Peiwen Tang & Minji Kim

Algorithm Comparison in Stock Price Prediction

Introduction:

Stock Price prediction is a kind of attractive forecast to perform as it can help to win money, but not as realistic, because there are so many influencing factors that affect the actual stock price. In this experiment, we are trying to forecast the stock price, not for the precise number, but to see if it is possible to predict a trend or a future range of stock market price using various algorithms. The dataset used for the experiment involves the past stock prices of the target company, which shows the historical performance of the company and if it will perform better or worse in the future, and the stock prices of other companies in the industry. The performance of other companies is a representation of both external factors and how the substitution or complementary will affect the stock price of the target company. We used Auto Arima, Prophet, and LSTM to perform prediction on the time series data, which is the past performance of the target company, and the Regressor to do the cross-company prediction, then use ANN to do a combination of prediction of past and cross-company forecast, and compare the run time, RMSE and accuracy of all the algorithms to see which one gives the most optimal performance.

Motivation:

Before doing the research and experiment, we already have in mind that forecasting on stocking marketing won't give us optimal prediction like training and learning in fields like image identification. Stock market price is determined by lots of external factors, like natural disaster or government policy. The trend or fluctuation of the stock market price could be even caused by no reason. However, we still want to examine if we can see some kind of pattern of the stock market price change using some kind of prediction, that gives us a hint of when the stock price could possibly go up or down and by how much range. Not only do we want to see if we can predict the stock price accurately, we also want to find an optimal method that utilizes the trade-off between the prediction accuracy and efficiency. Our idea of the project is to compare the different algorithms used for stock price forecasting to see which one gives the best balance between runtime and root mean square error.

Literature Survey:

The literature survey focused majorly on two parts, algorithm used and data applied. Singh's[1] article presents several algorithms doing forecasting on time series data, from naive approaches like averages and moving averages, to more advanced ones, like Auto Arima. While Selvamuthu's[2**]** paper introduces algorithms that are more likely to be used in stock price forecasting, including LSTM and Auto Arima, and Brownlee's[3] blog introduces Prophet, a open-source library used for time series data forecasting, and presents how to run the model on the time series data. Besides forecasting based on the time series data, which is the past performance of the company, **Roy's[4]** article also shows another possible prediction based on the performance of other companies using Regressor model, and how the combination of the combination cross-company prediction and past performance prediction could work.

Approaches:

Peiwen Tang & Minji Kim

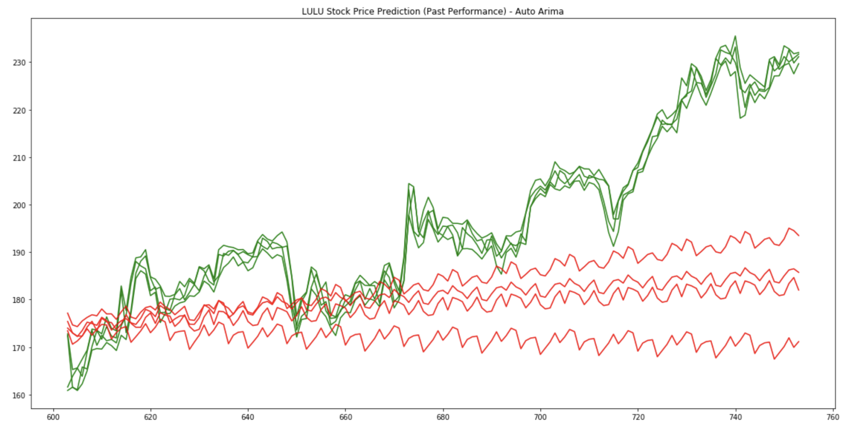
We started our project with data fetching. We picked Lululemon as our target company, which makes technical athletic clothes for workout. By using web scraping, we obtained the data from 01/01/2017-12/31/2018 as our training data and 01/01/2019-12/31/2019 as our testing data. We also picked nine other fitness companies (Nike, Peloton, Sprouts Family Market, United Natural Foods, Big Five Sporting Goods, Nautilus, Johnson Outdoors, Herbalife Nutrition Ltd, and Global X Health & Wellness Thematic ETF) to do cross company prediction.
Next is the selection of algorithms. As this project is for comparing different algorithms and observing which performs the best, choice of initial algorithms is important. For the prediction on the past performance of the company, from the research we have done, we first picked from LSTM using Keras, which is a familiar recurrent neural network model for us to implement. Then we chose Auto ARIMA and Prophet. As introduced in the article of Singh[1], Auto Arima stands for Auto-Regressive Integrated Moving Averages, and works on stationary data, and univariate series. It has three parameters: AR (autoregressive term), the past data, which is our training data, I (differencing term), and MA (moving average), number of past forecast errors. Prophet, introduced by Brownlee's[3] article, on the other hand, gives an additive time series forecasting model, and the key features are trends, seasonalities and holidays. Prophet also takes an univariate data set. These two methods can be implemented by importing libraries, and both of them are quite different from RNN, in terms of the model design, parameter turning, data intake (both are univariant, so both can only process one feature's prediction at a time), and output result, but are both suitable for time series prediction. Lastly, we chose to utilize Artificial Neural Network to do a more advanced prediction, which is to first take into the consideration of other companies' performance, then combine cross-company prediction results with the time series data prediction, to do a more comprehensive forecasting.

Experiment Design:

Our main comparison is based on the applications of the three models generally used for time series data to use the past performance of a company to predict the future stock price (Auto Arima, Prophet and LSTM). Specifically, the four attributes used are "High", "Low", "Open", and "Close", which refer to the maximum and minimum prices in a given time period, and the prices at which a stock began and ended trading in the same period. Other expected output would be the Root Mean Square Error and the graphs visualizing how the prediction deviates from the actual price. After that, we will use Regression model to perform prediction on the data of other companies stock prices, and combine the result data with the prediction results from the LSTM model, feed the data in the an Artificial Neural Network model, to generate a new prediction and see if it can improve the accuracy of prediction.

Evaluation:

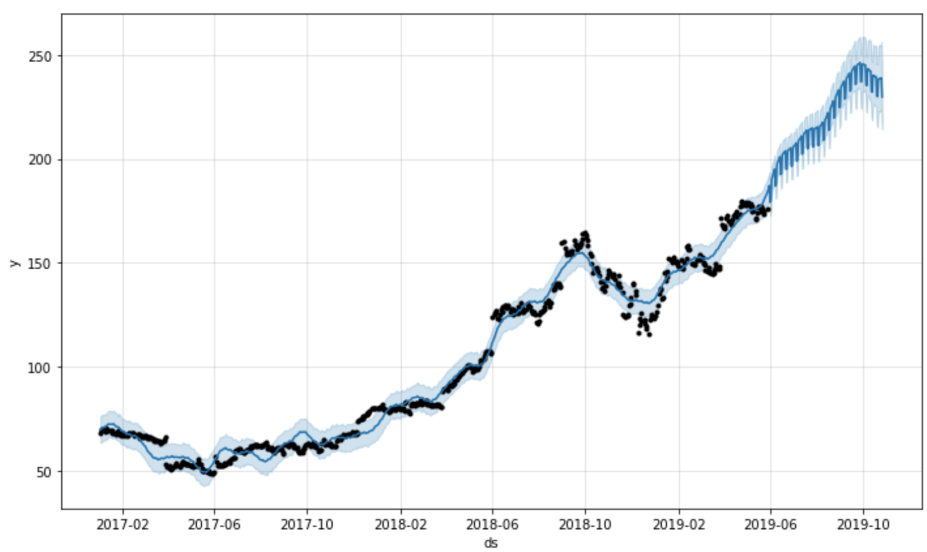
Evaluation 1 - Auto Arima:



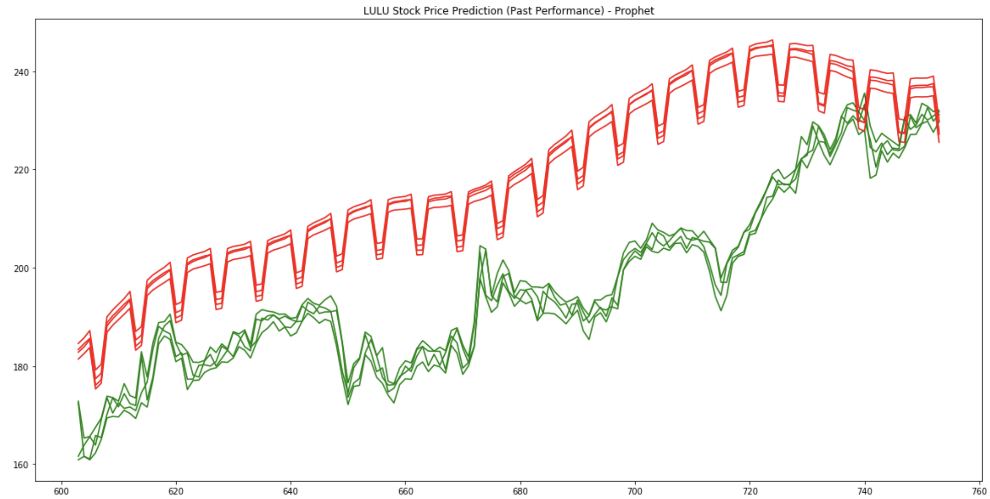
Graph 1: Test (green) vs Prediction (red) using Auto Arima

Graph 1 shows the prediction visualization when using Auto Arima, and the RMSE of the prediction is 24.045. The RMSE number itself is actually not bad as the actual data range approximately is around 170 to 220. However, when looking at Graph 1, we can see that it cannot forecast the trend very well, neither could it forecast the precision of the future stock price. We can only see an overall trend of the price going up, but no specific internal trend is learned. When looking at the individual line of the prediction, which stands for the maximum, minimum, open and close prices of the day, the prediction on the close price even has a negative trend: the forecasting prices are going down, but in reality, even with the fluctuation, all four prices eventually go up. Also, since Auto Arima works for univariate data series, we did the prediction one by one for each of the four features. Even with an appreciable RMSE, we don't think RMSE forecasted very well, neither in accuracy nor in precision.

<u>Evaluation 2 - Prophet:</u>



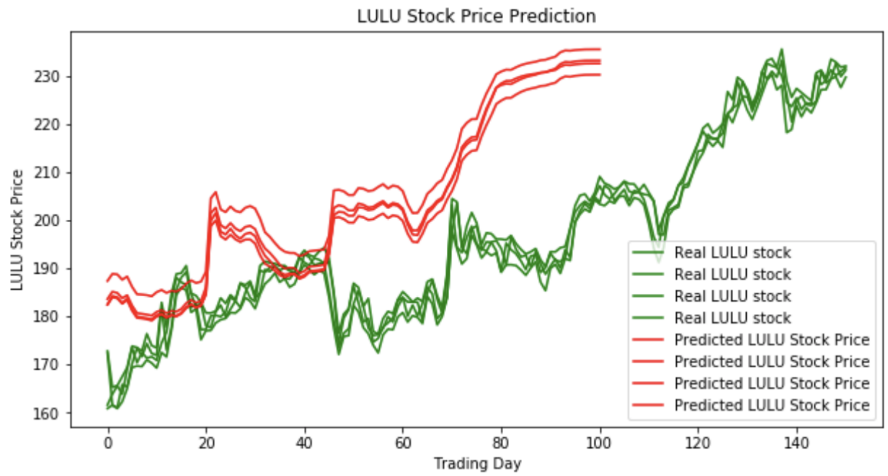
Graph 2: Prophet's graphing tool shows the combination of training data (black dots), prediction (dark blue line), and uncertainty interval (light blue shadow)



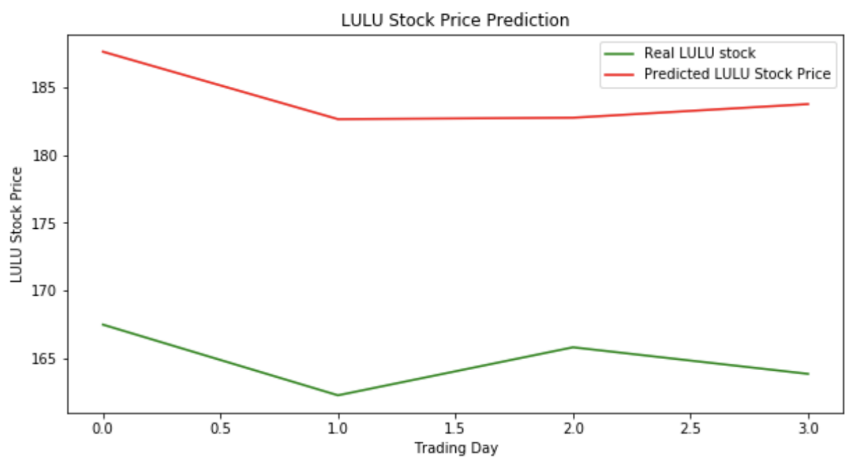
Graph 3: Test (green) vs Prediction (red) using Prophet

While using the prophet model, we can input no parameter, so it solely depends on the internal design of the model and the data set itself. Looking at the comparison of the prediction graph (Graph 3), we can see that overall, it achieves a better prediction compared to Auto Arima's forecasting, because it mimics the overall up and down trend of the stock price fairly well, not only the general trend of the stock price going up in during the time, but some kind of fluctuation during the period. However, from Graph 3, we can see a sequence of strange trapezoid shape trends. This is probably caused by the little dataset we feed in, which is a two-year dataset as training and one-year dataset as modeling, and no tuning on the parameters of the model. Overall, it gives a better trend prediction compared to Auto Arima. Prophet itself provides a graphing tool to visualize the train, test and predict data, which is shown in Graph 2. Prophet gives a possibility of giving a prediction result as a range instead of just a definite value, so if the goal of the prediction is not the value itself but seeing a trend or a range, Prophet could be a good choice. This result is generated by no parameter tuning, so if involving parameter factors (like seasonality and holiday), it could possibly give a better prediction. The RMSE is quite similar to the auto-arima prediction, which are all around 24. Prophet can give a good approximate prediction, but not as precise.

Evaluation 3 - LSTM:



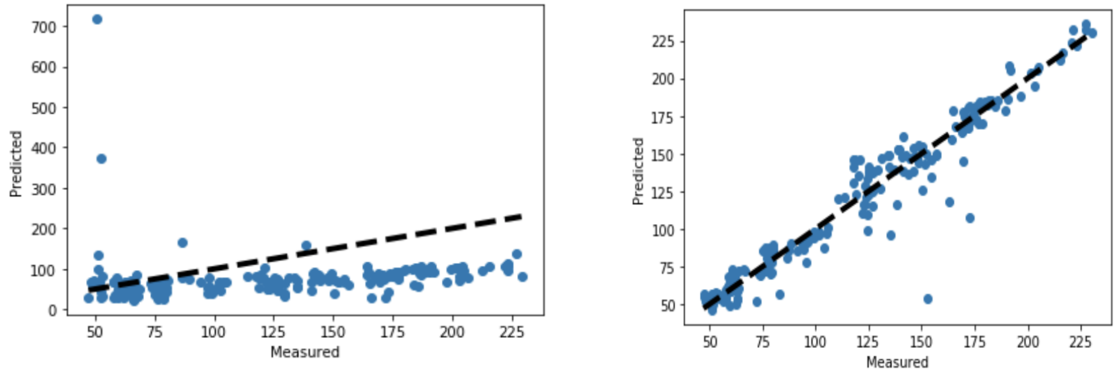
Graph 4: Test (green) vs Prediction (red) using LSTM



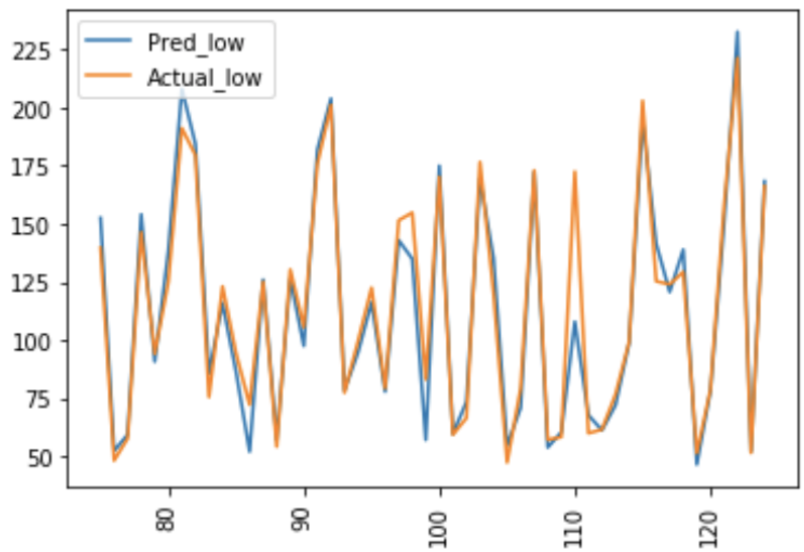
Graph 5: Test (green) vs Prediction (red) of close price using LSTM

From the prediction graph, we can see that it mimics the trend of the stock price pretty well. It shows precisely how the price could move up and down, and the overall up trend is also learned by the model. However, from the graph itself, we can see that the results actually deviate from the actual data for almost a fixed distance. Graph 6 gives a closer look at the prediction of the close price using LSTM. We can see that the precision is learned, but the prediction is always higher than the actual data for around 20. It shows that the Recurrent Neural Network performs pretty well on the precision of the data prediction compared to the other two, but not as good in terms of the accuracy. The RMSE for LSTM is 20.893. We can see an improvement from the number, but not as significant. Also, the runtime when using LSTM is significantly larger than using Auto Arima and Prophet. The trade-off between efficiency and accuracy may not be optimal for LSTM.

<u>Evaluation 4- Combined (ANN):</u>



Graph 6:(Low Predicition) Left: Regressor's Pred vs Accu; Right: Combined's Pred vs Accu



Graph 7: Prediction (blue) vs Test (orange) when using combined model

The combined model takes into account the prediction of the both a regressor model on the cross company prediction: predicting on the performance of other companies in the industry, and our previous prediction on the time series data of the company's past performance using LSTM. One of the prediction results is the Graph 6. Regressor's prediction is the one on the left, while the graph on the right is the combination of the regressor model and LSTM (the ANN model). Only using regressor model gives a good prediction, as we can see it does not deviate too much, but the combination gives a better result. Graph 7 gives a more detailed look of how the precision and accuracy are both achieved by using the combined model. The RMSE of the ANN is 9.227 for High price prediction and 7.276 for Low price prediction, which has a significant improvement compared to the time series data prediction. The run of the regressor model and the ANN model is neglectable, so the main time used on this prediction is on the LSTM prediction. We can see that the combined model achieves a better efficiency and accuracy compared to LSTM.

Conclusion:

Comparing the root mean squared errors of the three models on time series data forecasting, they all achieved similar results, Auto-arima and prophet is around 24 while LSTM is slightly better, achieving a RMSE of 20. However, each model has its own pros and cons. One disadvantage of auto-arima and prophet is that they are both univariant, so if we want to predict for multiple attributes, we need to run on each target separately. Both auto arima and prophet are easy to use and for a 2-year data set, their running time is almost neglectable, with very few parameters required for the user to tune. However, the three parameters of auto arima: p, the order of the auto-regressive (AR) model, d, the degree of differencing and q, the order of the moving average (MA) model still requires user testing. Prophet's parameters are more intuitive compared to auto arima and easy for users with no deep learning background to use, however, one of the important parameters of prophet is holiday. For the stock market, holidays have no trading action, so the parameter is not utilized in stock price forecasting. Also, from the diagram, we can see that the Prophet model tends to give a range rather than a specific number of predictions, so it's more vague compared to the other methods. LSTM, on the other hand, provides the best RMSE of the three, and from the result diagram, we can see that it's forecasted trend is much more accurate compared to the other two, but it requires a more complicated model design, plus a longer run time, but the improvement of the RMSE is not that significant. The combination of regressor and LSTM archives the best result during our experience experiment, which adds the market status data from the stock prices of other companies in the related field to the time series data to do the forecasting. The runtime of the method has no severe difference compared to solely using LSTM, but achieves a way better RMSE. The con of the model is that the design of the model would be more complicated compared to the pure time-series prediction method. Since our data picked do not cover periods like 2008 which has stock market crash and 2020 which has covid-19, which both have severe impact on the stock market, for the future work, we want to expand the data size to see how each model runs on a longer period and more complicated data set.

Reference:

[1] Aishwarya SinghAn. (2020, October 18). *Stock Price Prediction Using Machine Learning: Deep Learning*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/.

[2] Selvamuthu, D., Kumar, V. & Mishra, A. Indian stock market prediction using artificial neural networks on tick data. *Financ Innov* 5, 16 (2019). https://doi.org/10.1186/s40854-019-0131-7

[3] Brownlee, J. (2020, May 8). *Time Series Forecasting With Prophet in Python*. Machine Learning Mastery. https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/.

[4] Roy, A. (2021, January 9). *Stock Price Prediction Based on Deep Learning*. Medium. https://towardsdatascience.com/stock-price-prediction-based-on-deep-learning-3842ef697da0