Intelligence System. Natural Language Processing

Pablo Hernández Carrascosa

January 2022

GitHub → https://github.com/pablohdez98/NLP.Processing-and-sentiment-analysis

Problem description

The dataset is about tweets collected between April 9 and July 16, 2020. These tweets are related to the stock market and were obtained by using not only the SPX500 tag but also the top 25 companies in the index and "#stocks". It contains a total of 5000 tweets but only 1300 were manually classified and reviewed and 4 attributes.

- id: contains id used for the tweet.
- created_at: date and time when the tweet was tweeted.
- text: tweet/text written by the user.
- sentiment: whether the tweet was positive, neutral, or negative.

The goal of this assignment is to make a data processing and analysis to extract some useful knowledge and create a classification model to decide the sentiment of future tweets.

Methodology

All the experiments presented in this assignment have been conducted in R, version 4.1.2. First, we read the csv file by setting the encoding to UTF-8, remove the first and the second attributes (id and created_at) and encode the sentiments (negative → 0, neutral → 1 and positive → 2) for the following task.

Some transformations were carried out, such as removing emojis, links, punctuations, weird characters and stop words, lowercase the text, replace contractions, etc. to make the text ready for some NLP.

Before → "*RT @RobertBeadles: Yo💥\nEnter to WIN 1,000 Monarch Tokens✅\n\nUS Stock Market Crashes & what we can LEARN from them PT3!\n\nRETWEET, WATCH video…*"

After → "*rt @robertbeadl yo enter win monarch token us stock market crash & can learn pt retweet watch video*"

Next, we made a histogram which shows the length of the tweets and their occurrences (Figure I). Here, we saw that some tweets could have (before transformations) more than 280 characters, which it cannot be possible due to the maximum restriction of characters in Twitter. By reviewing the csv file, we suggested that it could be different tweets but coming from the same thread and same person.

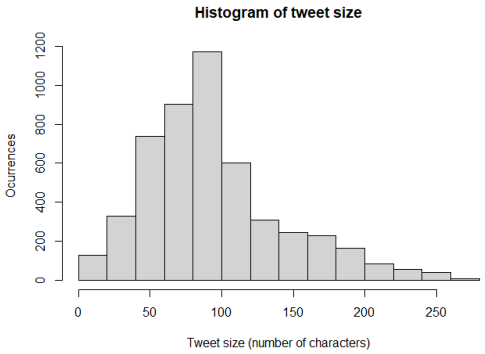


Figure I. Length of tweets

After this, we calculated the most popular hashtags and stocks and plotted in a bar chart (Figure II and III). Also, the frequency of each word (not considering hashtags and stock starting with $) was performed and displayed as word cloud in Figure IV. The most frequent hashtag is #stocks, but it is obvious as the majority of the tweets were recollected by searching that hashtag. And the most popular stocks are $SPX500, apple, amazon, Facebook, etc, also expected.

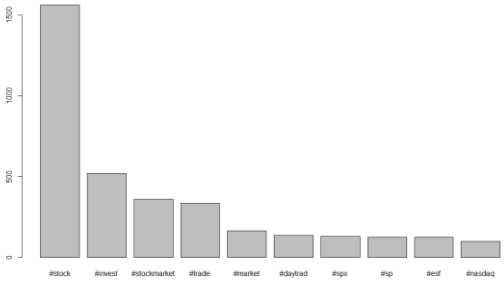


Figure III. Most popular hashtags

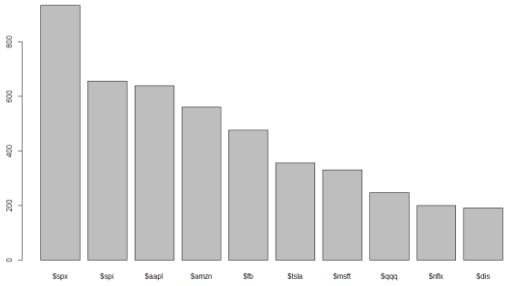


Figure II. Most popular stocks

Figure IV. Word map of most frequency words

In Figure V, we can see some analysis of some words of our dataset. Most of them are tagged correctly, for example, bank as a noun or slow as adjective.

token_id	token	lemma	pos	head_token_id	dep_rel	entity	nounphrase	whitespace
8	wfc	wfc	NOUN	9	compound			TRUE
9	stimulus	stimulus	ADJ	13	amod			TRUE
10	payment	payment	NOUN	11	npadvmod			TRUE
11	slow	slow	ADJ	13	amod			TRUE
12	onlin	onlin	NOUN	13	compound	GPE_B		TRUE
13	bank	bank	NOUN	13	ROOT			FALSE

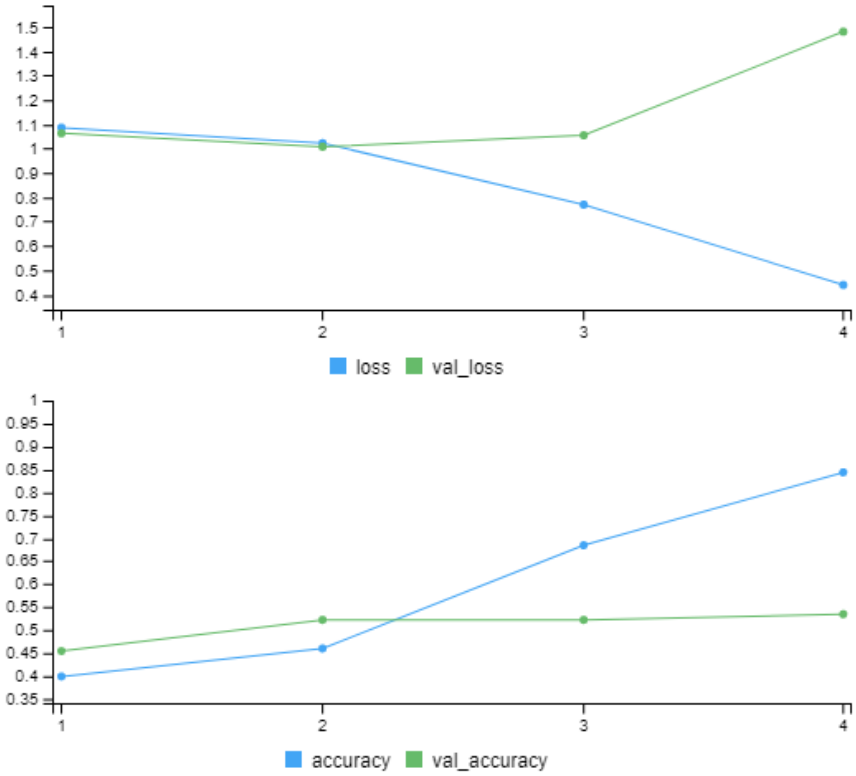
Figure V. Tags (labels) analysis for some words

For the classification model, we removed the rows which do not have any sentiments, leaving us with 1300 rows. The data frame was splitter in 0.75 for train set and 0.25 for test set. For this part, TensorFlow and Keras were installed. First, we tokenized every word in the text by converting them into an integer, according to their frequencies. The model used was a bidirectional Long Short-Term Memory (LSTM), it is an architecture of a recurrent artificial neural network. As it is bidirectional, in will run the inputs in two ways, from past to future and from future to past. The output layer has a 'Softmax' as activation function with 3 units, one per class or sentiment. It also has a dropout layer of 0.5 to avoid overfitting. The summary of the model can be seen in Figure VI.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_36 (Embedding)     (None, 10, 128)           128000
_________________________________________________________________
bidirectional_35 (Bidirectional) (None, 128)           98816
_________________________________________________________________
dropout_41 (Dropout)         (None, 128)               0
_________________________________________________________________
dense_51 (Dense)             (None, 3)                 387
=================================================================
Total params: 227,203
Trainable params: 227,203
Non-trainable params: 0
```

Figure VII. Model structure of LSTM

In the Figure VII is shown the loss and the accuracy in the train (blue colour) and test (green colour). The loss increases in the test set and the accuracy decrease does not improve too much in the test set.



Conclusions

The analysis of this dataset was very complicated as it contains a lot of uncommon things like emojis or hashtags. Once we could clean it, we could obtain a lot of useful information from it, such as, which stock is the most popular, or what word in positive tweets is the most repeated.

The results obtained in the LSTM model are not too good, with an accuracy of 0.5 in the test set. This could be first because the dataset was not too big and more tweets with labels are required. Also, there are words that can appear frequently in positive and negative sentiment tweets such as market, stock, price, etc.

In conclusion, NLP are too powerful and can be really useful in problems of real life. NLP requires a lot of pre-processing to became with a good and clean dataset, but as we have seen in other courses in other areas of machine learning, it is always a necessity to make a good pre-processing, taking approximately 60% or more of the project.

References

Stock Market Tweet | Sentiment Analysis lexicon. URL: https://www.kaggle.com/utkarshxy/stock-markettweets-lexicon-data [Accessed: 28-01-2022]

M. Rico. Language technologies Part 1, 2 and 3, in Data Science Master (Universidad Politécnica de Madrid)