Volume: 11 Issue: V Month of publication: May 2023

DOI: https://doi.org/10.22214/ijraset.2023.51757

Stock Price Prediction Using Stacked LSTM

Mohan Badhekar[1], Sujan Shaikh[2], Vaishnavi Jadhav[3], Prof. Balasaheb Gite[4]
[1, 2, 3, 4]Computer Engineering Department, ISBM College of Engineering Nande, Pune-412115, India
Savitribai Phule Pune University

Abstract: In this project we attempt to implement machine learning approach to predict stock prices. Machine learning is effectively implemented in forecasting stock prices. The objective is to predict the stock prices in order to make more informed and accurate investment decisions. We propose a stock price prediction system that integrates mathematical functions, machine learning, and other external factors for the purpose of achieving better stock prediction accuracy and issuing profitable trades.

There are two types of stock trading. You may know of intraday trading by the commonly used term "day trading." Intraday traders hold securities positions from at least one day to the next and often for several days to weeks or months. LSTMs are very powerful in sequence prediction problems because they're able to store past information. This is important in our case because the previous price of a stock is crucial in predicting its future price. 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 for next 30 days

I. INTRODUCTION

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. The stock market allows investors to own shares of public companies through trading either by exchange or over the counter markets. This market has given investors the chance of gaining money and having a prosperous life through investing small initial amounts of money, low risk compared to the risk of opening new business or the need of high salary career. Stock markets are affected by many factors causing the uncertainty and high volatility in the market. Although humans can take orders and submit them to the market, automated trading systems (ATS) that are operated by the implementation of computer programs can perform better and with higher momentum in submitting orders than any human. However, to evaluate and control the performance of ATSs, the implementation of risk strategies and safety measures applied based on human judgements are required. Many factors are incorporated and considered when developing an ATS, for instance, trading strategy to be adopted, complex mathematical functions that reflect the state of a specific stock, machine learning algorithms that enable the prediction of the future stock value, and specific news related to the stock being analysed.

Time-series prediction is a common technique widely used in many real-world applications such as weather forecasting and financial market prediction. It uses the continuous data in a period of time to predict the result in the next time unit. Many timeseries prediction algorithms have shown their effectiveness in practice. The most common algorithms now are based on Recurrent Neural Networks (RNN), as well as its special type - Long-short Term Memory (LSTM) and Gated Recurrent Unit (GRU). Stock market is a typical area that presents time-series data and many researchers' studies on it and proposed various models. In this project, LSTM model is used to predict the stock price.

II. STOCK MARKET OVERVIEW

Almost every country has one or more stock exchanges, where the shares of listed companies can be sold or bought. It is a secondary market place. When a company first lists itself in any stock exchange to become a public company, the promoter group sells substantial number of shares to public as per government norms. During incorporation of a company shares are bought by promoter groups or institutional investors in a primary market. Once promoter offloads major portion of the shares to public retail investors, then those could be traded in secondary market i.e., in stock exchanges. In India the BSE (Bombay Stock Exchange) and the NSE (National Stock Exchange) are the two most active stock exchange. The BSE has around 5000 listed companies where as NSE had around 1600. Both the exchange has similar trading mechanism and market open time, closing time and settlement process. Stock exchanges helps individual investors to take part in the share market and allows to buy even a single share of some listed company with the help of a trading account and Demat account. These online markets have revolutionized the Indian investment arena along with government initiative like tax benefit on equity investment, National Pension Scheme (NPS) investing in share market etc. Due to continuous reduction in bank interest rates and increasing inflation middle class investors are moving towards equity market from the safe haven of fixed deposits. All these have helped to grow the capitalization of both the exchanges.

III. RELATED STUDIES

A. Stock Market Prediction Using Machine Learning

The research work done by V Kranthi Sai Reddy Student, ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India. In the finance world stock trading is one of the most important activities. Stock market prediction is an act of trying to determine the future value of a stock other financial instrument traded on a financial exchange. This paper explains the prediction of a stock using Machine Learning. The technical and fundamental or the time series analysis is used by the most of the stockbrokers while making the stock predictions. The programming language is used to predict the stock market using machine learning is Python. In this paper we propose a Machine Learning (ML) approach that will be trained from the available stocks data and gain intelligence and then uses the acquired knowledge for an accurate prediction. In this context this study uses a machine learning technique called Support Vector Machine (SVM) to predict stock prices for the large and small capitalizations and in the three different markets, employing prices with both daily and up-to-the-minute frequencies

B. Forecasting the Stock Market Index Using Artificial Intelligence Techniques

The research work done by Lufuno Ronald Marwala A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering. The weak form of Efficient Market hypothesis (EMH) states that it is impossible to forecast the future price of an asset based on the information contained in the historical prices of an asset. This means that the market behaves as a random walk and as a result makes forecasting impossible. Furthermore, financial forecasting is a difficult task due to the intrinsic complexity of the financial system. The objective of this work was to use artificial intelligence (AI) techniques to model and predict the future price of a stock market index. Three artificial intelligence techniques, namely, neural networks (NN), support vector machines and neuro-fuzzy systems are implemented in forecasting the future price of a stock market index based on its historical price information. Artificial intelligence techniques have the ability to take into consideration financial system complexities and they are used as financial time series forecasting tools.

C. Stock Price Correlation Coefficient Prediction with ARIMA-LSTM Hybrid Model

The research work done by Hyeong Kyu Choi, B.A Student Dept. of Business Administration Korea University Seoul, Korea. Predicting the price correlation of two assets for future time periods is important in portfolio optimization. We apply LSTM recurrent neural networks (RNN) in predicting the stock price correlation coefficient of two individual stocks. RNN’s are competent in understanding temporal dependencies. The use of LSTM cells further enhances its long-term predictive properties. To encompass both linearity and nonlinearity in the model, we adopt the ARIMA model as well. The ARIMA model filters linear tendencies in the data and passes on the residual value to the LSTM model. The ARIMA-LSTM hybrid model is tested against other traditional predictive financial models such as the full historical model, constant correlation model, single-index model and the multi-group model. In our empirical study, the predictive ability of the ARIMA-LSTM model turned out superior to all other financial models by a significant scale. Our work implies that it is worth considering the ARIMALSTM model to forecast correlation coefficient for portfolio optimization.

IV. LSTM ARCHITECTURE

A. An overview of Recurrent Neural Network (RNN)

In a classical neural network, final outputs seldom act as an output for the next step but if we pay attention to a real-world phenomenon, we observe that in many situations our final output depends not only the external inputs but also on earlier output. For example, when humans read a book, understanding of each sentence depends not only on the current list of words but also on the understanding of the previous sentence or on the context that is created using past sentences. Humans don’t start their thinking from scratch every second.

As you read this essay, you understand each word based on your understanding of previous words. This concept of ‘context’ or ‘persistence’ is not available with classical neural networks. Inability to use context-based reasoning becomes a major limitation of traditional neural network. Recurrent neural networks (RNN) are conceptualized to alleviate this limitation are networked with feedback loops within to allow persistence of information. The Figure 1Error! Reference source not found. shows a simple RNN with a feedback loop and its unrolled equivalent version side by side.

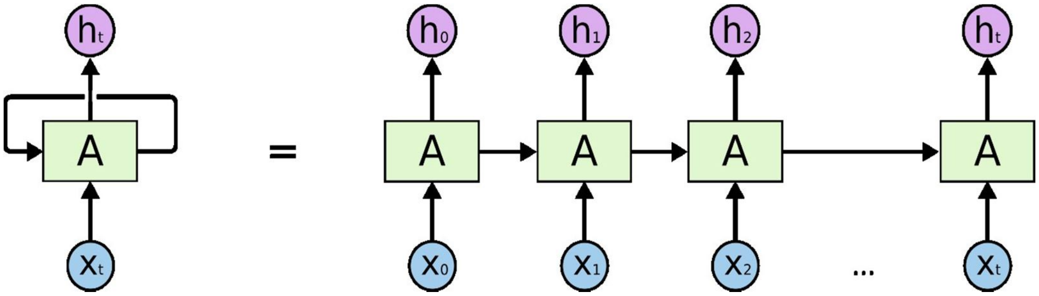

Figure 1: An unrolled recurrent neural network

Initially (at time step t) for some input Xt the RNN generates an output of ht. In the next time step (t+1) the RNN takes two input Xt+1 and ht to generate the output ht+1. A loop allows information to be passed from one step of the network to the next. RNNs are not free from limitations though. When the ‘context’ is from near past it works great towards the correct output. But when an RNN has to depend on a distant ‘context’ (i.e., something learned long past) to produce correct output, it fails miserably. This limitation of the RNNs was discussed in great detail by Hochreiter [8] and Bengio, et al. [9]. They also traced back to the fundamental aspects to understand why RNNs may not work in long-term scenarios. The good news is that the LSTMs are designed to overcome the above problem.

B. LSTM Networks

Hochreiter & Schmidhuber introduced a special type of RNN which is capable of learning long term dependencies. Later on, many other researchers improved upon this pioneering work. LSTMs are perfected over the time to mitigate the long-term dependency issue. The evolution and development of LSTM from RNNs are explained in given references. Recurrent neural networks are in the form of a chain of repeating modules of the neural network. In standard RNNs, this repeating module has a simple structure like a single tanh layer as shown in Figure 2

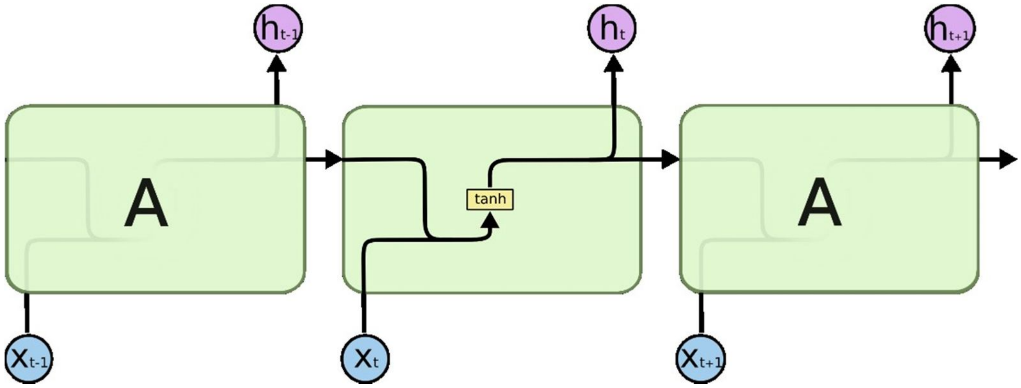

Figure 2: The repeating module in a standard RNN contains a single layer

LSTMs follow this chain-like structure; however, the repeating module has a different structure. Instead of having a single neural network layer, there are four layers, interacting in a very special way as shown in Figure 3.

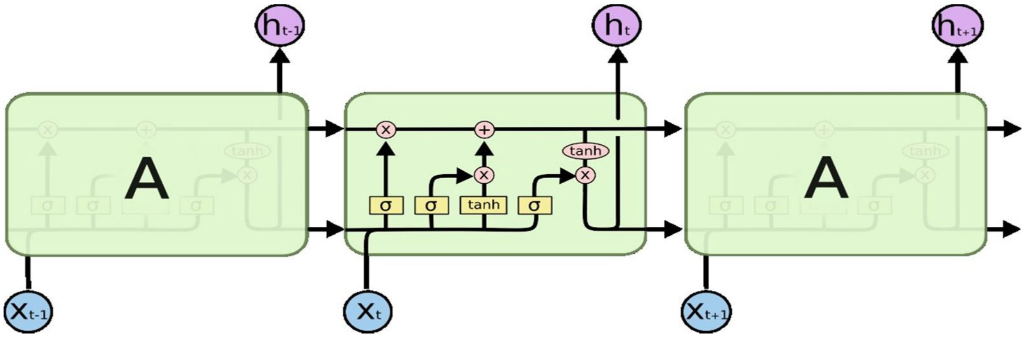

Figure 3: The repeating module in an LSTM contains four interacting layers

In Figure 3, every line represents an entire feature vector, from the output of one node to the inputs of others. The pink circles represent pointwise operations, like vector addition, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going to different locations.

C. The Working of LSTM

LSTM is a special network structure with three “gate” structures. Three gates are placed in an LSTM unit, called input gate, forgetting gate and output gate. While information enters the LSTM’s network, it can be selected by rules. Only the information conforms to the algorithm will be left, and the information that does not conform will be forgotten through the forgetting gate.

1) Forget Gate

A forget gate is responsible for removing information from the cell state.

- The information that is no longer required for the LSTM to understand things or the information that is of less importance is removed via multiplication of a filter.
- This is required for optimizing the performance of the LSTM network.
- This gate takes in two inputs; h_t-1 and x_t. h_t-1 is the hidden state from the previous cell or the output of the previous cell and x_t is the input at that particular time step.

2) Input Gate

- Regulating what values need to be added to the cell state by involving a sigmoid function. This is basically very similar to the forget gate and acts as a filter for all the information.
- Creating a vector containing all possible values that can be added (as perceived from h_t-1 and x_t) to the cell state. This is done using the tanh function, which outputs values from -1 to +1.
- Multiplying the value of the regulatory filter (the sigmoid gate) to the created vector (the tanh function) and then adding this useful information to the cell state via addition operation.

3) Output Gate

The functioning of an output gate can again be broken down to three steps:

- Creating a vector after applying tanh function to the cell state, thereby scaling the values to the range -1 to +1.
- Making a filter using the values of h_t-1 and x_t, such that it can regulate the values that need to be output from the vector created above. This f filter again employs a sigmoid function.
- Multiplying the value of this regulatory filter to the vector created in step 1, and sending it out as a output and also to the hidden state of the next cell.

V. FRAMEWORK TO FORECAST DIFFERENT SHARES PRICE

In this section, we shall first analyze some existing techniques and their merits to finally arrive at our methodology. Next, we shall discuss the algorithmic and implementation steps in detail.

A. Analyzing Different Methods

The prediction methods can be roughly divided into two categories, statistical methods and artificial intelligence methods. Statistical methods include logistic regression model, ARCH model, etc. Artificial intelligence methods include multi-layer perceptron, convolutional neural network, naive Bayes network, back propagation network, single-layer LSTM, support vector machine, recurrent neural network, etc. They used Long short-term memory network (LSTM).

For time series data, such as text, signals, stock prices, etc. LSTM is better suited to learn temporal patterns in deep neural networks. An LSTM solves the ‘vanishing gradient’ problem that exists in a RNN while learning long-term dependencies with time series dataset with the use of memory cell (states) and (input and forget) gates. So, LSTM may be a better option for future prediction of the company’s share price as well as growth.

B. Methodology

Used Long Short-term Memory (LSTM) with embedded layer and the LSTM neural network with automatic encoder.

LSTM is used instead of RNN to avoid exploding and vanishing gradients.

In this project python is used to train the model, MATLAB is used to reduce dimensions of the input.

The historical stock data table contains the information of opening price, the highest price, lowest price, closing price, transaction date, volume and so on.

C. Implementation Steps

1) *Step1: Data Collection:* Day-wise past stock prices of selected companies are collected from the Tiingo's official website.
2) *Step2: Data Pre-processing*: This step incorporates the following:
   a) Data Transformation: Normalization
   b) Data Cleaning: Fill in missing values.
   c) 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. Creating a data structure with 60 timesteps and 1 output.
3) *Step3: Feature Selection:* In this step, data attributes are chosen that are going to be fed to the neural network. In this study Closing Price is chosen as selected feature.
4) *Step 4: Train the model:* The model is trained by feeding the training dataset. The model is initiated using random weights and biases. Proposed LSTM model consists of a sequential input layer followed by 3 LSTM layers and then a dense layer with activation. The output layer again consists of a dense layer with a linear activation function.
5) *Step5: Output Generation:* The generated output is compared with the target values and error difference is calculated. The Backpropagation algorithm is used to minimize the error difference by adjusting the biases and weights of the neural network.
6) *Step 6: Test Dataset Update*: Step 2 is repeated for the test data set.
7) *Step 7: Error Calculation*: By calculating deviation we check the percentage of error of our prediction with respect to actual price.
8) *Step 8: Visualization*: Using Keras and their function APIs the prediction is visualized.

```
• # LSTM
• Inputs: dataset
• Outputs: RMSE of the forecasted data
• # Split dataset into 80% training and 20% testing data
• size = length(dataset) * 0.80
• train = dataset [0 to size]
• test = dataset [size to length(dataset)]
• # Procedure to fit the LSTM model
• Procedure LSTMAlgorithm (train, test, train_size, epochs)
• X_train = train
• Y_train = test
• model = Sequential ()
• model.add(LSTM(50,return_sequences=True,input_shape=(100,1)))
• model.add(LSTM(50,return_sequences=True))
• model.add(LSTM(50))
• model.add(Dense(1))
• model.compile(loss='mean_squared_error',optimizer='adam')
• model.fit(X_train, Y_train, validation_data=(X_test.ytest),epoch=100,batch_size=64,versbose=1)
• return model

• # Procedure to make predictions
• Procedure getPredictonsFromModel (model, X)
```

```
predictions = model.predict(X)
return predictions
epochs = 100
neurons = 50
predictions
# Fit the LSTM model
model = LSTMAlgorithm (train, epoch, neurons)

# Make predictions
pred = model.predict(train)

# Validate the model
n = len(dataset)
error = 0
for i in range(n): error += (abs(real[i] - pred[i])/real[i]) * 100
 accuracy = 100 - error/n
```

VI. PERFORMANCE MEASURE

A. Google

Epochs	Accuracy		RMSE	
	Train	Test	Train	Test
50	36.00%	97.34%	1864.57	77.42
100	34.21%	96.08%	1916.80	34.21
150	35.79%	95.36%	1870.68	135.07

Table 1: Epochs for Google Dataset

B. Apple

Epochs	Accuracy		RMSE	
	Train	Test	Train	Test
50	45.45%	62.71%	217.78	148.89
100	45.50%	62.97%	217.59	147.85
150	46.10%	62.36%	215.18	150.27

Table 2: Epochs for Apple Dataset

C. Amazon

Epochs	Accuracy		RMSE	
	Train	Test	Train	Test
50	27.06%	98.50%	2662.18	54.74
100	25.89%	98.32%	2704.82	61.27
150	26.67%	96.92%	2676.23	112.33

Table 3: Epochs for Amazon Dataset

D. Tesla

Epochs	Accuracy		RMSE	
	Train	Test	Train	Test
50	66.76%	90.88%	708.30	194.37
100	66.23%	89.80%	719.47	217.31
150	64.56%	87.62%	755.17	263.87

Table 4: Epochs for Tesla Dataset

VII. CONCLUSIONS

In this project, we are predicting closing stock price for the next 30 days of any given organization, we developed a model for predicting close stock price using LSTM algorithms for prediction. We have applied datasets belonging to Google, Apple, Microsoft and Tesla Stocks and achieved average 60% accuracy for training and average 80% accuracy for test datasets.

REFERENCES

[1] Stock Price Prediction Using LSTM on Indian Share Market by Achyut Ghosh, Soumik Bose1, Giridhar Maji, Narayan C. Debnath, Soumya Sen

[2] S. Selvin, R. Vinayakumar, E. A. Gopalkrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in International Conference on Advances in Computing, Communications and Informatics, 2017.

[3] Murtaza Roondiwala, Harshal Patel, Shraddha Varma, “Predicting Stock Prices Using LSTM” in Undergraduate Engineering Students, Department of Information Technology, Mumbai University, 2015.

[4] Xiongwen Pang, Yanqiang Zhou, Pan Wang, Weiwei Lin, “An innovative neural network approach for stock market prediction”, 2018

[5] Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan Department of Computer Science and Engineering National Institute of Technology, Hamirpur – 177005, INDIA - Stock Market Prediction Using Machine Learning.

[6] Pranav Bhat Electronics and Telecommunication Department, Maharashtra Institute of Technology, Pune. Savitribai Phule Pune University - A Machine Learning Model for Stock Market Prediction.

[7] Anurag Sinha Department of computer science, Student, Amity University Jharkhand Ranchi, Jharkhand (India), 834001 - Stock Market Prediction Using Machine Learning.

[8] V Kranthi Sai Reddy Student, ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India - Stock Market Prediction Using Machine Learning.

[9] Asset Durmagambetov currently works at the mathematics, CNTFI. Asset does research in Theory of Computation and Computing in Mathematics, Natural Science, Engineering and Medicine. Their current project is 'The Riemann Hypothesis-Millennium Prize Problems' - stock market predictions.

[10] Mariam Moukalled Wassim El-Hajj Mohamad Jaber Computer Science Department American University of Beirut - Automated Stock Price Prediction Using Machine Learning.