Stock Market Trends Prediction After Earning Release

Ran An - 06117810 (anran), Chen Qian - 06116065(cqian23), Wenjie Zheng- 05999176 (zhwj814)

***Abstract*— This project involves discovering how the company's stock performs in after-hours on the release day of its quarterly financial report. Project aims to use a variety of machine learning models to make predictions regarding the stock change based on the factors that reflect and affect investor reaction, namely, media coverage of financial news and companys quarterly financial report. The goal is to help finance companies to build their trading strategies, and furthermore, identify the key factors for company valuation.**

I. INTRODUCTION

It is said that accurately predicting a chaotic system like stock market is basically impossible, however, it is still a significant progress if machine learning and proper model selection can lower the risk and raise the probability of success. Public companies release quarterly and annual earnings statement, reporting financial items such as net income, earnings per share, and total revenue. By analyzing earnings report and compare against company's previous financial performances, it is possible to evaluate the financial health of the company and determine the corresponding investment strategies. General public shareholders' trading strategies can be influenced by analysis and articles from mainstream media. The project aims to take the financial report factors, consensus forecast and the sentimental analysis of earning report related articles into account to explore the possibility to predict stock price during the after market period, further, the stock price on the next trading date.

II. PROJECT MILESTONE

This section describes what we have achieved and what the future challenges are. It will be deleted in the final report.

Up till now, we have finalized the input feature selections for our trading strategies model as well as desired formats, and finished collecting all digital data from earnings report and stock price information to form our training set, test set and cross validation platform. We are actively downloading articles to be processed by Stanford NLP and we have already implemented the python script to pre-process the sentiment analysis according to NLP to the format we need for the input to our NLP learning model.

Meanwhile we have written prototype in Matlab for the learning model that is based on financial digital data using logistic regression. This helped us to understand the basic accuracy of our model assumption. We have been actively discussed about the model strategy for NLP and started to prototype it in Matlab as well, but there is no result for demonstration just yet. Also we have been discussing what we should learn from this project and decide that the accuracy is not the only outcome we are pursuing, but should analyze what are the factors that limit our design and how we can make better prediction. We came across some analyzing procedures that will help us to better understand the system behavior that will allow us to make (if possible) improvements to our design.

The outstanding challenges are to design the NLP learning model and how to combine both classifiers together to generate the final outputs. The error analysis procedures are listed but we need to integrate probing to our model so we are able to understand our model more. Since one of the limits we are experiencing is the size of sample set so we will need to decide if expanding sample size is necessary after doing bias/variances analysis.

III. MATERIALS AND METHODOLOGIES

A. Feature Selection

According to preliminary studies on previous years projects and related research results on trading strategy, data with less dimensions or from single source, i.e financial data from earnings report, is very challenging and nearly impossible to provide reliable and solid stock price prediction. Therefore multiple formats/sources of data have been selected to strengthen our model training.

1.Features from financial report: Companies financial reports include various of financial digits to reflect their financial achievements but not all of them are well concerned by the public or investment companies. We have consulted professional people with financial and economic background and researched online to obtain a list of financial factors in earnings report that considerably reflect company's operation and financial status, as well as the factors that most investment companies and financial analysts will look into to adjust their investment strategies. This list includes: total assets, total liability and equity, total revenue, net income (profit) and ear per share (eps).

2.Market Surprise: Market analysis companies usually provide consensus forecast on eps and the surprise here is the delta between actual eps and the forecast eps. The magnitude of surprise is one of the major factor that affects the aftermarket stock price. Therefore our team also record the esp forecast to obtain the market surprise, which is another features inputting to our model.

3.Sentiment Analysis result: The release of earning report introduces instant reaction to stock market and mainstream media will report news and write articles to analyze the report and even comments on companys future business strategies. These news and articles from mainstream media offer guidance of trading, affecting market confidence and even providing consensus to influence stock rating. Therefore

analyzing and decoding the characteristics of articles can provide valuable insights about how media coverage can affect investors. We decide to decode the sentiment of articles using Stanford NLP toolkit. An python script is implemented to invoke this toolkit with a txt formatted article as its input argument. The toolkit is able to output sentiment analysis score in the range from 0 - 4 of each sentence. The score and sentiment mapping is described in the following table:

TABLE I
SENTIMENT SCORE TABLE

Score	Sentiment
0	Very negative
1	Negative
2	Neutral
3	Positive
4	Very Positive

The toolkit is configured to output the sentiment results in JSON format and our script can get access to and process the JSON file using existing JSON library and obtain a normalized sentiment score distribution from 0 to 4 and input to our NLP machine learning model. The following figure is an visible result from the sentiment analysis of a sentence from one earning report news

The input to our learning model will be a vector with feature space 5 and and input from the example above will be X = [0.15, 0.58, 0.24, 0.02, 0.1]

B. Data Sources

1. Company Selections: A stock pricing prediction model will be highly inaccurate with poor generalization if we plan to train it with companies involves in different types of business because investors has different evaluation and expectation with companies in different industries. Narrowing down the adaptability of the model by choosing the companies that are only involved in Information technology is able to reduce the generalized error and improve the accuracy of the prediction. We have considered **Apple Inc, Tesla Motors, Google, Microsoft, Amazon, Facebook, Yahoo, Twitter and Oracle** as our targeted companies. Their businesses have relative large overlap and we have reasons to believe that their stock market reactions after earning report will have reasonable correlation and our model will be able to find out.

2. Digital Data: The digital data to be collected include historical stock prices, actual eps, forecast eps and earning report features. As this information are public so that we are able to obtain accurate results from Internet directly. Historical stock prices are collected from *Yahoo finance* and actual/forecast eps are from *StreetInsider.com*. Earning reports are collected from *Last10K.com* which formalizes the reports to a standard format thus makes key features easy to be collected.

3. News and Articles: Taking public influence, real-time property and authenticity into consideration, Earnings report-related news are downloaded from reliable mainstream media such as the *Wall Street Journal, Bloomberg, NASDAQ.com, Yahoo Finance and etc.*

C. Data Verification and Preprocessing

All date sources for model feature collection have been cross-validated by comparing data from different sources to verify correctness. Since company data and stock prices are volatile, data preprocessing may be necessary to normalize its mean and variance. It is obvious that article cannot be the input to our model directly therefore the python script is written to collaborate with Stanford NLP library to output normalized sentiment results to process our data.

IV. MODEL SELECTION

As stated in aforementioned introduction, we have two sets of input features, key items from quarterly earnings report, and articles from professional financial news as media reactions to the announcement. Therefore, the whole model is built up from two sub-models with separate input.

The process flowchart below shows explicitly how the model works. System A takes five key items from the financial report as feature inputs, and conduct data preprocessing before applying them to our learning algorithm. Same in system B, where feature input is a score vector with length of five from sentimental analysis, normalized in data preprocessing stage. To finalize the whole system, another learning algorithm (system C) should be applied to previous classifier results and conclude the final prediction.

In terms of model selection for both systems, we plan to experiment with three algorithms, namely logistic regression, boosting and Support vector machine. System B is quite similar to system A, as the only difference lies in the input feature, so we decide to start with the same algorithms.

However, currently, our biggest challenge is to integrate the classifier results from two models to generate the final prediction. We plan to apply hyperparameter vector, which is determined by the cross validation set to achieve the integration. For system C, only support vector machine is applied due to its strong capability of fitting nonlinear relationship.

In addition, we keep researching on possibilities of other learning algorithms that might be more applicable to our project. For instance, it has been pointed out that the stock market is roughly a chaos system, a nonlinear system with feedback. According to our research, empirically, Recurrent Neural Networks is empirically adaptable for such a system. Therefore, we would like to keep options open at this point.

V. MODEL AND ERROR ANALYSIS

This part has not been thoroughly considered yet but we would like to list some aspects and approaches to analyze our model.

1) Bias and Variances
   One of the limits of our model is that the number of training sample is relatively small and we will perform bias and variances analysis based on the number of

training set to understand if it is possible to improve accuracy/error with larger sample size.

2) Company Selection
   Though the companies we selected have businesses overlapped but based on other factors such as their company history, future business strategy and human resources, market usually have volatile reactions on their financial performance. We will investigate on how data from different company will affect model accuracy. It is valuable to perform cross validation on data from different company since the size of training set is relatively small.
3) Module Error Evaluation
   Since our model consists of multiple subsystems and their inputs is quite different. We would look into the train/general error from individual module to find out the bottle neck of the entire system. Further, investigatioin on how data preprocessing will affect the subsystem behavior and any other potential pre-processing schemes we may be able to use.

VI. CONCLUSIONS

We would like to pre-list what we plan to explore with this model and data.

- Accuracy of our model (training error and generalization error)
- Accuracy of each sub-model system
- If possible, evaluate how each learning algorithm can provide the best prediction
- correlation between each type of input sample with its corresponding label.
- The source of limit to our design
- Possible future improvements

APPENDIX

—

ACKNOWLEDGMENT

—

REFERENCES

[1] R. Socher, A. Perelygin, J.Y.Wu, J. Chuang, C.D. Manning, A.Y. Ng and C. Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

[2] Tomas Mikolov. 2010. Recurrent neural network based language model. Brno University of Technology. Johns Hopkins University.

[3] Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.