CSE352 Final Project Report
Group : Gabriello Lima, Paul Han
Professor Niranjan Balasubramanian

Introduction

Problem Definition- The goal of this project is to perform Sentiment Analysis on Twitter tweets relating to various stocks using sklearn. For input, we simply type in the stock, such as AAPL or TSLA, and the number of days we wish to analyze for.

```
Enter the stock you would like to analyze (Stock Symbol e.g. AAPL, TSLA): MSFT
Enter how many days you would like to analyze it for (0-21): 0
```

For output, we get the accuracy of the various ML algorithms we implemented (Logistic regression, Random Forest, and SVM).

```
Using TfidfVectorizer and Random Forest model:
              precision    recall  f1-score   support

          -1       0.71      0.66      0.68       417
           1       0.82      0.85      0.83       742

    accuracy                           0.78      1159
   macro avg       0.76      0.75      0.76      1159
weighted avg       0.78      0.78      0.78      1159

Accuracy using Random Forest: 0.7817083692838654

Time Elapsed: 21.8528 seconds

Using TfidfVectorizer and SVM:
              precision    recall  f1-score   support

          -1       0.64      0.76      0.69       417
           1       0.85      0.76      0.80       742

    accuracy                           0.76      1159
   macro avg       0.74      0.76      0.75      1159
weighted avg       0.77      0.76      0.76      1159

Accuracy using SVM: 0.7584124245038827

Time Elapsed: 0.1080 seconds

Using TfidfVectorizer and Logistic Regression:
              precision    recall  f1-score   support

          -1       0.80      0.59      0.68       417
           1       0.80      0.92      0.85       742

    accuracy                           0.80      1159
   macro avg       0.80      0.76      0.77      1159
weighted avg       0.80      0.80      0.79      1159

Accuracy using Logistic Regression: 0.8006902502157032

Time Elapsed: 1.1210 seconds
```

However, the most important thing is the observed polarities.

```
Trained and predicted polarity average: -0.21428571428571427
Polarity using TextBlob average: 0.053952345521541954
```

With the observed polarities (which exist in the range [-1, 1]) we can infer the general sentiment surrounding the stock, and perhaps determine whether or not to pursue the stock.

Motivation- Why should anyone care about this? The goal of this project was to provide some insight to the short term general market sentiment of any given stock. Hence, if you're interested in buying some TSLA stock, you could run this program to analyze how people are currently perceiving it. Do people like the stock right now, or do people hate it?

Contributions (Application)- In this project, we did Twitter sentiment analysis by using Tweepy (twitter API) to gather tweets. From there, we isolated tweets that were likely to involve sentiment by using keywords such as "I think" or "I feel" and then ran them through a spam protection algorithm to prevent overfitting to certain retweets or user created bots trying to influence others (something like "🚀🚀🚀TSLA TO THE MOON 🚀🚀🚀" would be filtered out). From there we moved to machine learning using sklearn and textblob. As for textblob, we calculated the average sentiment score of the literal text of according tweets. (e.g. "I am happy" would give us positive and "I am sad" would give us a negative polarity score). Then using sklearn, we trained and tested various models against a large data set of 5791 tweets and targeted the model to have over 85% accuracy. We moved on to predicting average sentiment scores on the tweets that we have received and furnished. We also used DataReader to receive the actual adjusted closing price of wanted stocks for evaluating the results.

Dataset- Our dataset consists of 5791 stock news tweets with text sections, containing emoji-clean, hashtag-clean texts of each tweet and sentiment sections that are scored either 1, which had positive effects on the stock price, or -1, which had negative effects.

Technology- Technologies we have used on this project includes:

- Tweepy: to get required tweets
- TextBlob: for the literal sentiment of the tweets (e.g. happy text vs sad text)
- Sklearn: for processing our dataset in terms of training, testing and predicting
- DataReader: to get the adjusted closing price of stocks.

Description- This project consists of two python files: main.py and tweets.py

- main.py - This file handles all of the machine learning elements and prints out the classification report and accuracy scores from training three different models: RandomForest, SVM, and Logistic Regression. It also prints out the average sentiment scores using textblob and the predicted sentiment scores from our model as well as the actual adjusted closing stock prices within our inputted days.
- tweets.py - Using Tweepy (twitter API) we gathered tweets that were likely to involve sentiment by using keywords such as "I think" or "I feel" or "I am." From there, we ran a

spam protection algorithm to prevent overfitting to certain retweets or user created bots trying to influence others. We also removed “garbage” values from the tweet, such as emojis and links, to better cater towards the ML. The getTweetsWrapper() function takes the ticker and number of days as parameters and does all of this to return the tweets in an array of string format for the ML.

Evaluation-

As for our results, a negative polarity average would mean a decrease in price and a positive polarity would mean an increase. We also had the polarity of the literal text using Textblob to compare if they are correlated in any ways with our own models. We expected that the negative polarity score of our own model would most likely predict the increase/decrease in stock price correctly. We tested the Tesla (TSLA), Apple Inc. (AAPL), and Facebook (FB) multiple times and achieved about 80% accurate prediction of increase/decrease. We concluded that the 20% inaccurate predictions could have multiple reasons such as insufficient number of tweets, false tweets that people post, etc. Another factor we observed is that sometimes, a positive textblob score (sentiments of the literal text) does not necessarily have a positive effect on the stock price increase/decrease. We predicted that this was due to false tweets that people post.

```
Enter the stock you would like to analyze (Stock Symbol e.g. AAPL, TSLA): TSLA
Enter how many days you would like to analyze it for (0-21): 2
```

```
Trained and predicted polarity average: 0.053929121725731895
Polarity using TextBlob average: 0.10980867526539898
Stock information:
                   High         Low        Open       Close    Volume   Adj Close
Date
2021-05-26  626.169983  601.500000  607.559998  619.130005  28639300  619.130005
2021-05-27  631.130005  616.210022  620.239990  630.849976  26314300  630.849976
2021-05-28  634.859985  622.380005  628.500000  625.380005  10403307  625.380005
Adjusted Closing Price in the beginning of analysis: 619.1300048828125
Adjusted Closing Price in the last day of analysis: 625.3800048828125
Difference in Adjusted Closing Price: 6.25
```

Below is an example of the inaccurate case where we predicted a positive polarity score but decrease in the stock prices.

```
Enter the stock you would like to analyze (Stock Symbol e.g. AAPL, TSLA): SNAP
Enter how many days you would like to analyze it for (0-21): 0
```

```
Trained and predicted polarity average: 0.6
Polarity using TextBlob average: 0.11108333333333334
Stock information:
                 High        Low       Open  Close   Volume  Adj Close
Date
2021-05-28  62.255001  61.075001  61.075001  62.25  5207527      62.25
Adjusted Closing Price in the beginning of analysis: 62.25
Adjusted Closing Price in the last day of analysis: 62.25
Difference in Adjusted Closing Price: 0.0
```

```
Enter the stock you would like to analyze (Stock Symbol e.g. AAPL, TSLA): FB
Enter how many days you would like to analyze it for (0-21): 1
```

```
Trained and predicted polarity average: 0.03597122302158273
Polarity using TextBlob average: 0.12091656977448345
Stock information:
                  High         Low  Open       Close    Volume   Adj Close
Date
2021-05-27  333.779999  326.760010   328  332.750000  20466700  332.750000
2021-05-28  332.868408  329.329987   331  331.920013   5778314  331.920013
Adjusted Closing Price in the beginning of analysis: 332.75
Adjusted Closing Price in the last day of analysis: 331.9200134277344
Difference in Adjusted Closing Price: -0.829986572265625
```

Conclusion- To conclude, we learned a great deal in this project. We learned how to properly query with twitter API. We also cemented our knowledge in machine learning basics from HW3. We also learned how to deal with large datasets that weren't necessarily right in front of us. Lastly, we got familiar with another sentiment analysis program, textblob. Some future work areas could be improving the twitter querying, or to use some more advanced ML tricks, or even a larger data set to train on. We could also advance the functionality to try and predict price movements as well.

Sources (Inspiration for the project)

1. Stock Prediction Using Twitter Sentiment Analysis
2. SemEval-2017 Task 4: Sentiment Analysis in Twitter

3. Stock-Market Sentiment Dataset