Comparative Performance Evaluation of Recurrent Neural Networks Using Pre- and Post-COVID Stock Market Data

Biswadarsi Biswal,Subhashree Sukla
Dept. of Computer Science and Engineering, GIFT Autonomous, Bhubaneshwar, 752054, India
Email: biswadarsi@gift.edu.in

Abstract:

The results of a neural network model depend on the selection of parameters. The parameters may include the quantity of the dataset, type of architecture, learning algorithm, data division ratio, etc. The idea of this research study is to find the optimal parameters of the Recurrent Neural Network (RNN) model for the prediction of the Bombay Stock Exchange market and analyze the performance of the RNN model by using a different range of datasets. The dataset range is classified as stock market index data during the COVID period, post-covid period, pre-covid period and the entire period, which includes pre-covid, post-covid and during the covid period. 150 candidate models were generated for every dataset; a total of 600 candidate models were generated, and the best models were selected for every dataset. The results of the RNN model perform well even though there is a fluctuation in stock market data.

Keywords: Recurrent Neural Network, Covid, Bombay Stock Market, Performance Analysis, Time Series Data.

Introduction:

The single word "corona" has terrified everyone in the world. The infectious disease was officially named COVID-19 by the World Health Organization on 2/11/2020 [1]. It was first reported in China, and spread quickly around the world. The researchers and data scientists are trying to develop a new model and validate the existing model to predict the confirmed case rate, death rate, transmission rate, etc., This research study is not aimed at developing a model for predicting things related to corona viruses. This research aims to test and analyze how the impact of a COVID virus is reflected in stock markets.

For the past several decades, the prediction of stock market indexes has been performed by machine learning algorithms with the help of historical data. Stock market data is not static. Any small change in any part of the world will affect the global stock market. During the pandemic periods of the Corona virus, there were fluctuations in the global stock market.

The author of this article have developed different neural network models [2]–[6] for the prediction of stock market data. A Multi-Layer Feed Forward Neural Network with Tracking Signal (MLFFNN-TS) approach has been developed in [4]. A MLFFNN-TS with fuzzy time series data has been developed in [5]. The NARXNN model with an iterative approach has been proposed in [6]. These models were tested on various time series data sets, like the Taiwan Stock Exchange Market, the BSE Stock Exchange Market, and the NN3 and NN5 forecasting competition time series data sets. All of the models produced results with greater than 99 percent accuracy.

The author also suggests implementing the methodology of Greg Heath [7] to avoid a bad random start to weight initialization. The selection of the best hidden layer's neurons is problem-dependent [8].

Adebiyi Ayodele [9] noted that the optimum model selection comes from training multiple networks with different configurations.

In the literature, train/validation/test datasets can be divided into 50/25/25 [10], 60/0/40 [11], 70/0/30 [12], or 80/0/20 [13]. The best neural network model is chosen based on the lowest error in the MAE, MSE, MAPE, RMSE, MPE, Theil'U, NMSE [14], or the highest value in POCID [10], R-Square [15], and R.

The RNN architecture is composed of feedback connections from every hidden neuron output to every hidden neuron input via a context layer. A detailed architecture and its basic concepts are found in [16]. The prediction performance is evaluated by the SMAPE; the reason is that the SMAPE is recommended by the NN3, NN5, and NNGC1 global forecasting competitions. A lower SMAPE value indicates better forecasting accuracy. In addition, the prediction performance is evaluated by the RMSE, R, and R-Square. A lower RMSE value and higher (ranging from 0 to 1) R and R-square values indicate better forecasting accuracy.

From the motivation of the study [2]–[6], this study develops a recurrent neural network model with an iterative approach to predict the closing index of a stock market. This research aims to conduct four different types of test cases. The first one is, historical data with the inclusion of covid period, i.e., POST-COVID with large dataset (9 years of dataset). Second one is, historical data with the exclusion of covid period, i.e., PRE-COVID with large dataset (7 years of dataset). Third one is, historical data inclusion of COVID period, i.e., POST-COVID with small dataset (2 years of dataset). The last one is, historical data with the exclusion of covid period, i.e., PRE-COVID with small dataset (3 years of dataset).

The remainder of this research work is structured as follows: Section II contains detailed reports on the proposed RNN model. Section III describes the experimental results attained by the RNN model with different test cases from the BSE dataset by using various performance measures and graphs. Finally, this study is concluded in section IV.

Methodology

This study presents a recursive approach to RNN models to reduce over fitting problems. The RNN model trains several networks by using various neurons with random initial weights, and the training function is Levenberg-Marq. Training parameters play a significant role in modeling a neural network to maximize forecasting accuracy. The proposed approach attempts to find optimal parameters, such as neuron counts in the hidden layer, an optimal weight, etc., for the prediction problem in a time series. This study aims to be one step ahead of predictions. Let y_1, y_2,... y_t represent a time series. As time t passes for t greater than one, the next value, y_{t+1}, is predicted based on the observed realizations of y_t, y_{t-1}, y_{t-2},..., y_1. The result of the network can be used for multi-step prediction by feeding the prediction back to the input of the network recursively. The proposed methodology of RNN is presented in Figure 1.

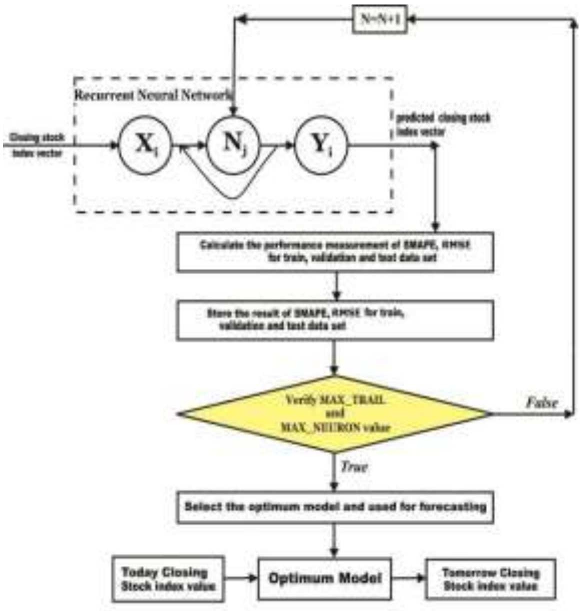
N=N+1
Recurrent Neural Network
Closing stock index vector
X_i
N_j
Y_i
predicted closing stock index vector
Calculate the performance measurement of SMAPE, RMSE for train, validation and test data set
Store the result of SMAPE, RMSE for train, validation and test data set
Verify MAX_TRAIL and MAX_NEURON value
False
True
Select the optimum model and used for forecasting
Today Closing Stock index value
Optimum Model
Tomorrow Closing Stock index value

Fig. 1: Recurent Neural Network Model

In Figure 1, x_i is the closing stock index vector, y_i is the predicted closing stock index from the neural network model, and N_i is the neuron size in the hidden layer. The execution process of the proposed RNN approach is modified from the previous study [6]. The only difference is that the previous study used the NARX neural network model and this study uses the RNN model. The execution process is explained in Algorithm 1.

Algorithm 1: Recurrent Neural Network.

```
1. Read the data file, which contains input and target values.
2. Normalize the data.
3. MAX_NEURON=10
4. MAX_TRIAL =15
5. FOR NEURON = 1 TO MAX_NEURON
6. FOR TRIAL = 1 TO MAX_TRIAL
7. Create RNN(input,target)
8. Define the number of hidden layers, training function, etc.
9. Define the data division ratio, training, validation and test data set.
10. Train the recurrent neural network.
11. Simulate the recurrent neural network
12. Denormalize the simulated neural network output data.
13. Calculate the performance measure SMAPE, RMSE, R, and R-Square for the training, validation and test data set.
14. Record the result of NEURON, TRIAL and performance measure specified in step 13.
15. END // TRIAL
16. END //  NEURON
17. From step 14, select the optimal RNN model, which provides less error in SMAPE of a validation dataset.
```

Results and Discussions

In this study, four test cases were conducted on a BSE stock index dataset BSE (http://www.bseindia.com /indices/IndexArchiveData.aspx ?expandable=3). Test Case 1 uses the dataset range spanning from 01.01.2013 to 15.02.2022, which includes a COVID period (i.e., POST-COVID with the large dataset). Test Case 2 uses the dataset range from 01.01.2013 to 30.01.2020, which does not include a COVID period (i.e., PRE-COVID with the large dataset). Test Case 3 uses the dataset range spanning from 31.01.2020 to 31.01.2022, which includes the COVID period (i.e., Post-COVID with the small dataset). Test Case 4 uses the dataset range from 31.01.2018 to 31.01.2020, which does not include a COVID period (i.e., PRE-COVID with a small dataset).

In this experiment, in every test case, 150 candidate models were generated. In total, 600 models were generated. Each RNN is created with a different neuron ranging from 1 to 10 in the hidden layer with 15 random initial weights. The data splitting ratio is 50/25/25. 50% is used for training; 25 % for validation, and 25 % for testing. From the ten architectures of different trials, optimum RNN model selection was based on the validation set with the lowest performance error, SMAPE. 1-7-1 with trial 3 is selected as the optimal model for Test Case 1. Here, 1-7-1, the first part indicates the neuron count in the input layer; second part indicates the neuron count in the hidden layer, and the third part indicates the neuron count in the output layer. Trial 3 indicates the random initial weight generation. Similarly, 1-2-1 with trial 6 is selected as the optimal model for Test Case 2. 1-2-1 with trial 3 is selected as the optimal model for Test Case 3. 1-9-1 with trial 3 is selected as the optimum model for Test Case 4.

For Every test case, the performance measures SMAPE, RMSE, R, and R-Square of the training, validation, and test datasets were analyzed, and the optimum results of every test case were tabulated as shown in Tables 1 to 4.

TABLE I. PERFORMANCE ANALYSIS OF POST-COVID WITH LARGE DATASET - (TEST CASE 1)

Performance Measure	Train	Val	Test	Accuracy Test
SMAPE	0.74	0.53	0.98	99.02
RMSE	72.60	67.50	180	
R-Square	1.00	1.00	1.00	
R	1.00	1.00	1.00	

TABLE II. PERFORMANCE ANALYSIS OF PRE-COVID WITH LARGE DATASET (TEST CASE 2)

Performance Measure	Train	Val	Test	Accuracy Test
SMAPE	0.75	0.70	0.65	99.35
RMSE	71.60	75.90	96.40	
R-Square	1.00	0.98	0.96	
R	1.00	0.99	0.98	

TABLE III. PERFORMANCE ANALYSIS OF POST-COVID WITH SMALL DATASET (TEST CASE 3)

Performance Measure	Train	Val	Test	Accuracy Test
SMAPE	2.02	0.78	0.76	99.24
RMSE	307	158	174	
R-Square	0.93	0.97	0.88	
R	0.98	0.98	0.94	

TABLE IV. PERFORMANCE ANALYSIS OF PRE-COVID WITH SMALL DATASET (TEST CASE 4)

Performance Measure	Train	Val	Test	Accuracy Test
SMAPE	0.70	0.56	0.70	99.30
RMSE	99.90	80.00	109.00	
R-Square	0.94	0.79	0.95	
R	0.98	0.89	0.98	

From Table 1 to Table 4, it is observed that the performance measures R and R-Square are closest to one in the training set, validation set, and test set of four test cases. It indicates that the RNN model is considered a perfect fit.

The performance measures of SMAPE in test sets of four test cases were 0.92, 0.65, 0.75, and 0.68, respectively. It indicates that the RNN model gives high-accuracy prediction results. Test case 1 produces 99.08 percent accuracy. Test case 2 produces 99.35 percent accuracy. Test case 3 produces 99.25 percent accuracy. Test case 4 produces 99.32 percent accuracy. The conclusion is that if the best parameter is identified by the neural network designer for the given problem, then the model will give a high-accuracy result. The performance measures of RMSE in test sets of four test cases were 180, 98.6, 174, and 109, respectively. It indicates that test case 2 produces the lowest error.

Figs. 2 and 3 show the line chart of actual versus predicted values for train and test datasets for POST-COVID with large datasets (Test Case 1). In the line chart, a solid line indicates forecasted data; a dotted line represents actual data. The X-axis indicates time periods, and the Y-axis indicates the closing index of the BSE stock market. From the figures, the RNN model predicted the future value perfectly.

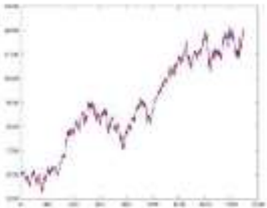

Fig. 2: Line Chart of Actual Versus Predicted for Train Data for POST-COVID with large dataset (Test Case 1)

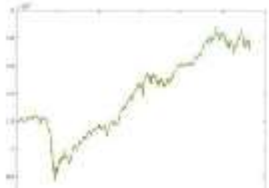

Fig. 3: Line Chart of Actual Versus Predicted for Test Data for POST-COVID with large dataset (Test Case 1)

Figs. 4 and 5 show the line chart of actual versus predicted values for train and test datasets for PRE-COVID with a large dataset (Test Case 2).

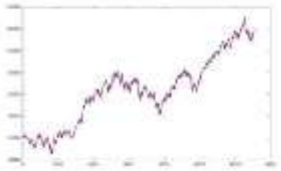

Fig. 4: Line Chart of Actual Versus Predicted for Train Dataset for

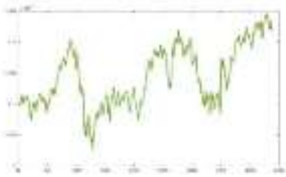

Fig. 5: Line Chart of Actual Versus Predicted for Test Dataset for PRE-

PRE-COVID with Large Dataset (Test Case 2)

COVID with Large Dataset (Test Case 2)

Figs. 6 and 7 show the line chart of actual versus predicted values for train and test datasets for POST-COVID with a Small Dataset (Test Case 3).

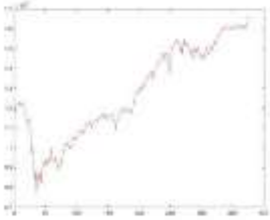

Fig. 6: Line Chart of Actual Versus Predicted for Train Dataset for POST-COVID with Small Dataset (Test Case 3)

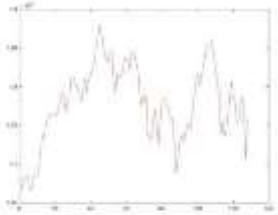

Fig. 7: Line Chart of Actual Versus Predicted for Test Dataset for POST-COVID with Small Dataset (Test Case 3)

Figs. 8 and 9 show the line chart of actual versus predicted values for train and test datasets for PRE-COVID with a Small Dataset (Test Case 4).

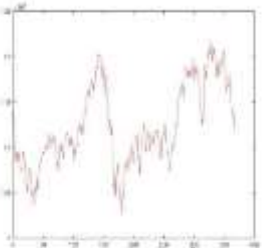

Fig. 8: Line Chart of Actual Versus Predicted for Train Dataset for PRE-COVID with Small Dataset (Test Case 4)

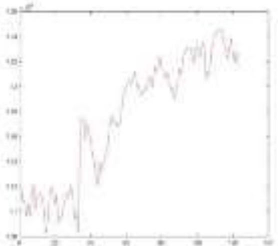

Fig. 9: Line Chart of Actual Versus Predicted for Test Dataset for PRE-COVID with Small Dataset (Test Case 4)

Figs. 10 and 11 show the regression plot to display the network output with respect to target for training and test dataset for POST-COVID with a large dataset (Test Case 1).

From Figs. 10 to 17, it is observed that, the dataset falls near 45 degrees for all test cases. It indicates that the RNN model generates a perfect fit. During the training phase, the dataset falls closest to 45 degrees for all test cases, whereas during the testing phase, the data slightly deviates from the straight line except for test case 1. It indicates that the neural network generates a perfect fit when the dataset is large.

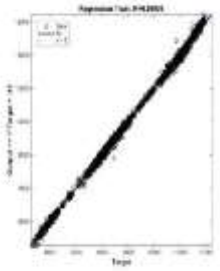

Fig. 10: Regression Plot for Train Dataset for POST-COVID with large dataset (Test Case 1)

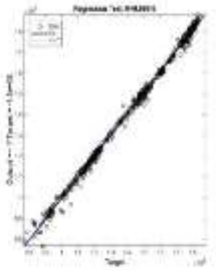

Fig. 11: Regression Plot for Test Dataset for POST-COVID with large dataset (Test Case 1)

Figs. 12 and 13 show the regression plot to display the network output with respect to target for training and test dataset for PRE-COVID with a Large Dataset (Test Case 2).

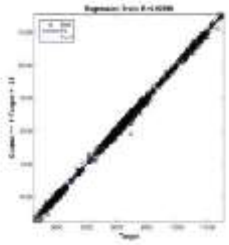

Fig. 12: Regression Plot for Train Dataset for PRE-COVID with Large Dataset (Test Case 2).

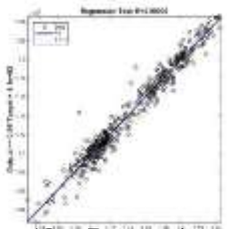

Fig. 13: Regression Plot for Test Dataset for PRE-COVID with Large Dataset (Test Case 2).

Figs. 14 and 15 show the regression plot to display the network output with respect to target for training and test dataset for POST-COVID with a Small Dataset (Test Case 3).

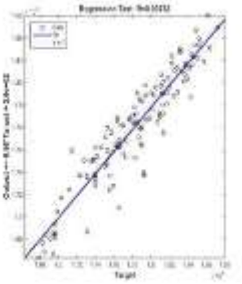

Fig. 14: Regression Plot for Train Dataset for POST-COVID with Small Dataset (Test Case 3).

Fig. 15: Regression Plot for Test Dataset for POST-COVID with Small Dataset (Test Case 3).

Figs. 16 and 17 show the regression plot to display the network output with respect to target for training and test dataset for PRE-COVID with a Small Dataset (Test Case 4).

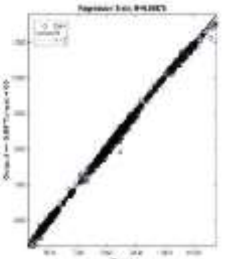

Fig. 16: Regression Plot for Train Dataset for PRE-COVID with Small Dataset (Test Case 4).

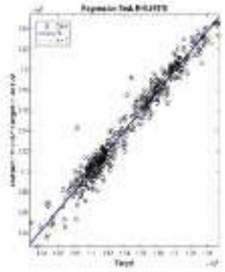

Fig. 17: Regression Plot for Test Dataset for PRE-COVID with Small Dataset (Test Case 4).

Conclusion

This study proposed a recurrent neural network model to predict the one-step-ahead closing index of the Bombay Stock Exchange. This research aims to test and analyze how the impact of the COVID virus is reflected in the stock market. In this study, four test cases were conducted on a BSE stock index dataset, namely POST-COVID with a large dataset, PRE-COVID with a large dataset, POST-COVID with a small dataset, and PRE-COVID with a small dataset. In this experiment, for every test case, 150 candidate models were generated; a total of 600 models were generated, and the best models were selected and reported for every dataset. The performance of the RNN was evaluated using the statistical performance measures SMAPE, RMSE, R, and R-Square in the training, validation, and test sets with varying dataset ranges; the results were also evaluated using a line chart and a regression plot. Test case 1 produces 99.08 percent accuracy. Test case 2 produces 99.35 percent accuracy. Test case 3 produces 99.25 percent accuracy. Test case 4 produces 99.32 percent accuracy. The results of the RNN model perform well even though there is a fluctuation in stock market data.

Reference

1. WHO "Online" https://www.who.int/news-room/q-a-detail/q-a-coronaviruses (Accessed on September-28, 2022)
2. D. Ashok kumar and S. Murugan, “'Performance Analysis of Indian Stock Market Index using Neural Network Time series Models”, IEEE Explore Digital Library, pp.72-78, 2013.
3. D. Ashok kumar and S. Murugan, “'Performance Analysis of MLPFF Neural Network Back propagation Training Algorithms for Time Series Data'”, IEEE Explore Digital Library, pp.114-119, 2014.
4. D. Ashok kumar and S. Murugan, “Performance Analysis of Different Data Division Ratio for Time Series Data Using Multi Layer Feed Forward Neural Network with Tracking Signal Approach”, , International Journal of Control Theory and Applications, Vol 10, Number 11, 2017.
5. D. Ashok kumar and S. Murugan, “A Novel Fuzzy Time Series Model for Stock Market Index Analysis using Neural Network with Tracking Signal Approach”, Indian Journal of Science and Technology Vol 10 (16), April 2017.
6. D. Ashok kumar and S. Murugan, “Performance Analysis of NARX Neural Network Back Propagation Algorithm by Various Training Functions for Time Series Data”, International Journal of Data Science, Vol 3 (4), November 20

7. Greg Heath, “Problem about getting optimum output in Neural Network MATLAB 2012a”, http://in.mathworks.com/matlab central/ newsreader/view thread/331714
8. Jeff Heaton , “Introduction to Neural Networks for Java”, Heaton Research, Inc, Second Edition, pp. 159-160, 2008.
9. A. Adebiyi Ayodele, K. Ayo Charles, O. Adebiyi Marion and O. Otokiti Sunday, “Stock Price Prediction using Neural Network with Hybridized Market Indicators”, Journal of Emerging Trends in Computing and Information Sciences, Vol. 3, pp. 1-9, 2012.
10. Ricardo de A.Araujo, and Tiago A.E. Ferreira, “An intelligent hybrid morphological-rank-linear method for financial time series prediction”, Neurocomputing, Vol. 72, pp. 2507-2524, 2009.
11. T. Victor Devadoss, and T. Antony AlphonnseLigori, “Stock Prediction Using Artificial Neural Networks”, International Journal of Data Mining Techniques and Applications, Vol. 2, pp. 283-291, 2013.
12. T. Victor Devadoss, and T. Antony Alphonnse Ligori, “Forecasting of Stock Prices Using Multi Layer Perceptron”, International Journal of Computing Algorithm, Vol. 2, pp. 440-449, 2013.
13. Zhang Yudong, and Wu Lenan, “'Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network”, Expert Systems with Applications, Vol. 36, pp. 8849-8854, 2009.
14. R. Adhikari, and R. K. Agrawal, “An introductory study on time series modeling and forecasting”, LAP Lambert Academic Publishing, Germany, pp. 1-67, 2007.
15. K. K. Suresh Kumar, “Performance analysis of stock price prediction using artificial neural network”, Global Journal of Computer Science and Technology, pp. 154-170, 2012.
16. Derrick T. Mirikitani, and Nikolay Nikolaev, “Recursive Bayesian Recurrent Neural Networks for Time-Series Modeling”, IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 21, NO. 2, FEBRUARY 2010.