Midterm Project: Sentiment Analysis of Twitter data and News Headlines

Team: Anuja Kapre and Raksha Kaverappa
Github repository: https://github.com/raksha592/Sentiment-Analysis-Midterm-Project

For the data provided, the problem was divided into two subtasks:

1. Sentiment Analysis of twitter and stocktwits data
2. Sentiment Analysis of News Headlines

For each of the subtasks, the following operations were carried out to obtain adequate accuracy and cosine similarity. The process was as follows:

i. Data Pre-processing
ii. Split data into train and test
iii. Tokenization of the words
iv. Building the model
v. Compare the predicted sentiment score with the actual sentiment score
vi. Obtain the cosine similarity
vii. Predict sentiment score for test data

Data Pre-processing:

Sub-task 1: For the twitter and stocktwits data, the data was cleaned considering various factors to improve the prediction accuracy. First, the keys containing the messages (‘spans’) was converted from lists to strings to help access the values of strings more easily. Next, the data in the dictionary was converted to lowercase. The punctuations and the unwanted characters were removed to clean the messages. To improve the prediction of the sentiment score, stopwords were removed from all the messages using NLTK library.

Sub-task 2: For the new and headlines data, the data was cleaned to remove the company name form the messages and replace it with a blank. Next, the keys containing messages were converted to lowercase in the dictionary. Punctuations and non-ascii characters were removed to clean the strings and the stopwords were removed using NLTK library. These factors and preprocessing steps helped improve the prediction of the sentiment score in the analysis.

Split data into train and test:

For the first step of the evaluation, we combined the Training data and the trial data into one complete training dataset.

We evaluated the model and tested the same model to compare the sentiment score to the predicted sentiment scores.

X: Messages/Spans

Y: Sentiment score

For the second part, we trained the model with the training dataset and tested it with the test dataset provided. The same data pre-processing was carried out for the test dataset.

Tokenization of the words:

We used the keras inbuilt library to tokenize the strings into words. We used the pad_sequences library from keras to convert the tokenized words into a numpy array of having the number of rows as the maximum length of the words in a string.
Sub-task 1: While padding these sequences, since dataset was larger, the maxLen attribute was set to the length of the longest sentence of the dataset (25). Also in tokenization, the num_words attribute was set to maxLen * 20 (500).

Sub-task 2: While padding sequences, since dataset was smaller, using the maxLen attribute as the length of the longest sentence of the dataset was overfitting the model.
The problem was solved by using average no of words per sentence (7) as the maxLen.While tokenizing, the num_words attribute was set to maxLen * 50 (350).

The same operation was performed for the test dataset.

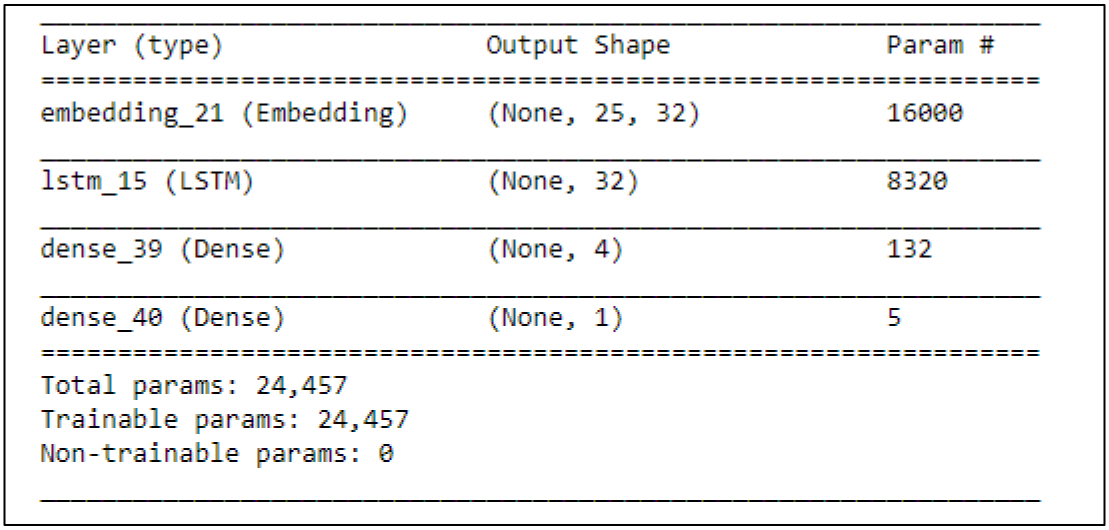
Building the model:
We implemented the model for both the datasets using two different models. The first model was using RNN and the second model was using CNN. We tested the model by changing several hyperparameters and observed the following:

- The model predicted very low accuracy of 3.75% with both RNN and CNN when the activation function was relu or softmax
- The model had good accuracy for activation function sigmoid, however, it was stuck and was unable to give proper prediction sentiment score
- The activation function was changed to tanh and we obtained a prediction accuracy above 60% using RNN and CNN for both the tasks

- We changed the dropout rate from 0.01 to 0.3 to get an improved accuracy
- The model for CNN was predicting maximum positive value (0.99) for all the messages having positive sentiment score.

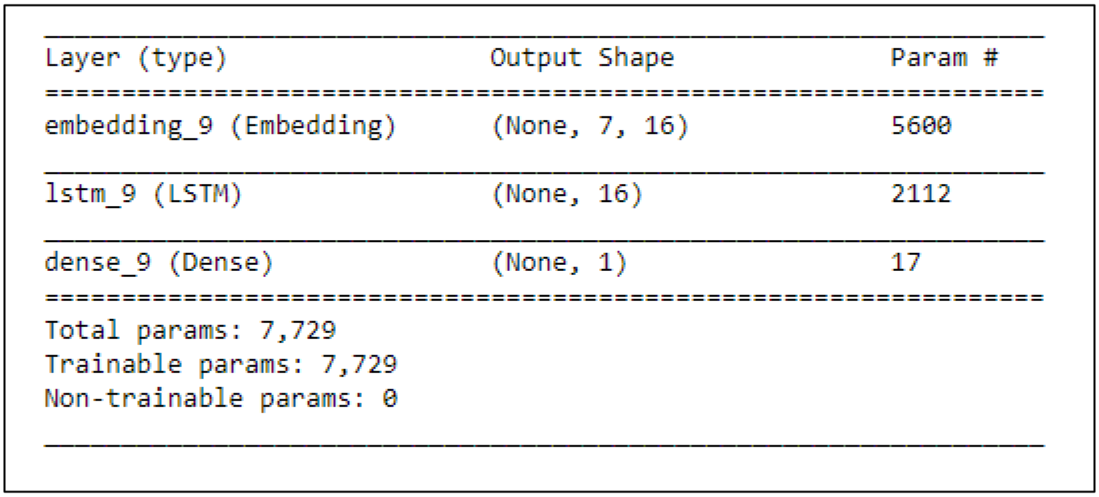
Based on the observations above, the RNN performed best giving the best accuracy and an accurate prediction of sentiment score. Our final model summary is as shown below:

Sub-task 1:



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
embedding_21 (Embedding)     (None, 25, 32)            16000     
_________________________________________________________________
lstm_15 (LSTM)               (None, 32)                8320      
_________________________________________________________________
dense_39 (Dense)             (None, 4)                 132       
_________________________________________________________________
dense_40 (Dense)             (None, 1)                 5         
=================================================================
Total params: 24,457
Trainable params: 24,457
Non-trainable params: 0

_________________________________________________________________
```

Sub-task 2:



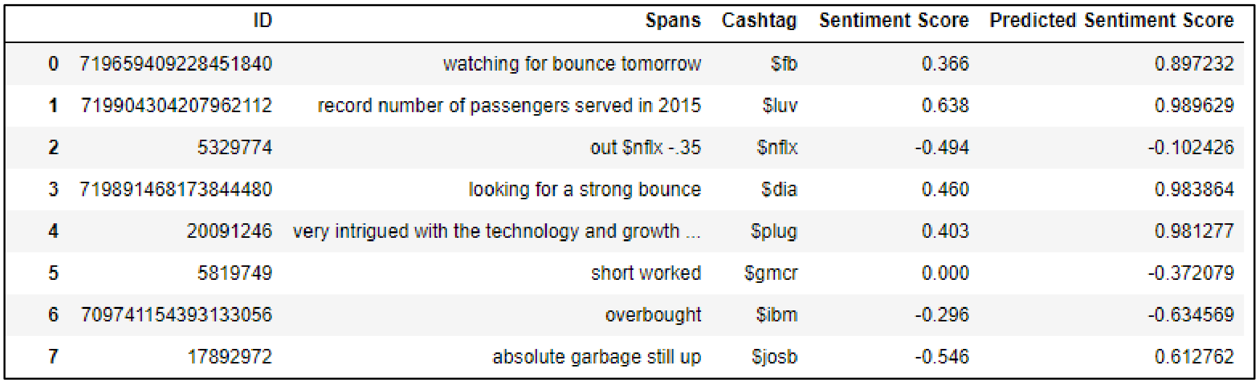
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
embedding_9 (Embedding)      (None, 7, 16)             5600      
_________________________________________________________________
lstm_9 (LSTM)                (None, 16)                2112      
_________________________________________________________________
dense_9 (Dense)              (None, 1)                 17        
=================================================================
Total params: 7,729
Trainable params: 7,729
Non-trainable params: 0

_________________________________________________________________
```

Compare the predicted sentiment score with the actual sentiment score:

The predicted sentiment score and the given sentiment scores were compared after testing the model on the training dataset.

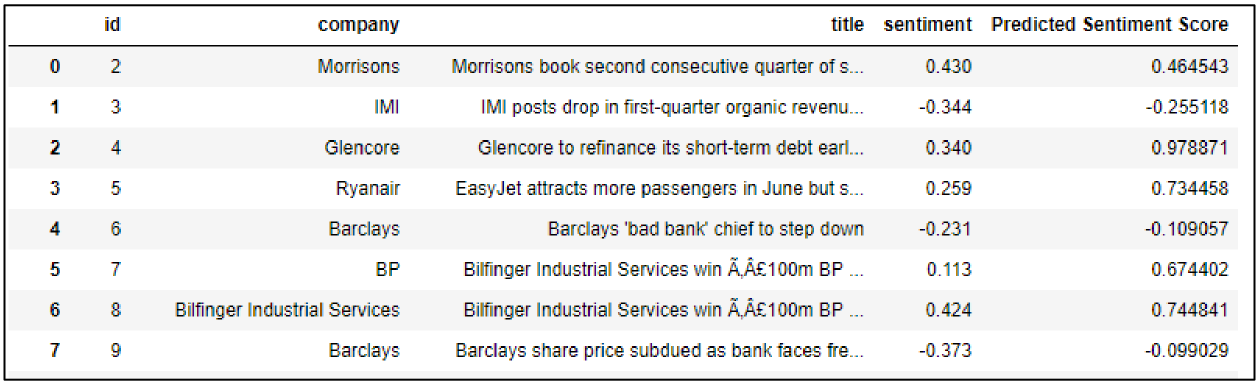
Following are the results of predicting sentiment score for training dataset for task1 and task2

Sub-task 1:



	ID	Spans	Cashtag	Sentiment Score	Predicted Sentiment Score
0	719659409228451840	watching for bounce tomorrow	$fb	0.366	0.897232
1	719904304207962112	record number of passengers served in 2015	$luv	0.638	0.989629
2	5329774	out $nflx -.35	$nflx	-0.494	-0.102426
3	719891468173844480	looking for a strong bounce	$dia	0.460	0.983864
4	20091246	very intrigued with the technology and growth ...	$plug	0.403	0.981277
5	5819749	short worked	$gmcr	0.000	-0.372079
6	709741154393133056	overbought	$ibm	-0.296	-0.634569
7	17892972	absolute garbage still up	$josb	-0.546	0.612762

Sub-task 2:



	id	company	title	sentiment	Predicted Sentiment Score
0	2	Morrisons	Morrisons book second consecutive quarter of s...	0.430	0.464543
1	3	IMI	IMI posts drop in first-quarter organic revenu...	-0.344	-0.255118
2	4	Glencore	Glencore to refinance its short-term debt earl...	0.340	0.978871
3	5	Ryanair	EasyJet attracts more passengers in June but s...	0.259	0.734458
4	6	Barclays	Barclays 'bad bank' chief to step down	-0.231	-0.109057
5	7	BP	Bilfinger Industrial Services win Ã‚Â£100m BP ...	0.113	0.674402
6	8	Bilfinger Industrial Services	Bilfinger Industrial Services win Ã‚Â£100m BP ...	0.424	0.744841
7	9	Barclays	Barclays share price subdued as bank faces fre...	-0.373	-0.099029

Evaluate model based on the cosine similarity:

The cosine similarity is computed using the cosine_similarity function from scikit learn. The cosine weight is the ratio of normalized vectors of Gold Standard score and Predicted score. The final cosine score observed for both the tasks were:

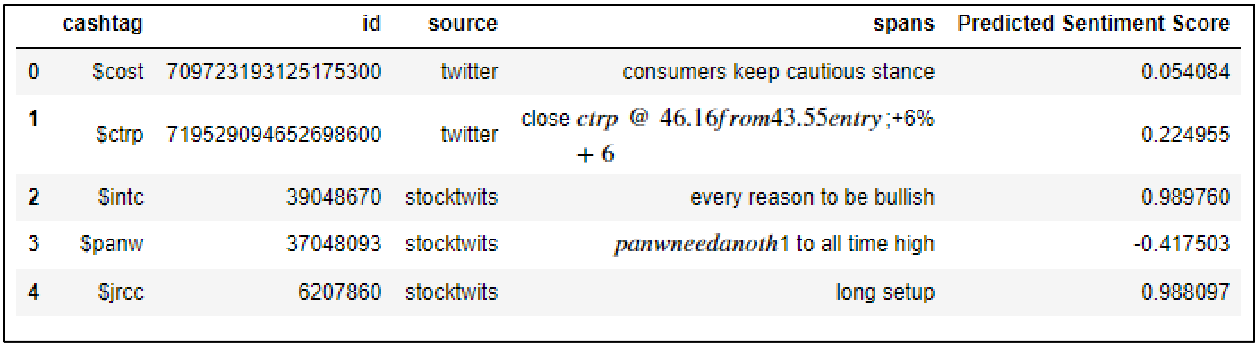
Sub-task 1: 0.6917

Sub-task 2: 0.6932

<u>Predict Sentiment score for test data:</u>

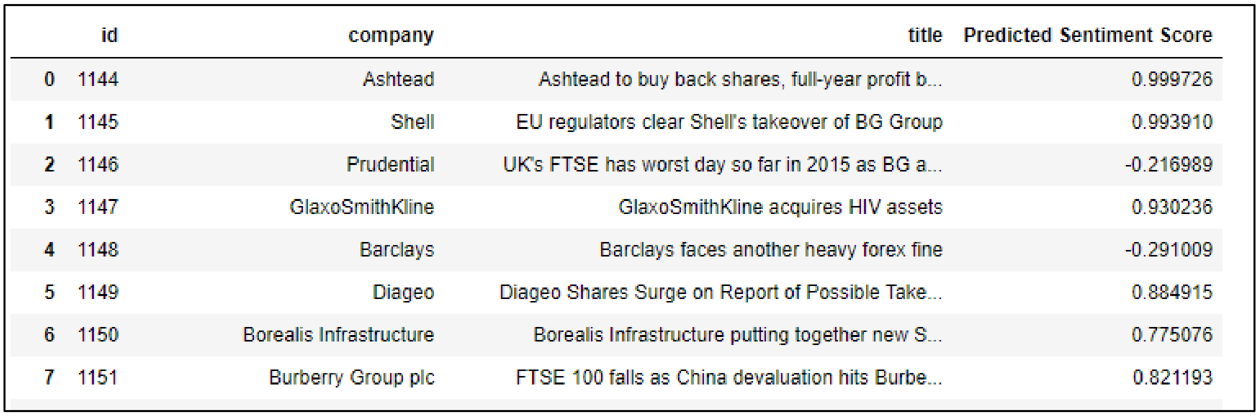
The predicted sentiment scores for the test dataset was also obtained. The results are as shown below:

Sub-task 1:



	cashtag	id	source	spans	Predicted Sentiment Score
0	$cost	709723193125175300	twitter	consumers keep cautious stance	0.054084
1	$ctrp	719529094652698600	twitter	close $ctrp\ @\ 46.16from43.55entry$;+6% $+6$	0.224955
2	$intc	39048670	stocktwits	every reason to be bullish	0.989760
3	$panw	37048093	stocktwits	$panwneedanoth$1 to all time high	-0.417503
4	$jrcc	6207860	stocktwits	long setup	0.988097

Sub-task 2:



	id	company	title	Predicted Sentiment Score
0	1144	Ashtead	Ashtead to buy back shares, full-year profit b...	0.999726
1	1145	Shell	EU regulators clear Shell's takeover of BG Group	0.993910
2	1146	Prudential	UK's FTSE has worst day so far in 2015 as BG a...	-0.216989
3	1147	GlaxoSmithKline	GlaxoSmithKline acquires HIV assets	0.930236
4	1148	Barclays	Barclays faces another heavy forex fine	-0.291009
5	1149	Diageo	Diageo Shares Surge on Report of Possible Take...	0.884915
6	1150	Borealis Infrastructure	Borealis Infrastructure putting together new S...	0.775076
7	1151	Burberry Group plc	FTSE 100 falls as China devaluation hits Burbe...	0.821193