Stock Price Prediction using LSTM Model

Team: Ojas Mandlecha, Nimisha Salve. **Project Mentor TA:** Ramya Ramalingam

1) Abstract

The objective of this project is to predict the stock price of Reliance Industries, an Indian company, based on the previous three day's data, the market indicators as well as technical indicators using the LSTM model. This would help people to invest wisely since they could know the next day's price. Our target contribution is in the application which includes the market indices like simple moving average, exponential moving average, and market index as they affect the stock price significantly. Moreover, we are evaluating our model's performance by using different combinations of the stock price with the indicators and choosing the combination which enhances the accuracy the most. We have collected our data, built a baseline model, and found the optimal combination of indicators that has the best validation set accuracy. Furthermore, we have successfully predicted the next day's price.

2) Introduction

Stock price, being highly volatile, is very difficult to predict. It is affected by several factors, most of which cannot be easily accounted for. But forecasting the stock price is very necessary for investors who want to earn a profit through this trade. It is one of the most important ways to invest your wealth but it can also be very complicated to capture the movement of stock prices. People usually do not have the time or capacity to understand the stock market. Our model understands and takes into consideration relevant and useful market factors as features to predict the stock price and reduce the risk as much as possible. This model can help any layman to grow their money, by making smart investments. It will also help people from getting scammed by fake predictions from unreliable sources, such as friends, relatives, etc. This project is our attempt to forecast the stock price using technical indicators like moving averages and market indices that can help to give a clearer picture of the stock market movements so that people across all backgrounds can invest and profit through this trade.

We are using historical stock prices from the period March 25, 2020, to March 26, 2021 which adds up to 252 rows (days). This data is divided into three parts: the first 80% would be the training dataset, the next 10% would be the validation data, and the remaining 10% would be the test data. The baseline model for our project takes the three-day historical stock price as the input. In addition to this, we have included the simple moving average, exponential moving average, and market index price as inputs to predict the price. We have chosen Nifty 50 as the market index for the project as Reliance Industries has been a part of Nifty 50 companies. The output of the model is the stock price of the company on the next day.

3) Background

We are closely following the procedures and ideas of the following paper.

1. Jon Cavallie Mester "Using LSTM Neural Networks To Predict Daily Stock Returns", VT2021.

The author has used daily stock trading data to let an LSTM train model at predicting daily returns for 60 stocks from the OMX30 and Nasdaq-100 indices, that is whether the next stock price goes up or down. The Author has taken an extensive input dataset having 10 years of stock prices and thus, the average accuracy of the model has come out to be nearly 50%, which is as good as randomly guessing the future movement of the price and for some companies it has performed worse than a random guess. We think that these shortcomings of the old paper can be corrected and improved by making changes in the inputs and tuning the model.

4) Summary of Our Contributions

1. **Contribution(s) in Application/Data: We have predicted the stock price, not just on the closing price but also using technical indicators such as moving averages (simple and exponential moving averages), and market index.**

2. **Contribution(s) in Algorithm: The prior work which we are referring to considers the past 1 day for the prediction but we would make use of the past 3 days.**

3. **Contribution(s) in Analysis: We have analysed the model by using different combinations of these features and chose a model corresponding to the set of features that has given the highest accuracy in the validation and test set.**

5) Detailed Description of Contributions

5.1 Methods

1) Data:

There are 4 types of data that are needed to build the dataset for this project - moving average, exponential moving average, market index, and historical stock price information.
The historical data and market index are much easier to acquire, and we pulled them from the Yahoo Finance Python API (yfinance)[I]. These tickers in our case are:

- Reliance Industries Ltd. (RELIANCE.NS)
- Nifty 50 (^NSEI)

In addition to this, the API also takes in the start and the end dates and returns a data frame for
all the stock price data available between these 2 dates. In our case,

- Start date: is 2020-03-25 (March 25, 2020)
- End date: is 2021-03-26 (March 26, 2021)

The simple moving average (SMA) and the exponential moving average (EMA) are calculated based on the historical data. The SMA and EMA can be calculated for any number of previous days. We have made these functions flexible such that by changing the 'period' we can get the desired SMA and EMA. The most followed one is 20, so we have taken 20 for this project. We are then using a function to create three more columns representing the previous three days' prices. This makes it easier for the LSTM model to use these columns as inputs for training.

5.2) Algorithm

The LSTM comprises four neural networks and numerous memory blocks known as cells in a chain structure. A conventional LSTM unit consists of a cell, an input gate, an output gate, and a forget gate. The flow of information into and out of the cell is controlled by three gates, and the cell remembers values over arbitrary time intervals. The LSTM algorithm is well adapted to categorize, analyze, and predict time series of uncertain duration making it an ideal model for stock price prediction.

The period for the simple moving average and the exponential moving average is considered to be 20 days.

Pseudocode for LSTM

```
def LSTM_cell (prev_ct, prev_ht, input):
        combine = prev_ht + input
        f_t  = forget_layer(combine)
        candidate = candidate_layer (combine)
        i_t = input_layer(combine)
        C_t = prev_ct * f_t + candidate * i_t
        o,t = output_layer(combine)
        h_t = o_t * tanh(C_t)
        return h_t, C_t
c_t = [0,0,0]
h_t = [0,0,0]


for input in inputs:
        C_t, h_t = LSTM_cell (c_t, h_t, input)
```

3)Analysis

First we considered the While experimenting for different periods, it was found that when the time period was large, the model tried to overfit and caused high variance. Also, the stock price of a company during its establishment is not significant to predict today's stock price. Therefore, to overcome this, the time period was selected as 1 year. We have found that tuning the parameters like adding more dense layers, increasing the number of epochs, and reducing the learning rate gives a better model. But we have taken care of the fact that too many layers can overfit the model and the model would perform badly on validation and test data. Our performance metrics include the root mean square error and mean absolute error. The evaluation of the model is done based on the root mean square error wherein the model with the least RMSE is considered optimal.

5.2 Experiments and Results

Our model will make use of many features such as closing price, moving averages (simple and exponential moving averages), market index, interest rate, etc. It will tell us which set of features gives us the highest accuracy or lowest mean square error. Our goal is to find the combination of technical features that would give us the best accuracy for predicting the next day's price. First, we have implemented a model which would predict the stock price based on just the closing price. This is our baseline model. Then, we began adding different features to the input and monitored how it affected the output. Along with this, we also tuned the parameters to enhance the model's performance further.

We have imported the dataset of 'Reliance Industries' from Yahoo Finance for the period March 25, 2020, to March 26, 2021. This data would include the closing price for the particular dates. This data is divided into three parts: the first 80% would be the training dataset, the next 10% would be the validation data, and the remaining 10% would be the test data. We have found the simple moving average and exponential moving average using 20

days. We expected that as we add more features and keep the other factors the same, the accuracy of the model would improve as these features help us to better capture the time series components. For predicting the stock price on closing price, we have used the ReLU activation function, x number of layers, Adam optimizer and tuned the learning rate = 0.0005 with epochs = 10000. In addition to this, we added the simple moving average, exponential moving average, and market index as inputs separately. Then, in a combination of 2 and finally, all features were included in the input.
The resulting RMSE values for each combination have been displayed in the table below:

Input	RMSE
Baseline	
Baseline + SMA	
Baseline + EMA	
Baseline + Market Index	
Baseline + SMA + EMA	
Baseline + SMA + Market Index	
Baseline + EMA + Market Index	
Baseline + SMA + EMA + Market Index	

rest of page 4 and a half page 5 for graphs and final prediction (can take entire page 5 as well)
mention GPU runtime

6) Compute/Other Resources Used

For out project, we have mad use of Google Collab with the GPU runtime since LSTM requires GPU. The collab notebook was used for code development and runtime. We also used Yahoo Finance to collect our data, i.e., stock and market index prices.

7) Conclusions

Outcomes: We learnt how the LSTM cell actually works on sequential data and how the volatility of the stock market can be best captured by using complex models. Also, how the size of the data, number of input parameters and other hyperparameters affect the output. The time period of the data is a crucial parameter since the pandemic brought some drastic changes in the timeseries data all over the world. This project has successfully attempted to solve one of the most interesting problems of forecasting the stock price that could increase the average wealth of investors. Ultimately, this promotes investments in stocks which can help mitigate the effects of wealth inequality and poverty as well.

In Hindsight: Initially the size of the data had proven to be a problem for us, so we decided to implement different time periods and found that the time one year was the most sufficient. As mentioned in our first report, we had planned to include the interest rate set by the Reserve Bank of India (RBI) but we noticed that since it is a constant term throughout the year, it is not affecting the output significantly. Thus, we decided to discard that feature.

(optional) For the Future: What would you recommend as potential directions of improvement for someone looking to extend your work (maybe even yourself)?

Ethical considerations and broader social impact: As mentioned in our introduction, if correct, our methods could help promote more participation in the stock market. So, this model can help even an uneducated person to grow their money, savings, and investments for their own better future. This model will also help people from getting scammed by fake predictions from unreliable sources, such as friends, relatives, etc. In terms of ethical considerations, it is necessary to mention that if our model is faulty in some way, it could give terribly wrong predictions and cause someone trying to utilize it for profit to suffer losses. Therefore, this model should not be the only factor considered when making an investment. Investors are urged to do due research and think carefully about their risk tolerance under varied market circumstances.