Singapore Covid19 Sentiment Analysis

1 Introduction

In recent days, Singapore has witnessed the escalating worldwide public health problem — COVID-19. The Singapore government has contributed a lot to control the epidemic. Twitter is one of the most popular microblog platforms on which users can express their opinions and thoughts freely. By analyzing the sentiment of tweets posted by Singapore residents about COVID-19, we can determine the trends of public opinions, according to which government can take some measures or even shift gears as soon as possible.

1.1 Objective of This Project

Our project includes several loosely coupled parts.

1. Collect Tweets data for sentiment analysis.
2. Train several deep learning models.
3. Use Spark for both sentiment analysis labeling and data visualization.
4. Analysis the result .

1.2 Report Organization

In section 2, we will talk about the method of data collection. In section 3, model training, evaluation details as well as Spark sentiment labeling will be introduced. In section 4, we will talk about the methodology we used in data visualization. Finally, in section 5 we will talk about the result analysis.

2 Data Collection

Twitter data is open to everyone and widely used in data analysing project. In this section, we will talk about the data collection method and information of the data.

Twitter Developer Account is the official approach provided by Twitter. This account offers several methods to scrap different kinds of data, such as user information, relationship, Tweets, etc. As our target is the Tweets posted by users from Singapore, we use the search method to get the metadata of the Tweets. Twitter also suggest the Tweepy as the official package from Python. The JSON format metadata can be collected by using an authentication method with the developer account token and a search method with a search query including keywords and several parameters.

To match the topic of COVID-19, we use 'covid', 'covid-19', '#covid19sg' and 'virus' as the search keywords. The function and setting of parameters are listed in Table 1 After implementing the search query, we get 84,012 pieces of Tweets in a consecutive period of seven days. The data is in JSON format with the attributes including id, time, user id, text, etc. To preprocess the data, we read it into the Spark Resilient Distributed Datasets (RDD). We perform deduplicatoin to eliminate the duplicate Tweets in different search. Finally, we retrieve the necessary attribute and data from the metadata using projection. The attribute retrieved are listed in Table 2.

Table 1. Search Parameters

Parameter	Value	Description
language	en	Select English data only.
geocode	1.31,103.83,30km	Filter the country to Singapore.
status	full_text	Get the full text.
since, until	changing	Collect data in seven days.

Table 2. Data Collection Attributes

Attributes	Description
created_at	time of creation
id	unique code for deduplication
hashtags	topics
full_text	text data
retweet_status	retweeted user and id

3 Sentiment Analysis

Sentiment analysis, also known as opinion mining or emotion AI refers to the use of natural language processing, text analysis, computational linguistics or other technologies to systematically identify, extract, quantify, and study affective states and subjective information. A basic task in sentiment analysis is classifying the polarity of a given text at the document. In common cases, negative and positive sentiment can be regarded as the poles of emotion. Quantitatively, sentiment analysis is to label a sentence with a sentiment score in the range $[0, 1]$. The closer the sentiment score is to 1, the more positive the sentiment of the text and vice versa.

In this section, we will talk about sentiment analysis model selection, model training and spark mapreduce details.

3.1 Model Training Dataset

In this project, there are two datasets used for different purposes. One is mentioned in section 2, which is the Tweets dataset we constructed used for sentiment analysis later. The other is used for deep learning model training. We used the IMDb movie review dataset for model training, which contains 25000 positive movie reviews labeled as 1 while 25000 negative movie reviews labeled as 0.

3.2 Deep Learning Model for Sentiment Analysis

Deep learning has matured in sentiment analysis recently. Commonly used deep learning model includes CNN, LSTM, CNN+LSTM and BidirectionalLSTM. The structures of the models are listed from Table 3 to Table 6. Here we only explain the structural principle details of LSTM because it is core model we used in the project while the rest are baseline for evaluation.

Table 3. Model Based on LSTM

Layer Type	Output Shape
InputLayer	500
Embedding	500×300
Dropout	500×300
LSTM	100
Dense	1

Table 4. Model Based on CNN

Layer Type	Output Shape
InputLayer	450
Embedding	450×300
Dropout	450×300
Conv1D	446×25
MaxPooling	223×25
Dropout	223×25
Conv1D	219×20
Flatten	4380
Dropout	4380
Dense	120
Dropout	120
Dense	1

All of these four models start with an "InputLayer", which accepts the whole text. The embedding layer is to encode words in the input text as real-valued vectors in a high-dimensional space where the similarity between words in terms of meaning translates to closeness in the vector space. For instance, the difference between word vector "men" and "women" is close to the difference between word vector "king" and "queen" shown as Equation 1.

$$\boldsymbol{v}_{men} - \boldsymbol{v}_{women} \approx \boldsymbol{v}_{king} - \boldsymbol{v}_{queen} \tag{1}$$

We used pretrained GloVe [1] to encode the word vector, which encodes a single word as a 300-D vector. So when text data passes through this layer, a text of length T will be converted a $T \times 300$ matrix.

Although the movie reviews in the IMDb dataset usually vary in length, their lengths are not longer than 450. Also, we used pre-padding to solve the length variation problem, which means adding several 300-D vectors whose elements are all 0s at the beginning of the sentence matrix.

Table 5. Model Based on CNN + LSTM

Layer Type	Output Shape
InputLayer	500
Embedding	500×300
Dropout	500×300
Conv1D	496×200
MaxPooling	248×200
LSTM	100
Dropout	100
Dense	1

Table 6. Model Based on Bidirectional LSTM

Layer Type	Output Shape
InputLayer	500
Embedding	500×300
Dropout	500×300
BidirectionalLSTM	200
Dropout	200
Dense	1

LSTM is, which refers to Long Short Term Memory and is a special kind of RNN, capable of learning long-term dependencies so that it is suitable for processing sequence data [2]. Every single LSTM unit contains several steps of arithmetic operation and will use the result of the previous LSTM unit. These arithmetic operations can be further distributed into a series of "gates". As Figure 1 shows, first gate is called as "Forget Gate" and computation formula for time step t is

$$f_t = \sigma(\boldsymbol{W}_f[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_f) \tag{2}$$

where σ means the sigmoid function, $\boldsymbol{x}_t$ is the input in time step t and $\boldsymbol{h}_{t-1}$ is one output from the previous time step. $\boldsymbol{W}_f$ and $\boldsymbol{b}_f$ are learnable parameters. [] means concatenation. Similarly, formula of "Input Gate" is

$$i_t = \sigma(\boldsymbol{W}_i[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_i) \tag{3}$$

$$\tilde{C}_t = tanh(\boldsymbol{W}_C[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_C) \tag{4}$$

and "Cell Gate"

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{5}$$

finally "Output Gate"

$$o_t = \sigma(\boldsymbol{W}_o[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_o) \tag{6}$$

$$h_t = o_t * tanh(C_t) \tag{7}$$

where o_t is the output of this unit and h_t is used in the next time step. Then after a dense layer, the LSTM based DNN model outputs a sentiment score $\in [0, 1]$.

3.3 Training Result

We trained all 4 models aforementioned in subsection 3.2 to compare their performances over the IMDb dataset with

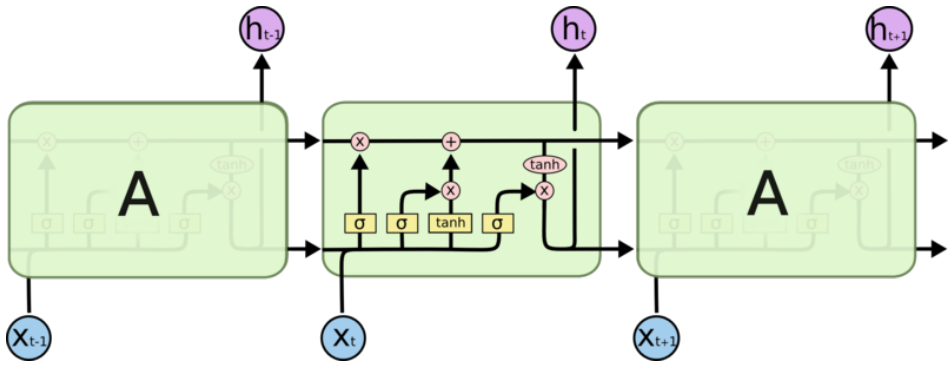

Figure 1. The inner structure of LSTM unit

10% as the validation using Tensorflow as the backend of framework Keras. As shown in Table 7, LSTM outperforms the rest models achieving accuracy of 89.90%. Figure 2 and Figure 3 shows its changes of accuracy and loss over training epochs respectively. So we chose it as the core sentiment analysis model in this project.

Table 7. Results Comparsion

	LSTM	CNN	CNN+LSTM	BiLSTM
Accuracy	89.90%	87.48%	88.70%	87.05%
Loss	0.2617	0.3009	0.2693	0.3047

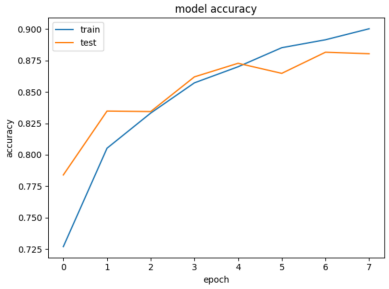

Figure 2. Accuracy Changes over Training Epochs

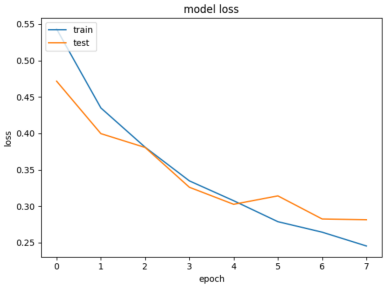

Figure 3. Loss Changes over Training Epochs

3.4 Spark MapReduce

By constructing pair RDD in the form of $(user\_id, full\_text)$, a new RDD in the form of $(sentiment\_score, full\_text)$ is obtained after using the LSTM model to label the sentiment scores of Tweets in Spark MapReduce.

4 Information Extraction and Visualization

4.1 Tweets emotion every day

Since we have the LSTM model, we can use it to score the text of tweets. The score is between 0 and 1. We make the score smaller than 0.5 to be negative and the score larger than 0.5 is positive. We construct the rdd of every tweet in step1, where the emotion(full-text) returns 0 or 1(0 is negative and 1 is positive), [w1,w2,...] is the word list of full-text, we construct it just like that in wordcount file and the stop words file is what given in the lab. By doing mapreduceByKey of first RDD, we get the emotion-rdd. For 7 days, we will have 7 emotion-rdd data.

$$Step1: RDD = (emotion(full-text), (1, [w1, w2, ...]))$$
$$Step2: EmotionRDD = ((emo = 1, count), (emo = 0, count))$$

The result is shown in Figure 4. The red one is positive and the black one is negative. We can find that usually positive tweets are slightly higher than negative. But on March 18th and 19th, the differences of positive and negative tweets are huge. We will analyze it in part 5.

4.2 Daily Frequent Words

We also want to get the frequent one word and two words of positive and negative tweets respectively, which is implemented in step3 and step4. For each emotion type, we use *mapreducebykey* and get top200 count word. Based on these top200 count word, we use the apriorior method and find the top200 pair words according to their count. So for each

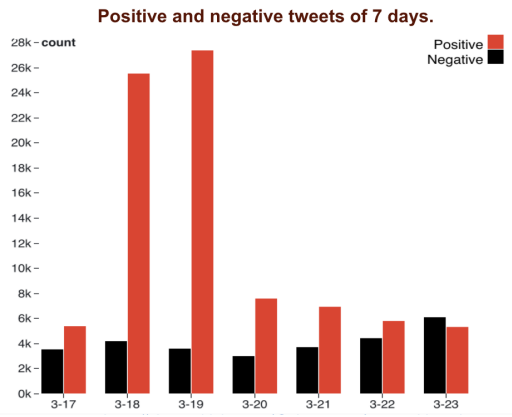

Figure 4. Daily Tweets Emotion

day, we will have four word frequency sets.

$$Step3: 1WordPos = (word, count)$$
$$1WordNeg = (word, count)$$
$$Step4: 2WordsPos = (word1, word2), count)$$
$$2WordsNeg = (word1, word2), count)$$

The result is shown in Figure 5 and Figure 6. Figure 5 is an example of one day, so it has 4 sets. Figure 6 is an example of positive 2 words frequency. The bigger the circle is, the higher the frequency is. More details can be found via [3].

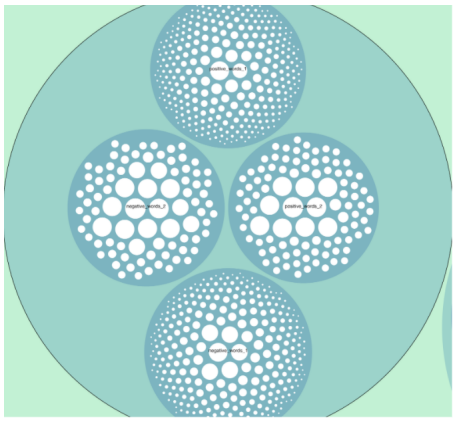

Figure 5. Word Count Per Day Example

4.3 Influential Users

What's more, we also find that in our twitter data, quite a lot of the twitters are retweets, which means it is very possible that someone can influence other people's opinions significantly. So we want to find the most influential twitter users and analyze everyday's retweet twitter and construct

Figure 6. Words Count Example

the RDD in this way:

$$Step1: if(retweets): userRDD = (originalusername, 1)$$
$$else: userRDD = (username, 1)$$
$$Step2: \text{Reduce the RDD by username and sort}$$
$$userRetweetcountRDD = (username, count)$$

If the tweet is retweet, we find the original user name, otherwise we just use the username. So for each tweet, what we care is that who spread the tweet. After we have the rdds that represent how many retweets of a user in one day, we can get the sum this count of 7 days, and use the count as influential factor, then we can get the top10 influential users. According to 7-days' data, we can select 10 most influential twitter uses who are SEACoronavirus, ChannelNewsAsia, STcom, tulunsokit, MothershipSG, TODAYonline, mrbrown, STForeignDesk, parrysingh and Liz-in-Shanghai.

4.4 Daily Tweets combination

If we only focus on every day's tweets, and collect those 10 users' RDDs, we can reconstruct tweets emotion every day data in this way, for every day's positive and negative data, we categorize it into 10 types(represents 10 users) and 1 other type. This will show the detail of tweets combination every day and we can find more potential information.
The result is shown in Figure 7. The legend shows the top10 influential users. The left part is negative tweets number, the right part is positive tweets number. Different color represents different users.

5 Result Analysis

5.1 How influencers influence on unusual day

From Figure 4, we have seen a dramatic increase on positive tweets on 18 March and 19 March, and the number of negative tweets has exceeded that of positive tweets on 23 March, which can also be reflected on Figure 7. We adopt the following approach to identify causes for these sentiment surges.

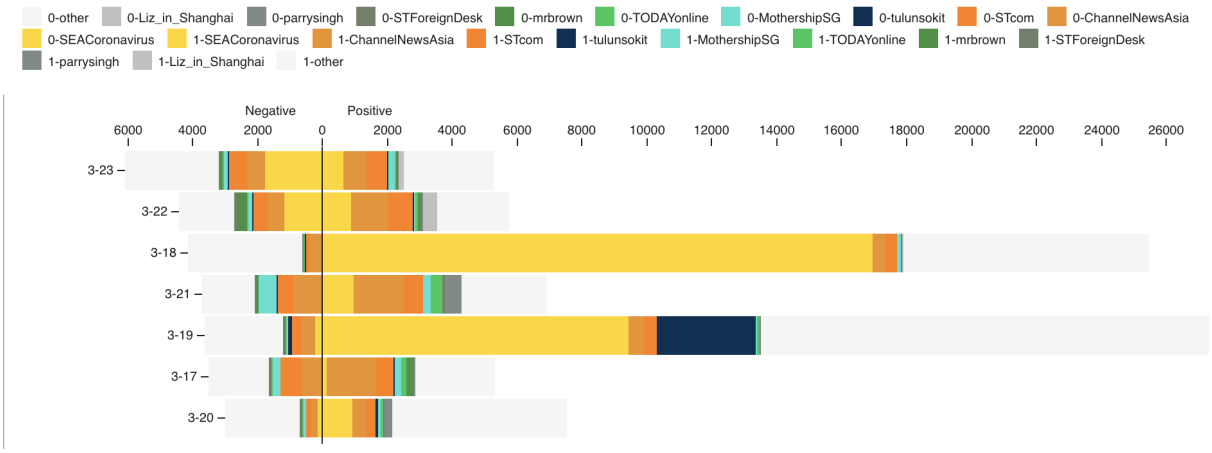

Figure 7. Daily tweets combination

For these three specific days, we went through Figure 7 and decided who was the dominant influencer on each day. We can easily tell that the positive emotion on 18 March, 19 March and negative emoition on 23 March were always dominanted by user SEACoronavirus. So we went through SEACoronavirus's tweets on these these days, and decided which specific tweets had the largest number of retweets, which may suggest which event causes the emotion surge on that day.

Figure 8 shows the tweets of SEACoronavirus with the largest number of retweets on the three specific days. On 18 March and 19 March, the most popular tweets are about the rapid increase on confirmed and death cases in ASEAN countries. Why the worsening condition of ASEAN countries will lead to positive attitudes of Singapore twitter users? We found a border restriction published by Singapore MOH on 15 March which will restrict visitors from ASEAN countries. So we guess on 18 and 19 March , Singapore residents may retweet to praise the SG government's foresight to restrict ASEAN countries in advance before the large-scale outbreak of those countries. On Mar 23, the most popular tweet of SeaCoronavirus is about an inefficient medicine to treat Coronavirus called Chloroquine, the increase on negative tweets may be caused by Singapore people's distrust upon Chloroquine.

5.2 Daily Frequent Words Analysis

Next part of our analysis is that we went through the government measures and events which hit the headlines during the rest four days and figured out Singapore people's attitudes towards them.

On March 17, Singapore Government handed out free sanitizer. And on our positive frequent words cloud on that day (top left corner of Figure 9), We can also see many noticeable 'sanitizer's, which means that people paid a lot attention to this event and had positive attitude towards it. On the same day, Malaysia locked down. And on our negative frequent words cloud (top middle of Figure 9), Malaysia and lock-down became the dominant words, which suggests that many Singapore residents tend to express negative emotions on twitter on this policy.

On March 20, big circles named App, Tracing and TaceTogether are noticeable on our positive frequent word cloud (top right corner of Figure 9), which suggests people were satisfied with this app. While on the same day, 'imported' became a most frequent negative word (bottom left corner of Figure 9) which means that people were worried about foreign visitors and in some way suggests the need to tighten the border.

On March 21, Singapore reported the first two deaths, so 'death' is the most frequent negative word on that day (bottom middle of Figure 9). On March 22, Singapore kind of "Lock-down", which makes people relieved because we see short-term, visitors, barred and entering have become high-frequency words in the positive tweets on that day (bottom right corner of Figure 9).

5.3 Feedback to Government

Based on the previous sentiment analysis of daily tweets posted by Singapore residents, we can give the following feedbacks to Singapore government's anti-virus measures.

We can see that during 15 Mar and 23 Mar, Singapore government has already taken timely and effective measures to control the situation. Internationally, before the outbreak and lockout of ASEAN countries, Singapore government has already banned on ASEAN visitors in advance. And shortly after imported cases increased and first death was reported, they closed the borders for visitors from all over the world. Domestically, they have delivered free sanitizers and masks. They also launched a new app to control community spread. All this measures are recognized by its people according to our twitter sentiment analysis.

Figure 8. Social influencer's most popular tweets

However, the following events increased the anxiety and fear of Singapore residents: lockdown of Malaysia, increasing imported cases, first death case and adoption of medicine Chloroquine. Accordingly Singapore government can take some measures or deliver a speech to explain these negative topics to appease people's negative emotions.

6 Conclusion

In this project, we first used web crawler to collect Tweets data via Twitter official APIs, then trained several neural network models to implement sentiment analysis task. We chose the model based on LSTM as our core model for sentiment score labeling. Next, we used data visualization technology to assist in the result analysis, based on which we proposed some feedback to the government.

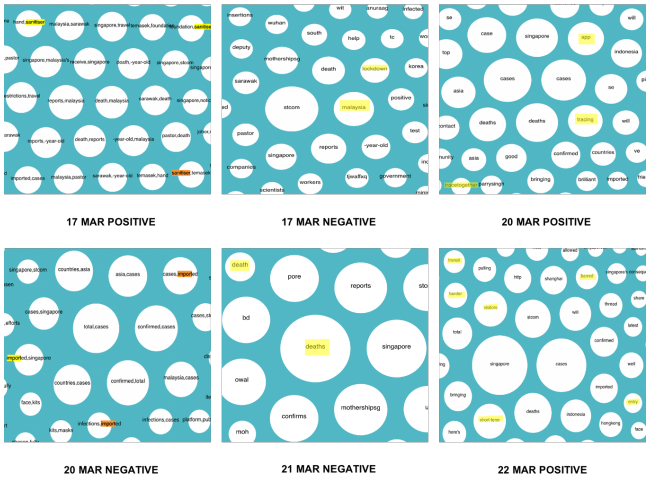

Figure 9. Daily word cloud

References

[1] Christopher D. Manning Jeffrey Pennington, Richard Socher. 2014. GloVe: Global Vectors for Word Representation. https://nlp.stanford.edu/projects/glove/

[2] Christopher Olah. 2015. Understanding LSTM Networks. https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[3] XU Yiqing. 2020. Zoomable Circle Packing. https://observablehq.com/@clearonexu/zoomable-circle-packing