Table of Contents

1. INTRODUCTION

1.1 Background Theory

Stock Market is a place where equity or small parts of all the public companies are bought and sold. Companies list their stock, the Securities Board regulates the prices, and buyers and sellers, as their name suggests, buy and sell the stocks. This act of trading is a factor that determines the price of any stock.

In today's day and time, the Stock Market is a common playground for numerous investors as well as multinational corporations. With the advancement of technology and rise of traders in the market, Stock Prediction has been mandatory for every investor wanting to leverage their profit and minimize loss. If anything, the rise of Machine Learning and Artificial Intelligence has only boosted this process. Banks, Hedge Funds, Market Makers and Brokerage Firms use Machine Learning and various complex algorithms to predict the prices of the stocks and their positions in the equities.

1.2 Objectives

The main objective of this project is to,

- Predict Stock Price of a given company in the given time using LSTM Neural Network.

1.3 Scope and Application

The model uses various features like stock date and previous closing price to predict the closing price for the given date. Applications of this project can be,

- In financial sector, to predict stock prices of any company,
- With slight modification, in sales prediction and analysis.

2. METHEDOLOGY

2.1 Model Architecture

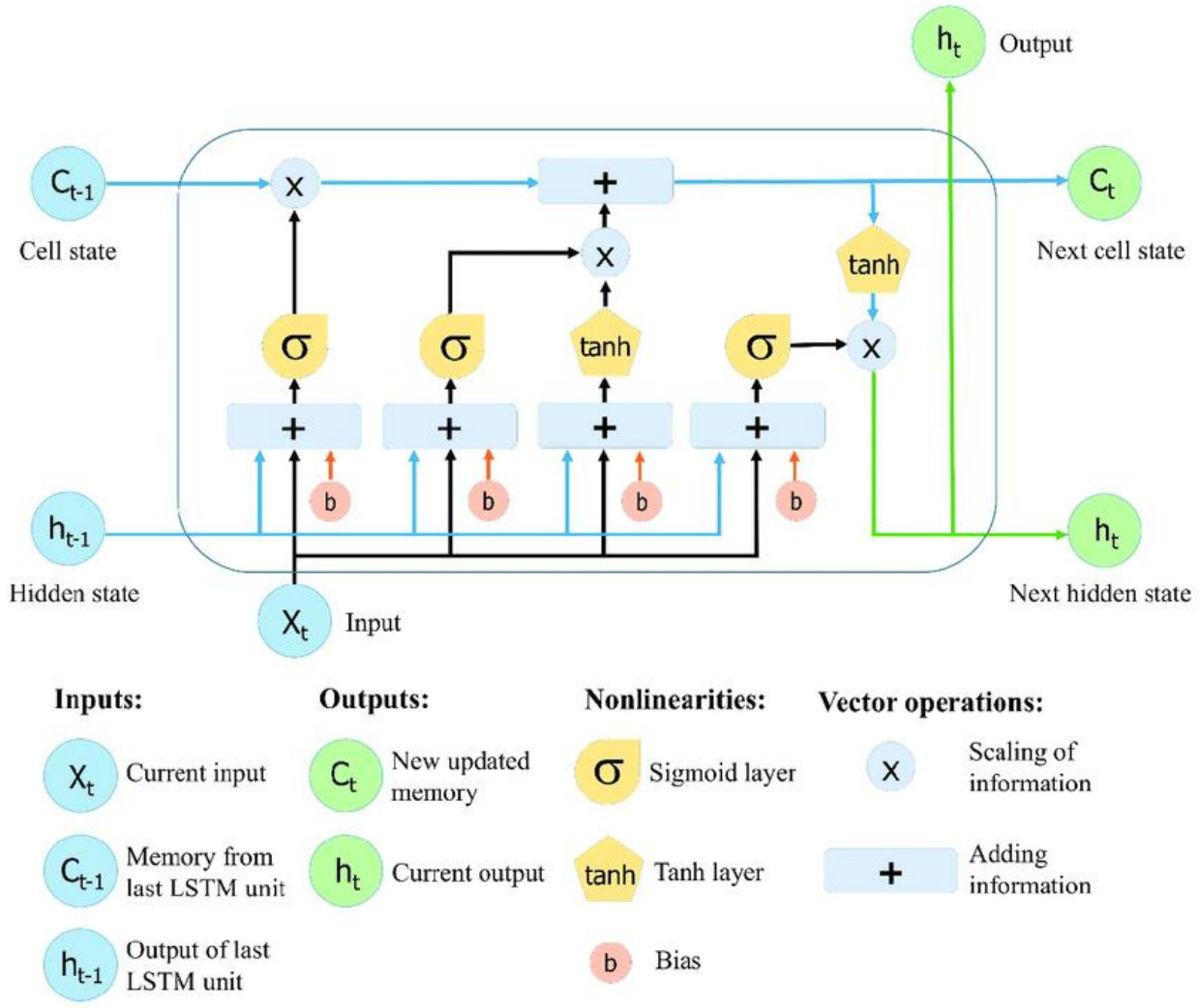

Fig 2.1.1. Model Architecture of LSTM Network

Long short-term memory is an artificial neural network used in the fields of artificial intelligence and deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. Such a recurrent neural network can process not only single data points, but also entire sequences of data.

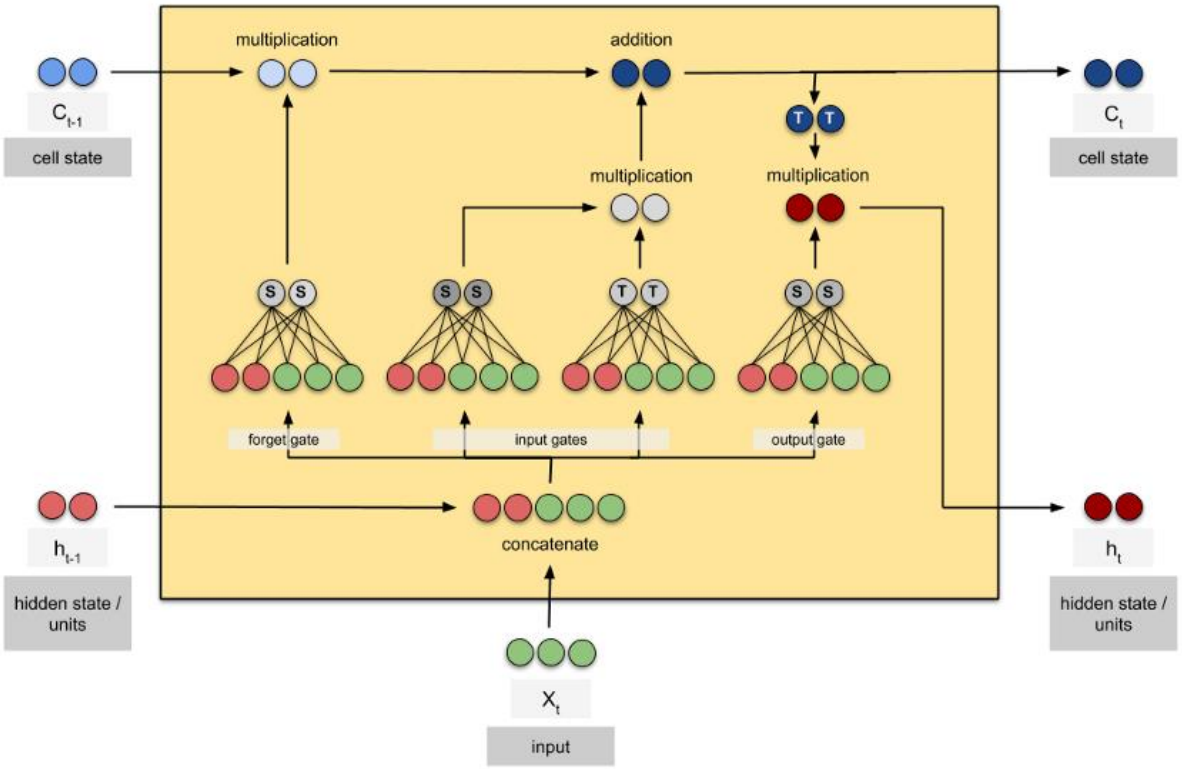

Fig. 2.1.2 Internal Architecture of LSTM with neural representation

2.2 Parameters

Firstly the 80% of data was divided for training purpose. Then the data was convert into numpy array.This data was then scaled using MinMaxScaler of scikit learn to [0,1].

```
[13] # Scaling the data
     scalar = MinMaxScaler(feature_range=(0,1))
     scaled_data = scalar.fit_transform(np_array)
     scaled_data
```

The model was trained in 7376 data, which were send in the batch of 60 to predict the 61th day. The model had two LSTM layer each with 50 perceptron and one with return sequence 'True' and one with 'False'.The model further contained two dense layer with 25 and 1 perceptron layer.

```
[17] # Model Creation
     model = Sequential()
     model.add(LSTM(50,return_sequences = True,input_shape= (X_train.shape[1],1)))
     model.add(LSTM(50,return_sequences = False))
     model.add(Dense(25))
     model.add(Dense(1))
```

The model used adam optimizer and mean_squared_error as loss function. The model had batch size as 1 and it was trained for 1 epochs.

```
[18] # Model Compilation
     model.compile(optimizer='adam',loss='mean_squared_error')
```

```
[19] # Model Training
     model.fit(X_train,y_train,batch_size=1,epochs=1)

     8006/8006 [==============================] - 292s 36ms/step - loss: 2.2332e-05
     <keras.callbacks.History at 0x7ff2a5bd0280>
```

2.3 Tools Used

Python

Python is a high-level, general-purpose programming language with a design philosophy that emphasizes code readability and the use of significant indentation which supports multiple program paradigms like functional, object oriented and structural. Python is used as the base language of all operations for this project everywhere from feature engineering to data process and visualization.

NumPy

NumPy is a python based mathematics library that has a specialization in dealing with arrays of multiple categories. In this project, we have used NumPy for the conversion of data in our dataset into an array to pass it as an argument in the model.

Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. Pandas is a core component for any data analysis and ML project and is used in almost every aspect of the process.

MatPlotLib

Matplotlib is a plotting library for Python programming language and its numerical mathematics extension NumPy. This was used frequently in the project for the visualization of the dataset and various features and their correlations.

Google Colab

Google Colaboratory or better known as Colab is an online web based IDE developed by Google Research. It allows remote test and execution of multiline codes. This tool was especially helpful for the project as the data set was very large, for which the processing would have taken an extensive time. But with the virtual resource available on Google Colab the training and testing of data was exponentially faster.

Yahoo Finance

Yahoo Finance is a media property that is a part of Yahoo Network that deals with the financial data, news and analysis. The dataset taken was part of the historic stock data offered by Yahoo Finance as a free to scrape csv file. The use of real life data sets from Yahoo Finance lead to a result that was easy to visualize and analyze.

3. DISCUSSION

3.1 Arguments on Parameter Values

Input layer:(60,1)

Loss Function: meant_squared_error

Optimizer: adam

Output layer shape :1

Neurons per hidden layer :50 for LSTM,

25 for Dense,

1 for Dense

4. CONCLUSION

4.1 Claims based on Parameter Values

4.1.1 Output

```
[32] valid
```

	Close	Prediction
8066	27.285000	29.638863
8067	28.827499	29.564745
8068	29.725000	29.780012
8069	29.290001	30.225052
8070	29.657499	30.529842
...	...	...
10077	141.110001	132.353592
10078	142.529999	133.619354
10079	141.860001	135.002060
10080	143.960007	135.958679
10081	145.929993	136.984009

2016 rows × 2 columns

Fig. 4.1.1.1 Closing Price Actual Vs Predicted for random sample data from the dataset.

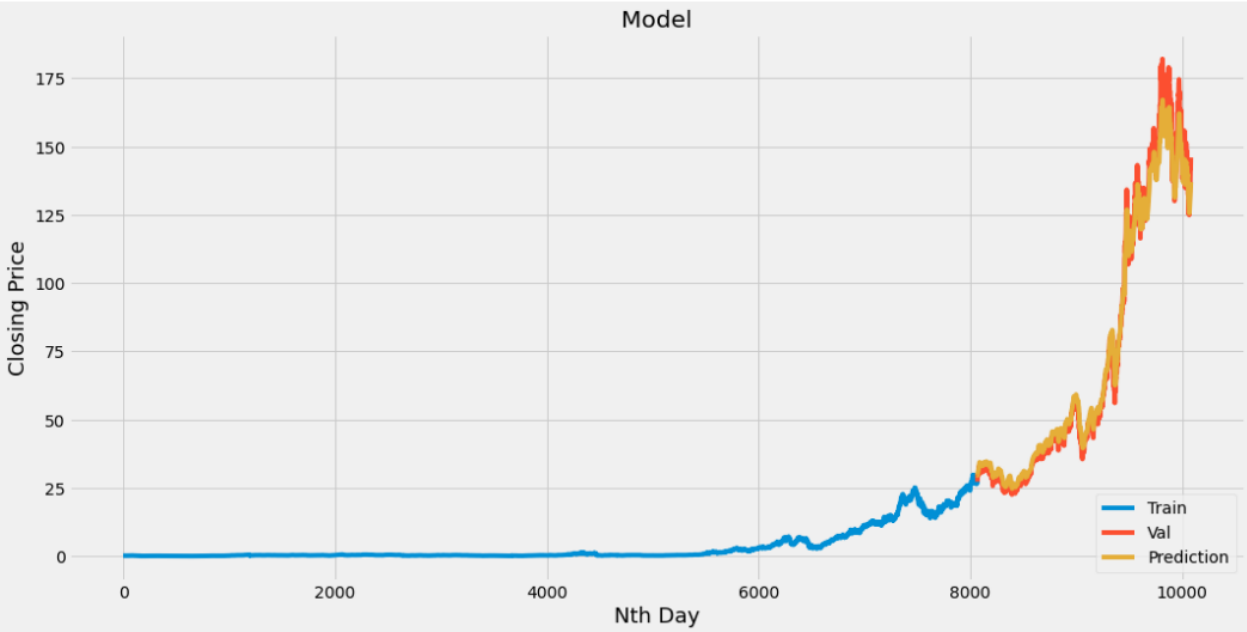

Fig 4.1.1.2. Graph between Closing Price (Train, Test and Predict)

4.1.2 Claims

```
[28] # Calculation of RMS Error
     rsme=np.sqrt(np.mean(prediction-y_test)**2)
     rsme

     0.2317599563573199
```

Fig. 4.1.2.1 Output showing a RMSE of 0.231

From the comparison column, we can see almost no discrepancy between the Closing and the Predicted Closing price of the stocks. The average deviation from actual value is marginal and well within the error metric of the model.

Hence from the graph, we can easily see that there is a major overlap between the validation data and the predicted data. This shows that the model is very accurate and was able to swiftly predict the price in the given time throughout the dataset.