Stock Market Movement Prediction Using Long Short Term Memory Recurrent Neural Networks

Pruthviraj Sandeep Deshmukh
B.E, Department of CSE
RV College of Engineering

Abstract - The stock market is known for its extreme complexity and volatility. People are always looking for an accurate and effective way to guide stock trading. Additionally, the art of forecasting the stock prices has been a difficult task for many of the researchers and analysts. In fact, investors are highly interested in the research area of stock price prediction. For a good and successful investment, many investors are keen on knowing the future situation of the stock market. Good and effective prediction systems for the stock market help traders, investors, and analysts by providing supportive information like the future direction of the stock market. Long short-term memory (LSTM) neural networks are developed by recurrent neural networks (RNN) and have significant application value in many fields. In addition, LSTM avoids long-term dependence issues due to its unique storage unit structure, and it helps predict financial time series. In this work, we present a recurrent neural network (RNN) and Long Short-Term Memory (LSTM) approach to predict stock market indices.

I. INTRODUCTION

Over the years, various machine learning techniques have been used in stock market prediction, but with the increased amount of data and expectation of more accurate prediction, the deep learning models are being used nowadays which have proven their advantage over traditional machine learning methods in terms of accuracy and speed of prediction. In this article, we will discuss the Long-Short-Term Memory (LSTM) Recurrent Neural Network, one of the popular deep learning models, used in stock market prediction. In this task, we will fetch the historical data of stock automatically using python libraries and fit the LSTM model on this data to predict the future prices of the stock. While predicting the actual price of a stock is an uphill climb, we can build a model that will predict whether the price will go up or down. Predicting stock prices is an uncertain task which is modelled using machine learning to predict the return on stocks. There are a lot of methods and tools used for the purpose of stock market prediction. The stock market is considered to be very dynamic and complex in nature. An accurate prediction of future prices may lead to a higher yield of profit for investors through stock investments. As per the predictions, investors will be able to pick the stocks that may give a higher return.

II. LITERATURE REVIEW

Predicting stock market prices is a complex task that traditionally involves extensive human-computer interaction. Due to the correlated nature of stock prices, conventional batch processing methods cannot be utilized efficiently for stock market analysis. We propose an online learning algorithm that utilizes a kind of recurrent neural network (RNN) called Long Short Term Memory (LSTM), where the weights are adjusted for individual data points using stochastic gradient descent. This will provide more accurate results when compared to existing stock price prediction algorithms. The network is trained and evaluated for accuracy with various sizes of data, and the results are tabulated. A comparison with respect to accuracy is then performed against an Artificial Neural Network.

Traditional approaches to stock market analysis and stock price prediction include fundamental analysis, which looks at a stock's past performance and the general credibility of the company itself, and statistical analysis, which is solely concerned with number crunching and identifying patterns in stock price variation. The latter is commonly achieved with the help of Genetic Algorithms (GA) or Artificial Neural Networks (ANN's), but these fail to capture correlation between stock prices in the form of long-term temporal dependencies. Another major issue with using simple ANNs for stock prediction is the phenomenon of exploding / vanishing gradient, where the weights of a large network either become too large or too small (respectively), drastically slowing their convergence to the optimal value. This is typically caused by two factors: weights are initialized randomly, and the weights closer to the end of the network also tend to change a lot more than those at the beginning. An alternative approach to stock market analysis is to reduce the dimensionality of the input data and apply feature selection algorithms to shortlist a core set of features (such as GDP, oil price, inflation rate, etc.) that have the greatest impact on stock prices or currency exchange rates across markets. However, this method does not consider long- term trading strategies as it fails to take the entire history of trends into account; furthermore, there is no provision for outlier detection.

III. METHODOLOGY

Long-Short-Term Memory Recurrent Neural Network belongs to the family of deep learning algorithms. It is a recurrent network because of the feedback connections in its architecture. It has an advantage over traditional neural networks due to its capability to process the entire sequence of data. Its architecture comprises the cell, input gate, output gate and forget gate. The cell remembers values over arbitrary time intervals, and the three gates regulate the flow of information into and out of the cell. The cell of the model is responsible for keeping track of the dependencies between the elements in the input sequence. The input gate controls the extent to which a new value flows into the cell, the forget gate controls the extent to which a value remains in the cell, and the output gate controls the extent to which the value in the cell is used to compute the output activation of the LSTM unit. However, there are some variants of the LSTM model such as Gated Recurrent Units (GRUs) that do not have the output gate. LSTM Networks are popularly used on time-series data for classification, processing, and making predictions. The reason for its popularity in time-series application is that there can be several lags of unknown duration between important events in a time series.

Various types of neural networks can be developed by the combination of different factors like network topology, training method etc. For this experiment, we have considered Recurrent Neural Network and Long Short-Term Memory. This section we will discuss the methodology of our system. Our system consists of several stages which are as follows: -

Stage 1: Raw Data:
In this stage, the historical stock data is collected from https://www.quandl.com/data/NSE and this historical data is used for the prediction of future stock prices.

Stage 2: Data Pre-processing:
The pre-processing stage involves
a) Data discretization: Part of data reduction but with particular importance, especially for numerical data.
b) Data transformation: Normalization.
c) Data cleaning: Fill in missing values.
d) Data integration: Integration of data files. After the dataset is transformed into a clean dataset, the dataset is divided into training and testing sets so as to evaluate. Here, the training values are taken as the more recent values. Testing data is kept as 5-10 percent of the total dataset.

Stage 3: Feature Extraction: In this layer, only the features which are to be fed to the neural network are chosen. We will choose the feature from Date, open, high, low, close, and volume.

Stage 4: Training Neural Network:
In this stage, the data is fed to the neural network and trained for prediction assigning random biases and weights. Our LSTM model is composed of a sequential input layer followed by 2 LSTM layers and a dense layer with ReLU activation and then finally a dense output layer with linear activation function.

A RNN takes input from two sources, one is from the present and the other from the past. Information from these two sources are used to decide how they react to the new set of data. This is done with the help of a feedback loop where output at each instant is an input to the next moment. Here we can say that the recurrent neural network has memory. Each input sequence has plenty of information and this information is stored in the hidden state of recurrent networks. This hidden information is recursively used in the network as it sweeps forward to deal with a new example.

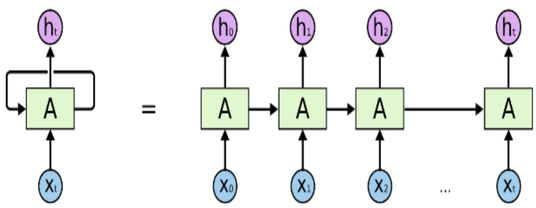

fig. 1 - Unrolled Recurrent Neural Network

In the case of RNN, the learned model always has the same input size, because it is specified in terms of transition from one state to another. Also the architecture uses the same transition function with the same parameters at every time step. LSTM is a special kind of RNN, introduced in 1997 by Hochreiter and Schmidhuber. In the case of LSTM architecture, the usual hidden layers are replaced with LSTM cells. The cells are composed of various gates that can control the input flow. An LSTM cell consists of input gate, cell state, forget gate, and output gate. It also consists of sigmoid layer, tanh layer and point wise multiplication operation. The various gates and their functions are as follows :

1. **Input gate:** Input gate consists of the input.

2. **Cell State:** Runs through the entire network and has the ability to add or remove information with the help of gates.

3. **Forget gate layer:** Decides the fraction of the information to be allowed.

4. **Output gate Layer:** It consists of the output generated by the LSTM.

5. **Sigmoid Layer:** generates numbers between zero and one, describing how much of each component should be let through.

6. **Tanh Layer:** generates a new vector, which will be added to the state.

Long-Short-Term Memory Function is a special type of RNN. These networks are proficient in learning about long-term dependencies. These networks are designed to evade the long term dependency problem. LSTM have a different structure compared to other neural networks. Conventional RNN has a very simple neural network with a feedback loop but LSTM consists of a memory block or cells instead of a single neural network layer. Each cell or block has 3 gates and a cell state tends to regulate the flow of data information through the cells.

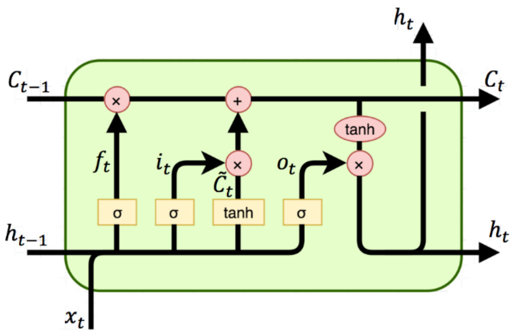

fig. 2 Internal Architecture of LSTM

In the above figure, the horizontal line passing through the top of the diagram is known as cell state (Ct−1, Ct). It acts like a conveyor belt that runs over the entire network. It carries the information from the previous cell to the present and so on. The decision for storing information in cell state is taken by the forget gate layer (ft) which is also known as sigmoid layer. The output from the forget gate is added to cell state using a pointwise multiplication operation. Next is the input gate which comprises a sigmoid layer (it) and tanh layer. Input gate combines these two into the cell state. Here C‘are the new values created by the tanh layer. Output (ht) is formed by a pointwise multiplication of sigmoid gate Ot and tanh.

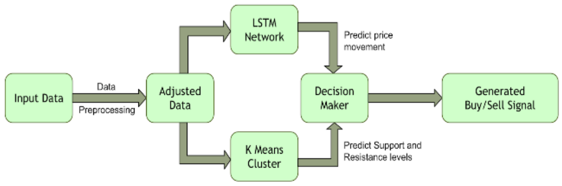

fig. 3 Data Flow Diagram

After crawling the Internet for specific open, high, low and closing prices for the stock of Infosys, we obtained the OHLC data for INFY in the National Stock Exchange. This data contains the daily OHLC prices and total amount of volume traded of INFY from 01-01-2000 to 31-12-2020.

On further analysis, the dataset is cleaned and pre-processed before any computation can be done on it. For this we utilise the data normalisation techniques. Once the data is formatted in the required order, we pass it through our two machine learning blocks.

The first block is the Long Short Term Memory Recurrent Neural Network. This block learns from the trend and the movement of the stock price and tries to predict the movement of the stock. It practically tries to predict the actual price of the underlying stock but we have tweaked it to perform such that it finds the difference between the previous price and tells us the movement instead of the price predicted by it.

The second is the K Means Clustering block. This performs a linear clustering on the Y coordinate values of the OHLC given to it and tries to predict the Support and Resistance levels of the stock. These key levels are then displayed on the screen along with the chart of the stock.

Finally, the predicted movement and support and resistance key levels are passed through a simple if and else block to determine what kind of a call should a trade take in order to enter a profitable trade. If the movement predicted is upwards and the price is near and above a considerable resistance level then a buy call is generated and if the movement predicted is downwards and the price is near and below a support level then a sell call is generated.

IV. RESULTS AND EXPERIMENTAL ANALYSIS

We have collected the individual results for each processing block and also showcased the accuracy of training for them.

```
train : 0.86
valid : 0.73
test : 0.72
```

fig. 4 Model Accuracy for the 3 datasets

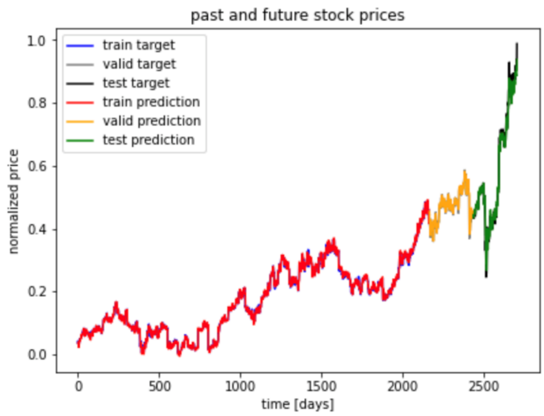

fig. 5 Graphical visualisation of model prediction

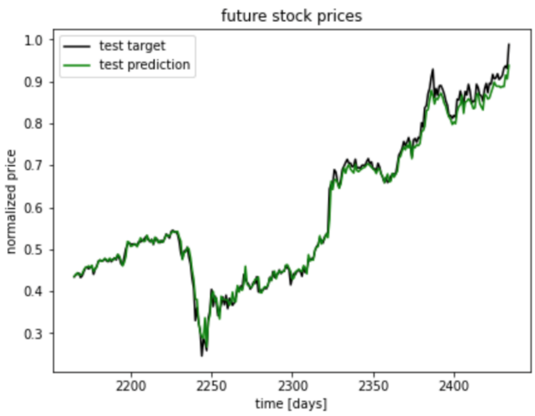

fig. 6 Model Prediction on test dataset

fig. 7 Plotting of Support and Resistance Levels on Graph

```
Predicted Signal : Long Position
Predicted Signal : Short Position
```

fig. 8 Signals generated after prediction

V. FUTURE WORK

Providing Target and Stop Loss values for the signals generated by finding relation between support and resistance level and strength of market movement. Currently our model only predicts the movement of the stock and the support and resistance levels to generate a signal for taking an appropriate position in the market. We have to optimise and tweak our system so that it can analyse the strength of the movement that is predicted and by taking the support and resistance levels into consideration, should provide the trader with an appropriate Stop loss and Profit target value. This can immensely help the trader because if the position is profitable, the trader should not exit the market too early. If the trader exits the market while the trend is still active, he ends up losing a portion of the profit that he could have collected otherwise. Having a Strong profit target can help the trader maximise profits. In the same way, if the trade is in loss, the trader should not wait in the market for too long because that way he ends up incurring more losses than he would have otherwise. Hence a strong Stop loss value is important so that the trader can cut the losses at the minimum.

Furthermore, instead of a trend following approach, we can also train the model to predict reversals in the market which can help in predicting long term swing positions and be useful for institutional investors. As of now, the model has only learned from the movement and trends that are showcased by the stock and hence the model can only predict the current trend of the stock. If we can train the model to only learn from the reversals and provide it the specific features of the stock during reversals, the model can learn to predict these reversal moves which can help long term swing traders to take appropriate positions in the market.

A sentiment analysis tool can be implemented to search for stocks which are currently trending in the news so that we can model our neural network to learn the movement of that stock and generate signals for it. Our model has been trained on the dataset that was provided to it which belongs to Infosys. As we move forward, it is important to consider that not all stocks have a good and volatile movement because of which taking positions in certain stocks can yield greater and quicker profits as compared to taking positions in other stocks. In order to understand which stock should be considered for modeling and analysis, we can implement a sentiment analysis algorithm. It can act like a web crawling tool which will keep tabs on the news that flows through the internet and twitter polls and ideas. By analysing this data, we can determine which stock is currently trending in the market. The more a stock is trending, the higher is the probability of that stock making big moves, which can be captured in the model. The sentiment analysis tool can also keep track of the stocks that are currently in our portfolio and change the stop loss or profit target values in real time according to the news, trend and current market sentiment of those stocks.

VI. CONCLUSION

This project proposes a RNN based on LSTM, which was built to forecast the movement of INFY. The prediction of our model has shown some promising results. The testing results confirm that our model is capable of tracing the trend in the market and by considering the key levels, can sufficiently predict the most logical position to take in the market to generate profits.

VII. REFERENCES

1. K. Hiba Sadia, Aditya Sharma, Adarrsh Paul, SarmisthaPadhi, Saurav Sanyal. “Stock Market Prediction Using Machine Learning Algorithms”, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-4, April 2019

2. Raghav Nandakumar, Uttam Raj K R, Vishal R, Y V Lokeswari. “Stock Price Prediction Using Long Short Term Memory”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issue: 03, March-2018

3. Nirbhey Singh Pahwa, Neeha Khalfay, Vidhi Soni, Deepali Vora. “Stock Prediction using Machine Learning a Review Paper”, International Journal of Computer Applications (0975 – 8887) Volume 163 – No 5, April 2017

4. V Kranthi Sai Reddy. “Stock Market Prediction Using Machine Learning”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issue: 10, Oct 2018

5. Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan. “Stock Market Prediction Using Machine Learning”, First Inter- national Conference on Secure Cyber Computing and Communications, National Institute of Technology (NIT), Jalandhar, DOI: 10.1109/ICSCCC.2018.8703332

6. Gareja Pradip, Chitrak Bari, J. Shiva Nandhini. “Stock market prediction using machine learning”, International Journal of Advance Research and Development (Volume 3, Issue 10)