Forecasting of Stock Prices Using Machine Learning Models

Albert Wong
Mathematics and Statistics
Langara College
Vancouver, Canada
0000-0002-0669-4352

Juan Figini
Mathematics and Statistics
Langara College
Vancouver, Canada
jfigini00@mylangara.ca

Amatul Raheem
Mathematics and Statistics
Langara College
Vancouver, Canada
araheem00@mylangara.ca

Gaétan Hains
LACL
Université Paris-Est
Créteil, France
0000-0002-1687-8091

Youry Khmelevsky
Computer Science
Okanagan College
Kelowna, Canada
0000-0002-6837-3490

Pak Chun Chu
Mathematics and Statistics
Langara College
Vancouver, Canada
jchu37@mylangara.ca

***Abstract*—Stock price prediction with machine learning is an oft-studied area where numerous unsolved problems still abound owing to the high complexity and volatility that technical-factors and sentiment-analysis models are trying to capture. Nearly all areas of machine learning (ML) have been tested as solutions to generate a truly accurate predictive model. The accuracy of most models hovers around 50%, highlighting the need for further increases in precision, data handling, forecasting, and ultimately prediction.**

In this paper we present the result of our work on high-frequency (every fifteen minutes) stock-price prediction r9y118(eaiiLe)15(y)ease25(or)18o(or)18o9(of)-3c9(wc(w)lfr)e.seb18(eswc(10(orwc(exog

Grants-in-Aid (GIA) Fund at Okanagan College as well as the Post Degree Diploma and the Work on Campus programs at Langara College for supporting our research.

I. Introduction

A profitable stock trading algorithm will benefit from a forecasting system that can produce accurate short-term forecasts. Based on this premise, we propose this research project to leverage our previous experience in building short-term forecasting models using machine learning (ML) algorithms [1]–[5].

The Algorithmic Trading World is a dynamic area involving forecasting competition that aims to encourage the development of new models to predict the stock market's short-term response following large trades.

We contemplate our central idea, "If I know the past price, could I anticipate the future value?"

To confirm our approach, we use stock price data, in fifteen minutes intervals, for Tesla stocks, extracted from the stock market for two years. We explore, clean, normalize, and build initial forecast models with training and testing datasets, then check our models' accuracy and efficiency.

Data Analysts are expected to use Machine Learning methodology to create forecasting models to predict the stock market behaviour based on specific indicators like SP500, stock behaviour, and seasonality.

The following sections provide a short survey of the most relevant literature, then the technical details of our approach (feature engineering, error measures, models, experimental comparisons), our conclusions about predictive accuracy and ideas for future work.

II. Existing Work

There exists an incredibly large body of research papers on algorithmic trading including testing and results of various ML models using neural networks (NN), random forest (RF), support vector regression (SVR), XGBoost, and long short-term memory (LSTM) algorithms. Here we briefly recall the ones that are most relevant and closest to our problem definition: stock-price prediction (independent from, but of course applicable to, trading performance) using structured technical-, structured company- and unstructured natural-language sentiment data.

Many models, such as that seen in [6], attempt to match predictions to real data, but a considerable price and time discrepancy exist between the predicted values and actual values. The discrepancy can be off by several hundred dollars (or any given unit of currency).

Other research groups, like that in [7], tested many neural networks (NN) and ML models to see which is the best for a given data set. They used back propagation NN, radial basis function NN, general regression NN, SVR, and Least Squares-SVR. The testing data was weekly adjusted close price of three individual stocks: Bank of China, Vanke A, and Kweichou Moutai with mean square error (MSE) and mean absolute percentage error (MAPE) as criteria. Though the authors gave largely inconclusive results, it was noted that back propagation had the best results, at least among the tested models, with an MAPE under 5%.

In 2018, Chen and He [8] investigated the reliability of deep learning methods based on a 6-layer convolutional neural network (CNN) at predicting prices on the Chinese stock market. They set the time scale to be a year and input to the opening price, high price, low price, closing price, and the volume for historical stock data sets from the Chinese stock market. The results obtained showed an accuracy of about 73%. Results at this level begin to approach a usable state, but still require fine-tuning to be considered truly reliable and predictive.

Many works have experimented, with success, on stock price predictions with financial news. In [9], the authors examined the relationship between measured sentiment from some media messages about a company and its stock price. The authors used natural language processing (NLP) to perform sentiment analysis on messages from Twitter, called tweets. This research was performed with a shorter duration than what was originally planned and has a noticeably small sample size, but it still highlights the effect of sentiment on market performance as a determining factor to be leveraged for profit.

Weng, Lu, Lang, Megahed, and Martinez [10] sought

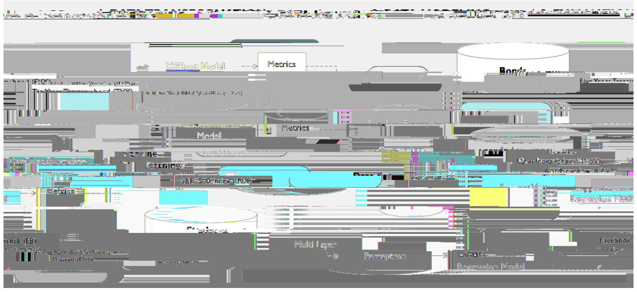

Fig. 1. Model Building Process

case, the model will be trained and re-trained using the “rolling” data set (dropping data from one fifteen minute interval at the “beginning” interval of the time series and adding data on the “most recent” fifteen minute interval.) Forecasts from the model for the next fifteen minute would then be generated and compared to the actuals roughly two thousands four hundred (4 months times 20 trading days each month times 30 trading period each day) times.

From a computational standpoint, this could be demanding if we were to produce forecasts every fifteen minutes for the entire stock universe in a sizable stock market. Therefore, the mean testing time is tracked as part of the evaluation metric. Furthermore, we also develop models and use them to produce forecasts over a longer period (one day or five days). We will discuss Further the training and evaluation process in detail below.

A. Dataset Description

To build the model, we collect data, in fifteen minute interval, numerical data on Tesla’s stock price as well as those for other exogenous variables that could have a material impact. These variables (features) are described in the following:

1) Numerical Features:

- Price of Five-year treasury bond
- Price of Ten-year treasury bond
- Value of Dow Jones Index
- Value of Nasdaq Index
- Value of S&P 500 Index
- Price of Facebook Stock
- Price of Alphabet(Google) Stock
- Price of Disney Stock
- Price of Tesla Stock; Target Variable

To capture the seasonal pattern and other calendar effects on stock prices, we created several indicator features for each fifteen minute interval:

2) One-Up Features:

- Year (3 one-up variables for 2020, 2021, and 2022)
- Months of the year (12 one-up variables)
- Day of the month (31 one-up variables)
- Week day (5 one-up variables for Monday to Friday)
- Hours of the day (6 one-up variables for hours 9 to 16)
- Minute Segment of the hour (4 one-up variable for minute segment 0,15,30, and 45)
- Whether the time period is in Monday morning (1 one-up variable)
- Whether the time period is in Friday afternoon (1 one-up variable)
- Whether the time period is in a ”Pre-holiday” afternoon (1 one-up variable)
- Whether the time period is in a ”post-holiday” morning (1 one-up variable)

For the purpose of this research, the data set for training and testing was created for the period of June 2020 to May 2022.

B. Feature Engineering

Once the data is collected, the min-max normalization process (refer to Equation 1) is applied to all numerical variables.

X

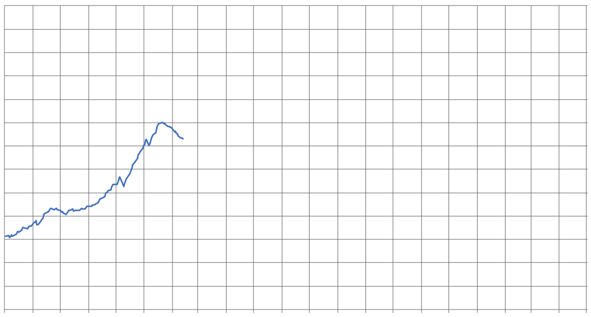

Fig. 4. SUPPORT VECTOR REGRESSION 60 DAYS TRAIN - 15 Mins PREDICTED.

Fig. 5. MULTILAYER PERCEPTRON 60 DAYS TRAIN - 5 DAYS PREDICTED.

and takes their majority vote for classification and average in case of regression. For this project, the RF models built have very simple structure with the following hyper-parameters: number of estimators = 100 and maximum depth = 100. The comparison between the actuals and predicted for all three scenarios is depicted in Figure 8-10.

D. Extreme Gradient Boosting Models

Extreme Gradient Booting (XGBoost) is a popular and efficient open-source implementation of the gradient boosted trees algorithm. Gradient boosting is a supervised learning algorithm, which attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models. For this project, the RF models

TABLE III
Result of RF Models

Model	RMSE	MAPE	MAE	MTT
RF				
Train 60 days Predict 5 days	65.598	5.9	45.649	1.278

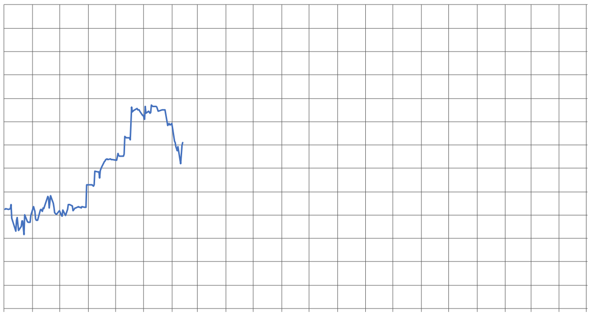

Fig. 8. RANDOM FOREST 60 DAYS TRAIN - 5 DAYS PREDICTED.

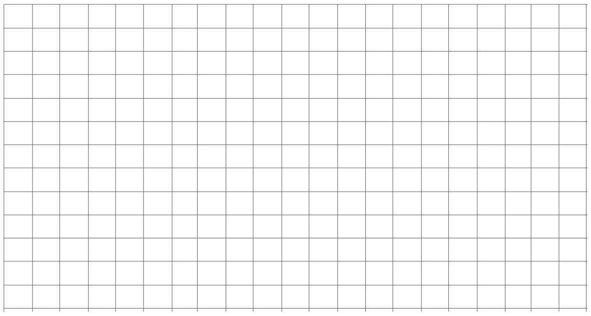

Fig. 9. RANDOM FOREST 60 DAYS TRAIN - 1 DAY PREDICTED.

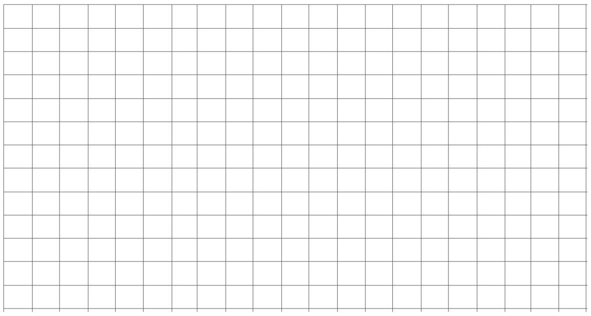

Fig. 10. RANDOM FOREST 60 DAYS TRAIN - 15 Mins PREDICTED.

Fig. 11. XGBOOST 60 DAYS TRAIN - 5 DAYS PREDICTED.

model is used to generate predictions for the next days comparing to those for the next five days

3) Creating a prediction with any of these models takes less than 2 seconds by projection.
4) It is possible to predict the whole week of stock in 15 minutes in less than 3 minutes using Google Colab.

In summary, we have learned how a careful choice of input variables and a rational comparison of basic ML engines can produce higher and somehow *explainable* prediction quality than has been previously published. On-the-fly training is possible at our time scale, even with non-optimized implementations so this counter-intuitive aspect of our approach could also be novel.

V. Future Work

The research will follow two defined paths. Building on the work presented here, fine-tuning some of the models and building more sophisticated machine learning models such as Long Short Term Memory (LSTM) models would be a natural extension. Adding other exogenous variables (features) such as short-term interest rate indicators (for example 2 years treasury bond prices), inflation related indicators (for example the price of gold) and sentiment data from social media and/or news reports would be another. Other possible and minor refinement would be the consideration of impact of dividend and other shareholder related events. We will also investigate the possible use of parallel or distributed computing on training time relative to the frequency of our data collection and stock-price predictions, and also the size of our models.

VI. Conclusions

In this paper, we presented results on forecasting the price of Tesla stocks using a historical price time series as well as several exogenous variables (features) that are considered relevant from a stock analysis standpoint. Four simple machine learning algorithms: support vector regression, multilevel perceptron, random forest, and XGBoost, were implemented to validate the appropriateness and accuracy of using these features for forecasting of stock prices. The outcome of this experiment confirms that this approach has merit even with machine learning models with simple structure. Future research would therefore focus on the inclusion of other relevant economic variables, such as inflation and short term interest rate, as well as sentiment data from social media and financial news sources. As well, the implementation of more sophisticated machine learning algorithms such as Long Short Term Memory would also be explored.

References

[1] G. Hains, C. Mazur, J. Ayers, J. Humphrey, Y. Khmelevsky, and T. Sutherland, “The WTFast’s Gamers Private Network (GPN®) Performance Evaluation Results,” in *2020 IEEE International Systems Conference (SysCon)*. IEEE, 2020, pp. 1–6.

Fig. 12. XGBOOST 60 DAYS TRAIN - 1 DAY PREDICTED.

Fig. 13. XGBOOST 60 DAYS TRAIN - 15 Mins PREDICTED.

[2] C. Mazur, J. Ayers, J. Humphrey, G. Hains, and Y. Khmelevsky, “Machine Learning Prediction of Gamer’s Private Networks (GPN®S),” in *Proceedings of the Future Technologies Conference*. Springer, 2020, pp. 107–123.

[3] A. Wong, C. Chiu, G. Hains, J. Humphrey, Y. Khmelevsky, C. Mazur, and H. Fuhrmann, “Gamers Private Network Performance Forecasting - From Raw Data to the Data Warehouse with Machine Learning and Neural Nets,” 2021. [Online]. Available: https://arxiv.org/abs/2107.00998

[4] A. Wong, D. Joiner, C. Chiu, M. Elsayed, K. Pereira, Y. Khmelevsky, and J. Mahony, “A Survey of Natural Language Processing Implementation for Data Query Systems,” in *2021 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*, 2021, pp. 1–8.

[5] A. Wong, C. Chiu, A. Abdulgapul, M. N. Beg, Y. Khmelevsky, and J. Mahony, “Estimation of Hourly Utility Usage Using Machine Learning,” in *SysCon 2022 (accepted for the publication)*, 2022.

[6] I. Parmar, N. Agarwal, S. Saxena, R. Arora, S. Gupta, H. Dhiman, and L. Chouhan, “Stock Market Prediction Using Machine Learning,” in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, 12 2018, pp. 574–576.