© July 2021| IJIRT | Volume 8 Issue 2 | ISSN: 2349-6002

Stock Market Prediction

Tarun Aggarwal[1], Rahul Maurya[2], Divyanshu Gupta[3], Ashish Singh[4], Amul Maurya[5]
[1]Assistant Professor, Computer Science, Raj Kumar Goel Institute of Technology, Uttar Pradesh, India
[2,3,4,5] B.Tech Student , Computer Science, Raj Kumar Goel Institute of Technology, Uttar Pradesh, India

***Abstract* - The prediction of stock price is a complex task which needs a dynamic algorithms background in order to calculate the longer-term share values. Stock prices are interconnected within the nature of market; hence it will be challenging to predict the costs. The planned algorithm using the market data to predict the share price using machine learning techniques like recurrent neural network named as Long Short-Term Memory, in that process weights are improved for each data points using stochastic gradient descent. This system will provide precise outcomes in comparison to currently available stock price predictor algorithms. The network is trained and calculated with various dimensions of input data to urge the graphical results.**

***Index Terms* - Machine Learning, Stock Price Prediction, Long Short- Term Memory, Stock Market, Artificial neural Networks, National Stock Exchange.**

1.INTRODUCTION

Numerous studies have been the subject of using machine learning in the quantitative financial, predicting prices of supervision and constricting entire portfolio of assets, as well as, stock process, and many other actions can be covered by machine learning algorithms. In general machine learning is a term used for all algorithm approaches using computers to expose outlines based only on data and not using any programming instructions. For quantitative finance and particularly assets selections several models supply a large number of approaches that can be used with machine learning to prediction future assets price. This type of models offers a mechanism that combine weak sources of information and make it a strange tool that can be used efficiently. Recently, the combination of statistics and learning models have refined several machine learning algorithms, such as artificial neural networks, gradient boosted regression trees, support vector machines and random forecast. This model considers the historical equity share prices of a company price and applies RNN (Recurrent technique called Long Short-Term Memory (LSTM). The proposed approach considers available historic data of a share and it provides prediction on a particular feature. The features of shares are Opening price, day High, day Low, previous day o price, Close price, Date of trading, Total Trade Quantity and Turnover. The proposed model uses the time series analysis in price for a required time span. The data in this paper contain of the daily opening prices of two stocks in the New York Stock Exchange NYSE .

2. LITERATURE REVIEW

The initial focus of our literature review was to estimate different algorithms and models to choose whether stock price forecasts could be made on real stock prices [11]. However, as we have not been able to detect a possible variation in this stock price forecast, we decided to look at current plans, examine major issues, and increase ourselves. A short-term search of common solutions to the above problem led us to LSTM. Stock markets price movement prediction with LSTM neural networks: The objective of this research is to study the applicability of recurrent neural networks, in particular the LSTM networks, on the problem of stocks market prices movements forecast. A recurrent neural network method in predicting day-to-day stock prices an application to the US Stock Exchange Recurrent Neural Networks (RNN) is a sub type of neural networks that use feedback connections. Some types of RNN models are employed in predicting financial time series. This study was lead in order to develop models to predict daily stock prices of selected listed companies of US Stock Exchange based on Recurrent Neural Network (RNN) Approach and to measure the precision of the models developed and identify the shortcomings of the models if present.

3. RECURRENT NEURAL NETWORK (RNN) AND LONG SHORT-TERM MEMORY (LSTM)

In a NN that only contains one hidden layer the number of nodes in the input layer always depend on the dimension of the data, the nodes of the input layer connect to the hidden layer via links called ‘synapses’. The relation between every two nodes from (input to the hidden layer), has a coefficient called weight, which is the decision maker for signals. The values obtained after this transformation constitute the output layer of our NN, this value may not be the best output, in this case a back propagation process will be applied to target the optimal value of error, the back propagation process connect the output layer to the hidden layer, sending a signal conforming the best weight with the optimal error for the number of epochs decided. This process will be repeated trying to improve our predictions and minimize the prediction error. After completing this process, the model will be trained. The classes of NN that predict future value base on passed sequence of observations is called Recurrent Neural Network (RNN) this type of NN make use of earlier stages to learn of data and forecast futures trends. The earlier stages of data should be remembered to predict and guess future values, in this case the hidden layer act like a stock for the past information from the sequential data. The term recurrent is used to describe the process of using elements of earlier sequences to forecast future data. RNN can’t store long time memory, so the use of the Long Short-Term Memory (LSTM) based on “memory line” proved to be very useful in forecasting cases with long time data. In a LSTM the memorization of earlier stages can be performed trough gates with along memory line incorporated. The following diagram-1:

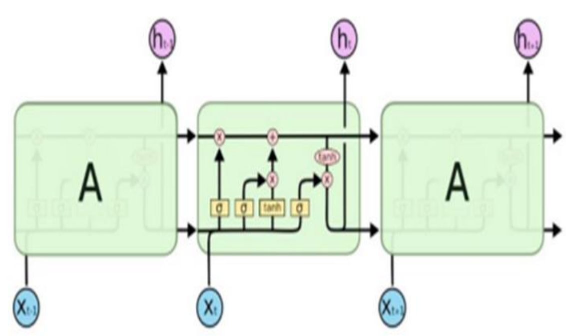

Figure 1. The internal structure of an LSTM [5].

Describe the composition of LSTM nodes. The ability of memorizing sequence of data makes the LSTM a special kind of RNNs. Every LSTM node most be consisting of a set of cells responsible of storing passed data streams, the upper line in each cell links the models as transport line handing over data from the past to the present ones, the independency of cells helps the model dispose filter of add values of a cell to another. In the end the sigmoidal neural network layer composing the gates drive the cell to an optimal value by disposing or letting data pass through. Each sigmoid layer has a binary value (0 or 1) with 0 “let nothing pass through”; and 1 “let everything pass through.” The goal here is to control the state of each cell, the gates are controlled as follow: - Forget Gate outputs a number between 0 and 1, where 1 illustration “completely keep this”; whereas 0 indicates “completely ignore this.” - Memory Gate chooses which new data will be stored in the cell. First, a sigmoid layer “input door layer” chooses which values will be changed. Next, a tanh layer makes a vector of new candidate values that could be added to the state. - Output Gate decides what will be the output of each cell. The output value will be based on the cell state along with the filtered and freshest added data.

4.A STOCK PRICE PREDICTOR USING LSTM

The proposed framework that studies online anticipating the close prices of the stock with the support of Long Short-Term Memory (LSTM). The Long Short-Term Memory (LSTM) is a counterfeit recurrent neural system (RNN) design [1] used in the field of deep learning, Unlike normal feed forward neural systems, LSTM has input associations. Not only does the procedure not focus on single information (e.g. pictures) but also on full information arrangements, (For example, a speech or a video). For example, LSTM is material for undertakings, such as un partitioned, associated penmanship recognition, speech recognition and recognition of peculiarities in arranged traffic or IDS (interruption location frameworks).

Algorithm 1: Stock prediction using LSTM
Input: Significant stock data
Output: prediction of stock price using price variation
Step 1: Start.

Step 2: Data Preprocessing after getting the historic data from the market for a specific share.

Step 3: import the dataset to the data structure and read the open price.

Step 4: do a feature scaling on the data so that the data values will vary from 0 and 1

Step 5: Creating a data structure with 60 timestamps and 1 output.

Step 6: Building the RNN (Recurrent neural network) for

Step 5 data set and Initialize the RNN by using sequential repressor.

Step 7: Adding the first LSTM layer and some Dropout regularization for removing unwanted values.

Step 8: Adding the output layer.

Step 9: Compiling the RNN by adding adam optimization and the loss as mean_squared_error.

Step 10: Making the predictions and visualizing the results using plotting techniques

Before processing the data there is a significant step that is to collect the information from market. Information assortment is the principle step in our proposed framework importing of the information from advertise clearing organizations like BSE (Bombay Stock Exchange) and NSE (National Stock Exchange). The dataset that will be utilized in the market expectation must be utilized to be separated dependent on different perspectives. Information assortment additionally supplements to upgrade the dataset by including more information that is outside. Our information for the most part comprises of the earlier year stock costs. For python available packages for retrieving the data from NSE is NSEpy .

5. METHODOLOGY AND DATA

The data in this paper comprise of the daily opening prices of two stocks in the New York Stock Exchange NYSE extracted from yahoo finance, for GOOGL our data series cover the period going from 8/19/2004 to 12/19/2019 and for NKE the data cover the period from 1/4/2010 to 12/19/2019. To build our model we are going to use the LSTM RNN, our model uses 75% of data for training and the other 25% of data for testing. For training we use mean squared error to improve our model. Also, we used different Epochs for training data (12 epochs, 25 epochs, 50 epochs and 100 epochs) our model will be structured as follow:

Table 1: the LSTM model summery

Layer (type)	Output Shape	Parameters
lstm_1 (LSTM)	(None, 50, 96)	37632
dropout_1 (Dropout)	(None, 50, 96)	0
lstm_2 (LSTM)	(None, 50, 96)	74112
dropout_2 (Dropout)	(None, 50, 96)	0
lstm_3 (LSTM)	(None, 50, 96)	74112
dropout_3 (Dropout)	(None, 50, 96)	0
lstm_4 (LSTM)	(None, 96)	74112
dropout_4 (Dropout)	(None, 96)	0
dense_1 (Dense)	(None, 1)	97
Total number of parameters : 260,065 Trainable parameters : 260,065 Non-trainable parameters :0		

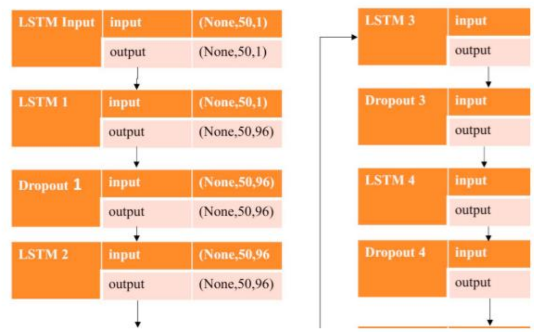

Figure 2: the LSTM model structure

6. RESULT AND DISCUSSION

After training our NN the result of our testing has shown different results, the number of epochs as well as the length of the data have both significant impact on the result of testing. For example, if we changed the dataset for NKE giving it a time set going from 12/2/1980 to 12/19/2019 the results will appear as follow:

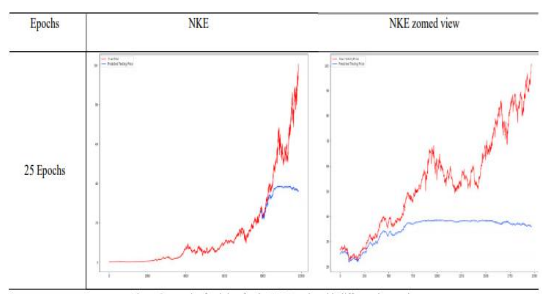

Figure 3: result of training for the NKE stocks with different dataset time

After observing our data, we can see that at first the data was less volatile and have lower values, in the figure, the red lines represent the real market value and the blue lines represent the predicted price value, after the NKE start peeking bigger values, the asset become more volatile, then the nature of this asset changed. In our case is better to avoid this type of change. Our model has lost trace of opening prices around 600 to 700 day of testing which conform the change in data nature. The result for our dataset for different number of epochs is giving by the following Figure:

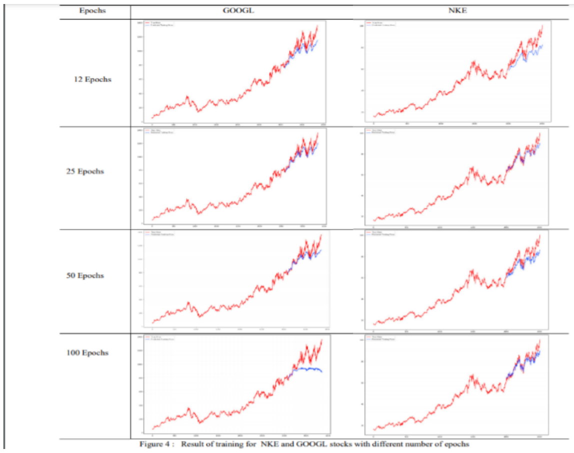

Figure 4 : Result of training for NKE and GOOGL stocks with different number of epochs

Table 2: the value of loss for GOOGL and NKE for deferrent number of epochs

	GOOGL		NKE	
	processing Time / sec	Loss	processing Time / sec	Loss
12 epochs	264	0.0011	132	0.0019
25 epochs	550	0.001	275	0.0016
50 epochs	1100	6.57E-04	550	0.001
100 epochs	2200	4.97E-04	1100	8.74E-04

7. CONCLUSION

This paper proposes RNN based on LSTM built to forecast future values for both GOOGL and NKE assets, the result of our model has shown some promising result. The testing result conform that our model is capable of tracing the evolution of opening prices for both assets. For our future work we will try to find the best sets for bout data length and number of training epochs that beater suit our assets and maximize our predictions accuracy.

In future enhancement the inclusion of sentimet analysis from social media to understand what the market thinks about the price variation for a particular share, and it can be implement this by adding twitter and Facebook it can be implement this by adding twitter and Facebook API to our program as Facebook is a leading social media which has lots of market trend information posted by users.

REFERENCES

[1] Batres-Estrada, B. (2015). Deep learning for multivariate financial time series.
[2] Emerson, S., Kennedy, R., O'Shea, L., & O'Brien, J. (2019, May). Trends and Applications of Machine Learning in Quantitative Finance. In 8th International Conference on Economics and Finance Research (ICEFR 2019).
[3] Heaton, J. B., Polson, N. G., & Witte, J. H. (2017). Deep learning for finance: deep portfolios. Applied Stochastic Models in Business and Industry, 33(1), 3-12.
[4] Moritz, B., & Zimmermann, T. (2016). Tree-based conditional portfolio sorts: The relation between past and future stock returns. Available at SSRN 2740751.
[5] Olah, C. (2015). Understanding lstm networks–colah's blog. Colah. github. io.
[6] Paiva, F. D., Cardoso, R. T. N., Hanaoka, G. P., & Duarte, W. M. (2018). Decision-Making for Financial Trading: A Fusion Approach of Machine Learning and Portfolio Selection. Expert Systems with Applications.
[7] Patterson J., 2017. Deep Learning: A Practitioner's Approach, O'Reilly Media.
[8] Siami-Namini, S., & Namin, A. S. (2018). Forecasting economics and financial time series: Arima vs. lstm. arXiv preprint arXiv:1803.06386.
[9] Takeuchi, L., & Lee, Y. Y. A. (2013). Applying deep learning to enhance momentum trading strategies in stocks. In Technical Report. Stanford University.
[10] Wang, S., and Y. Luo. 2012. "Signal Processing: The Rise of the Machines." Deutsche Bank Quantitative Strategy (5 June).
[11] Nandakumar, R., Uttamraj, K. R., Vishal, R., & Lokeswari, Y. V. (2018). Stock price prediction using long, short term memory. International Research Journal of Engineering and Technology, 5(03).
[12] Fig-2,3,4 is taken from www.elsevier.com.
[13] Table -1 and 2 are taken from www.elsevier.com. Using Random Forests", IEEE, 2017. Fang1, Philip Yu2, "Stock Market Prediction via