Stock Price Prediction

Author : Miki Katyal
MikiKatyal@arizona.edu

Abstract

The Stock Market is a very unpredictable domain. Conventionally, the Prediction of stock Price is made manually based on the study of technical charts and fundamental study of the company's financial status. With improvement In machine learning algorithms, neural networks are used to learn stock price patterns and predict the price for validation. When this validation is done in past periods, it is called "Back-testing." Forecasting future prices is also one function where neural networks can be used.

1 Problem statement

In this project, daily data (OHLC—Open, High, Low, and Closing price) of a stock are given as input to a neural network to understand the following :

- How does this process work?
- How can stock price forecasting be done?
- Can stock price by itself contribute to the best possible prediction?

2 Outcome

The project will demonstrate :

- The process of using Neural Networks for stock price prediction and forecasting.
- The outcome of Prediction and Forecasting of stock price.

3 Understanding the stock price within stock exchange

The price of the stock at a given time (smallest unit of time being second), on an operating day of the stock exchange where stock is listed is the most elementary unit that is studied. This elementary unit is studied over a sequence of time as a time series data. Patterns in this time series data help to identify the trends, seasonality, and irregular fluctuation, which are essential for understanding the stock price behavior and forecasting it.

4 Understanding the Time Series Data

Time series data is a collection of observations recorded sequentially over some time. Every data point represents the value of a variable at a specific time, making time an essential factor in the dataset. In a time series, observations are spaced at regular intervals like minute-by-minute, daily, weekly, monthly, or bigger intervals based on the context. Some major characteristics related to

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

time series data are Trends, seasonality, Cyclic patterns, and Noise. Time series data is essential for Forecasting, Anomaly Detection, and Pattern recognition as this analysis of time series allows for observing how the phenomena change over time, revealing insights that help with understanding the underlying patterns.

5 Process followed in this project

- **Data Collection:**
  The stock price is downloaded from the public website of NSE (National Stock Exchange), https://www.nseindia.com/all-reports

- **Perform EDA (Exploratory Data Analysis):** Plot the data and visualize the price trend of the selected stock.

- **Select quantity of data input to Neural Network:** Select the number of days those are used as input to the Neural Network.

- **Data PreProcessing:** Fill blank columns. Use MinMaxScaler and convert stock price values between -1 and 1.

- **Split data for training and testing:** Fill blank columns, Use MinMaxScaler to convert stock price values between -1 and 1.

- **Train the Model:** on training data

- **Run Model for Prediction:** Use the Model with test data to make a Prediction. Use this to validate the Model with unseen data

- **Run Model for Forecasting:** Make a prediction one day beyond the test data and add this new prediction to the next set of input data. This process is repeated for the count of forecasting days.

- **Visualize the Prediction and Forecasting:** Plot the Prediction and the Forecasting data together to visualize the stock price trend in the future for the count of forecasting days.

- **Visualize and Compare Forecasting trend with ground truth:** Wait for some days to reach the Forecasting end date, and capture the ground truth from the public website of NSE.

6 What is LSTM ?

This project uses LSTM to predict and forecast the stock price. Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) that remembers information for a long time and applies that sorted data to future calculations.

6.1 Why LSTM ?

Early RNN suffered from the vanishing gradient problem, limiting their ability to learn long-range dependencies. LSTM was proposed in 1997 (by Hochreiter and Schmidhuber) as RNN to tackle the vanishing gradient problem. While the exploding gradient problem was solved with a technique named Gradient Clipping, this was easier than other techniques for preventing the vanishing gradient problem.

6.2 Architecture of LSTM

The LSTM network architecture consists of three parts in each cell.

- **First Part:** The first part chooses whether the information coming from the previous timestamp should be remembered or is irrelevant and can be forgotten.
- **Second Part:** The second part tries to learn new information from the input to this cell.
- **Third Part:** At last, in the third part, the cell passes the updated information from the current timestamp to the next timestamp. One cycle of LSTM is considered a single-time step.

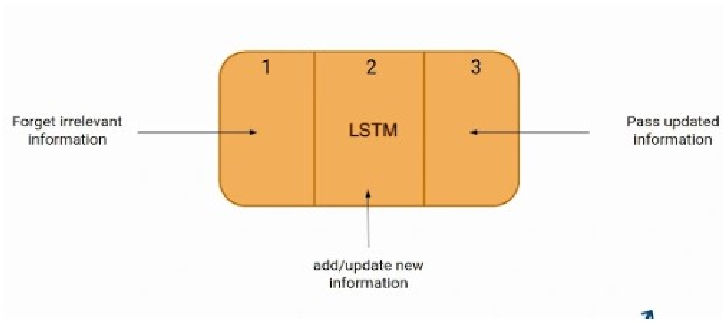

Figure 1: A cell in LSTM.

6.3 What is Long-Term and Short-Term Memory ?

LSTM also has a hidden state where H(t-1) represents the hidden state of the previous timestamp and Ht is the hidden state of the current timestamp.

In addition, LSTM's cell state is represented by C(t-1) and C(t) for the previous and current timestamps, respectively.

Here the hidden state is known as Short-term memory, and the cell state is known as Long-term memory. Refer to the following image.

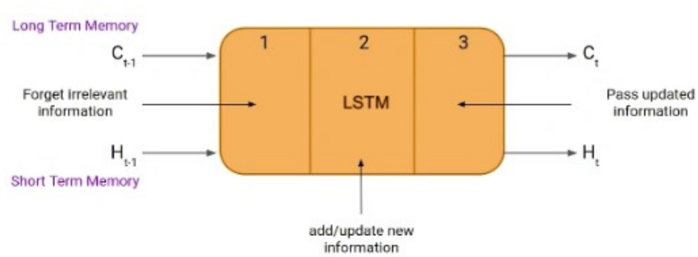

Figure 2: Cell Gates and Memory.

6.4 First Part - Forget Gate

The first step is to decide if to keep the information from the previous time step or forget it. The equation for the forget gate is

$$f_t = \sigma \left(x_t * U_f + H_{t-1} * W_f \right)$$

- Xt: input to the current timestamp.
- Uf: weight associated with the input

- Ht-1: The hidden state of the previous timestamp
- Wf: It is the weight matrix associated with the hidden state

A sigmoid function is applied to it. That will make ft a number between 0 and 1. This ft is later multiplied with the cell state of the previous timestamp, as shown below.

$C_{t-1} * f_t = 0 \;\; ...iff_t = 0$ (Forget everything)

$C_{t-1} * f_t = C_{t-1} \;\; ...iff_t = 1$ (Forget nothing)

6.5 Second Part - Input Gate

The input gate is used to quantify the importance of the new information carried by the input. The equation of the input gate is

$i_t = \sigma\left(X_t * U_i + H_{t-1} * W_t\right)$

- Xt: Input at the current timestamp t
- Ui: weight matrix of input
- Ht-1: A hidden state at the previous timestamp
- Wi: Weight matrix of input associated with hidden state

The sigmoid function is used. As a result, the value of I at timestamp t will be between 0 and 1.

New information

The New information that is required to be passed to the cell state is the function of a hidden state at the previous timestamp t-1 and input x at timestamp t.tanh activation function is used,as a result, the value of of new information will be between -1 and 1.

In case the value of Nt is negative, the information is subtracted from the cell state, whereas with a positive value of Nt the new information is added to the cell state at the current timestamp.

$N_t = tanh\left(X_t * U_c + H_{t-1} * W_c\right)$ (New Information)

The Nt won't be added directly to the cell state. An updated equation is used for that.

$C_t = f_t * C_{t-1} + i_t * N_t$ (Update cell state)

Here, Ct-1 is the cell state at the current timestamp, and the others are the values we have calculated previously.

6.6 Third Part - Output Gate

The final output gate has equations that is similar to the equations of the last two gates.The value will be between 0 and 1 as the sigmoid function is used.

$O_t = \sigma \left(x_t * U_o + H_{t-1} * W_o \right)$

To compute the current hidden state, Ot and tanh will be used to update the cell state.

$H_t = O_t * tanh \left(C_t \right)$

To get the final output of the current state, Use Softmax activation on hidden state HT.

$Output = Softwmax \left(C_t \right)$

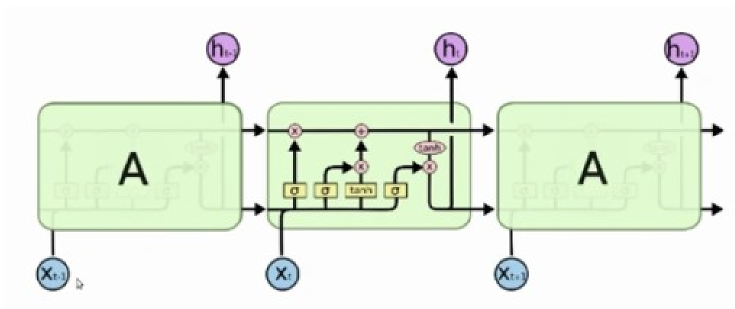

Figure 3: LSTM Cell in RNN.

7 Related Work

The Work in this project is not related to any existing publications. The goals of this project are :

- Compare the Neural Network's price trend prediction with the actual price trend and achieve the prediction closest to the actual price trend of the selected stock.
- Can the Price of stock by itself contribute to the best possible prediction ?

8 Experimental Results

1. Historical data with Forecasting

Figure 4: Historical Data with Forecast Data

2. Compare Ground Truth (from Trading View Application) with Forecasted data

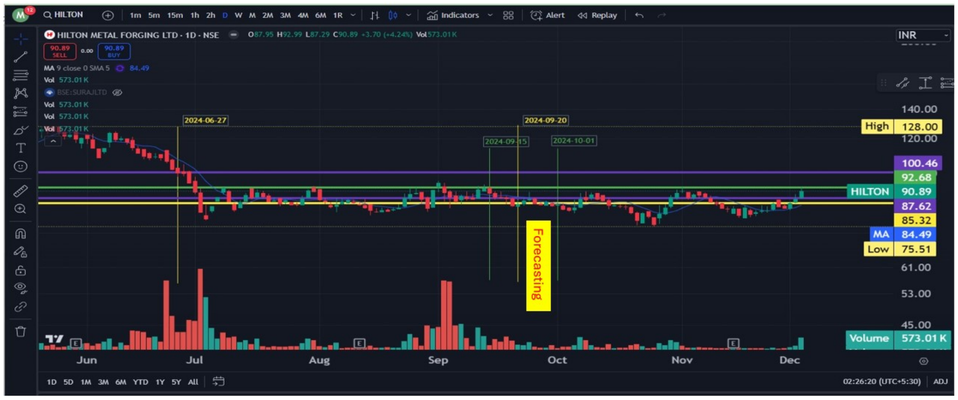

Figure 5: Ground Truth

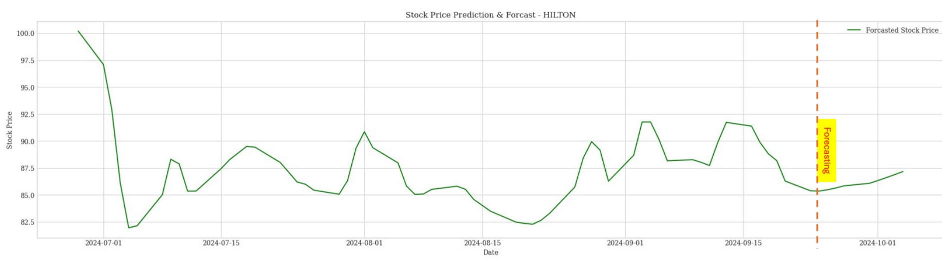

Figure 6: Forecast data

9 Conclusion

Stock price in isolation can contribute substantially to the prediction and forecasting of price trends, in a short period of 2 weeks.

10 References

- `https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/`
- `https://colah.github.io/posts/2015-08-Understanding-LSTMs/`