Predicting Stock Prices Using FbProphet

Dinesh Deivamani, *Northeastern University, deivamani.d@husky.neu.edu*

Kanika Nama, *Northeastern University, nama.k@husky.neu.edu*

Abstract: This paper proposes the use of an interesting prediction technique named FBProphet for predicting and forecasting Stock Price data. After initial pre-processing and training, we are predicting the stock price by analyzing historical data as accurately as possible. So, we are developing a FbProphet model using Spark and Python. Furthermore, we extended our approach to spark clusters on google cloud data proc to improve our model's performance. We perform visual hyperparameter tuning on our model to improve the accuracy. Later we performed cross validation on our model and calculated the accuracy, which resulted in 96.5% for the set of parameters we used.

1.Introduction Time series analysis is an approach to analyze time series data to extract meaningful characteristics of data and generate other useful insights applied in business situation. Time series analysis helps understand time-based patterns of a set of metric data points that are critical for any business. Time series forecasting techniques can answer business questions. The Important objective of time series analysis is to establish a model that describes the time series pattern and could be used for forecasting FbProphet helps us to overcome difficulties caused by different seasonalities like weekly, monthly, yearly, holidays, changes in stock trend due to several events and launch of new products. Also, Prophet can be customized easily that allows us to improve the quality of our model. The most important thing is that, the parameters are quite understandable even for non-experts in time series analysis, which usually requires good amount of skill and experience.

2.Methodology

2.1 Block diagram

Figure 1.1 represents the block diagram of our proposed system. The figure explains the process cycle of our FbProphet model. We have currently less number of tools for designing the forecasting process.

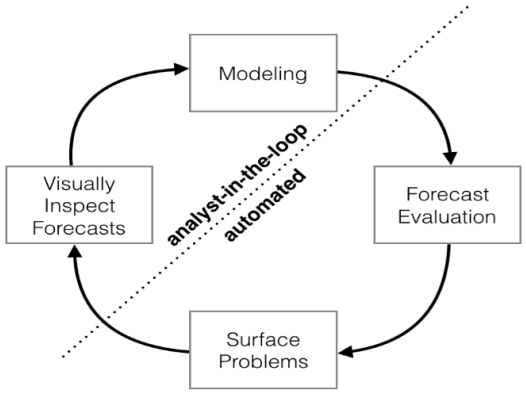


Fig. 1.1 Block diagram of Fbprophet

2.2 Data acquisition:

In this project, the Stock price dataset is used. The dataset consists of close prices for stocks of Amazon from 2016/01/01 to 2019/04/26 as an example to have better understanding about what we are doing. With "**fix_yahoo_finance"**, we can access stock price easily. The documentation we can see here is a strong trend of growing price from 2016. However, there are still a lot of up and down or cycles during these years.

2.3The Prophet Forecasting Model- Our time series model consists of three main components: trend, seasonality, and holidays. All three components are combined in the following equation:

$$y(t)= g(t)+s(t)+h(t)+\varepsilon t$$

g(t): piecewise linear growth curve

s(t): periodic changes

h(t): effects of holidays

εt: error term accounts for noise

2.4 Model Architecture and workflow:

At its core, the Prophet procedure is an additive regression model with four main components:

- Prophet automatically detects changes in trends by selecting changepoints from the data.
- A yearly seasonal component modeled using Fourier series.
- A weekly seasonal component using dummy variables.
- A user-provided list of important holidays.

2.5 Important Hyperparameters:

1) ***Seasonality***- The seasonal element increases the flexiblility of the model with intervallic changes because of weekly and yearly seasonality. Weekly seasonalities are Monday, Tuesday, Wednesday, Thursday, Friday. Saturday and Sunday is not added because the stock market is closed. And the Yearly seasonality consists of all the months in a year.

2) ***Holidays and Events***- Holidays and events are special days which can affect the sales of the particular business. e.g., Black Fridays.
To use this, we need to declare a custom list of important events.
Error- The error term signifies information that was not replicated in the model. Typically it is demonstrated as usually dispersed noise.

3) ***Trend*** – Trend shows the growth of historical data over the years:

$$g(t) = \frac{C}{1 + e^{-k(t-m)}},$$

where:

- *C* is the carrying capacity.
- *k* is the growth rate.
- *m* is an counterpoise parameter.

2.6 Model Fitting & Forecasting:

We are creating our own model with necessary seasonalities, holidays and tuning required parameters. We can also define the number of days we need to predict.

The forecast contains several columns of dataframe which includes ds which are the dates, y is the actual value, yhat is the predicted value. It also has y_upper and y_lower which gives us a range of maximum and minimum values.

2.7 LSTM Model:

To implement a deep learning technique in TensorFlow, we are implementing a simple LSTM model.

2.8 Databricks Platform:

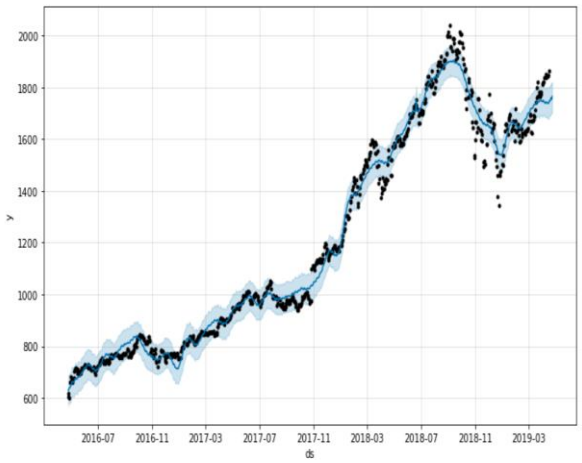
Databricks is a convenient platform to run Apache Spark. The code is deployed in Databricks Spark Cluster which has advantages of computation efficiency and handling bigdata.

2.9 GCP Dataprocs:

We set up a storage bucket in GCP dataprocs in which our code was imported to. A cluster was created to run the code from the storage bucket. However, we were not successful in completion of GCP Dataprocs deployment.

2.7 Plotting the Forecast

Here we plot the obtained forecast to get predictions.



Cross Validation:

In this part we are evaluating our model by comparing actual and the predicted values using below parameters:

- Horizon
- Initial the size of the initial training period
- Period the spacing between cutoff dates

.

	horizon	mse	rmse	mae	mape	coverage
128	7 days	5202.346849	72.127296	47.533108	0.035233	0.342342
45	7 days	5226.322570	72.293309	47.990197	0.035789	0.333333
420	7 days	5251.406851	72.466591	48.378168	0.036165	0.324324
500	7 days	5245.612608	72.426602	48.304951	0.036188	0.315315
707	7 days	5250.609425	72.461089	48.411493	0.036178	0.324324

Results-

Finally we are comparing the actual and predicted values of our model with the LSTM model and Arima model.

Date	FbProphet_Predicted	FbProphet_Actual
2019-04-22	1893.47	1887.31
2019-04-23	1898.39	1923.77
2019-04-24	1895.39	1901.75
2019-04-25	1889.51	1902.25
2019-04-26	1917.81	1946.19

Date	LSTM_Predicted	LSTM_Actual
2019-04-22	1865.43	1887.31
2019-04-23	1870.62	1923.77
2019-04-24	1877.36	1901.75
2019-04-25	1890.42	1902.25
2019-04-26	1904.37	1946.19

Date	ARIMA_Predicted	ARIMA_Actual
2019-04-22	1861.05	1887.31
2019-04-23	1886.67	1923.77
2019-04-24	1923.12	1901.75
2019-04-25	1901.11	1902.25
2019-04-26	1901.61	1946.19

LSTM_Accuracy	FbProphet_Accuracy	Arima_Accuracy
78.29	96.5	91.8

Conclusion:

1.We created the FB Prophet model including the required seasonality's for our timeseries data.

2.Holiday parameters are included which contains long weekend holidays and Prime days for our Amazon stocks as they can have an impact on the company's sales.

3.We have tuned different hyperparameters of different components and chose the set of parameters that gave better results for every component. Later we exported the predicted vs actual data in a csv file for comparison.

4.Also, we performed cross validation using performance metrics which gave MSE, RMSE, MAE, MAPE values.

5.We used MAPE value to check the performance of our model and we got approximately 0.08% error for a horizon of 30 days.

6.We have achieved an approximate accuracy of 96.5% for the dataset on our created model.

7. The comparative results of both ARIMA and LSTM models are tabulated as in Figure 3.1. When deployed on Databricks, the computation time reduces because of the Spark clustering compared to usual Jupyter Notebook.

8.The advantage of using Databricks lets you have a Spark environment already loaded as setting up of Spark environment on local machine is time consuming and complicated.

References:

1.https://www.analyticsvidhya.com/blog/2018/05/generate-accurate-forecasts-facebook-prophet-python-r/

2. https://blog.exploratory.io/is-prophet-better-than-arima-for-forecasting-time-series-fa9ae08a5851?gi=91057e2befad

3. https://heartbeat.fritz.ai/sales-forecasting-using-facebooks-prophet-f9ae0214f196?gi=3256df5bb1a4

4. https://blog.exploratory.io/an-introduction-to-time-series-forecasting-with-prophet-package-in-exploratory-129ed0c12112?gi=9093ce75d6b1

5. https://www.datascience.com/blog/introduction-to-forecasting-with-arima-in-r-learn-data-science-tutorials

6. https://www.r-bloggers.com/forecasting-pm2-5-with-forecast-and-prophet/

7. https://pythondata.com/forecasting-time-series-data-with-prophet-part-1/

8. https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-9-part-3-predicting-the-future-with-facebook-prophet-3f3af145cdc

9. https://towardsdatascience.com/a-quick-start-of-time-series-forecasting-with-a-practical-example-using-fb-prophet-31c4447a2274

10. https://www.kdnuggets.com/2018/11/sales-forecasting-using-prophet.html

11. https://research.fb.com/prophet-forecasting-at-scale/