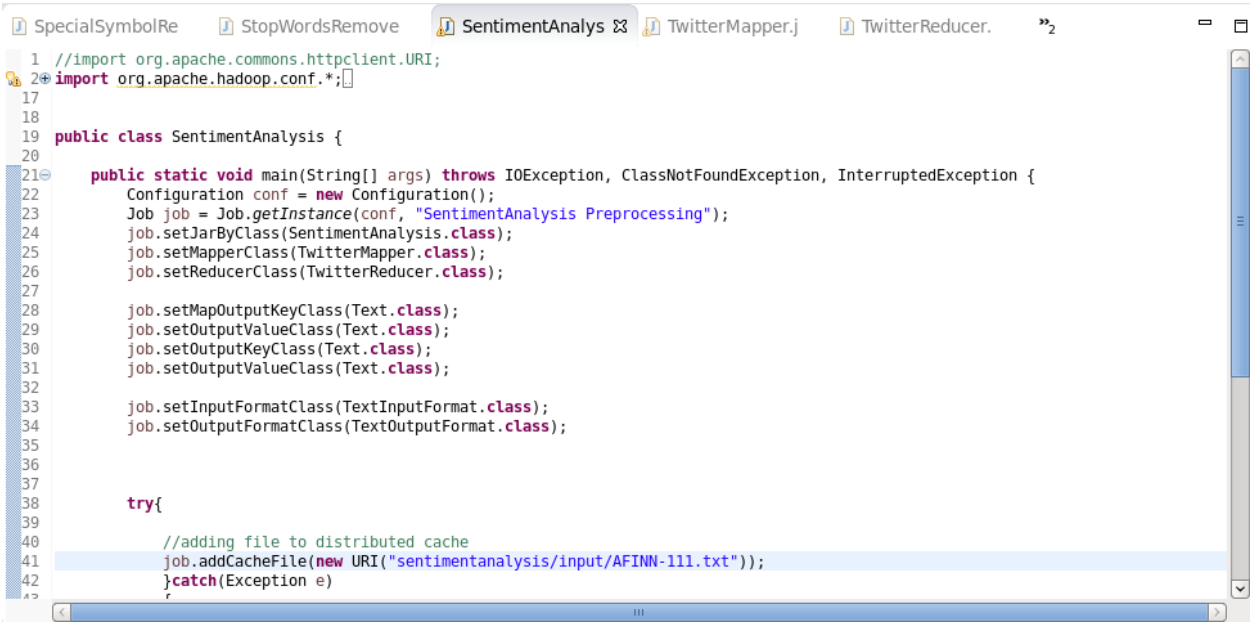
Created class- StopWordsRemover for data preprocessing to remove stop words from data. Stop words like- a, to, pronouns are removed as they do not define tweet sensitivity.




```
public  class StopWordsRemover {

public static String[] stopWords = {"a", "as", "able", "about", "above", "according", "accordingly",
        "across", "actually", "after", "afterwards", "again", "against", "aint", "all", "allow", "allows", "almost",
        "alone", "along", "already", "also", "although", "always", "am", "among", "amongst", "an", "and", "another",
        "any", "anybody", "anyhow", "anyone", "anything", "anyway", "anyways", "anywhere", "apart", "appear",
        "appreciate", "appropriate", "are", "arent", "around", "as", "aside", "ask", "asking", "associated", "at",
        "available", "away", "awfully", "be", "became", "because", "become", "becomes", "becoming", "been",
        "before", "beforehand", "behind", "being", "believe", "below", "beside", "besides", "best", "better",
        "between", "beyond", "both", "brief", "but", "by", "cmon", "cs", "came", "can", "cant", "cannot", "cant",
        "cause", "causes", "certain", "certainly", "changes", "clearly", "co", "com", "come", "comes", "concerning",
        "consequently", "consider", "considering", "contain", "containing", "contains", "corresponding", "could",
        "couldnt", "course", "currently", "definitely", "described", "despite", "did", "didnt", "different", "do",
        "does", "doesnt", "doing", "dont", "done", "down", "downwards", "during", "each", "edu", "eg", "eight",
        "either", "else", "elsewhere", "enough", "entirely", "especially", "et", "etc", "even", "ever", "every",
        "everybody", "everyone", "everything", "everywhere", "ex", "exactly", "example", "except", "far", "few",
        "ff", "fifth", "first", "five", "followed", "following", "follows", "for", "former", "formerly", "forth",
        "four", "from", "further", "furthermore", "get", "gets", "getting", "given", "gives", "go", "goes", "going",
        "gone", "got", "gotten", "greetings", "had", "hadnt", "happens", "hardly", "has", "hasnt", "have", "havent",
        "having", "he", "hes", "hello", "help", "hence", "her", "here", "heres", "hereafter", "hereby", "herein",
        "hereupon", "hers", "herself", "hi", "him", "himself", "his", "hither", "hopefully", "how", "howbeit",
        "however", "i", "id", "ill", "im", "ive", "ie", "if", "ignored", "immediate", "in", "inasmuch", "inc",
        "indeed", "indicate", "indicated", "indicates", "inner", "insofar", "instead", "into", "inward", "is",
        "isnt", "it", "itd", "itll", "its", "its", "itself", "just", "keep", "keeps", "kept", "know", "knows",
        "known", "last", "lately", "later", "latter", "latterly", "least", "less", "lest", "let", "lets", "like",
        "liked", "likely", "little", "look", "looking", "looks", "ltd", "mainly", "many", "may", "maybe", "me",
        "mean", "meanwhile", "merely", "might", "more", "moreover", "most", "mostly", "much", "must", "my",
        "myself", "name", "namely", "nd", "near", "nearly", "necessary", "need", "needs", "neither", "never",
        "nevertheless", "new", "next", "nine", "no", "nobody", "non", "none", "noone", "nor", "normally", "not",
        "nothing", "novel", "now", "nowhere", "obviously", "of", "off", "often", "oh", "ok", "okay", "old", "on",
        "once", "one", "ones", "only", "onto", "or", "other", "others", "otherwise", "ought", "our", "ours",
        "ourselves", "out", "outside", "over", "overall", "own", "particular", "particularly", "per", "perhaps",
        "placed", "please", "plus", "possible", "presumably", "probably", "provides", "que", "quite", "qv",
        "rather", "rd", "re", "really", "reasonably", "regarding", "regardless", "regards", "relatively",
        "respectively", "right", "said", "same", "saw", "say", "saying", "says", "second", "secondly", "see",
        "seeing", "seem", "seemed", "seeming", "seems", "seen", "self", "selves", "sensible", "sent", "serious",
        "seriously", "seven", "several", "shall", "she", "should", "shouldnt", "since", "six", "so", "some",
        "somebody", "somehow", "someone", "something", "sometime", "sometimes", "somewhat", "somewhere", "soon",
        "sorry", "specified", "specify", "specifying", "still", "sub", "such", "sup", "sure", "ts", "take", "taken",
        "tell", "tends", "th", "than", "thank", "thanks", "thanx", "that", "thats", "thats", "the", "their",
        "theirs", "them", "themselves", "then", "thence", "there", "theres", "thereafter", "thereby", "therefore",
        "therein", "theres", "thereupon", "these", "they", "theyd", "theyll", "theyre", "theyve", "think", "third",
        "this", "thorough", "thoroughly", "those", "though", "three", "through", "throughout", "thru", "thus", "to",
        "together", "too", "took", "toward", "towards", "tried", "tries", "truly", "try", "trying", "twice", "two",
        "un", "under", "unfortunately", "unless", "unlikely", "until", "unto", "up", "upon", "us", "use", "used",
        "useful", "uses", "using", "usually", "value", "various", "very", "via", "viz", "vs", "want", "wants",
        "was", "wasnt", "way", "we", "wed", "well", "were", "weve", "welcome", "well", "went", "were", "werent",
        "what", "whats", "whatever", "when", "whence", "whenever", "where", "wheres", "whereafter", "whereas",
        "whereby", "wherein", "whereupon", "wherever", "whether", "which", "while", "whither", "who", "whos",
        "whoever", "whole", "whom", "whose", "why", "will", "willing", "wish", "with", "within", "without", "wont",
        "wonder", "would", "would", "wouldnt", "yes", "yet", "you", "youd", "youll", "youre", "youve", "your",
        "yours", "yourself", "yourselves", "zero" };
```

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.



```
SpecialSymbolRe  StopWordsRemove  SentimentAnalys  TwitterMapper.j  TwitterReducer.
34        job.setOutputFormatClass(TextOutputFormat.class);
35
36
37
38        try{
39
40            //adding file to distributed cache
41            job.addCacheFile(new URI("sentimentanalysis/input/AFINN-111.txt"));
42            }catch(Exception e)
43            {
44                System.out.println("file not added my dear");
45                System.exit(1);
46            }
47
48
49        FileInputFormat.addInputPath(job, new Path(args[0]));
50        FileOutputFormat.setOutputPath(job, new Path(args[1]));
51
52        System.exit(job.waitForCompletion(true)?0:1);
53
54
55    }
56
57
58
59
60 }
61
```

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)

```
SpecialSymbolRe  StopWordsRemove  SentimentAnalys  *TwitterMapper.  TwitterReducer.
12 public class TwitterMapper extends Mapper<LongWritable,Text,Text,Text> {
13
14     Map<String,String> dictionary = null;
15
16
17     @Override
18     protected void setup(Context context)throws IOException,InterruptedException
19     {
20         dictionary = new HashMap<String,String>();
21
22
23
24             URI[] cacheFiles = context.getCacheFiles();
25
26             if (cacheFiles != null && cacheFiles.length > 0)
27               {
28                 try
29                 {
30                     String line ="";
31                     FileSystem fs = FileSystem.get(context.getConfiguration());
32                     Path path = new Path(cacheFiles[0].toString());
33                     BufferedReader reader = new BufferedReader(new InputStreamReader(fs.open(path)));
34
35                     while((line = reader.readLine())!=null)
36                     {
37                         String []tokens = line.split("\t");
38                         dictionary.put(tokens[0], tokens[1]);
39                     }
```

SpecialSymbolRe | StopWordsRemove | SentimentAnalys | *TwitterMapper. | TwitterReducer.

```
56    @Override
57    public void map(LongWritable key, Text value,Context context) throws IOException,InterruptedException
58    {
59        long sentiment_value = 0;
60        try{
61            String quoted_text=null,tweet_text=null;
62            String[] tweet = value.toString().split("-");
63            if(tweet.length==6)
64                quoted_text = tweet[5];
65            if(quoted_text == null && tweet.length==5)
66                tweet_text = tweet[4];
67            String processedTweet= null;
68
69
70
71
72                if(quoted_text!=null)
73                {
74
75                    processedTweet = SpecialSymbolRemover.remove(quoted_text);
76
77                    processedTweet = StopWordsRemover.remove(processedTweet);
78
79                    String []words = processedTweet.split(" ");
80
81                    for(String temp:words)
82                    {
83                        if(dictionary.containsKey(temp))
```

SpecialSymbolRe | StopWordsRemove | SentimentAnalys | *TwitterMapper. | TwitterReducer.

```
81                    for(String temp:words)
82                    {
83                        if(dictionary.containsKey(temp))
84                        {
85                            sentiment_value+=Long.parseLong(dictionary.get(temp));
86                        }
87                    }
88
89                }
90
91            else if (tweet_text!=null)
92            {
93
94                processedTweet = SpecialSymbolRemover.remove(tweet_text);
95
96                processedTweet = StopWordsRemover.remove(processedTweet);
97
98                String []words = processedTweet.split(" ");
99
100               for(String temp:words)
101               {
102                   if(dictionary.containsKey(temp))
103                   {
104                       sentiment_value+=Long.parseLong(dictionary.get(temp));
105                   }
106               }
107
108           }
```

SpecialSymbolRe | StopWordsRemove | SentimentAnalys | *TwitterMapper. | TwitterReducer.

```
                    {
                        sentiment_value+=Long.parseLong(dictionary.get(temp));
                    }
                }

            }


        if(quoted_text!=null || tweet_text!=null)
        {
            if(sentiment_value>0)
            context.write(new Text("Positive"),new Text("1"));
            else if(sentiment_value<0)
                context.write(new Text("Negative"),new Text("1"));
            else
                context.write(new Text("Neutral"),new Text("1"));

        }


        }catch(Exception e)
        {
            e.printStackTrace();
        }

    }

}
```

Reducer class- TwitterReducer

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

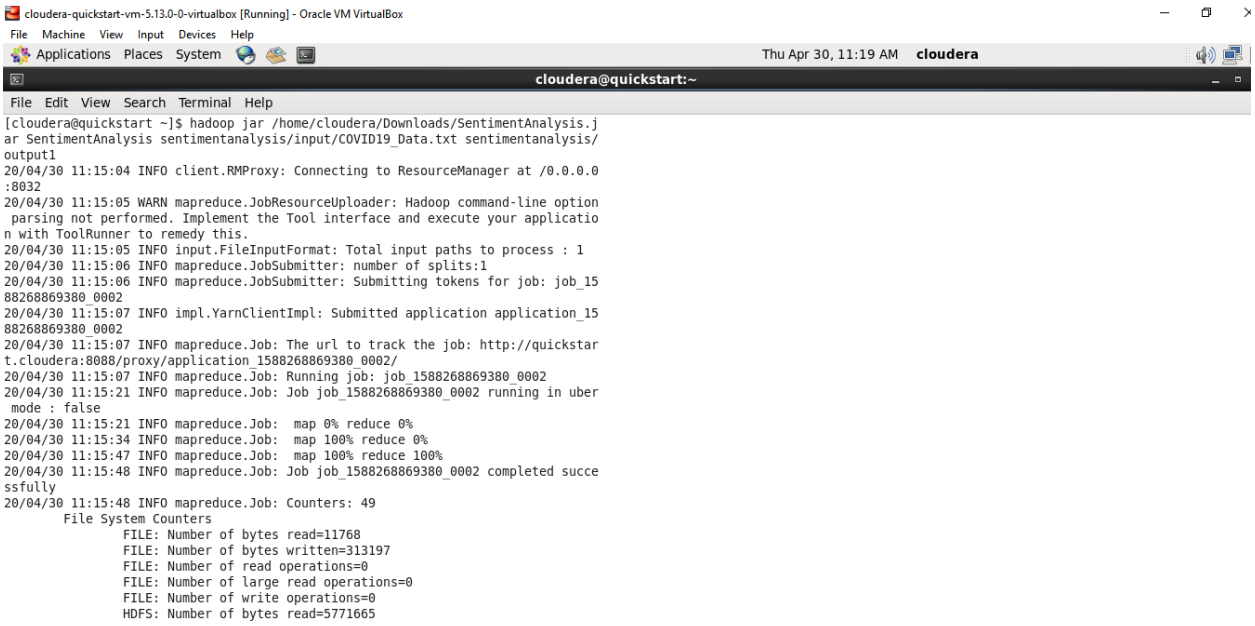
SpecialSymbolRe | StopWordsRemove | SentimentAnalys | *TwitterMapper. | TwitterReducer.

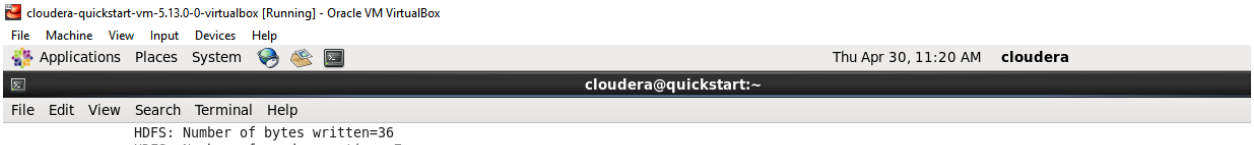
```
import java.util.HashMap;
import java.util.List;

import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.Reducer.Context;


public class TwitterReducer extends Reducer<Text, Text, Text, Text>
{

    private Text res = new Text();

    public void reduce(Text key, Iterable<Text> values, Context context) throws IOException, InterruptedException
    {
        int counter = 0;

        for (Text val : values)
        {
           counter = counter + 1;
        }

        context.write(key, new Text(String.valueOf(counter))); //Generating the reduced output.
    }
}
```

AFINN-111.txt

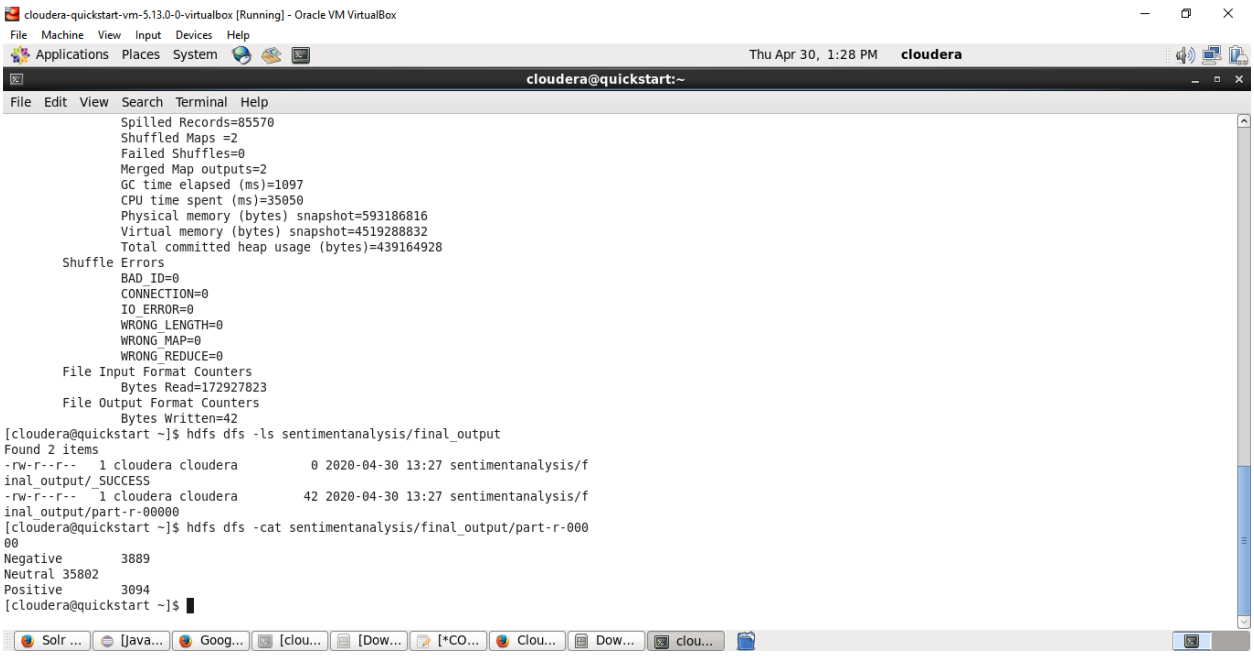
```
worrying    -3
worse   -3
worsen  -3
worsened    -3
worsening   -3
worsens -3
worshiped   3
worst   -3
worth   2
worthless   -2
worthy  2
wow 4
wowow   4
wowww   4
wrathful    -3
wreck   -2
wrong   -2
wronged -2
wtf -4
yeah    1
yearning    1
yeees   2
yes 1
youthful    2
yucky   -2
yummy   3
zealot  -2
zealots -2
zealous 2
```

Output:






```
                HDFS: Number of bytes written=36
                HDFS: Number of read operations=7
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=11276
                Total time spent by all reduces in occupied slots (ms)=11046
                Total time spent by all map tasks (ms)=11276
                Total time spent by all reduce tasks (ms)=11046
                Total vcore-milliseconds taken by all map tasks=11276
                Total vcore-milliseconds taken by all reduce tasks=11046
                Total megabyte-milliseconds taken by all map tasks=11546624
                Total megabyte-milliseconds taken by all reduce tasks=11311104
        Map-Reduce Framework
                Map input records=52473
                Map output records=965
                Map output bytes=9832
                Map output materialized bytes=11768
                Input split bytes=151
                Combine input records=0
                Combine output records=0
                Reduce input groups=3
                Reduce shuffle bytes=11768
                Reduce input records=965
                Reduce output records=3
                Spilled Records=1930
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1



                Map output materialized bytes=11768
                Input split bytes=151
                Combine input records=0
                Combine output records=0
                Reduce input groups=3
                Reduce shuffle bytes=11768
                Reduce input records=965
                Reduce output records=3
                Spilled Records=1930
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=288
                CPU time spent (ms)=4550
                Physical memory (bytes) snapshot=352935936
                Virtual memory (bytes) snapshot=3016089600
                Total committed heap usage (bytes)=226365440
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=5740945
        File Output Format Counters
                Bytes Written=36
```



Use Case 4: Twitter data sentimental analysis using Cassandra

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

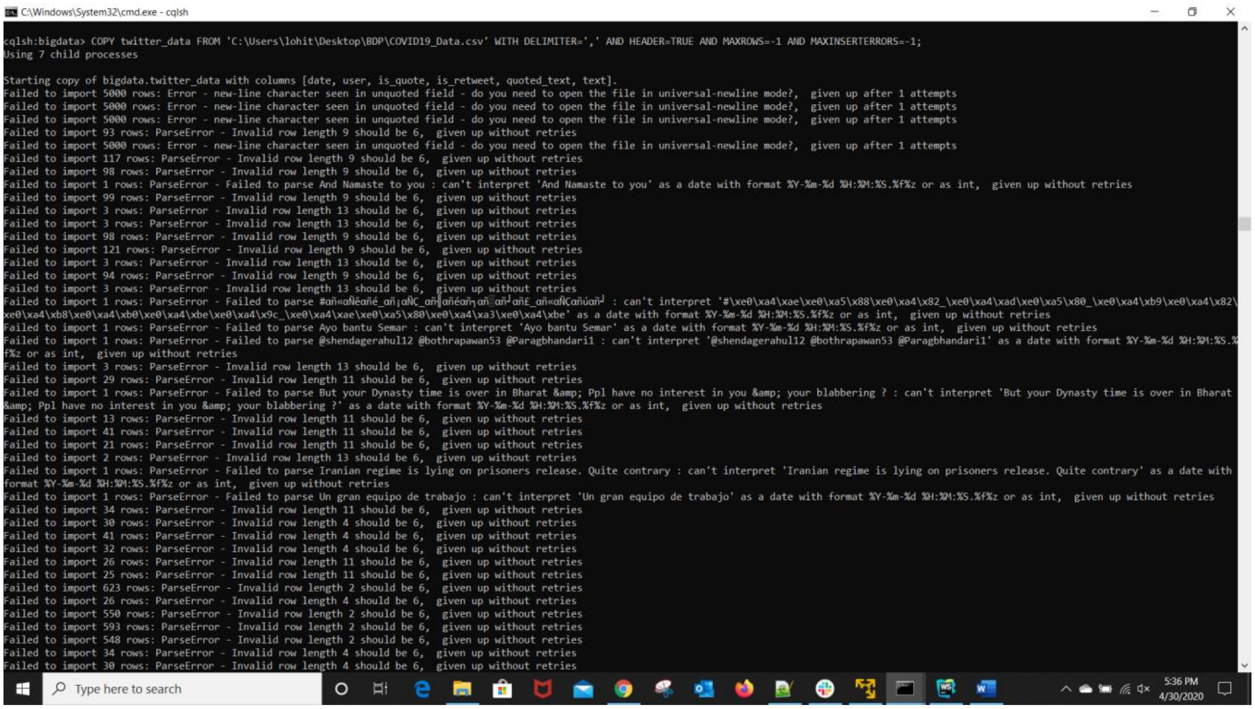
Create Table:

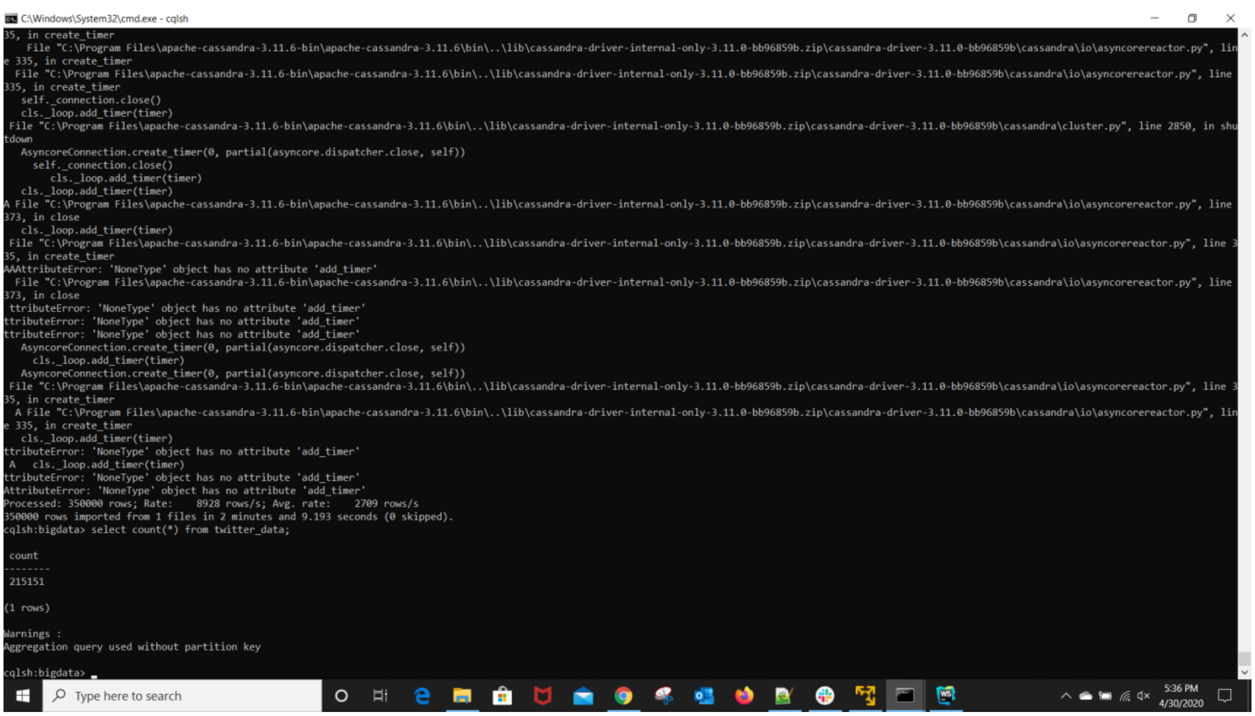
Twitter_data table is created with 6 columns- date, user, is_retweet, is_quote, text, quoted_text. Primary key is a composite key with columns (date,user).

```
cqlsh:bigdata> CREATE TABLE twitter_data(
           ...     date timestamp,
           ...     user varchar,
           ...     is_retweet boolean,
           ...     is_quote boolean,
           ...     text text,
           ...     quoted_text text,
           ...     PRIMARY KEY (date,user)
           ...     );
```

Load Twitter data:

COVID19 csv file is loaded into twitter_data table with more than 2 lakh records.

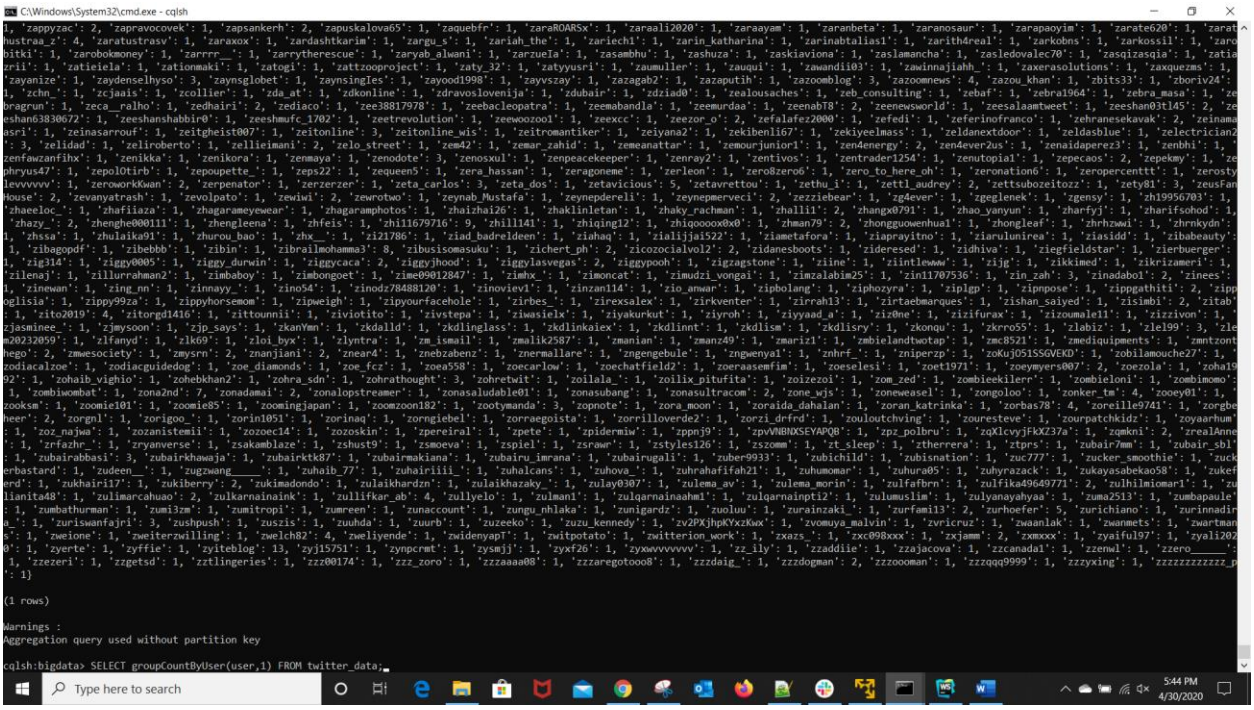




Analysis 1: Display the users and their tweet count

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.

```
cqlsh:bigdata> CREATE OR REPLACE FUNCTION cumulateCounter(state map<varchar,bigint>, user varchar, count counter)
          ... RETURNS NULL ON NULL INPUT
          ... RETURNS map<varchar,bigint>
          ... LANGUAGE java
          ... AS '
          ... if(state.containsKey(user)) {
          ...   state.put(user, state.get(user) + count);
          ... } else {
          ...   state.put(user, count);
          ... }
          ... return state;
          ... ';
cqlsh:bigdata> CREATE OR REPLACE AGGREGATE groupCountByUser(varchar, counter)
          ... SFUNC cumulateCounter
          ... STYPE map<varchar,bigint>
          ... INITCOND {};
cqlsh:bigdata>
```



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.

```
cqlsh:bigdata> CREATE OR REPLACE FUNCTION cumulateCounter2(state map<boolean,int>, is_retweet boolean)
           ... RETURNS NULL ON NULL INPUT
           ... RETURNS map<boolean,int>
           ... LANGUAGE java
           ... AS '
           ... if(state.containsKey(is_retweet)) {
           ...   state.put(is_retweet, state.get(is_retweet) + 1);
           ... } else {
           ...   state.put(is_retweet, 1);
           ... }
           ... return state;
           ... ';
cqlsh:bigdata> CREATE OR REPLACE AGGREGATE groupCountByRetweet(boolean)
           ... SFUNC cumulateCounter2
           ... STYPE map<boolean,int>
           ... INITCOND {};
cqlsh:bigdata> SELECT groupCountByRetweet(is_retweet)
           ... FROM twitter_data;

 bigdata.groupcountbyretweet(is_retweet)
-----------------------------------------
            {False: 179528, True: 35623}

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:bigdata>
```

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

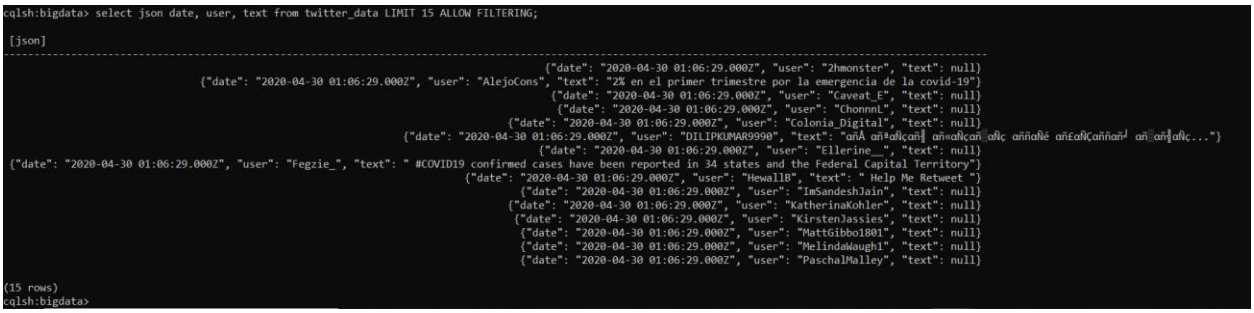
Cassandra directly does not support LIKE operator. To display user tweets whose username starts with 'L', SASI index is created on user column.

```
cqlsh:bigdata> DROP INDEX user_idx;
cqlsh:bigdata> CREATE CUSTOM INDEX user_idx ON twitter_data (user) USING
           ... 'org.apache.cassandra.index.sasi.SASIIndex' WITH
           ... OPTIONS = {'analyzer_class': 'org.apache.cassandra.index.sasi.analyzer.StandardAnalyzer', 'case_sensitive': 'false'};

Warnings :
SASI indexes are experimental and are not recommended for production use.

cqlsh:bigdata> SELECT user, text FROM twitter_data WHERE user LIKE 'L%' LIMIT 15;

 user            | text
-----------------+------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------

       Liberbot_ |
                                                                          null
   LiekoIchihara |
                                                                          null
       LauraRizo |
                                                                          null
      Linomopeko |
                                                                          null
     LauraCalvoA |                                                                          La Secretar├¡a General de Industria y Pyme @IndustriaGob ofrece formaci├│n y asesoramiento a #
pymes a trav├®s de EOI para contrarrestar los efectos COVID-19.
        LoriSums |
                                                                          null
       Lartyfat1 |
                                                                          null
     Lloydfinger |
                                                                          null
      Lerato_Kea |
                                                                          null
      LDPhilippe | 4./ #Covid19 que ┬½ la vitesse de r├®action des #Chinois dans la gestion des #├®pid├®mies a ├®t├® stup├®fiante... que la #chloroquine ... peut-├¬tre le meilleur #traitement des #coronavirus et
 la meilleure #pr├®vention ┬╗ ≡ƒôû@raoult_didier https://t.co/8x0lsKZaXn
 Lizbeth12808221 |
                                                                          null
  LordSriRama333 |
                                                                          null
     LancsSocial |
                                                                          null
        Liv3_Ron |                                                                          We were under a month long lockdown &amp; cases of Covid-19 DID NOT DECREA
SE. They INCREASED to 5000. The CURVE DID NOT FLATTEN AT ALL
    LADominatorr |
                                                                          null

(15 rows)
cqlsh:bigdata>
```

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

```
cqlsh:bigdata> select * from twitter_data where date >= '2020-04-13' LIMIT 5 ALLOW FILTERING;

 date                            | user            | is_quote | is_retweet | quoted_text
                                                                                  | text
---------------------------------+-----------------+----------+------------+------------------------------------------------------------------------------------
--------------------------------------------------------------------------------+-----------------------------------------------

 2020-04-30 01:06:29.000000+0000 |       2hmonster |    False |      False | How Milwaukee represents the COVID-19 pandemic for African Americans Race explains why cities across the country are seeing a dispropo
rtionate amount of black deaths - https://t.co/8g0LwxRCZE https://t.co/a6srL9NWwC |                                   null
 2020-04-30 01:06:29.000000+0000 |       AlejoCons |     True |      False |
          RT @publico_es: ≡ƒö┤ ├ÜLTIMA HORA | La econom├¡a espa├▒ola se desploma un 5 | 2% en el primer trimestre por la emergencia de la covid-19
 2020-04-30 01:06:29.000000+0000 |        Caveat_E |     True |      False |                                   RT @linamichi: Good news from South Korea. Looks like reco
vered #COVID19 patients testing positive again was due to RNA fragments releasedΓÇª |                                   null
 2020-04-30 01:06:29.000000+0000 |         ChonnnL |     True |      False |                                   RT @cnpnnatt: [illegible]
[illegible]
[illegible]ΓÇª |                                   null
 2020-04-30 01:06:29.000000+0000 | Colonia_Digital |    False |      False |                                   Consulta aqu├¡ el avance del Cov
id-19 en tu municipio: hay diagn├│sticos en 823 localidades https://t.co/QFS4LBDeDz |                                   null

(5 rows)
cqlsh:bigdata>
```

Query 5: Display table data in json format



Use Case 5: Twitter data sentimental analysis using Spark SQL

Load json data into SparkContext

Queries.scala | COVID19_Data1.json | COVID19_Data.csv

```scala
import org.apache.log4j._
import org.apache.spark._
import org.apache.spark.sql.SparkSession

object Queries {

  def main(args: Array[String]): Unit = {

    //Setting up the Spark Session and Spark Context
    val conf = new SparkConf().setMaster("local[2]").setAppName("Task2")
    val sc = new SparkContext(conf)
    val spark = SparkSession
      .builder()
      .appName( name = "Task2")
      .config(conf =conf)
      .getOrCreate()

    Logger.getLogger( name = "org").setLevel(Level.ERROR)
    Logger.getLogger( name = "akka").setLevel(Level.ERROR)

    // We are using all 3 Fifa dataset given on Kaggle Repository
    //a.Import the dataset and create df and print Schema

    val df1 = spark.read
      .format( source = "json")
      .option("header", "true") //reading the headers
      .option("mode", "DROPMALFORMED")
      .load( path = "dataset/COVID19_Data1.json")

```

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

```
val Q1 = spark.sql( sqlText = "select count(1) retweet_count from Covid WHERE isretweet LIKE '%True%' ")
Q1.show()
```

```
+-------------+
|retweet_count|
+-------------+
|        19964|
+-------------+
```

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

```
var Q2=spark.sql( sqlText = "select tweetuser user, count(1) tweets from Covid GROUP BY tweetuser ORDER BY tweets DESC LIMIT 5")
Q2.show();
```

```
+--------------------+------+
|                user|tweets|
+--------------------+------+
|"tweetuser":"auro...|    31|
|"tweetuser":"maxp...|    29|
|"tweetuser":"BotR...|    27|
|"tweetuser":"PHLN...|    25|
|"tweetuser":"Vala...|    16|
+--------------------+------+
```

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

```
val Q3 = spark.sql( sqlText = "select tweetuser user,count(1) quoted_tweets,isquote from Covid where isquote LIKE '%True%' " +
  "GROUP BY tweetuser,isquote ORDER BY quoted_tweets DESC LIMIT 5")
Q3.show()
```

```
+--------------------+-------------+----------------+
|                user|quoted_tweets|         isquote|
+--------------------+-------------+----------------+
|"tweetuser":"shei...|           13|"isquote":"True"|
|"tweetuser":"EndTax"|           11|"isquote":"True"|
|"tweetuser":"Rosa...|            7|"isquote":"True"|
|"tweetuser":"Eliz...|            6|"isquote":"True"|
|"tweetuser":"rwwo...|            5|"isquote":"True"|
+--------------------+-------------+----------------+
```

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

```
val Q4 = spark.sql( sqlText = "select count(1) tweets_USA from Covid WHERE lower(tweettext) LIKE '%usa%' OR" +
  " lower(tweettext) LIKE '%united states%' OR lower(tweettext) LIKE '%america%' ")
Q4.show()
```

```
+----------+
|tweets_USA|
+----------+
|       836|
+----------+
```

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