SENTIMENT ANALYSIS ON STREAMING TWEETS

Laura Soto

University of Miami
lms338@miami.edu

ABSTRACT

This project analyses the sentiment classification on tweets that share the same keyword. The implementation consists of live streaming tweets that contain the given keyword into Google Cloud Storage, to then perform data preprocessing and sentiment analysis. Finally, after comparing and applying the algorithms, the obtained results were stored and analyzed. Five different keywords were analyzed, but overall a pattern between the obtained results was identified. The datasets for each keyword were obtained on three different timeframes and had different sizes. The performance on raw data and on saved results was analyzed, as well as the computational time on different datasets sizes.

1. INTRODUCTION

Twitter is one of the most known, and most used social media platforms. Indeed, according to Twitter Usage Statistics, on average, every minute 350,000 tweets are produced, which equivalents to 500 million tweets a day. The main usage of this social media platform is to share ideas and opinions. Therefore, analyzing the sentiments behind these tweets can provide numerous knowledge. In fact, it allows exploring how a topic is overall being observed.

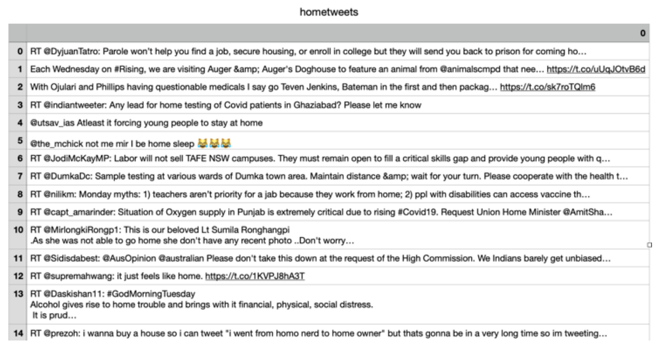
This project analyses the sentiment behind five different keywords. The initial thesis was that these keywords were clearly going to follow the predicted sentiment. In fact, the keywords used with the initial belief to have a positive sentiment behind them are 'family', 'day', and 'home'. On the other hand, the words that were predicted to have most likely a negative sentiment behind them are 'trump' and 'lockdown'.

In order to prove or contradict this initial thesis, sentiment analysis on collected datasets needed to be performed. In order to do so, accessing the Twitter API was needed, as well as stabling a Dataflow between Google Cloud Storage, and Twitter.

2. IMPORTING THE TWEETS

Twitter provides developer access, which allows people like me to perform algorithms and analysis on their data. That is why the first step to start this project was to set up the Twitter Developer account and to ensure access to the Twitter API. After this account was created, some personal access keys were provided. By using these personal keys, a simple python code was created to download a given amount of tweets that contained the given keyword. Even though a Tweet has different components like creation date, username, and others, for the objective of this project only the text of the tweet was imported and stored as a CSV file.

Initially, in order to import these datasets into the Google Cloud Storage, a manual command was inputted. By using this first approach, I was able to both create the tweets dataset and import them into a Bucket. On the other hand, this initial approach presented some challenges, since there is a restriction of around 2500 tweets that can be downloaded at a time, and the process of manually importing these tweets can be tedious. That is why overcoming these challenges was the main goal of my second and final approach.



hometweets

	0
0	RT @DyjuanTatro: Parole won't help you find a job, secure housing, or enroll in college but they will send you back to prison for coming ho…
1	Each Wednesday on #Rising, we are visiting Auger &amp; Auger's Doghouse to feature an animal from @animalscmpd that nee… https://t.co/uUqJOtvB6d
2	With Ojulari and Phillips having questionable medicals I say go Teven Jenkins, Bateman in the first and then packag… https://t.co/sk7roTQlm6
3	RT @indiantweeter: Any lead for home testing of Covid patients in Ghaziabad? Please let me know
4	@utsav_ias Atleast it forcing young people to stay at home
5	@the_mchick not me mir I be home sleep 😹😹😹
6	RT @JodiMcKayMP: Labor will not sell TAFE NSW campuses. They must remain open to fill a critical skills gap and provide young people with q…
7	RT @DumkaDc: Sample testing at various wards of Dumka town area. Maintain distance &amp; wait for your turn. Please cooperate with the health t…
8	RT @nilikm: Monday myths: 1) teachers aren't priority for a jab because they work from home; 2) ppl with disabilities can access vaccine th…
9	RT @capt_amarinder: Situation of Oxygen supply in Punjab is extremely critical due to rising #Covid19. Request Union Home Minister @AmitSha…
10	RT @MirlongkiRongp1: This is our beloved Lt Sumila Ronghangpi .As she was not able to go home she don't have any recent photo ..Don't worry…
11	RT @Sidisdabest: @AusOpinion @australian Please don't take this down at the request of the High Commission. We Indians barely get unbiased…
12	RT @supremahwang: it just feels like home. https://t.co/1KVPJ8hA3T
13	RT @Daskishan11: #GodMorningTuesday Alcohol gives rise to home trouble and brings with it financial, physical, social distress. It is prud…
14	RT @prezoh: i wanna buy a house so i can tweet "i went from homo nerd to home owner" but thats gonna be in a very long time so im tweeting…

Fig. 1. Screenshot of how the tweets were stored in the local machine before automation.

For my second and final approach, a Dataflow Job in Google Cloud was applied. This Dataflow job automates the inputting the tweets into the Google Cloud Storage. In order to do so, the initial retrieving of the tweet's python code was modified in order to push each tweet into a Pub Topic. The

Dataflow Job imports these tweets at the same time they are being exported from Twitter into the specified Bucket. The advantage of this approach is that the process of streaming tweets is automated, as well as the limit of tweets that can be downloaded.

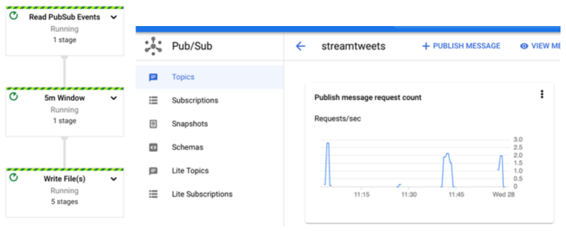
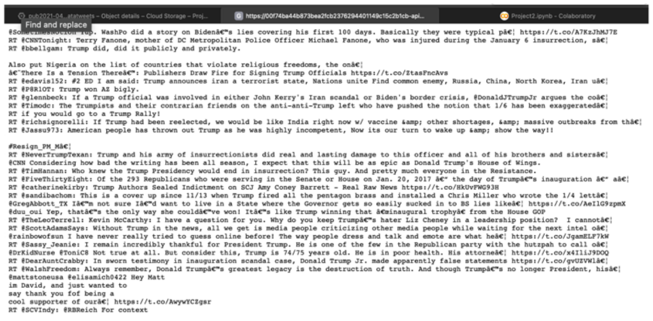


Fig. 2. Screenshot of Dataflow job on Pub/Sub, and the Pub/Sub

Finally, after automating and optimizing the streaming of the tweets, the result is different text files containing the imported tweets, into the Google Cloud Storage that are ready to be clean and analyzed.


pub2021-04...atatweets – Object details – Cloud Storage – Proj...
https://00f74ba44b873bea2fcb2376294401149c15c2b1cb-api...
Project2.ipynb - Colaboratory
Find and replace

```
@So[illegible]p. WashPo did a story on Bidenâ€™s lies covering his first 100 days. Basically they were typical pâ€¦ https://t.co/A7KzJhMJ7E
RT @CNNTonight: Terry Fanone, mother of DC Metropolitan Police Officer Michael Fanone, who was injured during the January 6 insurrection, sâ€¦
RT @bbellgam: Trump did, did it publicly and privately.

Also put Nigeria on the list of countries that violate religious freedoms, the onâ€¦
â€˜There Is a Tension Thereâ€™: Publishers Draw Fire for Signing Trump Officials https://t.co/ZtasFncAvs
RT @edavis152: #2 ED I am said: Trump announces iran a terrorist state, Nations unite Find common enemy, Russia, China, North Korea, Iran uâ€¦
RT @P8R1OT: Trump won AZ bigly.
RT @glennbeck: If a Trump official was involved in either John Kerry's Iran scandal or Biden's border crisis, @DonaldJTrumpJr argues the coâ€¦
RT @Timodc: The Trumpists and their contrarian friends on the anti-anti-Trump left who have pushed the notion that 1/6 has been exaggeratedâ€¦
RT if you would go to a Trump Rally!
RT @richsignorelli: If Trump had been reelected, we would be like India right now w/ vaccine &amp; other shortages, &amp; massive outbreaks from thâ€¦
RT @Jassu973: American people has thrown out Trump as he was highly incompetent, Now its our turn to wake up &amp; show the way!!

#Resign_PM_Mâ€¦
RT @NeverTrumpTexan: Trump and his army of insurrectionists did real and lasting damage to this officer and all of his brothers and sistersâ€¦
@CNN Considering how bad the writing has been all season, I expect that this will be as epic as Donald Trump's House of Wings.
RT @TimHannan: Who knew the Trump Presidency would end in insurrection? This guy. And pretty much everyone in the Resistance.
RT @FiveThirtyEight: Of the 293 Republicans who were serving in the Senate or House on Jan. 20, 2017 â€" the day of Trumpâ€™s inauguration â€" aâ€¦
RT @catherinekirby: Trump Authors Sealed Indictment on SCJ Amy Coney Barrett - Real Raw News https://t.co/HkUvFWG93H
RT @sandibachom: This is a cover up since 11/13 when Trump fired all the pentagon brass and installed a Chris Miller who wrote the 1/4 lettâ€¦
@GregAbbott_TX Iâ€™m not sure Iâ€™d want to live in a State where the Governor gets so easily sucked in to BS lies likeâ€¦ https://t.co/AeI1G9zpmX
@duu_oui Yep, thatâ€™s the only way she couldâ€™ve won! Itâ€™s like Trump winning that â€œinaugural trophyâ€ from the House GOP
RT @TheLeoTerrell: Kevin McCarthy: I have a question for you. Why do you keep Trumpâ€™s hater Liz Cheney in a leadership position?  I cannotâ€¦
RT @ScottAdamsSays: Without Trump in the news, all we get is media people criticizing other media people while waiting for the next intel oâ€¦
@rainbowofsun I have never really tried to guess online before! The way people dress and talk and emote are what heâ€¦ https://t.co/JgamELF7kW
RT @Sassy_Jeanie: I remain incredibly thankful for President Trump. He is one of the few in the Republican party with the hutzpah to call oâ€¦
@DrKidNurse @ToniC8 Not true at all. But consider this, Trump is 74/75 years old. He is in poor health. His attorneâ€¦ https://t.co/x4IliJ9DOQ
RT @DearAuntCrabby: In sworn testimony in inauguration scandal case, Donald Trump Jr. made apparently false statements https://t.co/gvUZVWlâ€¦
RT @WalshFreedom: Always remember, Donald Trumpâ€™s greatest legacy is the destruction of truth. And though Trumpâ€™s no longer President, hisâ€¦
@mattstoneusa @elisamich0422 Hey Matt
im David, and just wanted to
say thank you fof being a
cool supporter of ourâ€¦ https://t.co/AwywYCZgsr
RT @SCVIndy: @RBReich For context
```

Fig. 3. Screenshot of the text file containing the tweets after automation

3. DATA PREPROCESSING

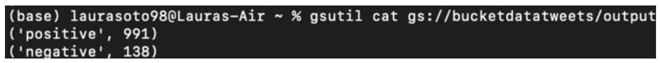
As mentioned before, the data files imported into the Google Cloud Bucket only contain the text of a tweet which is not only composed of English words. In fact, in many cases, it contains numbers, emojis, symbols, links, and others. Therefore, in order to prepare the data for a sentiment analysis algorithm all the non-alphabet and non-key words needed to be cleaned.

3.1 Data Cleaning

Because of the magnitude of many of these datasets, the most effective time and space complexity approach to clean this data is to use PySpark. A python code was created to perform this task. This code initially converts the dataset text file into an RDD, and then applies three different functions. The first function removes all the non-alphabet things on the text, therefore after applying this initial function the remaining RDD is composed only by string text. The second function removes all non-English words as well as empty spaces. Finally, after cleaning the datasets, the function distinct was applied to make sure we remove all the duplicates. As a result of these functions, the remaining RDD was composed of cleaned and ready-to-be analyzed data.

4. SENTIMENT ANALYSIS

The main objective of this project was to be able to identify the sentiment behind a tweet. In order to achieve so, the python library Textblob was used. This library contains a well-known algorithm to identify the polarity and the subjectivity of a given text.
Since most of our datasets have a large size, applying PySpark was also observed to be the best approach. The code to implement this sentiment analysis algorithm consists of initially using the cleaned RDD to perform the analysis. Moreover, Textblob was used as a function, which was mapped on each element of the RDD. As a result of this mapping, each RDD element had a polarity value. This polarity value consisted of a float number between negative one and positive one. A zero value for this polarity represents neutral so that no sentiment could be identified by the input. On the other hand, a negative value represents a negative sentiment, and a positive value represents a positive sentiment. Furthermore, a second function was applied to identify each RDD element sentiment based on the above criteria. After applying this function a reduced map function was applied to the RDD, which allowed to have as an output the count of tweets per sentiment. This output was stored as a text file in the Bucket.



```
(base) laurasoto98@Lauras-Air ~ % gsutil cat gs://bucketdatatweets/output
('positive', 991)
('negative', 138)
```

Fig. 4. Screenshot of the output stored after performing the sentiment analysis

Applying the described above algorithm was performed through a Dataproc Job which was executed using the following command.

```
gcloud dataproc jobs submit pyspark sentimentanalysis.py --cluster="cluster-
twitter" --region="us-central1"  --project="projecttwitter-311814" --
gs://bucketdatatweets/input/textfilename gs://bucketdatatweets/output/name
```

Fig. 5. Command executed to perform a Dataproc Job

5. RESULTS

For each keyword, three different sizes datasets were stored on different dates. Overall, the percentage of positive and negative sentiments through these different timeframes and sizes followed a pattern. On the other hand, the results showed that the initial prediction of the most prominent sentiment was wrong for some keywords. These results analyses were performed by importing the output of the sentiment analysis into Google Colab and using the Matplot library and are the following.

5.1 Number of Tweets Vs Sentiments

The number of tweets versus the number of sentiments varies depending on each keyword. In general, contradicting the initial hypothesis, all the keywords had more positive sentients than negative. In fact, the average proportion was 70% positive 30% negative.

5.1.1 Lockdown

The initial belief for this keyword was that the prominent sentiment was going to be negative. On the other hand, the results show that there are more positive tweets that involve this word than negative. The three initial datasets, before the data cleaning and dropping neutral values, had a size of 500, 5000, and 2000 instances.

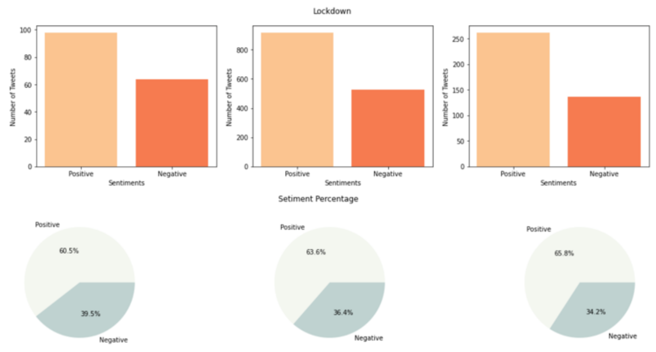


Fig. 6. Plots obtain from keyword 'Lockdown'

5.1.2 Home

This keyword had a proportion of 70% positive 30% negative, which proved the initial hypothesis. On the other hand, the still high percentage of negative sentiment when using this keyword may be caused by covid impacts during the past months. The initial datasets had a size of 5000, 3000, and 2000 tweets.

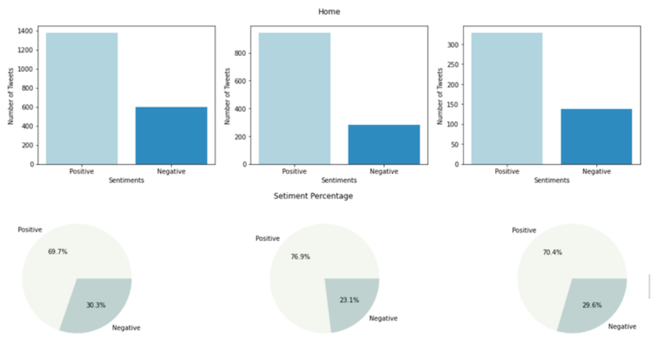


Fig. 7. Plots obtain from keyword 'Home'

5.1.3 Trump

Contrary to the belief this keyword was involved in more positive tweets than negative. On the other hand, out of the used keywords, this was the one with the highest negative percentage, with a distribution of 60% positive 40% negative. The initial dataset sizes are 3000, 5000, and 1000 respectively.

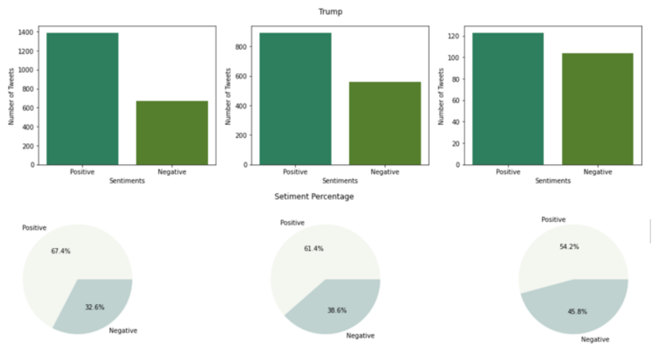


Fig. 8. Plots obtain from keyword 'Trump'

5.1.4 Family

As predicted, this keyword is most likely used on a positive tweet. Its original datasets contain 10000, 5000, and 300 tweets.

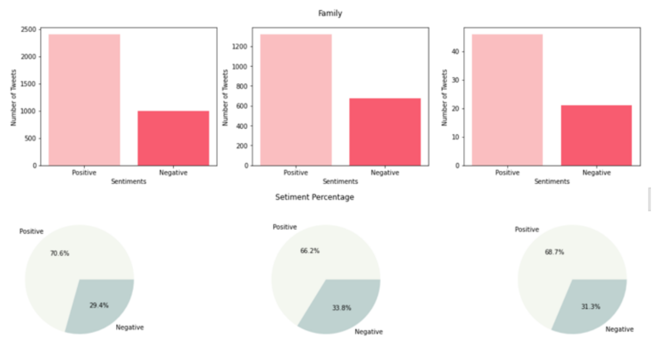


Fig. 9. Plots obtain from keyword 'Family'

5.1.5 Day

Finally, the last keyword did not only prove to be most likely used on positive tweets, but it was also the one with the highest positive rate. The initial datasets for this keyword contained 5000, 5000, and 2000 tweets.

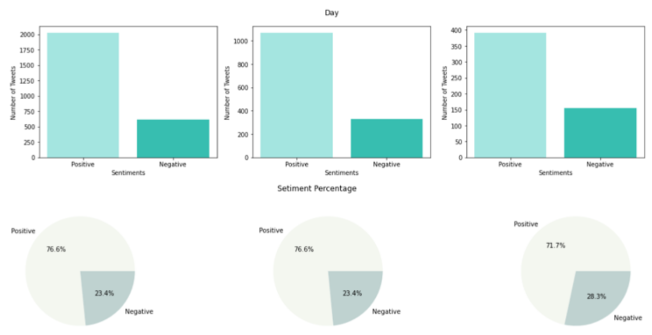


Fig. 10. Plots obtain from keyword 'Day'

5.2 Time vs Amount of Tweets

When measuring the computational time when locally downloading different sizes of datasets, we obtained the following results:

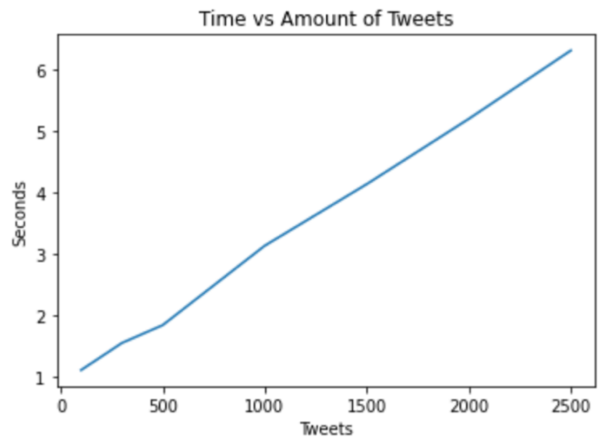
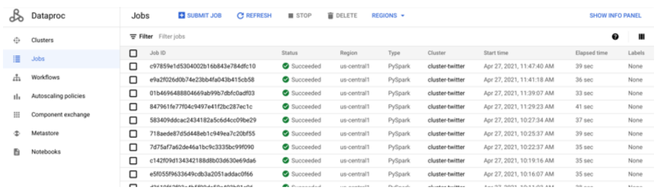


Fig. 11. Plot obtain from imputing the time vs the amount of tweets

This time measure was obtained using the command 'time' before compiling the code. What we can observe in this figure is that the more tweets we want to download, the longer it takes. Overall, it follows almost a straight line.
On the other hand, since Twitter API has a request limit, I wasn't able to download more than 2500 tweets per request locally.

5.3 Performance on Raw Data vs Saved Results

When measuring the computational time on raw data and on saved results, I used the time it took to complete the Dataproc job, which is displayed on Google Cloud Dataproc Jobs. Overall, I did not notice a big change between the performance on raw data or on saved results. In fact, the average time that it takes to completes the job is 36s no matter if it was performed on the 10.000 or the 300 datasets.



Jobs

Job ID	Status	Region	Type	Cluster	Start time	Elapsed time	Labels
c97859e1d5304002b16b843e784dfc10	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 11:47:40 AM	39 sec	None
e9a2f026d0b74e23bb4fa043b415cb58	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 11:41:18 AM	36 sec	None
01b4696488804669ab99b7dbfc0adf03	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 11:39:07 AM	33 sec	None
847961fe77f04c9497e41f2bc287ec1c	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 11:29:23 AM	41 sec	None
583409ddcac2434182a5c6d4cc09be29	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 10:27:34 AM	37 sec	None
718aede87d5d448eb1c949ea7c20bf55	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 10:25:37 AM	39 sec	None
7d75af7a62de46a1bc9c3335bc99f090	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 10:22:37 AM	35 sec	None
c142f09d134342188d8b03d630e69da6	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 10:19:16 AM	35 sec	None
e5f055f9633649cdb3a2051addac0f66	Succeeded	us-central1	PySpark	cluster-twitter	Apr 27, 2021, 10:16:07 AM	35 sec	None

6. CONCLUSION

In conclusion, this project was able to perform sentiment analysis on stream tweets. Moreover, using Dataflow and Pub to live stream tweets into a Google Cloud Storage Bucket to perform data cleaning and sentiment analysis using a PySpark code applied as a Dataproc Job. The results of these executions showed in some cases different results of what it was initially predicted. Proving that for these keyword cases the sentiment percentage proportion was on average 70%-30%, prevailing the positive tweets. On the other hand, when comparing the computational time for the different dataset sizes, and for new or already used datasets, on average they all took 36s to compute. Furthermore, as expected the more tweets are downloaded the more time it takes. Finally, this project is an example of how to optimize the performance of applying a sentiment analysis algorithm to big tweets dataset by using the Google Cloud tools, and PySpark.

7. REFERENCES

[1
https://github.com/Laurasoto98/SentimentAnalysisonStreamingTweets

[2]
https://colab.research.google.com/drive/1fo2xMNmkasUP6lxae3IDOlq0GjTQAaLv?authuser=1#scrollTo=MuuA_E6E7NZg