

**The University of Hong Kong**  
**Department of Statistics and Actuarial**  
**Science**

STAT7008 Programming for Data Science

Group project

Sentiment Analysis and Retweeting Prediction Regarding  
Tweets about Hong Kong Typhoon

<b>Student Name</b>	<b>Student ID</b>
Yang Yingyi	3035584500
Long You	3035584562
Liu Zhengyang	3035085982
Zhou Xiao	3035586924

# Contents

1. Introduction
2. Outline
3. Data Collection: Web Crawler
4. Sentiment Analysis
  - 4.1. Data Preprocessing
  - 4.2. Sentiment Polarity Scores
5. Classification and Prediction for Retweeting
  - 5.1. Method
  - 5.2. Analysis and Results
6. Conclusion
7. Appendix
8. Reference

# 1. Introduction

Nowadays, more and more people are using social media to share their life and mood due to its rapid development, especially when a widely influential event just happened. Twitter, as the most famous social media platform which integrates various functions including chatting, news reporting, live streaming, online shopping, etc., has 350 million active users worldwide, and generates over 80 million tweets per day. The huge information flow of twitter is valuable source for analyzing social consensus. For example, the Typhoon Mangkhut, was an extremely powerful tropical cyclone that brought widespread damages to Guam, the Philippines and South China in mid-September. In this report, we mainly discuss three parts.

The first part is obtaining huge information flow from twitter about typhoon by web crawler, which is the basis of subsequent process.

In the second part, we will discuss how can we quantitatively analyze the polarity and intensity change of people's attitude towards different typhoon events during its influence period. In fact, we found that in different days of typhoon's influential period, the proportion of negative, neutral, and positive words are different. The negative comment will increase from the time it is about to landing to the time its damage reached the peak, and then decrease as typhoon gradually vanish. In addition, if the destructive power of a certain typhoon is larger, the polarity score of sentimental analysis towards it will be more negative as result.

The number of retweets also reflects, no matter positive or negative, the intensity of people's emotion toward an event and their degree of recognition to the original tweet. The third part of our project is to predict whether a tweet will be retweeted by LSTM network model. In general, we transform different words into numerical vectors, according to the corresponding lexicon. The distance of 2 word-vectors will be closer if their corresponding words have similar meaning. After we have enough word vectors, we can fit LSTM network model by training set, and check the accuracy, adequacy and probable over-fitting problem of our model by comparing the prediction value and testing set.

## 2. Outline

During typhoon events, social media serves as an effective and important tool to transmit and acquire disaster information. Texts from social media can be used as a way of extracting disaster loss information, analyzing human behaviors and formulating responses. Since there is a word limit of 140 for each single tweet, the moods or attitudes people tend to share within such a short paragraph are usually very intense and emotional. We can conduct sentiment analysis, which ‘computationally’ determining whether a piece of writing is positive, negative or neutral to study people’s reaction towards a significant event. <sup>[1]</sup>

Firstly, the information in the hashtag is helpful for us to retrieve relevant information needed. We used Web crawlers to obtain text comment data about different typhoons. Secondly, We got the polarity (positive/neutral/negative) of people’s comment on typhoons by using sentiment analysis method. Also, we measured the destructive power of the typhoon indirectly since the reason for the negative comments is that the typhoon will damage the infrastructure and cause inconvenience to people. Finally, the more the number of retweeting, the more representative the tweet. In the future work, we paid more attention to the comments with more retweeting. Here we build a classification model to recognize the pattern of retweeting. That’s very meaningful. In brief, our project is mainly divided into three parts: data extraction, sentiment analysis and classification for retweeting.

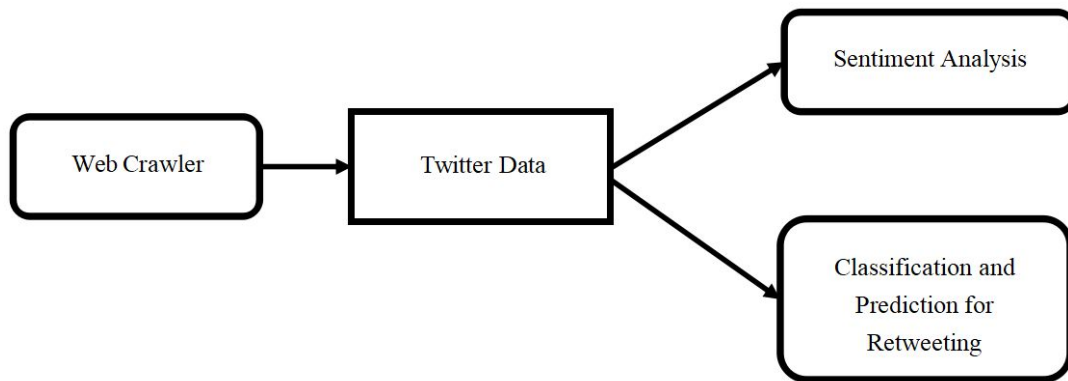


Fig.1 Project Flow Chart

### 3. Data Collection: Web Crawler

Here, we concentrate on typhoons around Hong Kong over the past two years. We use a web crawler package called *got3* to get text data for each selected typhoon by searching the exact name of the typhoon on twitter. The details of the web crawler will be clarified in the Appendix part. In total, we select 14 typhoons and crawl their information on Twitter. The specific description of our original data set is as follows.

Table 1. Data Set Description

Typhoon	Starting time for web crawler	End Time for web crawler	Maximum number of Twitters
Mangkhut	2018-09-08	2018-09-20	130000
Yutu	2018-10-31	2018-11-02	40000
Barijat	2018-09-11	2018-09-13	40000
Bebinca	2018-08-13	2018-08-16	60000
Son-tinh	2018-07-17	2018-07-23	100000
Ewiniar	2018-06-08	2018-06-09	30000
Khanun	2017-10-13	2017-10-16	60000
Marwar	2017-09-02	2017-09-04	50000
Pahkar	2018-08-25	2018-08-27	50000
Hato	2017-08-22	2018-08-23	30000
Merbok	2017-06-11	2017-06-12	30000
Haima	2016-06-11	2017-06-12	30000
Nida	2016-07-30	2017-08-01	50000
Linfa	2015-07-09	2015-07-10	30000

For the convenience of later analysis, we selected some data for some specific dates and saved them according to the date and the name of typhoon. We save them as the project attachment. We can see high frequency word in the word cloud. For example, Fig.2 shows the word cloud of Mangkhut typhoon comment on 2018/09/14.



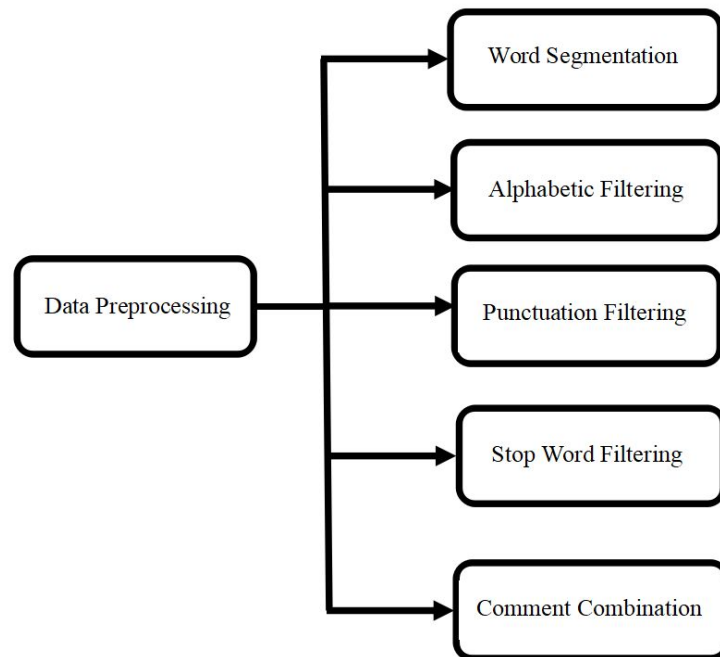


Fig.3 Data Preprocessing

➤ Word Segmentation

Words are the basic components of a sentence. In English, a sentence is often composed of some words, and the meaning expressed by the whole sentence is organically combined by the meaning of these words. Sentiment analysis aims to analyze the emotional orientation of these texts, and it is inseparable from the analysis of basic element words. Therefore, we need to cut a sentence into several words according to the specific English language rules thoroughly, and then we can analyze the words. <sup>[2]</sup> This process is called word segmentation. Here we use the `word_tokenize` method from “`nltk`” package.

➤ Punctuation Filtering: Filter out all punctuation

➤ Alphabetic Filtering

After observing the dataset, the review data is often mixed with non-English languages such as Latin and Spanish. In order to facilitate the establishment of the sentiment analysis model, we filter the text data and delete the non-English text.

## ➤ Stop Word Filtering

After we have performed the above operation, all the comment texts are in the format of "XX XX XX...", where XX represents English words. We can regard each comment text as a vector of words:

$$d = (w_1, w_2, \dots, w_i, \dots, w_n)$$

where  $d$  denotes a sentence and  $w_i$  denotes the  $i$ -th word <sup>[3]</sup>.

Some words in the sentence  $d$  are meaningless and worthless but appear very frequently. These words are called stop words, such as “the”, “a” and “that”. If we use these words in the work of text mining, it will inevitably lead to redundant workload. <sup>[1]</sup> Also because of the high frequency of stop words, the weights of stop words in the text will be very high, thus affecting the final result. So we need to remove the stop words from the comment text data. Here, we use the stop words from the “nltk” package and add some words into stop words, such as “b”, “http”.

## ➤ Comment Concatenation

In this sentiment analysis part, we want to know the polarity score of a typhoon for a specific date. So we need to combine all the comment text for a specific date in order to get a whole string text for each file. Fig.4 illustrates the whole string text after data preprocessing.

```
In [7]: mangkhut20180916

Out[7]: 'mangkhut philippines counts cost deadly news mangkhut wreaks havoc philippines leaving least dead new york times mangkhut powerful storm world far wreaks havoc philippines relief nt even worse mangkhut photos mangkhut cnnhttps mangkhut killed storm batters philippines bbc news china issues red alert mangkhut bears hong kong photo ap hongkongers prepare worst super mangkhut nears mangkhut philippines mangkhut ompongph super mangkhut hong kong observatory issued increasing gale storm signal meaning expected increase significantly mangkhut approaches scene outside one roads thankfully ompongph meanwhile philippines philippines typhoonmangkhut mangkhut killed storm batters philippine s mangkhut philippines counts cost deadly waiting super mangkhut pass hope safe update sa mangkhut sa hong kong ingat po kabayan facebookcomstoryphp mangkhut philippines counts cost deadly mangkhut philippines hit flash flooding landslides september front page philippines reporting first two fatalities mangkhut locally known ompong death toll risen least badly hit areas could take three days reach mangkhut business typhoonmangkhut name super mangkhut date sep hkt position e km southeast hong kong maximum sustained wind near centre kmh mangkhut move towards coast western guangdong todayhttps wwwhkgovhkwxinfocurrwtcgisehtm mangkhut philippines counts cost deadly news mangkhut wreak s havoc philippines leaving least dead new york times mangkhut powerful storm world far wreaks havoc philippines relief nt even worse mang khut photos mangkhut cnnhttps mangkhut killed storm batters philippines bbc news china issues red alert mangkhut bears hong kong photo ap hongkongers prepare worst super mangkhut nears mangkhut philippines mangkhut ompongph super mangkhut hong kong observatory issued increasing gale storm signal meaning expected increase significantly mangkhut approaches scene outside one roads thankfully ompongph meanwhile philippines philippines typhoonmangkhut mangkhut killed storm batters philippines mangkhut philippines counts cost deadly waiting super mangkhut pass hope safe update sa mangkhut sa hong kong ingat po kabayan facebookcomstoryphp mangkhut philippines counts cost deadly mangkhut philippines hit flash flooding landslides september front page philippines reporting first two fatalities mangkhut locally known ompong death toll risen least badly hit areas could take three days reach mangkhut business typhoonmangkhut name super mangkhut date sep hkt position e km southeast hong kong maximum sustained wind near centre kmh mangkhut move towards coast western guangdong todayhttps wwwhkgovhkwxinfocurrwtcgisehtm mangkhut philippines counts cost deadly news mangkhut wreaks havoc philippines leaving least dead new york times mangkhut
```

Fig.4 Mangkhut Typhoon Text Data (2018/09/16) after Data Preprocessing



## 4.2 Sentiment Polarity Scores

### ➤ Methods

Sentiment analysis is simply the process of working out (statistically) whether a piece of text is positive, negative or neutral. The majority of sentiment analysis approaches take one of two forms: polarity-based, where pieces of texts are classified as either positive or negative, or valence-based, where the intensity of the sentiment is taken into account. For example, the words ‘good’ and ‘excellent’ would be treated the same in a polarity-based approach, whereas ‘excellent’ would be treated as more positive than ‘good’ in a valence-based approach.

We choose to use **VADER** to handle sentiment analysis with twitter media text data.<sup>1</sup> VADER belongs to a type of sentiment analysis that is based on lexicons of sentiment-related words especially in the context of social media<sup>[2]</sup>. In this approach, each of the words in the lexicon is rated as to whether it is positive or negative, and in many cases, positive or negative to what extent. Below we can see an excerpt from VADER’s lexicon, where more positive words have higher positive ratings and more negative words have lower negative ratings<sup>2</sup>.

Table 2. Sentiment Polarity Scores Example Using VADER

Word	Compound Rating
sad	-0.4767
nervous	-0.2732
insane	-0.4019
good	0.4404

---

<sup>1</sup> Source:

<http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf>

<sup>2</sup>Source:

<http://t-redactyl.io/blog/2017/04/using-vader-to-handle-sentiment-analysis-with-social-media-text.html>

To figure out whether these words are positive or negative, developers of these methods need to have a group of people manually evaluate them, which is obviously quite expensive and time consuming. In addition, the dictionary needs to cover the words in the text you are interested in, otherwise it will not be very accurate. On the other hand, this method is accurate when there is a good match between the dictionary and the text, and the result can be quickly returned even on a large amount of text. Fortunately, we can use the pre-trained model here.

VADER produces four sentiment metrics from these word ratings, which is shown below. The first three, positive, neutral and negative, represent the proportion of the text that falls into those categories. As it shows, the example sentence “I feel sad” was rated as 75.6% negative, 24.4% neutral and 0% positive. The final metric, the compound score, is the sum of all of the lexicon ratings (-0.4767 in this case) which have been standardised to range between -1 and 1. In this case, the example sentence has a rating of -0.4767, which is pretty strongly negative.

Table 3. Sentiment Polarity Scores of “I feel sad”

Sentiment Metric	Value
Positive	0
Neutral	0.244
Negative	0.756
Compound	-0.4767

## ➤ Analysis and Results

We get each typhoon's sentiment polarity scores for each selected date. The result is posted in the conclusion section. Fig.5 and Fig.6 show the average scores of each typhoon.

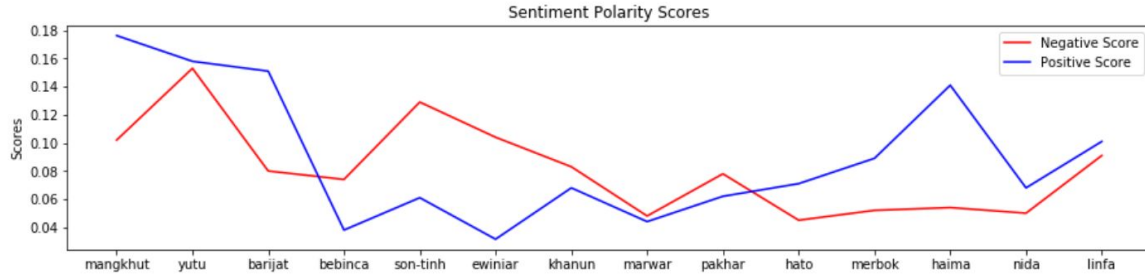


Fig.5 Sentiment Polarity Scores for Different Typhoons (only negative and positive scores)

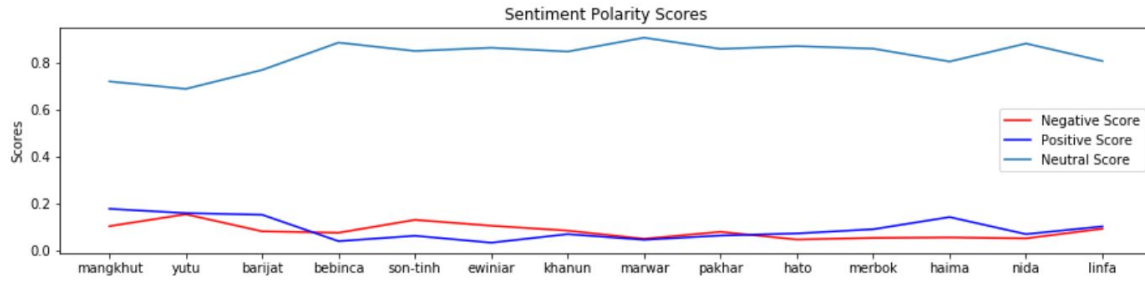


Fig.6 Sentiment Polarity Scores for Different Typhoons (averaging)

Here we want to find something more perceivable by using our sentiment analysis result. Since the reason for the negative comments is that the typhoon will damage the infrastructure and cause inconvenience to people, we can measure the destructive power of the typhoon by:

$$p_i = \frac{\text{negative\_score}_i - \text{positive\_score}_i}{\text{negative\_score}_i + \text{positive\_score}_i}$$

$$power = \begin{cases} 100 * (\max_{i \in date} p_i - \min_{j \in date} p_j) \leftarrow \exists p_x > 0 \\ 0 \leftarrow \neg \exists p_x > 0 \end{cases}$$

where i is a specific date for the typhoon.

Hence, we can get the destructive power for each typhoon. Fig. 7 illustrates the coefficient of destructive power for some typhoons.

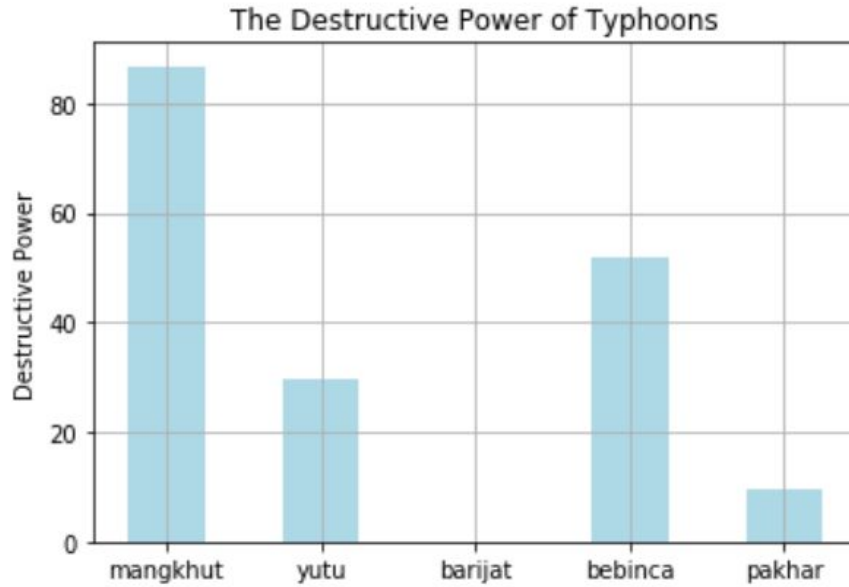


Fig. 7 Destructive Power Coefficients for Some Typhoons

Also, it is meaningful to recognize the extent to which typhoons affect people's mood. Here the effects of typhoons includes good effects and bad effects. We define psychological effect coefficient as:

$$PEC = \max_{i \in date} (positive\_score + negative\_score) * 100$$

where i is a specific date of the typhoon.

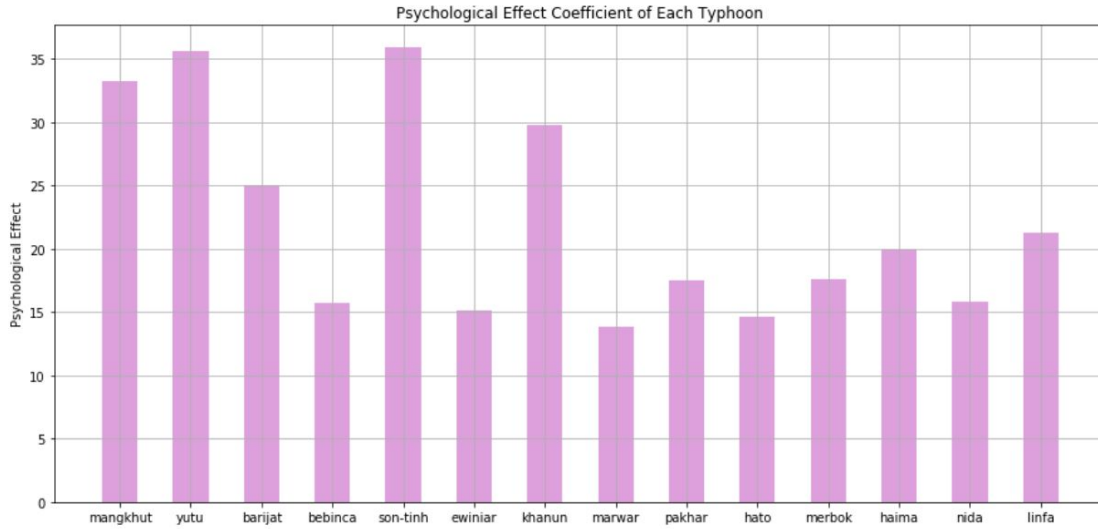


Fig. 8 Psychological Effect Coefficients for Each Typhoon

## 5. Classification and Prediction for Retweeting

The more times a tweet is retweeted for, the more recognized the tweet text data. In our crawled data, there is also an attributes called “retweets”, which means the number of retweeting of the tweet. In this part we predict whether a tweet is retweeted based on the content of the comment data. We need to deal with a binary classification.

### 5.1 Method

Because text data is sequential, we choose the LSTM neural network model, which is designed for processing sequential data. LSTM neural network, which stands for Long Short-Term Memory, is a particular type of recurrent neural network that got a lot of attention recently within the machine learning community.

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn. All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer. LSTMs also have this chain-like structure, but the

repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way [4].

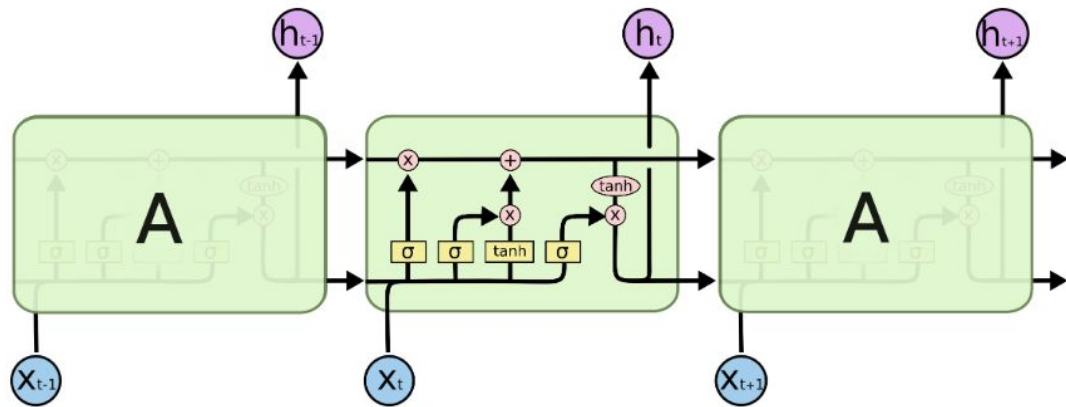


Fig. 9 The Repeating Module in an LSTM Contains Four Interacting Layers<sup>3</sup>

In the above diagram, each line carries an entire vector, from the output of one node to the inputs of others. The pink circles represent pointwise operations, like vector addition, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denote its content being copied and the copies going to different locations.

The key to LSTMs is the cell state, the horizontal line running through the top of the diagram. The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It's very easy for information to just flow along it unchanged.

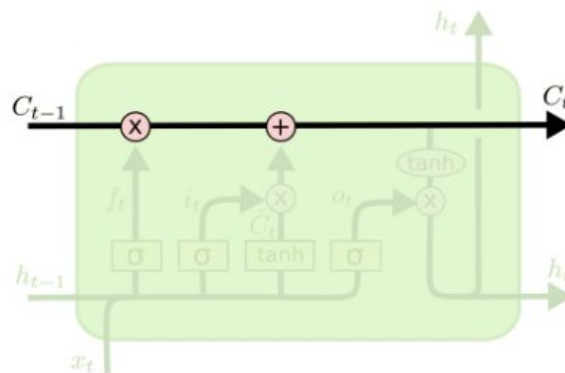


Fig. 10 LSTM Core Idea

<sup>3</sup> Source: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

## 5.2 Analysis and Results

One problem may affect the predictability of our model is that the raw data is imbalanced. About 79% of the tweets are not retweeted, which will decrease our model performance dramatically. Therefore, we need to make a data augmentation first. We manually select some tweets data as our training set, which data distribution is around 1:1. (i.e. about half of tweets are retweeted). Later we will show the difference between the model performance with data augmentation and the performance without that.

Neural network is easy to be overfitting. So we use dropout method to optimize the model. Dropout is an approach to regularization in neural networks which helps reducing interdependent learning among the neurons. For each hidden layer, for each training sample, for each iteration, ignore (zero out) a random fraction ( $p$ ) of nodes and corresponding activations. And then use all activations, but reduce them by a factor  $p$  (to account for the missing activations during training) .

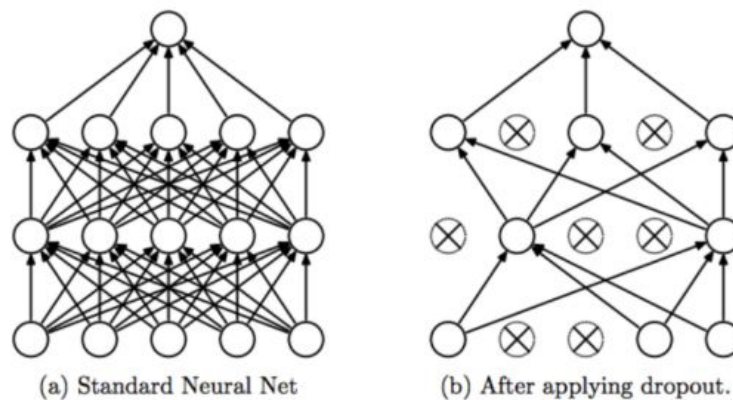


Fig. 11 Dropout Idea<sup>4</sup>

Here we choose  $p=0.1$  and use the “relu” and “sigmoid” function as the activation function. By using 20% of the training data as the validation data set, our model performance is as follows:

---

<sup>4</sup>Source: <https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5>

```

In [41]: inp = Input(shape=(maxlen,))
x = Embedding(max_features, embed_size, weights=[embedding_matrix])(inp)
x = Bidirectional(LSTM(50, return_sequences=True, dropout=0.1, recurrent_dropout=0.1))(x)
x = GlobalMaxPool1D()(x)
x = Dense(50, activation="relu")(x)
x = Dropout(0.1)(x)
x = Dense(1, activation="sigmoid")(x)
model = Model(inputs=inp, outputs=x)
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

In [39]: model.fit(X_train, y, batch_size=30, epochs=2, validation_split=0.2)

Train on 20772 samples, validate on 5193 samples
Epoch 1/2
20772/20772 [=====] - 440s 21ms/step - loss: 0.0299 - acc: 0.9906 - val_loss: 1.3747 - val_acc: 0.7237
Epoch 2/2
20772/20772 [=====] - 438s 21ms/step - loss: 1.5738e-04 - acc: 1.0000 - val_loss: 0.9825 - val_acc: 0.7789



Out[39]: <keras.callbacks.History at 0x240b7bdlac8>

```

Fig. 12 Model Performance

We can clearly see the promotion after we make the data augmentation. Finally we get 77.89% accuracy in our validation set whose data distribution is 1:1.

Table 4. The Model Performance

	Benchmark Accuracy	Accuracy in Validation set
Before Data Augmentation	79%	80% 
After Data Augmentation	50%	77.89% 

Note: In our original data set, 79% of the tweets are not retweeted. So the benchmark accuracy before data augmentation is 79%. Similarly, after data augmentation our label distribution is 1:1 and then the benchmark accuracy after data augmentation is 50%.

## 6. Conclusion

Firstly, we extract related information from Twitter using web crawler, select and organize datasets to prepare for further analysis. Data preprocessing is necessary here in order to carry out the sentiment analysis. In this part, we conduct word segmentation, alphabetic filtering, punctuation filtering, stop word filtering and comment concatenation.

We quantitatively measured the attitude of netizens towards different typhoons at different time points during its influential period. With the help of VADER, the sentiment polarity scores of each typhoon on each date can be calculated. 3 examples are shown below and the complete table of results are shown in the appendix.



Table 5. Sentiment polarity scores

Name of Typhoon	date	compound	negative	neutral	positive
<b>Mangkut</b>	2018-09-14	1	0.022	0.751	0.227
	2018-09-16	1	0.155	0.662	0.183
	2018-09-17	-1	0.13	0.75	0.119
<b>Yutu</b>	2018-10-31	-1	0.157	0.707	0.136
	2018-11-01	-1	0.157	0.715	0.127
	2018-11-02	1	0.144	0.644	0.212
<b>Barijat</b>	2018-09-12	1	0.109	0.751	0.141
	2018-09-13	1	0.051	0.789	0.16

In sentiment analysis part, we also discovered that, the magnitude of polarity score is highly correlated with the destructive power of the corresponding typhoon. Hence, we create the “destructive power coefficient” to estimate the influence that people actually perceived from their emotions in tweets. For the typhoon with higher destructive power, both the absolute value of its positive score and negative score will be higher, which indicates that the fluctuation of public sentiment will be more significant under the external stimulation from an extreme event. Also, to measure how sentimental polarity correlated with psychological influence typhoons have on people, we constructed psychological effect coefficient to statistically show the extent of such psychological effect.

Additionally, in last part we predicted whether certain tweets would be retweeted using binary prediction method, meaning that on validation set, which is the test set of our model with 1:1 data distribution (half retweeting, half not retweeting), have a prediction accuracy rate of 77.9% approximately, which is a pretty decent prediction but still has some room to improve.

There exists some extensions for future study :

1. Due to hardware and time restriction, we only selected a small portion of comment data as training set. If more data could be acquired in terms of volume and time dimension, we

would generate a result to represent the degree of destructive power and psychological influence.

2. In the last part of classification problem, we only select the maximum feature (i.e. the number of unique words) of 800. If more features are selected and more data are added into the training set, the accuracy will be significantly improved.

3. The higher frequency of retweeting, the stronger the representativeness it will be. If the frequency of retweeting can be used as the weight of text data in the future, the efficiency of the prediction model will be greatly improved.

## 7. Appendix

The results of Sentiment Analysis:

Name of Typhoon	date	compound	negative	neutral	positive
<b>Mangkhut</b>	2018-09-14	1	0.022	0.751	0.227
	2018-09-16	1	0.155	0.662	0.183
	2018-09-17	-1	0.13	0.75	0.119
<b>Yutu</b>	2018-10-31	-1	0.157	0.707	0.136
	2018-11-01	-1	0.157	0.715	0.127
	2018-11-02	1	0.144	0.644	0.212
<b>Barijat</b>	2018-09-12	1	0.109	0.751	0.141
	2018-09-13	1	0.051	0.789	0.16
<b>Bebinca</b>	2018-08-13	-1	0.071	0.875	0.053
	2018-08-14	-1	0.122	0.843	0.035
	2018-08-15	1	0.029	0.944	0.027
<b>Son-Tinh</b>	2018-07-17	0.9999	0.027	0.94	0.033
	2018-07-18	1	0	0.972	0.028
	2018-07-23	-1	0.359	0.641	0

<b>Ewiniar</b>	2018-06-08	-1	0.151	0.849	0
	2018-06-09	-0.999 7	0.056	0.881	0.063
<b>Khanun</b>	2017-10-13	-1	0.226	0.701	0.072
	2017-10-14	1	0	0.864	0.136
	2017-10-15	-1	0.042	0.934	0.023
	2017-10-16	-1	0.064	0.895	0.041
<b>Marwar</b>	2017-09-02	1	0	0.913	0.087
	2017-09-03	-1	0.106	0.862	0.032
	2017-09-04	-1	0.039	0.948	0.013
<b>Pahkar</b>	2018-08-25	-1	0.093	0.825	0.082
	2018-08-26	0.9999	0.048	0.907	0.045
	2018-08-27	-1	0.094	0.847	0.059
<b>Hato</b>	2017-08-22	1	0.017	0.854	0.129
	2017-08-23	-1	0.073	0.889	0.038
<b>Merbok_</b>	2017-06-11	-1	0.058	0.897	0.046
	2017-06-12	1	0.045	0.824	0.131
<b>Haima</b>	2016-10-19	1	0.07	0.812	0.118
	2016-10-20	1	0.037	0.8	0.163
<b>Nida</b>	2016-07-30	0.9999	0.026	0.945	0.03
	2016-07-31	-1	0.112	0.861	0.027
	2016-08-01	1	0.011	0.842	0.147
<b>Linfa</b>	2015-07-09	1	0.029	0.787	0.184
	2015-07-10	-1	0.153	0.829	0.018

All codes are saved in the attachment uploaded on Moodle.

## 8. Reference

- [1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011, June). Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media* (pp. 30-38). Association for Computational Linguistics.
- [2] Gilbert, C. H. E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*. Available at (20/04/16) <http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.Pdf>.
- [3] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1* (pp. 142-150). Association for Computational Linguistics.
- [4] Sundermeyer, M., Schlüter, R., & Ney, H. (2012). LSTM neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.