PREDICTING TOP MOVERS IN STOCK MARKET USING MARKET NEWS SENTIMENT ANALYSIS AND RNN-LSTM

[#1]Akash Kandarkar, [#2]Vedant Rawat

[1]aakandar@syr.edu

[2]vrawat@syr.edu

[#12]School of Information Studies, Syracuse University

Abstract— People who are able to predict the future are often compared with god, as no common entity can predict the future correctly on a regular basis. Predicting the future correctly for financial gains can make a person get everything from nothing. As per Arthur Clarke's third law, "Any sufficiently advanced technology is indistinguishable from magic" [1]. Using complex machine learning algorithms and Artificial Neural Networks, we can not only mimic the human brain, but also achieve what a human brain could never do humanly i.e., predict the stock prices and more importantly predicting Top Movers stock for maximum financial gains, which is nothing less than magic. To support this idea, RNN architecture LSTM models are used in combination with the market sentiment information and stock market quantitative data to understand the effect and further categorize stocks dynamically into whether the stock will be a Top Mover; with good enough percentage change in stock value from the previous day or will the stock perform in a regular manner.

Index Terms— RNN: LSTM, Sentiment Analysis, Stock Market Prediction, Top Movers

1 INTRODUCTION

Primary objective as part of this project, was to be able predict if a stock will make large movement in terms of price, to be categorized as top mover, when a user provided a particular stock on a particular date. For this, we used market sentiment scores generated from top 25 headlines on a particular day. This data was combined with the stock market quantitative data from Yahoo Finance where a user could dynamically select stock and the LSTM models would predict if the stock will perform well to be categorized as a Top Mover stock based on threshold percentage also provided by the user.

Further, we have experiemened with different types of market sentiment scores and various LSTM model hyper-parameters to get the least prediction error and best model performance. For this we experiemened with Real -Time market News scrapped from web, Kaggle market news and Sentiment scores data [2] and Yahoo Finance stock market data using Python libraries. We will be further, going through these experiments in detail.

2 PROBLEM AND DATA DESCRIPTION

Market sentiment has a considerable effect on the stock market prices, and in this project, we would further like to predict the "Top Movers" stocks (stock with the highest percentage change in a day) using Sentiment analysis. This market sentiment data is derived using the expert sentiment scores obtained from Kaggle dataset [2].

Further, we will also be using the stock market data along with the previously gathered sentiment scores to finally predict which stocks might fall in the "Top Movers" category. This project could bring a better investment strategy for intra-day traders, where traders could gain good monetary benefits in one day due to the highly volatile nature of certain stocks partially impacted by market sentiment.

This problem solution can also be beneficial for investment risk analysis by understanding which stocks could go down rapidly, as the percentage change of stock is being predicted.
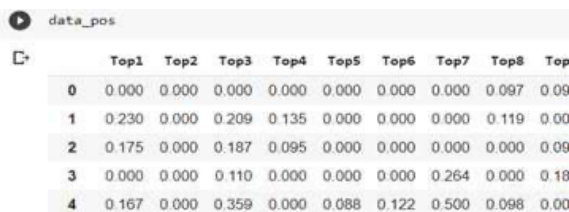
For this project, sentiment scores could be generated using the text data from Kaggle which comprises the news headlines from 2008-08-08 to 2016-07-01. There is a total of 27 columns in the CombinedNewsDJIA.csv [2]. However, we had to switch to considering the expert label column for further analysis and discard the sentiment scores we had generated. Further, we used the stock data from 2008-08-08 to 2016-07-01 downloaded using Yahoo Finance python library. The columns are OPEN, HIGH, LOW, VOLUME, CLOSE, ADJ CLOSE. We dropped the CLOSE column as it is same as Adjusted close price column. Combining this data, we performed extensive experimentation to get the best results.

3 APPROACH/ALGORITHM

3.1 Vader Sentiment v/s Expertise Sentiment

This section contains the methodology which we implemented for the project in order to attain quality results. Since one of the goals of this project was to check the effect of the sentiment on the stock values, we had to have the sentiment scores/values of the market news headlines.

Vader stands for Valence Aware Dictionary for Sentiment Reasoning. Vader is a model which is used to generate the polarity of the text data for checking the sentiment of the text data. For this project we generated the sentiment scores (positive, negative, neutral and compound) using Vader. The effect of these scores on the overall stock price prediction was then compared with the expert sentiment. The expertise sentiments were collected from the same Kaggle dataset [2]. The performance of the Vader generated sentiment score and the expert generated sentiment were compared so that we could achieve the best result from the models and also check the effect of the sentiment for the prediction of the stock prices. The results will be discussed in the results section (refer 4.1 section).

data_pos									
	Top1	Top2	Top3	Top4	Top5	Top6	Top7	Top8	Top
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.097	0.09
1	0.230	0.000	0.209	0.135	0.000	0.000	0.000	0.119	0.00
2	0.175	0.000	0.187	0.095	0.000	0.000	0.000	0.000	0.09
3	0.000	0.000	0.110	0.000	0.000	0.000	0.264	0.000	0.18
4	0.167	0.000	0.359	0.000	0.088	0.122	0.500	0.098	0.00

Figure 1 Positive Sentiment scores using Vader

Figure 1 shows the generation of the positive sentiment scores for the individual headlines using Vader. Similarly, neutral, negative and compound scores were also generated using Vader.

3.2 Sentiment v/s Non-Sentiment Data

Based on the results of section **3.1** and **Figure 3**, we were able to deduce that the expert labelled sentiment scores were more helpful for achieving our goal. Now that the sentiment scores to consider were finalized, we tried to prove that the sentiment scores are actually helpful for the top mover prediction task. For this we considered 10 different stocks some volatile in nature and some stable in nature and checked which performed better and gave an evaluation metric score for the task. Based

on the results, in section **5.2,** we could observe that the majority of the stocks had an impact of sentiment scores while predicting stock prices. These observations were helpful for our deep learning RNN-LSTM model favoring the consumption of the sentiment information and hence we considered to include the sentiment scores for the further tasks in the project (refer 5.2 section).

3.3 Dynamic Stock List & Threshold

Intially, we were experimenting with some selected stock, but we wanted to make this project more dynamic in nature so that it could be used for analyzing and predicting any stocks and any threshold values set by any user for Top Movers category. So, we decided to get user inputs for three dynamic variables:

- List of stocks to be checked
- Date Range
- Top Mover Percentage Threshold

Using the above three dynamically set run time variables, we tried predicting the Top Mover category of stocks from a given set of stocks for the provided percentage threshold. The date variable had some limitations as we had to use the Kaggle Sentiment scores [2] for our models which had data from year 2008 to 2016 only, as real-time news could not be stored and most of the websites had similar news on them giving us co-related news sentiment score features.

Figure 2 Runtime user input option

4 DATA

For any machine learning tasks to work effectively, we must modify the data appropriately. Preparation of the data was carried out before implementing the deep learning techniques in the following manner. Standardization: we needed to enforce standardization because the dataset comprised of values with different scales. Therefore, to avoid giving features extra weightage we im-

plemented standardization for the overall data. Next, we performed the train-test split. This helped in checking how well the model performs on unseen data. Next, we incorporated the time lags which were essential for performing the time-series analysis and capturing the lag effect.

Finally, we reshaped the dataset in order to fit it into the deep learning model.

The data had to be limited from year 2008 to 2016 as mentioned earlier, as scrapping live news data from the web was not possible due to the news sentiment score features being correlated in nature, as similar news would be published on multiple news websites. Also, we could not trust getting news on a daily basis for each stock due to which we used top 25 headlines data which was available on a daily basis and would be applicable to the overall market.

5 RESULTS

5.1 MSE Results Vader v/s Expertise Sentiment Scores

Figure 3 Vader v/s Expertise Sentiment

Figure 3 results show the comparison of MSE scores while experimenting the Vader model scores against the Expert labelled sentiment scores, based on these results we discarded the Vader generated sentiment scores and used the Kaggle labelled sentiment scores data [2].

5.2 Comparison of MSE scores for models using Senitment scores and models without using sentiment scores

Table 1 shows the initial experimentation results of MSE scores of prediction models using the market news sentiment scores and excluding the usage of market sentiment scores in LSTM prediction model. Highlighted in grey shows stocks which improved using the sentiment scores in LSTM model
(refer: ComparisonSentimentVSNonSentiment python file)

Table 1 Comparison of MSE scores

Stocks	WithoutSenti-ment Scores MSE	WithSentiment Scores MSE
NKTR	0.0043	0.0036
AMD	0.0021	0.0017
KGC	0.0051	0.0042
ZION	0.0012	0.0013
AMGN	0.0016	0.0015
SOHU	0.0062	0.0064
T	0.0091	0.0091
AAPL	0.0039	0.0031
BLIN	0.0071	0.0073
WMT	0.0295	0.0321

6 DISCUSSION – FUTURE WORK

Multiple tasks were possible, but those tasks could not be carried out either due to high computational efforts or feasibility challenges keeping in mind the scope of the project. Below are some of the tasks which we think could further help this project:

1) Usage of dynamic real time news data was not possible, as the news data is not available on a daily basis, but if we could store the news data for individual stocks on a daily basis that would give us better Sentiment scores and possibly a better prediction model.

2) The tuned hyper-parameters for the LSTM models had to be restricted to few combinations, due to higher computational time. If proper computational resources were available, we could optimize the predicition models in a much better way by testing larger combinations of hyper-parameters.

3) This project module could be put onto cloud hosted website for better user interaction and usage, giving the users a GUI to use the prediction results and choose the list of stocks and Top Movers threshold.

4) Although, we tried to generate our own Sentiment scores using Vader model, we had to use an expert labelled sentiment scores from Kaggle [2] for our project as it gave us better results. This issue could be tackled, if a better sentiment score generating module is built for LSTM model consumption.

7 APPENDICES

7.1 Individual stock hyper-parameter tuning:
After the initial implementation, we were able to perform hyper-parameter tuning on individual stocks, instead of using the DJIA index prediction model hyper-parameters for each stock. This required higher computational effort due to which

we had to limit the hyper-parameter combinations to 48 possible models for each stock.

Figure 4 Individual stock hyper-parameter tuning

By checking these hyper-parameter combinations as shown in **Figure 4,** we could further get better results if better computing resources were available.

7.2 Negative Threshold Alert
Figure 5 shows implementation of a small alert module for users, when the stock prices go down on the negative side, below the absolute value of the threshold percentage.

Figure 5 Negative Threshold Alert

8   CONCLUSIONS

We used market sentiment data points to generate Sentiment scores for each day and used the stock market data to predict if a stock would be a Top Mover or not, based on the user defined threshold and stocks selected by user.

Mostly, the results indicated that using market sentiment for such prediction was really helpful and getting more market news information for such predictions could define new success levels in stock market prediction and make this effort lucrative.

REFERENCES
[1] https://en.wikipedia.org/wiki/Clarke%27s_three_laws (Clark's three laws) Third Law by Arthur C. Clarke (1917-2008), downloaded on 12-08-2020
[2] Sun, J. (2016, August). Daily News for Stock Market Prediction, Version 1. Retrieved [12-08-2020] from https://www.kaggle.com/aaron7sun/stocknews

AUTHORS
#1 Akash Kandarkar
Student, CIS.731 M001 FALL21 Artificial Neural Networks
#2 Vedant Rawat
Student, CIS.731 M001 FALL21 Artificial Neural Networks