Computer Engineering Department
CMPE-255 | Data Mining | Professor David C. Anastasiu

Final Project Report

Future Stock Prediction Using historical data

Team 7

Ajith Balaji Nagarajan(013727246)

Pooja Ramaswamy(013738868)

Divjot Singh Dhody(013854737)

Spring 2019

Table of Contents

Chapter 1. Introduction 3

Chapter 2. System Design & Implementation details 3
Algorithms, Technologies, and Tools 3
Moving Average 3
Prophet 4
LSTM 4
LSTM RNN 4
Bidirectional LSTM 4
Dilated CNN 5
Chapter 3. Experiments and Proof of Concept Evaluation 5
Model Flow Diagram 6
Datasets 6
Data Preprocessing 6
Train Valid Split 7
Results 7
Moving Average Graph 7
LSTM Graphs 8
Bidirectional LSTM Graphs 8
LSTM RNN Graphs 9
Prophet Graphs 9
Dilated CNN 9
Chapter 4. Discussion and Conclusions 10
Decisions, Difficulties, and Discussion 10
Conclusion and Future Work 10
Chapter 5. Project Plan / Task Distribution 10
References 10

Chapter 1. Introduction

Stock market is a place where shares or stocks of a firm are traded. It can be split into two components: primary market and secondary market. Primary market is where new issues are introduced to the market through Initial Public Offerings. Secondary market is where investors trade securities that they already own. Stock market is having a highly fluctuating and non-linear time series data. Due to the equivocal and unforeseeable nature of stock market, stock market forecasting takes higher risk compared to other sectors. It is one of the most important reason for the difficulty in stock market prediction [3]. Here are where various machine learning models such as Linear Regression and Nearest neighbor regression and time series models such as Moving average, ARIMA, Prophet and some deep learning techniques involving TensorFlow and LSTM are widely used to solve this problem. In this study, we apply several machine learning and deep learning algorithms to forecast future stock closing values and they were being evaluated based on Root Mean-Squared Error (RMSE) and average loss depending upon the model as the performance metric. Here individual stock prices were derived directly from yahoo_stocks(), yahoo client API that has real time stock values of over 60 different companies with stock values ranging for more than 10 years. From the various features present in the data such as High value, Low value, Close value, Adj. close value, Date etc., Date and normalized version of close values were found to give better results for most of the models and hence were chosen for the final analysis. A web application is also built, that could forecast stock values, apart from suggesting the user about buying and selling.

Chapter 2. System Design & Implementation details

Algorithms, Technologies, and Tools

Six different models were implemented and their performances were evaluated using Root Mean-Squared Error (RMSE) and average loss such as Moving Average, time series model Prophet, Deep Learning models such as library version of Long Short Term Memory (LSTM) and modified versions of LSTM such as LSTM RNN, Bi-directional LSTM and Dilated CNN and LSTM RNN. The Moving average being a very basic model performed extremely poor compared to all other models mentioned above. Since five other high performing models were selected, techniques such as Linear Regression, Nearest Neighbor Regression and Support vector regression weren't performed. The deep learning models gave much better results of all, but they were computationally in efficient. Prophet being a time series model gave good results, although not as high as the deep learning models, was a good compromise computationally. All the models were implemented, trained, visualized and evaluated in Jupyter Notebook iPython environment. A brief description about the pros and cons and the metric used for evaluating each models will be explained below.

1. **Moving Average:**

   Average' is easily one of the most common things we use in our day-to-day lives. For instance, calculating the average marks to determine overall performance, or finding the average temperature of the past few days to get an idea about today's temperature – these all are routine tasks we do on a regular basis. Hence, this is one of the very basic model to start our evaluation. Also, this algorithm tells us about the nature of the dataset being used. It simply uses the average value of a set of stocks to predict the future value. Moving average uses the average of the latest set of values for predicting the future values. Although this model is simple to implement, it is not really useful for predicting future values.

2. **Prophet:**

   Prophet is an open source software released by Facebook's core data science team. Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well. Prophet is used for producing accurate, reliable forecast for planning and goal setting at a faster rate. It is fully automatic and requires no special preprocessing steps for analyzing. It can handle large datasets more efficiently.

3. **Long Short-Term Memory (LSTM):**

   This is an improvised version of Recurrent Neural Network (RNN). LSTM has feedback network that makes it 'general purpose computer'. It is well suited for classification, processing and making prediction on time series data. The results of LSTM are far better compared to all other models, as it can efficiently handle lags of unknown duration associated with Time series data. However, it slows down drastically when the dataset is large, even though it gives a good result.

4. **Long Short-Term Memory Recurrent Neural Network (LSTM RNN):**

   A slightly modified manual version of the LSTM library was used for the analysis. Here the hyperparameters were tuned manually in order to improve the speed of performance of LSTM for large datasets. It was succeeded to give better performance on 500 epochs at a rate slightly faster than the library version of LSTM performed above.

5. **Bi-directional Long Short-Term Memory (Bi-directional LSTM):**

   This is an improvised version of LSTM RNN from the previous step which apart from back propagation also includes forward propagation. It has lots of advantages compared to the unidirectional LSTM (LSTM RNN), with the major point being the ability to store both past and the future values. The makes the model give more accurate and close

prediction of future stock values compared to normal LSTM. This is much faster compared to unidirectional LSTM.

6. Dilated CNN:

Dilated convolutional neural networks (dilated CNN) have recently enjoyed a great success in image segmentation and dense prediction (semantic segmentation), text-to-speech, and text classification. Dilated CNN and convolutional neural networks max-pooling (CNN Max Pooling) pairs to predict the future prediction of stock values. Max pooling CNN can capture high level features automatically. The dilated CNN can be used for capturing temporal dependencies. Dilated CNN supports exponential expansion of the receptive field without loss of resolution or coverage and suffer less from vanishing gradient problem observed in back-propagation through time approaches and they are also easily parallelizable. The give the best performance of all the models used for evaluation, with the predicted plot very closely following the actual plot.

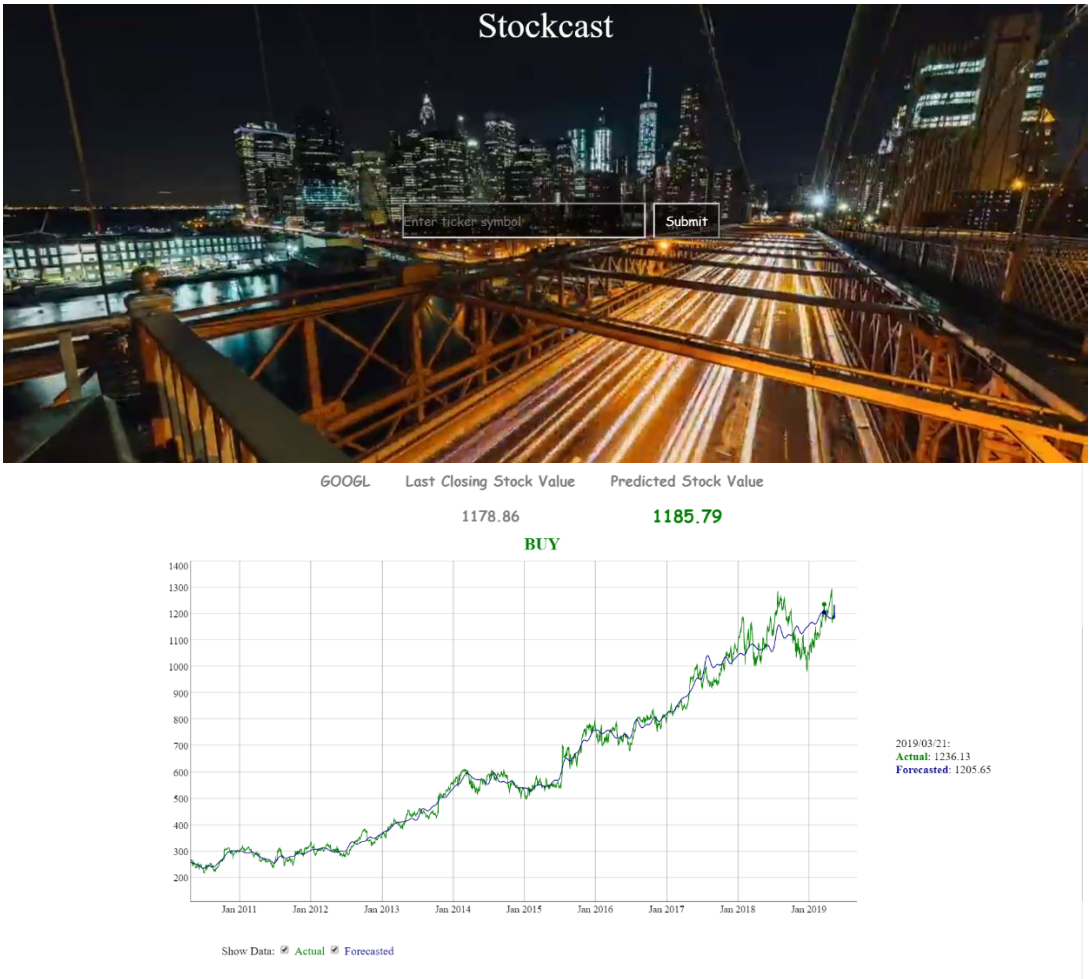


Fig 1. Snapshot of the Application developed using prophet algorithm in backend and HTML&CSS front end.

Chapter 3. Experiments and Proof of Concept Evaluation

Model Flow Diagram

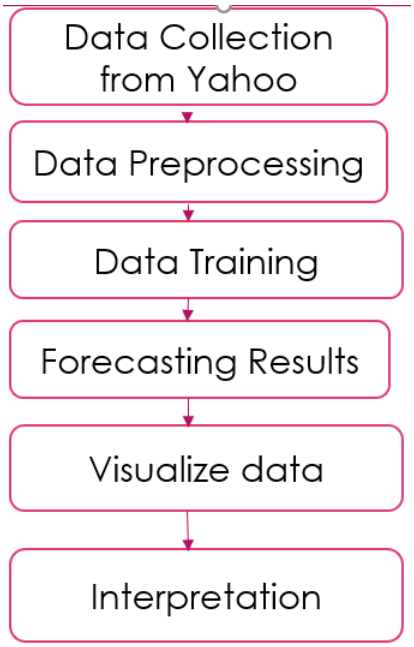


Fig 2. Project's model flow diagram

Datasets

Almost 15 years of daily stock price values were fetched from yahoo finance API for evaluation of the model. The evaluation for prophet, Dilated LSTM was performed using the stock value with the name FORD. The evaluation of Moving average and library version of LSTM was performed using the stock value with the name IBM and the evaluation of manual version of LSTM RNN and Bidirectional LSTM was performed using the stock value with the name PEP.

Preprocessing

After loading the datasets, of the features that were present high, low, last, close, date, Adj. close, features such as date and close values were selected as it results in more accurate future prediction of stock values than the other factors. The close values of stock data were standardized using min max scaler, before being fed to classifier for training and prediction.

Train-Validation Split

For the purpose of evaluation, 10 years of stock data were used for the purpose of training and the remaining 5 years of data were used for validation. This was used as a standard for evaluating all the models.

Results

The table below shows the RMSE evaluation of Moving average, prophet and LSTM:

Model	**RMSE**
Moving Average	32.9939
Prophet	12.2484
LSTM	5.7396

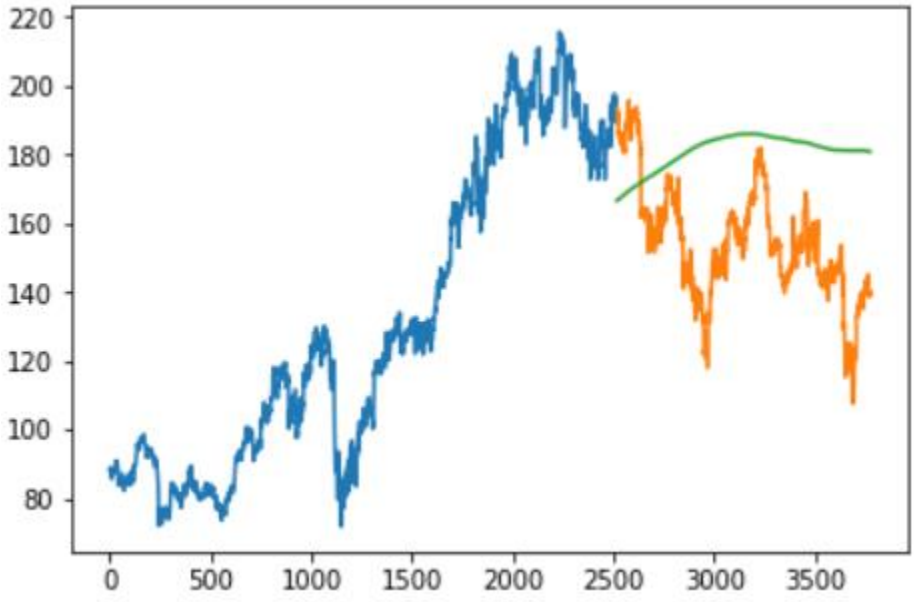
Table 1. RMSE evaluation

The table below shows the average loss over 500 epochs for the manual versions of LSTM:

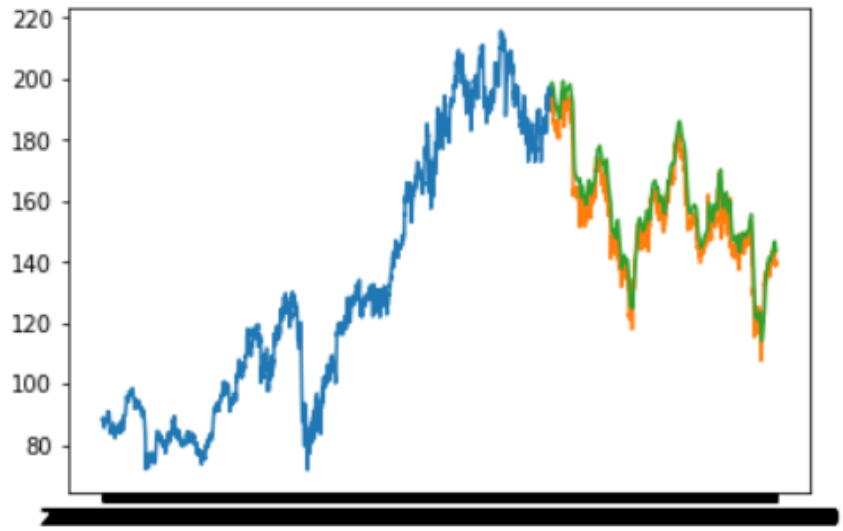
Model	**Average Loss over 500 epochs**
LSTM RNN	0.0027
Bi-directional LSTM	0.0026
Dilated CNN	0.0005

Table 2. Average loss evaluation

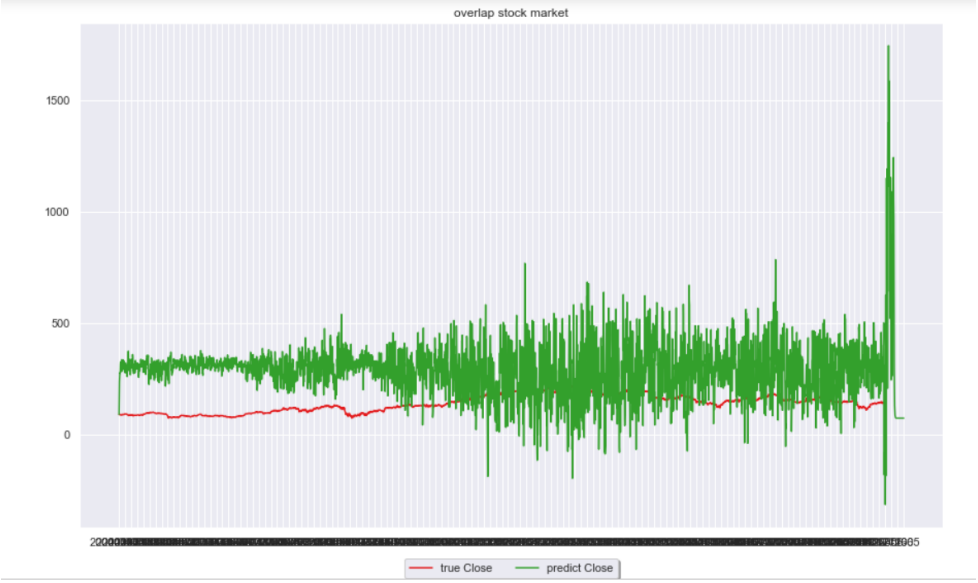
Moving Average Graph



LSTM

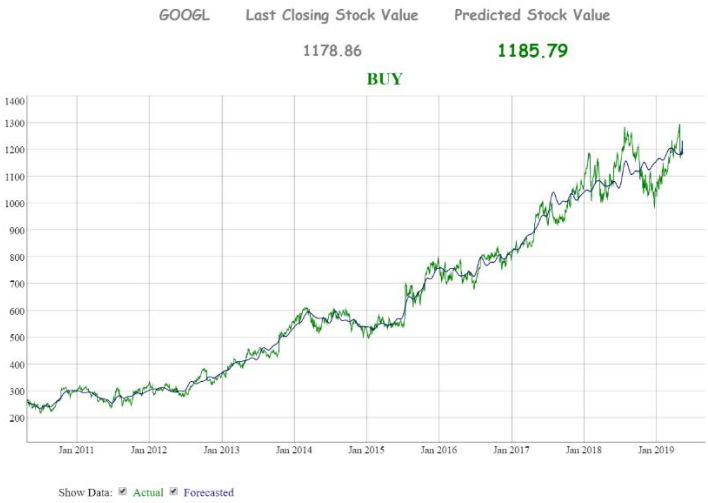


Bi-directional LSTM

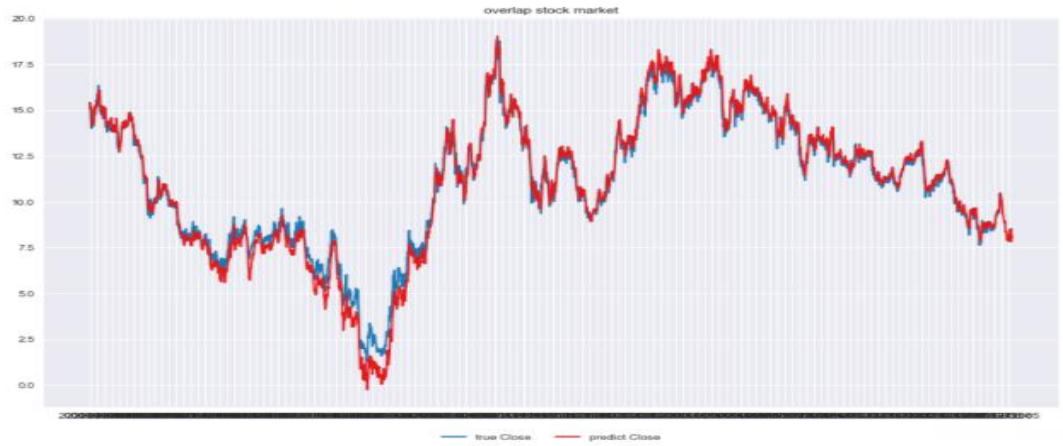


LSTM RNN

Prophet



Dilated CNN



Chapter 4. Discussion and Conclusions

Decisions, Difficulties, and Discussion

Time-series data analysis, especially stock market, is a highly sought after and hot area. Considering the inherent complexity of such markets, and large number of factors that can affect its behavior, it is not an obvious or a routine data mining task to come up with a model that performs consistently well.

For the above-mentioned reason, we decided to build on an application that could prove as useful tool for people who wants to invest in stock market. Although all of the deep learning models gave us much better results, they were computationally very costly approaches. Hence we decided to use Prophet model, which had the next lowest RMSE score to build an application. Also the fact that prophet works really well and at a faster rate for high dimensional dataset helped us to decide on that. The application built by us can forecast future stock values using real time stock values of close to 60 different companies, apart from giving suggestions on when to buy or sell a particular stock value.

Conclusion and Future Work

We are planning on improvising our application by replacing yahoo finance API with APIs that have much broader stock database such as quandl. Also, we are planning on optimizing LSTM in such a way that it can predict the results at a faster rate apart from giving more accurate results.

Chapter 5. Project Plan / Task Distribution

Initially we decided to add another factor by performing sentiment analysis on stock news data on the prediction of stock values. However due to time constraints, we concentrated more on real time stock prediction. Pooja worked on Prophet and Dilated CNN, Ajith worked on manual version of LSTM RNN and Bidirectional LSTM and Divjot worked on moving average and LSTM models. We evaluated the results and decided to use prophet for our application for the above mentioned reasons. The frontend HTML was developed by Ajith, designed with CSS by Pooja and integrated with python in the backend with flask by Divjot.

Project Github repository: https://github.com/PoojaR24/StockCast
The Datasets: https://finance.yahoo.com/

References

1. https://skymind.ai/wiki/lstm
2. https://facebook.github.io/prophet/
3. https://web.ics.purdue.edu/~cmousas/papers/conf18-IEEE-ITSC.pdf
4.https://reader.elsevier.com/reader/sd/pii/S1877050918307828?token=37D0B9955CC55A68CF024E51767813AF874A09D2817F70D16DCDBA915D2C340461DFB2CCFBC46D37505F9C00964B9F45