Study on the prediction of stock price based on the associated network model of LSTM

(PDF) Study on the prediction of stock price based on the associated network model of LSTM

Mentor- Aditya Pramod Nahata

2017A8PS0399P, Shantanu Nigam
2017A3PS0309P, Nirav Bhandari
2017A8PS0456P, Sarthak Mahapatra

Paper Summary

- The main purpose of this paper is to design a deep network model to predict simultaneously the **opening price**, the **lowest price** and the **highest price** of a stock on the next day according to the historical price of the stock and other technical parameter data.
- An LSTM based deep recurrent neural network model is proposed to predict the three associated values (so it is called the associated neural network model, and abbreviated as associated net model).
- The associated net model is compared with LSTM and LSTM-based deep recurrent neural network, and the feasibility of the model is verified by comparing the accuracy of the three models.
- The paper claims that this method has higher accuracy in predicting daily stock price than the technical analysis method.

Agenda

01 Problem Statement

02 Data

03 Model

04 Results

1. Problem Statement

The Stock Market

Main methods for predicting Stock Market

The Stock Market has always been a hot spot for investors and investment companies. The many methods for prediction are broadly divided in 2 classes:

Statistical

This is based on mathematical and statistical analysis of data. For eg-

1. Logistic Regression Model
2. ARCH Models

Artificial Intelligence

This includes using multi layer perceptron, LSTM, RNN, CNN and other models for prediction.

How's this method different from existing methods

Most of the existing Stock Market Predictors use LSTM layers to predict the Open, Low and High price of the stock individually. However, these values are deeply related to each other. The paper has presented the idea of a model which predicts all these values in an associated manner and has claimed better results.

2. Data

Collection and Processing

Data Collection

- In most Machine Learning/ Deep Learning applications, data collection is the major task. However, there was no problem in collection of data as stock prices are freely available and well documented by the National Stock Exchange (NSE) of India.
- The stock price of Tata Motors (ISIN: INE155A01022) was taken for the past 15 years for training of the model.
- The data had values of open price, low price, high price, total quantity traded, last price, close price, average price and turnover.
- Once trained, given the trends of past "N" days, the model would predict the Open, Low and High Price for the next day.

Data Pre-Processing

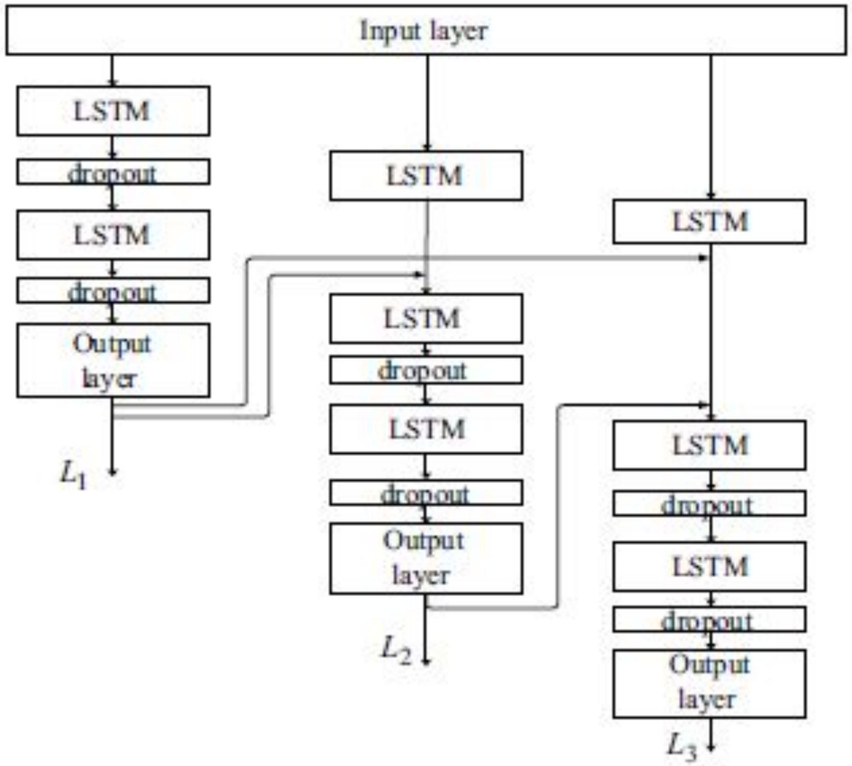
- The data collected was already clean. The inputs and outputs were scaled between 0 and 1 for faster training and convergence of the model. Scaling used was min_max scaling.

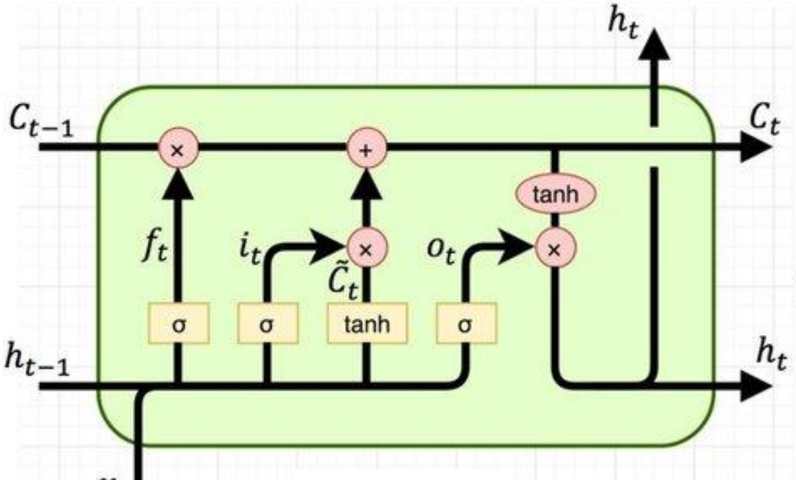
$$X' = \frac{X - min}{max - min}$$

- The input and output data was then created for the model to train. It is all done under the function preprocess_df( ) . A sequence of 20 time frames having 7 columns of information of stock data from stock prices was created, and the output was a 1x3 array having OP, LP, HP in this order.

3. Models

Design and Working





LSTM Cell

$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right)$$
$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right)$$
$$o_t = \sigma\left(x_t U^o + h_{t-1} W^o\right)$$
$$\tilde{C}_t = \tanh\left(x_t U^g + h_{t-1} W^g\right)$$
$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right)$$
$$h_t = \tanh(C_t) * o_t$$

- LSTM stands for long short term memory. This is a type of RNN block, used for series/sequence type data prediction.
- The memory blocks in LSTM helps in solving the problem of information loss in RNN blocks with time.
- When the series is long, it becomes difficult to relate information between starting and ending part of the series while making a prediction. LSTM memory blocks store this vital information and deletes them using forget gates when the information is not required.

The Algorithm

STEP 1

During the training phase, the Adam optimization algorithm is used in the model, and L_{total} is used as the evaluation function

STEP 2

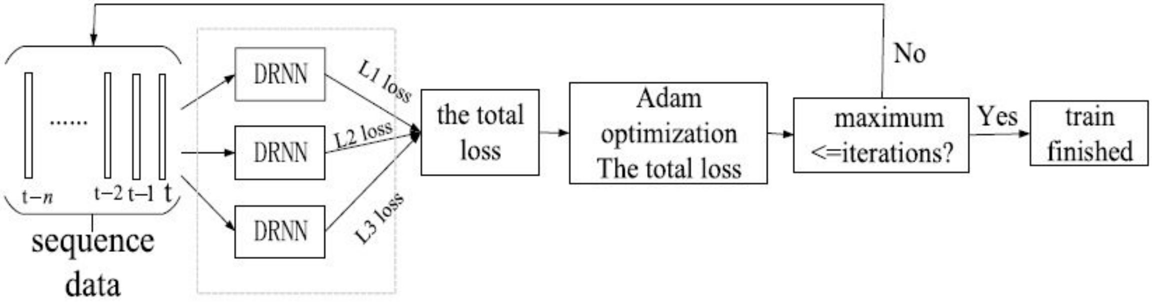
The first input to the Associated Net model (which contains three DRNN networks) is the sequence data

STEP 3

Each DRNN network produces a loss, and the losses sum of these three DRNN networks is the total loss.

STEP 4

Total number of epochs are checked, and if they have been completed the training stops, else the algorithm runs again.



Hyperparameter Optimization

The hyperparameters at our jurisdiction were the widths of the LSTM layers and dropout probability -

1) LSTM width - This was solved by creating small models (predicting only open price) with different widths (of power of 2) and considered the best result in 40 epoch.
2) Dropout - A standard 0.2 dropout probability was taken.

Width vs Loss at 40 epochs

Sr. NO	First LSTM Layer	Second LSTM Layer	Train Loss	Validation Loss
1	32	32	9.459e-4	2.427e-3
2	64	32	1.876e-2	3.164e-2
3	64	64	6.0734e-4	1.1246e-3
4	128	64	5.2012e-4	9.9201e-4
5	128	128	4.927e-4	1.1143e-3

Data starts to overfit here (Train loss has reduced but Validation loss increased). So Model-4 was chosen.

Problem faced

The major problem that was encountered during the training phase was that the model wasn't treating the three LSTM layers differently, and considering it as one connected layer. Due to this our weights were not trained effectively which in turn gave a very high loss (~ 30%).

Therefore the three different layers were trained separately, and this approach gave the expected results and the loss was drastically improved.

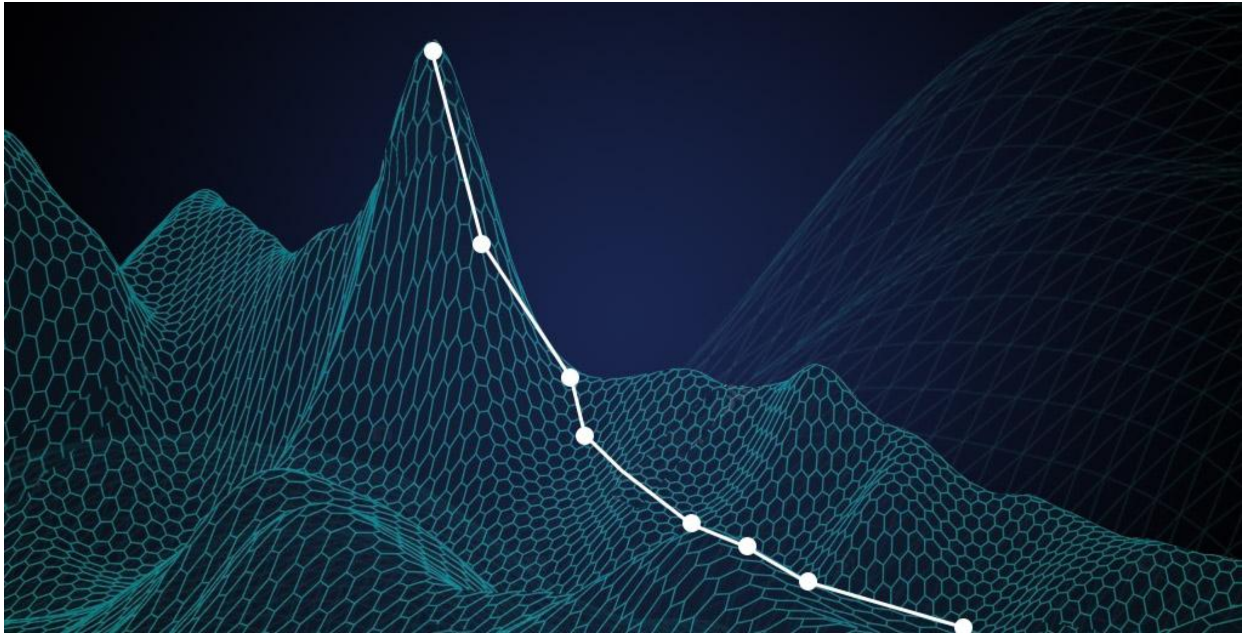
Callbacks

The callbacks used were -

1) Tensorboard - This was used in order to analyse the the loss as well as the predicted values visually.
2) Checkpoint - This callback gave an idea about the decrement in loss with each epoch, so that more epochs than necessary were not used and the best model was obtained.

4. Results

Quantitative and Qualitative



Quantitative Results

Measuring Performance

What was felt was that in Stock Market Prediction models, performance measurement based on **Loss** is much more better than **Accuracy**, because for high accuracy, the predicted values need to be exactly equal to the test values. But this is extremely difficult in case of stock market prediction so only reduction in the loss of the predicted values can be done.

And so the major performance measurement criteria of our model was **chosen as Loss**, which has been calculated using **MSE**.

Validation Loss vs Epochs for main model

Epochs	Open Price	Low Price	High Price	Average Loss
20	6e-5	5e-5	4e-5	5e-5
40	3e-5	4.75e-5	3e-5	3.58e-5
60	2e-5	4e-5	2.83e-5	2.94e-5

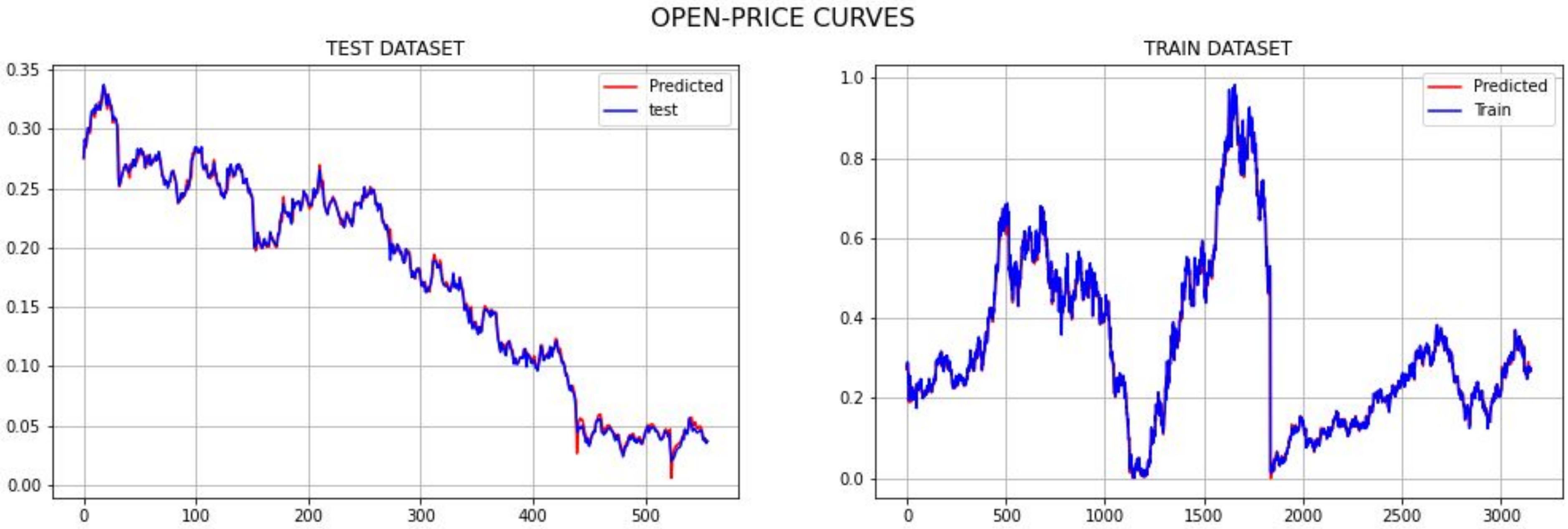
Output Parameters

The model predicts a total of 3 parameters -

1) Open Price
2) Low Price
3) High Price

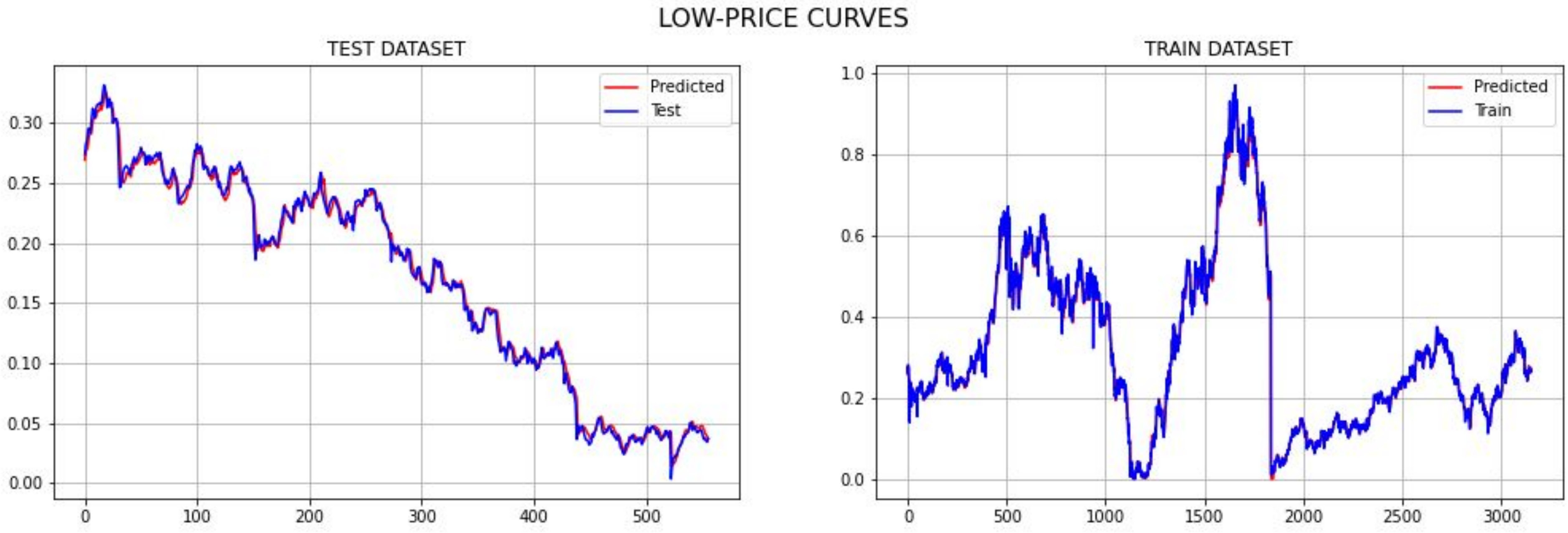
The following 3 slides give an idea about the losses in these parameters that the model predicted.

Open-Price Plots



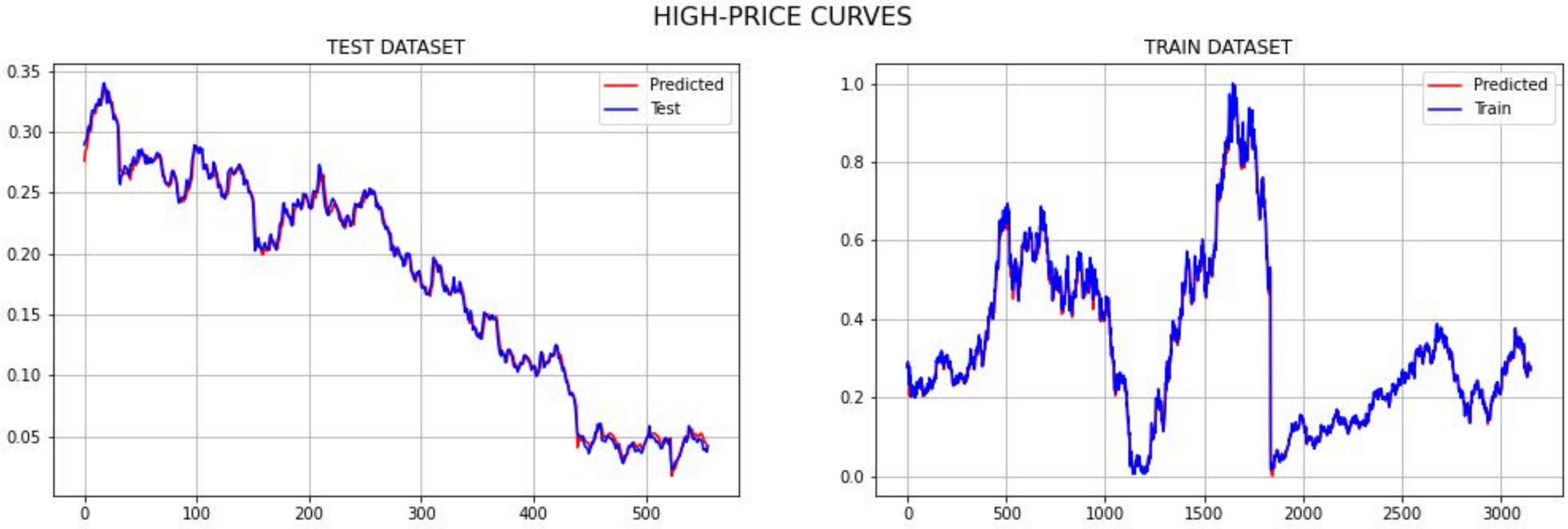
Observation - The predicted curves very well map the Train as well as the Test set.

Low-Price Plots



Observation - The predicted curves very well map the Train as well as the Test set.

High-Price Plots

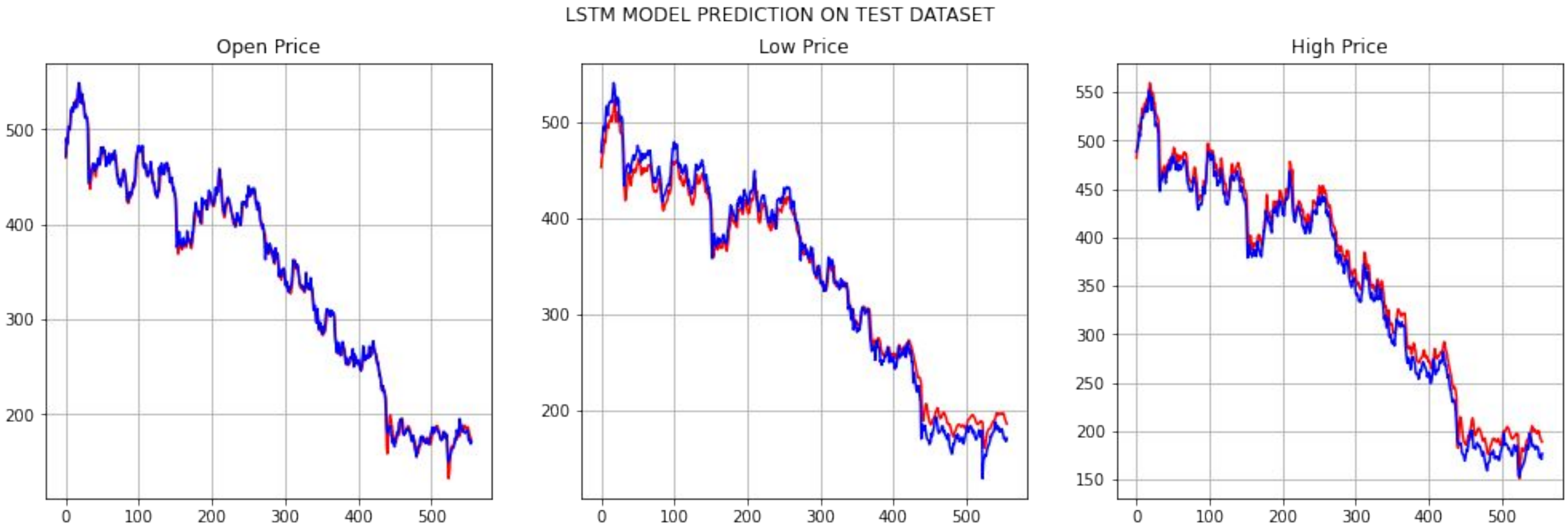


Observation - The predicted curves very well map the Train as well as the test set.

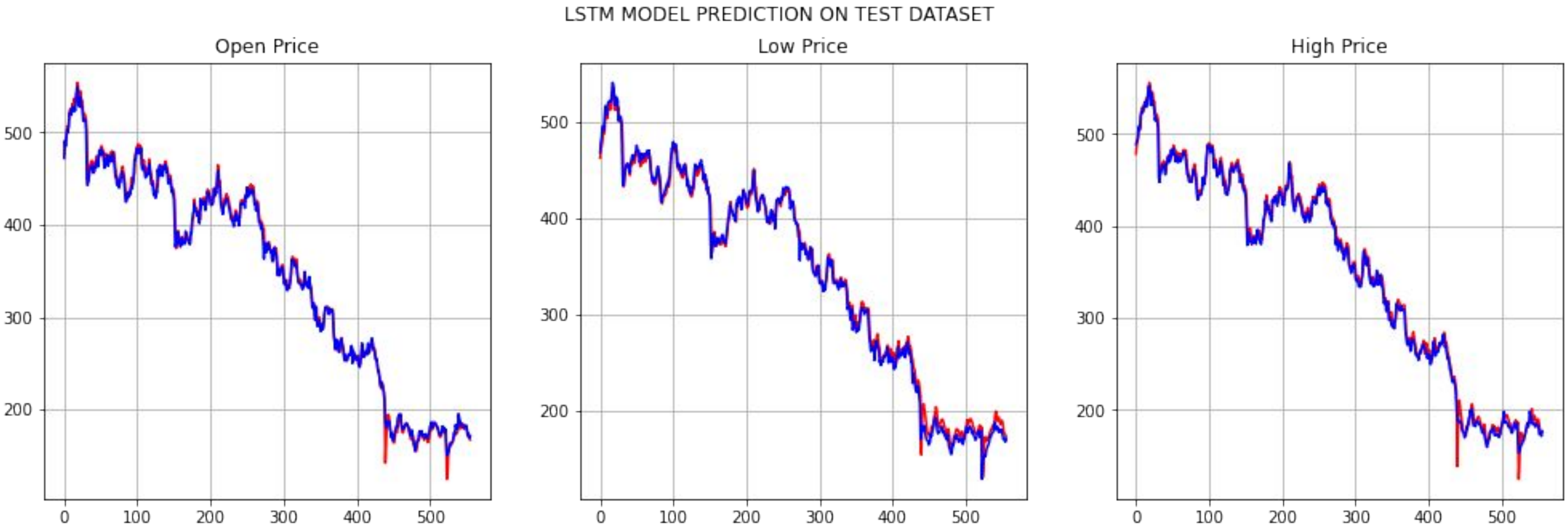
This model was compared with DRNN and LSTM


Comparing Different Models
1.60E-04
1.40E-04
1.20E-04
1.00E-04
8.00E-05
6.00E-05
4.00E-05
2.00E-05
0.00E+00
LOSS
OP
LP
HP
OP
LP
HP
OP
LP
HP
ADRNN
DRNN
LSTM
EPOCHS
20
40
60

Prediction by LSTM Model



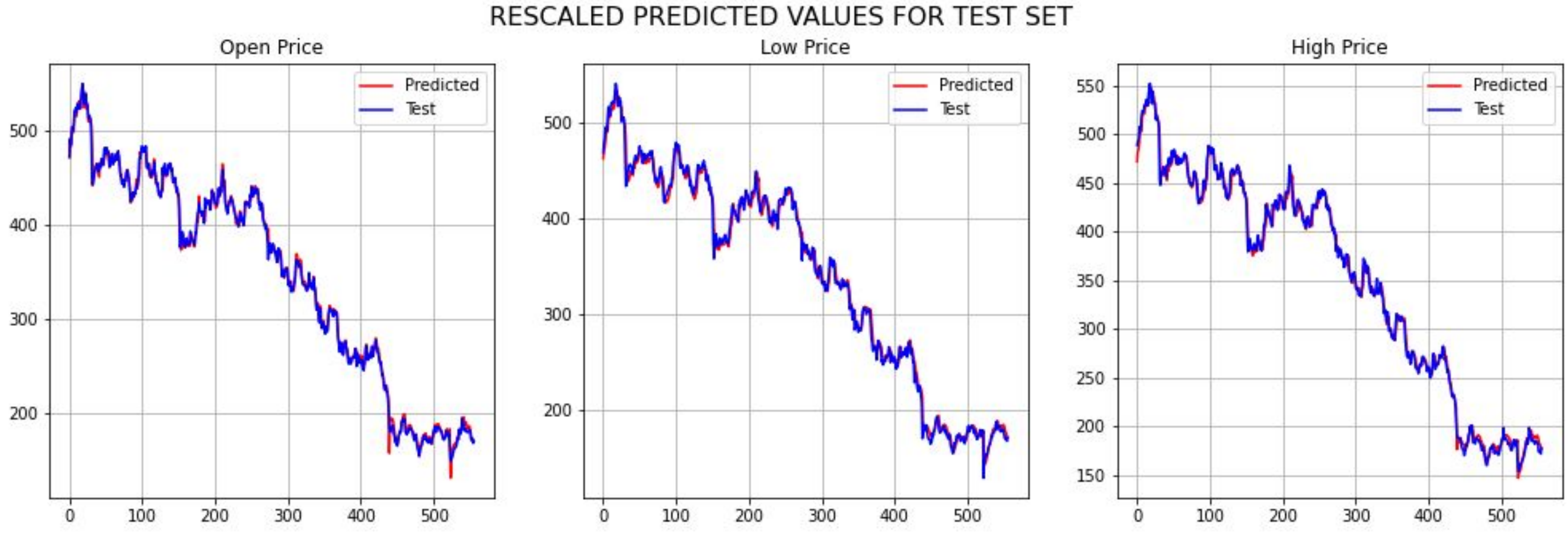
Prediction by DRNN Model



Qualitative Analysis

Predicted Values for each Output

The following figure shows the values predicted by our model. They show how perfectly well the model performs on the Test dataset.



Validation Loss vs Epochs for Associated Net

	Open Price	Low Price	High Price
20	6e-5	5e-5	4e-5
40	3e-5	4.75e-5	3e-5
60	2e-5	4e-5	2.83e-5

Interpretation of different models...

Associated net and DRNN gives comparable performance, but DRNN is outperformed by the former.

01

Associated net
This model gives the best accuracy and least loss for all the 3 prediction values

02

DRNN
This model also gives very good performance but is outperformed by associated net.

03

LSTM
Single LSTM layer has very less trainable parameters as compared to other models, but its loss is also more.

Conclusion

1. The Associated Net model which was proposed in the paper was replicated and the results were verified.
2. The model made by us showed even lesser loss as compared to that in the paper (data and hyperparameter dependent), but gave similar results against other models for comparison.
3. The model was able to closely predict the actual price of the stock.

Takeaway

There were many things we learned, among which a few are as follows-

1. Using external server to run heavy computational codes (Google Colab).
2. Using Tensorflow and numpy
3. About RNN, LSTM, and GRU.
4. We also looked at word predictor for sentence completion, to understand this problem better
5. Basics of stock market

Thank you!

Does **anyone** have any questions?