Stock Market prediction

-using RNN & Time series data

# N

*Eric Jonsson School of Engineering and Computer Science University of Texas at Dallas*

Richardson, USA [NXS210021@utdallas.edu](mailto:NXS210021@utdallas.edu)

# C

*Eric Jonsson School of Engineering and Compstock market importnceuter Science University of Texas at Dallas*

Richardson, USA [SAL200000@utdallas.edu](mailto:SAL200000@utdallas.edu)

# B

*Eric Jonsson School of Engineering and Computer Science University of Texas at Dallas*

Richardson, USA [PXR210008@utdallas.edu](mailto:PXR210008@utdallas.edu)

# D

*Eric Jonsson School of Engineering and Computer Science University of Texas at Dallas*

Richardson, USA [SXM200273@utdallas.edu](mailto:SXM200273@utdallas.edu)

***Abstract*—An efficient indication of market analysis demonstrates a country’s economy. Almost every industry is now valued by its stocks and thus stock markets have become an essential element in today’s world. As they serve as a criterion of what cycle the economy is in, and hopes and fears of people who generate wealth resulting in the greatest number of investments inclining towards stock markets. With such a growing demand and involvement of masses in stock markets, there comes a necessity for systems which could accurately provide predictions of future stock prices by learning on the past stock price given the time series data. Our designed network aims to do so for a certain institution.**

***Index Terms*—Stock, RNN, Time Series data**

1. INTRODUCTION

A stock is a financial instrument that represents ownership in a company or corporation and represents a proportionate claim on its assets (what it owns) and earnings (what it generates in profits). Stocks are also called shares or a com- pany’s equity. These are ownership equity in an organization or company and give shareholders voting rights as well as a residual claim on corporate earnings in the form of capital gains and dividends. More specifically, stock ownership im- plies that the shareholder owns a slice of the company equal to the number of shares held as a proportion of the company’s total outstanding shares. Investors come together on stock exchanges to buy and sell shares in a public venue. Share prices are set by supply and demand as buyers and sellers place orders.

The prices of shares on a stock market can be set in several ways. The most common way is through an auction process where buyers and sellers place bids and offers to buy or sell. A bid is the price at which somebody wishes to buy, and an offer (or ask) is the price at which somebody wishes to sell. When the bid and ask coincides, a trade is made. Millions of traders and investors, have differing ideas about the value of a specific stock and thus the price at which they are willing to buy or sell it come together to form the stock market. The thousands of transactions that occur as these investors and traders convert their intentions

to actions by buying and/or selling a stock cause minute-by- minute gyrations in it over the course of a trading day.

With millions of traders in the stock market it is desirable for systems with predictive capability of stock price values for various institutions. The stock prices are time series data where stock prices vary every business day. On some days it rises by a certain amount while on other days it falls by a certain amount. Hence, we can utilize any time series prediction model to develop a system which can be used for prediction of stock prices for a future date.

The purpose of this report is to build a predictive system that has learnt well from the historical data in order to perform better forecasting of stock prices. A time series data can be defined as a historical sequence of observations for a selected variable. In our case the variable is stock price. One of the most powerful and proven models for processing sequential data has been Recurrent neural networks since the past decade. One of the most successful RNN’s architectures is Long Short-Term memory (LSTM), which consists of the memory cell, which is a unit of computation that replaces traditional artificial neurons in the hidden layer of the network. LSTM is suitable to operate over data dynamically with high forecasting capacity because the memory cells help the network to associate memories and input remotely in time. [1]

The remaining report is organized as follows: Section II describes the background work in the field of machine learning in regards with building time-series prediction systems. Section III provides details about the theoretical and conceptual study of the proposed algorithm. Furthermore, Section IV describes the experimental setup required to implement the architecture including the libraries used, the dataset details and required preprocessing of the data. Section V includes the test results and analysis of the application. Lastly, Section VI and Section VII conclude the performance of this application and present research areas that can be addressed.

1. BACKGROUND WORK

Time series forecasting has been widely used to determine future prices of stocks, and the analysis and modeling of finance time series is an important task for guiding investors’ decisions and trades. Nonetheless, the prediction of prices by means of a time series is not trivial and it requires a thorough analysis of indexes, variables, and other data [2]. A set of data measured over time to acquire the status of some activity is known as a time series [3]. Linear models like AR, ARMA, ARIMA [4] [5] have been used for stock market forecasting. The only problem with these models is, that they work only for a particular time series data, i.e. the model identified for a particular company won’t perform well for another.

Over the past few years RNN and LSTM has been used a lot for problems involving time series data. A recurrent neural network (RNN) can be considered as multiple copies of the same NN, each copy sends the information to its successor using backpropagation through time. RNN is simple but it has a drawback which is known by Curse of dimensionality where gradient can be either vanishing or exploding. LSTM is a solution to that problem; it is suitable because of its ability to have memory and to distinguish between recent and older data using gates. 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.[6]

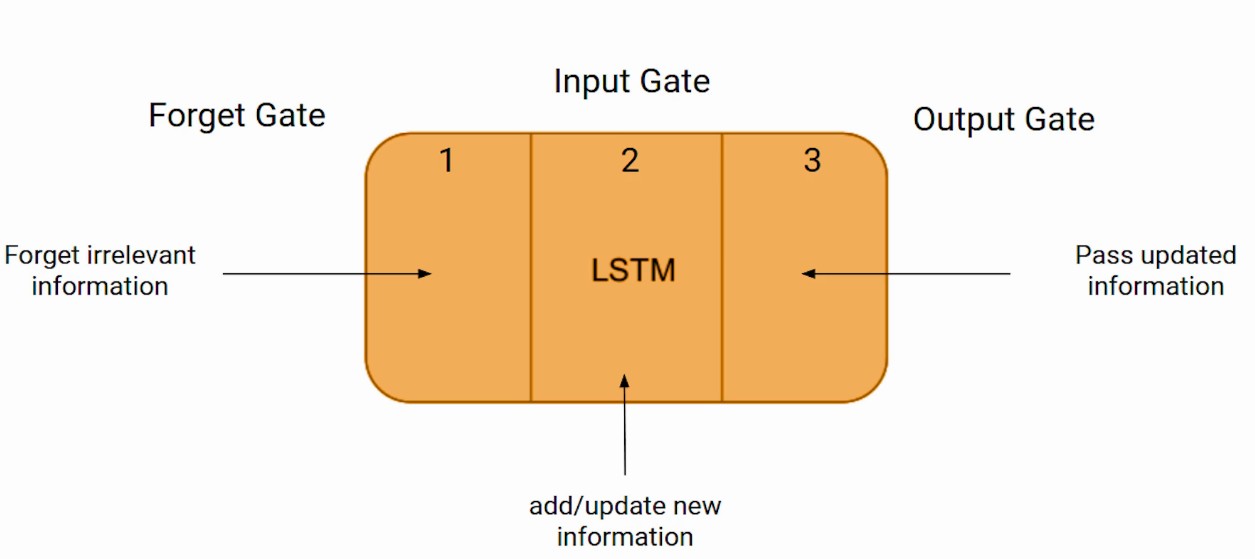


Fig. 1. Basic LSTM

1. THEORETICAL AND CONCEPTUAL STUDY

Neural networks are a subset of machine learning that recognize patterns from the given data and generalize from it. This model is composed of a network of neurons, just as human brain that transmit data and information based on the input from neighboring neurons. Neural networks computationally approximate a mapping between the inputs and outputs. They can be best considered as a function approximation algorithm to generalize and draw statistics from the input.

An Artificial Neural Network (ANN) is structured into layers of neurons and connections:

1. Input Layer
2. Hidden Layers
3. Output Layer [7]

Each neuron is connected to another neuron with an associated weight and threshold. Further, these association weights are adjusted to improve the performance of the network. An activation function enables the ANN to learn any nonlinear function. The output at each neuron is the activation of the weighted sum of its input.



Fig. 2. Artificial Neural Network

However, ANNs cannot be used while dealing with sequence data as they do not capture the sequential information. Hence, we have used Recurrent Neural Networks. The classes of NN that predict future value based 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.

Diagram, schematic

Description automatically generated

Fig. 2. Recurrent Neural Network

RNN takes two inputs, one is the present data and the other is from the past. The output for the new set of data is dependent on these two sources. Feedback loop enables us to send the output of each instant as the input to the next. Each input sequence has plenty of information and this information are 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. [8]

In a recurrent neural network, the input layer receives the input and then activations are applied in each hidden layer successively to produce the output. Each hidden layer is characterized by its own weights and biases. The commonly used activation functions are:

1. Sigmoid
2. Tanh
3. ReLU

Recurrent networks use the backpropagation learning methodology. Backpropagation networks are independent of the sequence in which the inputs are presented whereas the recurrent networks consider the sequence. Thus, the recurrent networks represent the idea of predicting stock market returns based on recent history more closely. The main difference between a feedforward backpropagation network and a recurrent network is the existence of a feedback mechanism in the nodes of the recurrent network. This feedback mechanism facilitates the process of using the information from the previous pattern along with the present inputs. [9]

The formula for the current state can be written as –

ht = f (ht-1, it)

where,

ht : New state

ht-1 : Previous state

it : Input

Applying the weights, we can update the above equation as –

ht = f (whhht-1, wxhit)

where,

whh : weight at the recurrent neuron

wxh : weight at the input neuron

After multiple hidden states, the final output state can be written as –

yt = whyht

All RNNs have feedback loops in the recurrent layer. This lets them maintain information in 'memory' over time. But for use cases with long-term temporal dependencies, it is difficult to train standard RNNs. This is because the gradient of the loss function decays exponentially with time (called the vanishing gradient problem). Hence 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.

As LSTM maintains a strong gradient over many time steps it can be trained with relatively long sequences. An LSTM unit has four major elements, “memory cell”, write, read, and forget gates.

Diagram

Description automatically generated

Fig. 3. Layers of a single LSTM module

Memory cell holds the data, and the three logic gates define the flow. The Write Gate is responsible for writing data into the memory cell. The Read Gate reads data from the memory cell and sends that data back to the recurrent network. The Forget Gate maintains or deletes data from the information cell, or in other words, determines how much old information to forget. In fact, these gates are the operations in the LSTM that execute some function on a linear combination of the inputs to the network, the network’s previous hidden state and previous output. [10]

1. DATASET DETAILS & PREPROCESSING

The dataset we are using in this project is the New York Stock Exchange [11]. This dataset contains historical stock prices with fundamental data of over 500 companies such as Yahoo Finance, fundamentals are from Nasdaq Financials, extended by some fields from EDGAR SEC databases.

We are using the prices-split-adjusted.csv file from this dataset which consists of raw daily prices where maximum data spans from 2010 to 2016 of companies new in the stock market. There have been 140 stock splits with some adjustments in this data.

For dataset preprocessing, we have dropped Null and NA values. We have also processed the data through min max scalar normalization for values to be in range. Furthermore, for better results and training our model we are focusing on 4 prime features – Open, Close, Low and High and dropped the remaining features. We have split our dataset in an 80:20 ratio for training and testing respectively.

V. RESULTS & ANALYSIS

While running our algorithm, we came across a major issue- on training the model with the dataset the Root Mean Square Error of the model kept increasing. As we increased the size of the dataset, the error increased exponentially. This is because of a fundamental issue of using RNN on timeseries data set. With the RNN model arises the exploding gradient issue which was causing the error to increase significantly.

To overcome this problem, we have defined a range for updating the weights, so that the calculated weights do not go beyond range specified.

For our eventual outcomes, we have attempted many ways to deal with get the most elevated precision conceivable. To standardize the information, we initially utilized the min/max standardization strategy, which permitted us to fairly precisely anticipate the future stock qualities, yet as it is stock costs, the mistake we were getting was excessively high.

Because of the contrasts between the actual information and our predicted information, we chose to attempt various methodologies and techniques as far as normalizing the information. At last, we chose utilizing 10 cycles on the preparation information with a learning rate of 0.00002 on 200k data. The accompanying two tables show the different boundaries we explored different combination of various parameter values with their testing and training accuracies as well.

Training Results and Graph

Table

Description automatically generated

Chart, line chart

Description automatically generated

Testing Results and Graph

Table

Description automatically generated

Line chart

Description automatically generated with medium confidence

# V1. CONCLUSION & FUTURE WORK

By making this time series model utilizing a recurrent neural network, we have took into account a technique wherein we can all the more precisely anticipate the stock costs later on. With respect to true applications, this calculation can be utilized to conclude which stocks to put resources into, and possibly at which times too. Simultaneously, in any case, it should be thought about that while this model predicts future stock costs basically dependent on noteworthy costs, there are numerous different variables that do go into the progressions in stock qualities, and as a future work, those extra factors can be carried out as a piece of this calculation. By executing those extra factors, we can all the more likely foresee stock costs, particularly as they might change because of financial slumps or monetary development. Besides, as future work, this model can likewise be extended to more readily fit stock trade value patterns in different nations, as patterns might change dependent on the space of the world.

REFERENCES

1. Stock Market Prediction by Recurrent Neural Network on LSTM Model (https://blog.usejournal.com/stock-market-prediction-by-recurrent- neural-network-on-lstm-model-56de700bff68)
2. https://www.sciencedirect.com/science/article/abs/pii/S156849461500352X
3. Hamzaebi C., Akay D. and Kutay F. (2009). ”Comparison of direct and

iterative artificial neural network forecast approachesin multi-periodic time series forecasting.” Expert Systems with Applications 36 (2): 3839- 3844.

1. Zhang G. P. (2003). ”Time series forecasting using a hybrid ARIMA and neural network model.”Neurocomputing 50:159-175.
2. Menon V. K., Vasireddy N. C., Jami S. A., Pedamallu V. T. N., Sureshku- mar V., and Soman K. P. (2016, June). ”Bulk Price ForecastingUsing Spark over NSE Data Set.”In International Conference on Data Mining and Big Data :137-146.
3. Zineb Lanbouri, Said Achchab,Stock Market prediction on High frequency data using Long-Short Term Memory,Procedia Computer Science,Volume 175,2020,Pages 603-608,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2020.07.087.
4. Adil Moghar, Mhamed Hamiche, Stock Market Prediction Using LSTM Recurrent Neural Network,Procedia Computer Science,Volume 170,2020,Pages 1168-1173,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2020.03.049.
5. Hiransha M, Gopalakrishnan E.A., Vijay Krishna Menon, Soman K.P.,NSE Stock Market Prediction Using Deep-Learning Models,Procedia Computer Science,Volume 132,2018,Pages 1351-1362,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2018.05.050.
6. J. Roman and A. Jameel, "Backpropagation and recurrent neural networks in financial analysis of multiple stock market returns," Proceedings of HICSS-29: 29th Hawaii International Conference on System Sciences, 1996, pp. 454-460 vol.2, doi: 10.1109/HICSS.1996.495431.
7. <https://medium.com/@nehapatil364/fundamentals-of-recurrent-neural-network-rnn-and-long-short-term-memory-lstm-network-b53f747fefaf>
8. Dataset source - https://www.kaggle.com/dgawlik/nyse