Stock price prediction based on financial news using lexicon-based sentiment analysis

Hanbo Yu

hanbo.yu@queensu.ca

1. Abstract

The primary goal of this project is to predict the stock price movement based on the sentiment analysis of financial news. Stock price prediction has a great influence on quantitative finance. To implement the solution, the sentiment analysis part is processed by a lexicon-based model-VADER while two different traditional machine learning regression methods-random forest and XGBoost are implemented for better performance as well as root mean squared error as the evaluation metrics. The study shows that compared with the traditional regressing approach, it is not quite reliable to predict the stock price only with sentiment analysis output due to various possible reasons.

2. Introduction

The stock market is known for being highly volatile and potentially high return. It is hard to predict what will happen to the market every second. Nowadays, large financial institutions are seeking quantitative methods to forecast the price of the stock, two common methods are technical analysis, which considers past price and volume to predict the future price, and fundamental analysis involves analyzing its financial data to get some insights, which involves analyzing its financial data to predict the market trend. Sentimental Analysis of Financial News is an approach that follows the fundamental analysis technique by applying machine learning techniques to classify financial news into different categories. Then these data can be used for predicting the stock price.

As an investor, daily financial news is an important resource to acquire the latest information about the market and individual stock to help decision making. A typical example is when US Federal Reserve announced that they would hold a meeting to discuss interest rate adjustment, the whole market would start to fall. Such examples include the new Apple product release that raised Apple’s stock price and Elon Musk’s announcement of his confidence in Bitcoin, which indirectly raised Tesla's stock price as he said to the public that Tesla had bought a great number of Bitcoins. Therefore, my idea is to use this information not only to predict the stock trend but also its price.

The contribution of this paper is to propose a new aspect to predict the stock price by using the sentiment analysis only and comparing it with the traditional approach to see if this is a good idea. There are some challenges during the research process of this topic:

1. In the beginning I was thinking to use reinforcement learning to predict the price, however, I couldn't implement it as it requires far more knowledge than what I have learnt now, and I don't have enough time for it.
2. Most of the study is to predict the trend instead of the price, I don't know if predicting the exact price is worthy of research

3. Problem Statement

To achieve the goal, the problem is split into several small tasks:

1. There are news, opinions and tweets related to the stock market, what dataset should be used?
2. How to choose a model to conduct sentiment analysis?
3. How to properly use the data for prediction?
4. What models will be used for prediction?
5. What method will be used to evaluate the proposed approach?

4. Proposed Solution

For sentiment analysis, I used VADER as the language processing model to score each news. It is an unsupervised learning model, which means we don't need any labels for the dataset to train it as in the real world, once people get all the data from social media or news websites, they don't have time to label it, or the cost can be expensive. Therefore, unsupervised learning can be a better approach compared to supervised learning.

Once the sentiment scores are output, two traditional machine learning approaches – random forest and XGBoost are applied on top of the scores for regression prediction. Compared with deep learning, the traditional machine learning approaches are easily interpretable but have lower accuracy. In my opinion, the purpose is to verify the idea so that we can first use the machine learning approach for a general result. If the bottleneck is the regression model, the deep learning model can replace the original one.

VADER

VADER (Valence Aware Dictionary for sEntiment Reasoning) is a lexicon and rule-based model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. Introduced in 2014, VADER text sentiment analysis uses a human-centric approach, combining qualitative analysis and empirical validation by using human raters and the wisdom of the crowd. [1]

5. Experiment and Results

5.1. Dataset

After careful selection, the dataset is acquired from https://www.kaggle.com/datasets/lykin22/stock-headlines

Daily News for Stock Market Prediction.

This dataset provides 25 headlines of the news for each day and the stock numerical data for the same day.

DJIA_table.csv: This has all the stock numerical data from open price to adj close price. All the data is downloaded directly from Yahoo Finance.

Stock_Headlines.csv: This combined dataset has 27 columns. The first column is "Date", the second is "Label" for trend prediction, and the following ones are news headlines ranging from "Top1" to "Top25".

5.2. Experiment

Step1. Preprocessing and cleaning

The two csv files are first combined into one by adding the extracted close price of each day to the news file. Then the missing headlines are replaced with whitespace as there is no need to drop these data since sentiment analysis will give 0/Neutral to the empty string. Since the data is fetched from the internet by the crawler, the HTML character such as ‘b’ and ‘\’ should be cleaned from the original string. I didn’t do much preprocessing since VADER doesn’t just do simple matching between the words in the text and its lexicon. It also considers certain things about the way the words are written as well as their context. Preprocessing such as punctuation removal and stemming may cause the bad or wrong result output.

Step2. VADERize the data

After the data is cleaned, we use VADER to give each headline a score for a day and calculate the mean compound value as the sentiment analysis score for that day. The process is repeated for each row, and we have three more columns of data: compound_mean, compound_max and compound_min.

Step3.Train the regression model with only sentiment analysis score and evaluate

The data is split into train and test data, with a rate of 0.8/0.2. Two models have their own parameter grid and gridsearchCV is used with eight-fold of cross-validation to find the best hyperparameter. The result is evaluated by the root mean squared error function.

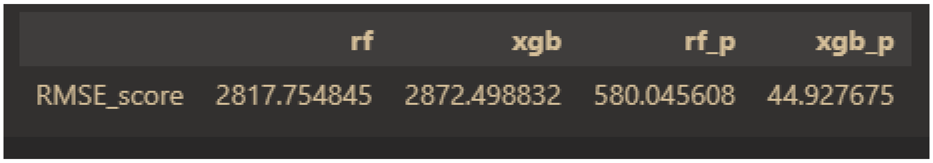
Step4. Train the regression model with sentiment analysis score and the previous data and evaluate

This time I added the sentiment analysis data as well as the close price data for yesterday and the day before yesterday. The training process is the same as step3.

Step5. Predict

All the trained models will be used to predict test data. The output will be put into the same pandas dataframe for horizontal comparison.

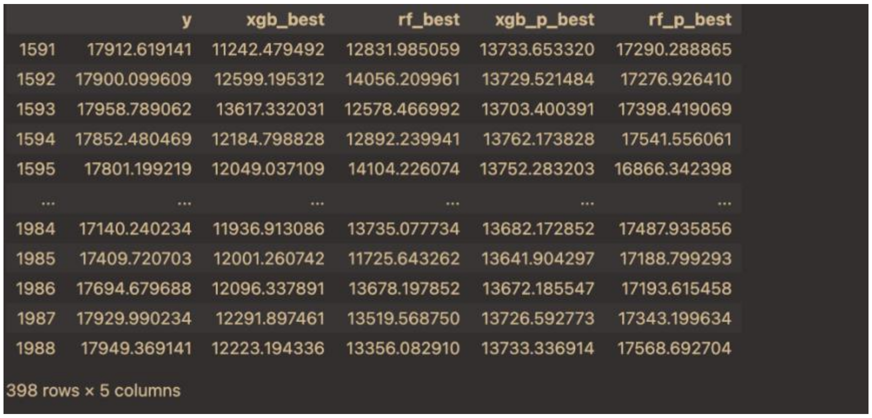
5.3. Result



	rf	xgb	rf_p	xgb_p
RMSE_score	2817.754845	2872.498832	580.045608	44.927675

Figure 1. The RMSE comparison table

The RMSE value (Figure 1) shows that for both XGBoost and random forest, the sentiment analysis score only method has a big loss of estimation. Observing the comparison table (Figure 2) also confirms the poor prediction of this approach. However, the general trend of the prediction is in the right direction. After adding the look-back data, both models are improved a lot based on the RMSE value. In the prediction table, the random forest regressor even shows an output close to the result.



	y	xgb_best	rf_best	xgb_p_best	rf_p_best
1591	17912.619141	11242.479492	12831.985059	13733.653320	17290.288865
1592	17900.099609	12599.195312	14056.209961	13729.521484	17276.926410
1593	17958.789062	13617.332031	12578.466992	13703.400391	17398.419069
1594	17852.480469	12184.798828	12892.239941	13762.173828	17541.556061
1595	17801.199219	12049.037109	14104.226074	13752.283203	16866.342398
...	...	...	...	...	...
1984	17140.240234	11936.913086	13735.077734	13682.172852	17487.935856
1985	17409.720703	12001.260742	11725.643262	13641.904297	17188.799293
1986	17694.679688	12096.337891	13678.197852	13672.185547	17193.615458
1987	17929.990234	12291.897461	13519.568750	13726.592773	17343.199634
1988	17949.369141	12223.194336	13356.082910	13733.336914	17568.692704

398 rows × 5 columns

Figure 2. The comparison table

This result also shows some problems. Although the random forest has a good performance in predicting the value, it didn't give an accurate trend. For example, in 1593 and 1594, the random forest gave a 200-300 point raise while the actual value is going down. The same problem also occurs on XGBoost. The problem must have occurred in the sentiment analysis process. Then I plotted the distribution of the score (Figure 3), most of the compound_mean is below 0, which is weird because if the stock market follows this score, it should go down almost every day.

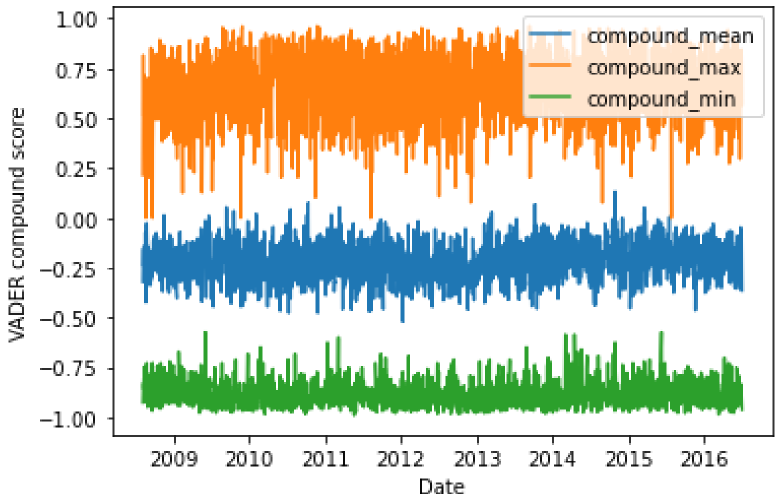


Figure 3. VANDER compound score distribution

6. Conclusion

The result shows that only simply using a sentiment analysis score to predict the price of the stock is not efficient. For future work, the score needs to be converted to the percentage of increasing or decreasing the price, which needs some more concrete formulas to implement this task. A possible reason for the poor prediction could also be as VADER is optimized for social media, not for formal languages, the default lexical is not suitable for formal financial news. It could be necessary to increase the VADER sentiment lexicon with financial words for a greater identification of financial terms. A more powerful model such as FinBERT may be a better approach to processing the financial news. The conclusion is that the sentiment analysis for financial news is more suitable for predicting stock trends rather than price.

7. Reference

1. Hutto, Clayton, and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." *Proceedings of the international AAAI conference on web and social media*. Vol. 8. No. 1. 2014.