A Relative Study Of Arima With Cdrann For Finding Best Future Predictor

G. Sundar[1], P. Patchaiammal[2], Dr. K Satyanarayana[3]

[1]Research Scholar, Bharath University, Chennai sundarganapathy66@gmail.com
[2]Research Scholar, Bharath University, Chennai sarandsk1@gmail.com
[3]Principal, Sindhi College, Chennai

Article History Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

Abstract: Prediction analysis is used for gaining the optimal value in real world problems. Artificial Neural Network with deep learning forms a deep neural network. It will help to create a novel future prediction for real world problem. For time-series problem like stock market prediction, there is a need for best predictor for future price. To achieve this, a relative analysis is must. This research work helps to identify the best future predictor for stock by relating the proposed model CDRANN with the standard ARIMA model. The performance result shows that for time series dataset CDRANN model was able to perform better than the ARIMA method.

Key words: ARIMA, CDRANN, AI, DL, Time Series, Future Prediction, Error Rate

1. INTRODUCTION

1.1 Artificial Intelligence

Artificial intelligence is the science and engineering of intelligence systems, which simulates human intelligence in learning information to use it. Artificial Intelligence means the ability of computers to think and act like human so as to perform as extremely intelligent manner. It has created a huge impact on human interaction with system. It has enhanced the experience in a big way. AI collects real time data and processes the data so that the system learns by their own. AI lets the system gain information and rules. Based on that, patterns are formed. By using these previous patterns, AI helps the system to predict the value in future. It is used to maximize the chances of success of intelligence system. These systems are designed so as to save the time in data handling. Data science is the major process of handling data for future benefits. AI is the major perspective to manage the data for problem solving techniques. AI helps to integrate various activities to support decision making in future research communities. It is good for real time accurate reporting and processing of large volume of historical data to make decisions. By using AI, patterns are formed with historical data. By understanding these patterns, knowledge set is formed so that better decisions related to the dataset are taken. This helps to the implementation of proper prediction techniques. AI is designed and trained for specific type of task. It is because AI has programming techniques, which classify the things accordingly. AI acts as human brain because it has a cognitive ability, which helps to perform unfamiliar task. To find the problem solution, AI works beyond an algorithm.

With the predictive analysis by AI, the stock price can be predicted. The application derived using AI is able to predict the price patterns and alert the investor when you buy the stocks. So, the reduced rate can be known before the investors buy the stock.

The price trend is analysed on the basis of the historical data recoded on each stock. So, the investor can get the notification of when to buy a stock. Buying stock at the right time at the right price is possible with the help of AI techniques.

In general, AI is a concept of intelligence to describe the ability to distinguish information and to maintain knowledge base. This knowledge is applied in accordance with the adaptive behaviour within an environment.

1.2 Deep Learning

The deep learning system performs the task repeatedly to gain experience and to improve the outcome. This system has various layers in deep to enable and gain the learning knowledge. For any given problem, the "idea" is required to solve it as deep learning.

Deep learning needs a lot of data for learning. Deep learning allows solving complex data structures like unstructured and interconnected. If the Deep learning algorithm learns more data, then its performance will be better. Deep Learning makes one understand patterns so as to give the ability of the next set of sequence value prediction. Deep Learning Neural Networks are characterized by making the layers deep. This is useful to solve vanishing gradient problems.

1.3 Stock Market

Stock market is an accumulation of buyers and sellers of shares. The buyer is known as an investor and the seller is known as a trader. Every company has a certain quantity of shares listed for public for trading and it is listed out in Stock Exchanges. The stock price keeps on fluctuating on the basis of company value. The fluctuation of stock price needs to be analysed to predict the prices of stock. To gain maximum profit in share the investor or trader must know the right time for the investment. Even though many techniques are available to predict the stock price, the stock prediction is quite difficult.

2. TIME SERIES PREDICTION

The dataset measured and formed in the base of consistent time intervals are known as time series dataset. Stock market dataset is one of the time series dataset. These time series are used for predicting the future values by using past historical dataset. It is available in various formats like hour, day, week, month, and year. In this research work the long-term dataset of HDFC Bank Life Insurance is used. Prediction by time series is done by using the historical past data observations.

2.1 AI in Prediction

Artificial Neural Network works in non-linear transformation of its inputs from one form to another. This non-linear transformation is used to identify the input so as to project complex decision boundaries in order to perform the higher dimensionality as linearly separable. Learning by Artificial Intelligence means one has to teach the computer to learn from the input data, and process it for the output decision. The component known as ANN is used for learning functions in Artificial Intelligence. ANN has only three layers like input, hidden and output layers. By adding more hidden layers, we form a multi-layer neural network.

2.2 DL in Prediction

Deep learning is good to solve problems with no labelled data. Deep learning is a process of ML algorithm, used to employ deep neural network with multiple hidden layers to solve a particular problem. Deep learning model helps to perform more non-linear transformation of the input with more complex decision boundaries. Adding Deep Learning to Recurrent ANN back track solver makes the future stock price prediction more accurate, but in deep learning, the computer trains itself to learn and process the data for output. A deep learning system takes these multiple hidden layers from the base to form a deep neural network. This deep learning system forms a self-learning system of the human brain like structure.

In Deep Learning, RNN, LSTM, and MLP plays a major role. The combination of these algorithms creates a major change in prediction. ARIMA is one of the time-series modelling, which is used for improving the predictions. It is a combination of multiple linear regressions and MLP (Multi-Layer Perceptron). The proposed CDRANN model is a combination of RANN Back track Solver with LSTM. This research work helps to identify the best Deep Learning technique for time series dataset.

3. PROPOSED CDRANN MODEL

CDRANN means Comprehensive Deep Recurrent Artificial Neural Networks with nonlinearity in nature. It is the comprehensive evolutionary model based on Recurrent Neural Networks (RNNs) with Back Track Solver in corporation with LSTM Deep Learning Structure, which has been widely adopted in research areas concerned with sequential dataset. RNN only consists of sigma cells or tanh cells. So it is unable to learn the relevant information of input data when the input gap is large. By introducing gate functions into the cell structure, the long short-term memory (LSTM) could handle the problem of long-term dependencies very well.

CDRANN is mainly divided into three layers like forget, input, and output layers. The input layer is also known as memory gate layer. In this, historical dataset is combined with the previous layer output and it is dense through tanh activation function. Compared with multiple activation functions linear, relu, tanh and sigmoid, tanh performed well in CDRANN input structure. CDRANN works from left side to right side with forget gate, input gate, and output gate. In the forget gate, the learning historical and previous data started and the value to be retained/remembered or forgotten is decided by using sigmoid function. Then the retained value is combined with the current input value so as to solve gradient vanishing problem.

CDRANN forget gate takes RNN Backtrack Solver input along with LSTM previous stage output to decide the elimination of unnecessary data in historical study. It releases the number between 0 and 1, 0 means the values are ignored and 1 means the values are saved and retained. This is stored and passed to the input gate. In the memory input gate, Sigmoid and tanh activation functions are combined with the connected input value to form a vector for new data value. Here, the memory stage is updated newly formed vectors. The sigmoid function act as door for input to choose changed values. The tanh function used to make the vector as of new candidate value. The vector formed in the memory gate moved to the output gate. The output gate will decide the resultant value from the vectors. The final result is decided by sigmoid function along with tanh in output gate. Now, the final filtered improved predicted value is received from CDRANN. This filtering structure makes CDRANN model as fine-tuning structure in compared with other deep learning models.

4. ARIMA MODEL

ARIMA produces better prediction for short term dataset modelling. It is best for univariate ARIMA model. It is a generalized form of 'AR' integrated with 'MA'. It means AR + I + MA. In this 'AR' means Auto-Regressive. 'I' means Integrated. 'MA' means Moving Average. By combining all these ARIMA means Auto-Regressive Integrated Moving Average. It is based on linear regression equation. ARIMA is a forecasting model, which is best fit for short-term modelling time-series dataset.

In ARIMA model 'AR' used for find the auto regression with current and trailed observations. 'I' used to make the time series as a fixed value. This is done by measuring the observation difference at various time intervals. 'MA' used to measure error term in relation of observed and residual values in trail.

The ARIMA value is categorized as three terms p, d, and q. In which 'p' is the order of AR, which is the number of lagged observations, 'd' is the number of differences between past and current time values, which is the degree of difference to make stationary time series and 'q' is the error order of MA term, which is the moving average window.

The value of 'p' refers to the number of lags to be used as predicted, 'q' is the number of lagged forecast error. The general form of ARIMA (p, q, and d) which means that p is the number of lagged observations in training model, d is the number of timing difference, and q is the size of moving average window. If we take ARIMA (3, 1, 0) means that the Auto-regression is 3, l the difference degree is 1 and the moving average window is 0. By increasing the 'p' value term means increase the dataset. For increased dataset ARIMA will produce poor results.

5. IMPLEMENTATION

5.1 Data Preparation

In this section HDFC Life Insurance Data set of BSE market is collected by using the Keras command gh(symbol='HDFCBANK',start=start,end=end) in python library nsepy. The result is shown in the below diagram 1.

	Open	High	Low	Close	VWAP	Volume	Turnover	Trades	Deliverable Volume	Date
Date										
2015-01-01	951.00	954.40	945.05	952.05	949.13	886235	8.411519e+13	12603	468601	2015-01-01
2015-01-02	950.40	969.30	950.40	965.30	965.25	1475096	1.423838e+14	23437	774971	2015-01-02
2015-01-05	970.00	970.55	955.10	957.15	959.15	1199000	1.150019e+14	25544	716913	2015-01-05
2015-01-06	954.00	956.55	937.55	942.25	948.51	2054920	1.949110e+14	30301	1410336	2015-01-06
2015-01-07	939.70	951.35	936.25	945.00	946.73	1436528	1.360005e+14	34047	703837	2015-01-07
...	...	...	...	...	...	...	...	...	...	...
2021-02-22	1545.05	1573.90	1539.45	1548.00	1557.44	14725919	2.293473e+15	299769	7734714	2021-02-22
2021-02-23	1553.75	1557.70	1522.65	1529.15	1537.17	9119953	1.401896e+15	230064	3676937	2021-02-23
2021-02-24	1526.50	1613.95	1516.25	1606.45	1561.95	7157166	1.117910e+15	144539	3318706	2021-02-24
2021-02-25	1609.75	1636.25	1602.00	1606.40	1620.67	10054785	1.629546e+15	231264	4200723	2021-02-25
2021-02-26	1587.05	1588.90	1521.00	1534.40	1543.86	13956423	2.154672e+15	397166	5926438	2021-02-26

1526 rows × 10 columns

Diagram 1 Data report of HDFC Life Insurance

5.2 ARIMA Execution

The ARIMA model is executed by using the available built-in function defined known as auto_arima with the help of built-in library pmdarima.

auto_arima(data2['Close'], start_p =1, start_q = 1, max_p = 3, max_q = 3, m = 12, d = 1, trace = True, error_action ='ignore', suppress_warnings = True, stepwise = True)

The ARIMA model is implemented and the result of the performance is shown in the diagram 2. The ARIMA model in the result takes the p value as 1 to 3, d value as 1 and the q value as from 1 to 3. The HDFC Life Insurance dataset is considered for the ARIMA model execution.

```
Performing stepwise search to minimize aic
 ARIMA(1,0,1)(0,1,1)[12] intercept   : AIC=inf, Time=20.38 sec
 ARIMA(0,0,0)(0,1,0)[12] intercept   : AIC=18562.270, Time=2.48 sec
 ARIMA(1,0,0)(1,1,0)[12] intercept   : AIC=15519.806, Time=6.19 sec
 ARIMA(0,0,1)(0,1,1)[12] intercept   : AIC=17278.679, Time=6.07 sec
 ARIMA(0,0,0)(0,1,0)[12]             : AIC=18563.130, Time=1.39 sec
 ARIMA(1,0,0)(0,1,0)[12] intercept   : AIC=15980.181, Time=1.15 sec
 ARIMA(1,0,0)(2,1,0)[12] intercept   : AIC=15381.809, Time=21.90 sec
 ARIMA(1,0,0)(2,1,1)[12] intercept   : AIC=inf, Time=17.28 sec
 ARIMA(1,0,0)(1,1,1)[12] intercept   : AIC=inf, Time=7.94 sec
 ARIMA(0,0,0)(2,1,0)[12] intercept   : AIC=18565.281, Time=3.80 sec
 ARIMA(2,0,0)(2,1,0)[12] intercept   : AIC=15383.533, Time=27.24 sec
 ARIMA(1,0,1)(2,1,0)[12] intercept   : AIC=15383.519, Time=27.41 sec
 ARIMA(0,0,1)(2,1,0)[12] intercept   : AIC=17279.079, Time=14.00 sec
 ARIMA(2,0,1)(2,1,0)[12] intercept   : AIC=inf, Time=38.99 sec
 ARIMA(1,0,0)(2,1,0)[12]             : AIC=15379.935, Time=7.14 sec
 ARIMA(1,0,0)(1,1,0)[12]             : AIC=15517.925, Time=3.44 sec
 ARIMA(1,0,0)(2,1,1)[12]             : AIC=inf, Time=10.35 sec
 ARIMA(1,0,0)(1,1,1)[12]             : AIC=inf, Time=4.15 sec
 ARIMA(0,0,0)(2,1,0)[12]             : AIC=18566.138, Time=1.93 sec
 ARIMA(2,0,0)(2,1,0)[12]             : AIC=15381.655, Time=16.25 sec
 ARIMA(1,0,1)(2,1,0)[12]             : AIC=15381.641, Time=42.25 sec
 ARIMA(0,0,1)(2,1,0)[12]             : AIC=17279.641, Time=170.23 sec
 ARIMA(2,0,1)(2,1,0)[12]             : AIC=inf, Time=95.81 sec

Best model:  ARIMA(1,0,0)(2,1,0)[12]
Total fit time: 589.026 seconds
==============================================================================
Dep. Variable:                      y   No. Observations:                 1526
Model:        SARIMAX(1, 0, 0)x(2, 1, 0, 12)   Log Likelihood            -7685.967
Date:                Tue, 20 Apr 2021   AIC                          15379.935
Time:                        21:40:28   BIC                          15401.225
Sample:                             0   HQIC                         15387.862
                               - 1526
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.9602      0.003    290.594      0.000       0.954       0.967
ar.S.L12      -0.6827      0.004   -193.069      0.000      -0.690      -0.676
ar.S.L24      -0.3006      0.004    -77.889      0.000      -0.308      -0.293
sigma2      1495.1030      3.870    386.340      0.000    1487.518    1502.688
==============================================================================
Ljung-Box (L1) (Q):              0.23   Jarque-Bera (JB):          9978007.80
Prob(Q):                         0.63   Prob(JB):                        0.00
Heteroskedasticity (H):         21.76   Skew:                          -12.39
Prob(H) (two-sided):             0.00   Kurtosis:                      399.93
==============================================================================
```

Diagram 2 Execution result of ARIMA model for HDFC Life Insurance Dataset

The Diagram 3 shows the performance history of HDFC Life Insurance Data with original_close_price and predicted_close_price values in graph by using ARIMA model. The blue line in the graph shows original value in the data set and the red line in the graph shows the predicted value.

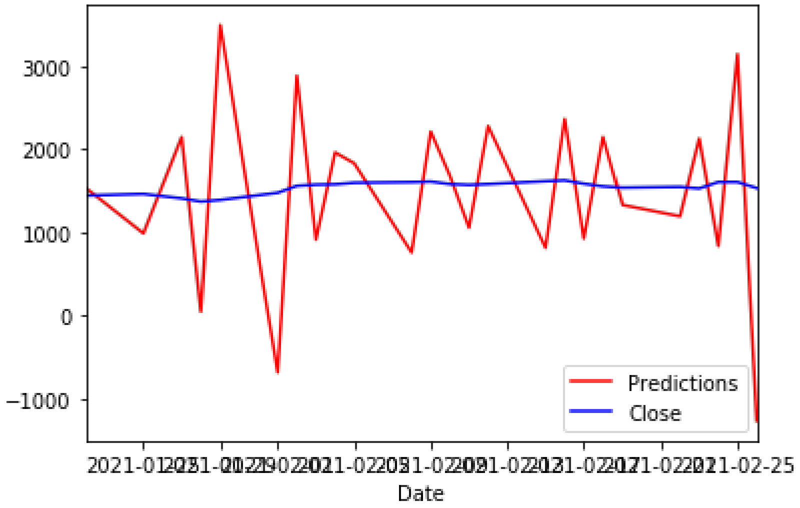

Diagram 3 Performance graph for ARIMA dataset

From the performance graph it is clear that there is a major difference between the original and the predicted values. Only few values are matched. This is because of consideration of long term historical dataset like HDFC Life Insurance Dataset.

The error value by using mean_squared_error built in function is shown below. The error rate for ARIMA model is 10.8%. It is shown in the below executed result diagram 4.

```
print(np.sqrt(metrics.mean_squared_error(test['Close'],predictions))/100)
```

```
10.805768927800475
```

Diagram 4 Mean_squared_error of ARIMA

5.3 CDRANN Execution

The CDRANN model is executed by using the available built in function defined known as LSTM with the help of built in library tensorflow.keras.layers.

The CDRANN model is implemented and the result of the performance is shown in the diagram 5. The CDRANN model with the epochs of 15 execution is shown for the HDFC Life Insurance dataset is considered for the CDRANN model execution.

```
Epoch 1/15
45/45 [==============================] - 5s 120ms/step - loss: 0.0177
Epoch 2/15
45/45 [==============================] - 5s 121ms/step - loss: 0.0078
Epoch 3/15
45/45 [==============================] - 5s 120ms/step - loss: 0.0065
Epoch 4/15
45/45 [==============================] - 5s 118ms/step - loss: 0.0063
Epoch 5/15
45/45 [==============================] - 5s 118ms/step - loss: 0.0051
Epoch 6/15
45/45 [==============================] - 6s 122ms/step - loss: 0.0055
Epoch 7/15
45/45 [==============================] - 5s 118ms/step - loss: 0.0047
Epoch 8/15
45/45 [==============================] - 5s 120ms/step - loss: 0.0036
Epoch 9/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0040
Epoch 10/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0038
Epoch 11/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0034
Epoch 12/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0032
Epoch 13/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0033
Epoch 14/15
45/45 [==============================] - 5s 119ms/step - loss: 0.0034
Epoch 15/15
45/45 [==============================] - 5s 120ms/step - loss: 0.0031
```

Diagram 5 Execution result of CDRANN model for HDFC Life Insurance Dataset

The Diagram 6 shows the performance history of HDFC Life Insurance Data with original_close_price and predicted_close_price values in graph by using CDRANN model. The blue line in the graph shows original value in the data set and the red line in the graph shows the predicted value.

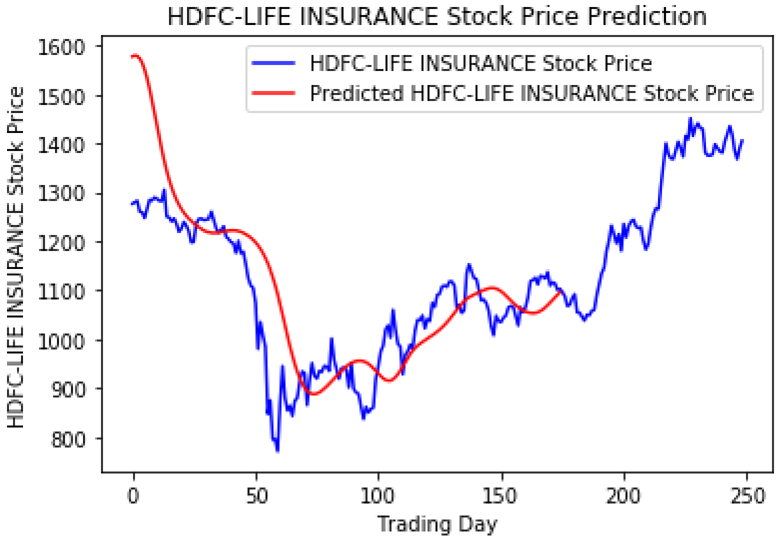

Diagram 6 Performance graph for ARIMA dataset

From the performance graph, it is clear that there is a minor difference between the original and the predicted values. Only few values are not matched. This is because of consideration of long-term historical dataset like HDFC Life Insurance Dataset.

The error value by using mean_squared_error built in function is shown below. The error rate for CDRANN model is 3.8%. It is shown in the below executed result diagram 7.

```
lstm_error = sum(forecast_errorss) * 1.0/len(real_stock_price)
print(sqrt(abs(lstm_error)))
```

```
3.847553634813434
```

Diagram 7 Mean_squared_error of ARIMA

5.4 Error rate comparison

In the section the relativity of error rate between the models ARIMA and CDRANN with the dataset is compared by using bar graph. The diagram 8 shows the final comparison of the models ARIMA error rate with CDRANN error rate by means of mean_squared_error. The bar graph clearly shows that CDRANN out performs ARIMA for long-term prediction. The error rate of ARIMA is 10.8% and of CDRANN is 3.8% which is very less.

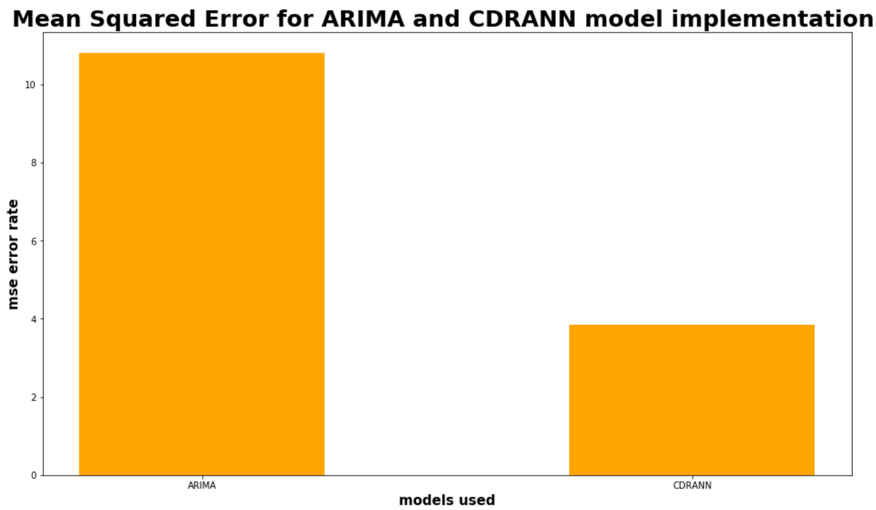

Diagram 8 Relative error rate analysis of ARIMA Vs CDRANN

6. CONCLUSION

In the large dataset, if any gap identified means deep learning models like RNN, ARIMA are not able to learn and predict exactly. ARIMA model is only best for short term dataset. If the dataset used is large then its error rate is increased. In this research work, the relativity of ARIMA model with LSTM based CDRANN model is implemented and the performance results are compared. The final error rate shows that CDRANN model out performs in long term time series data prediction with minimum error rate.

7. REFERENCES

1. G.Sundar, K.Satyanarayana “Stock Prediction Scrutiny using Artificial Neural Network” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7 Issue-6S3 April, 2019 Published By: Blue Eyes Intelligence Engineering & Sciences Publication Retrieval Number: F1021376S19/19©BEIESP
2. Rajesh, M. "A Review on Excellence Analysis of Relationship Spur Advance in Wireless Ad Hoc Networks." International Journal of Pure and Applied Mathematics 118.9 (2018): 407-412.
3. G.Sundar, K.Satyanarayana “Multi Layer Feed Forward Neural Network Knowledge Base to Future Stock Market Prediction” Published By: Blue Eyes Intelligence Engineering & Sciences Publication Retrieval Number: K121809811S19/2019©BEIESP DOI: 10.35940/ijitee. K1218.09811S19
4. Capability of Artificial Neural Network for ForwardConversion of Geodetic Coordinates (φ, λ, h) to Cartesian Coordinates (X, Y, Z), Yao YevenyoZiggah, • Hu Youjian, Xianyu Yu, • Laari Prosper Basommi, International Association for Mathematical Geosciences 2016
5. https://bdtechtalks.com/2020/06/08/what-is-recurrent-neural-network-rnn/
6. G.Sundar, K.Satyanarayana “Performance Analysis of Stock Prediction by PLANN Techniques” January - February 2020 ISSN: 0193 - 4120 Page No. 6303 – 6313 Published by: The Mattingley Publishing Co., Inc.
7. https://www.analyticsvidhya.com/blog/2017/12/introduction-to-recurrent-neural-networks/
8. http://neuralnetworksanddeeplearning.com/chap2.html
9. http://www.codeproject.com/Articles/175777/Financial-predictor-via-neural-network
10. https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/
11. http://www.ijcaonline.org/volume22/number2/pxc3873497.pdf
12. http://www.codeproject.com/Articles/175777/Financial-predictor-via-neural-network
13. http://www.ijcaonline.org/volume22/number2/pxc3873497.pdf
14. http://journal.seo.ir/page/download-GLgsPnnRG_A.artdl
15. G.Sundar, K.Satyanarayana “Improvement of Stock Price Prediction by Synthesizing Recurrent ANN Back Track Solver” International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 01, 2020 ISSN: 1475-7192 DOI: 10.37200/IJPR/V24I1/PR200297

.http://www.researchgate.net/publication/228607056_Using_neural_networks_to_forecast_stock_market_prices.
https://www.machinelearningplus.com/time-series/ARIMA

https://people.duke.edu/~rnau/411arim.htm

http://www.researchgate.net/publication/228607056_Using_neural_networks_to_forecast_stock_market_prices

https://www.tutorialspoint.com/time_series/time_series_lstm_model.htm