Hybrid ARIMA-Deep Belief Network model using PSO for Stock Price Prediction

Shaikh Sahil Ahmed[1], Mahesh Kankar[1], Biju R. Mohan[1] and Nagaraj Naik[1]

National Institute of Technology, Karnataka
sahilahmed786001@gmail.com mahesh15kankar@gmail.com

Abstract. Forecast analysis is in very high demand in many fields for improving sales and operation planning in various industries and enterprises. So, accuracy is a major factor in forecasting stock market prices. We already know there are existing deep learning models for stock market prediction such as Gated Recurrent Unit (GRU), Support Vector Machine (SVM), Multilayer Perceptron (MLP), etc. In this paper, we enhanced the prediction of stock prices using series hybrid models over single deep learning models. The models we used are Autoregressive Integrated Moving Average (ARIMA), Deep Belief Network (DBN), Long Short-Term Memory (LSTM) and performed analysis on hybrid models in comparison with single models. We have chosen a single model as ARIMA, LSTM, and hybrid as ARIMA-DBN and ARIMA-LSTM. For finding the best fit parameter for ARIMA and DBN, Particle swarm optimization (PSO) technique is used. We compared the various models based on performance errors like MSE, RMSE, MAPE, etc. As already existing ARIMA and LSTM is not good enough for forecasting and so we worked over ARIMA-DBN model to overcome the limitations of other models. After research, we found out that series hybrid ARIMA-DBN is effectively better as compared to other single models for stock market prediction.

Keywords: Deep Learning, Time Series forecasting, ARIMA, Linear and non-linear models, Particle Swarm Optimization.

1 Introduction

In decision making, mostly the factors are based on accuracy and therefore, improving the accuracy plays a major role for decision-makers. From many people, they find better combining various models or using hybrid models, it will give enhanced forecasting accuracy as compared to single models. In other words, we can say that single models may not be able to take out all the features having in time series. While using the series hybrid model, it consists of two major parts. In the first phase, the first model is owned to solve one of the components of the time series model and in the second phase, the other model is owned to solve the other component present in the time series model which was unable to extract by the first model.

There may be various components in a time series where it may not be answerable by single deep learning or other models. Most of the time, a time series contains two components, one is a linear component and the other is a non-linear component. So, one with a hybrid model can solve the linear pattern using the first model and the non-linear pattern with the other model.

The main motive of the paper is to identify how much, the hybrid models are able to evaluate in comparison with single models in terms of various performance errors.

2 Literature Review

Biju R Mohan, G Ram Mohana Reddy [1] described the estimation of resource exhaustion in the server virtualization system using a hybrid ARIMA-ANN model. As the ARIMA model alone cannot extract all the features i.e non-linearity pattern present in residuals which will be tackled by the Artificial Neural Network (ANN) model further. Finally, their accuracy shows five times improvement for forecasting resource usage.

Mehdi Khashei, Zahra Hajirahimi [2] described the series hybrid ARIMA-MLP model for solving the linearity and non-linearity pattern present in the time series model. They have chosen three datasets and performed the evaluation over various error performance analysis finding Mean Squared Error, Mean Absolute Error, Root Mean Squared Error and Mean Absolute Percentage Error.

Zhihang Li, Mengjiao Qin, Zhenhong Du [3] described the red tide analysis forecasting based on a series hybrid ARIMA-DBN model that suggests for the red tide prediction before occurrence. They have chosen various environmental factors such as pH, salinity, nitrite, ammonia, silicate, chlorophyll, etc. Based upon these factors they overcome the forecasting accuracy with better results.

Iyan E. Mulia, Harold Tay K., Roopsekhar Pavel Tkalich [4] described the hybrid algorithm using ANN-GA (Genetic Algorithm) for forecasting up to 14 days ahead of the factors they have taken are turbidity and chlorophyll. With this model, they got good accuracy with high temporal variability.

Ina Khandelwal, Ratnadip Adhikari, Ghanshyam Verma [5] described DWT (Discrete Wavelet Transform) which was used to decompose the linearity and non-linearity pattern of the in-sample data for ARIMA and ANN respectively. They got better results using a hybrid ARIMA-ANN model as compared to individual ARIMA and ANN models.

3 Methodology

We observed our results based on four models, first is Autoregressive Integrated Moving Average (ARIMA), second is Long Short-Term Memory (LSTM), third is Autoregressive Integrated Moving - Deep Belief Network (ARIMA-DBN) and the fourth is ARIMA-LSTM.

3.1 Particle Swarm Optimization

It is one of the efficient optimization algorithms which is inspired by a group of birds with some pattern in the sky. PSO is initialized with particles having a starting position and velocity that will be used to calculate the fitness value for every particle. It finds the best fitness for every particle currently executing also said as pbest.

Another value is taken as the best value also called gbest. PSO is very much helpful in the hybrid ARIMA-DBN model for finding the p,d,q of ARIMA parameters in an optimized way. It is an optimization algorithm that is used to find an appropriate solution and speed up the training process.

3.2 Autoregressive Integrated Moving Average

ARIMA is a model for predicting values that described a given time series based on its past values, i.e with the help of own historical data(with own lag and lagged forecast errors). It is modeled using 3 factors i.e p, d and q. Here, p represents the lag order of Autoregressive (AR) term, d shows the lag order of Differencing and q is known for the lag order of Moving Average (MA). Here, the Autoregressive process is a sample of a stochastic process which is also called transition models or conditional models.

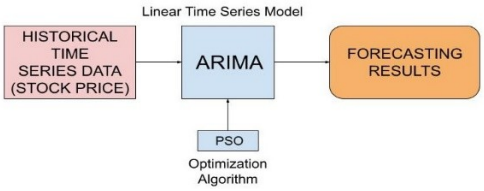

Fig. 1. Flow Diagram of ARIMA model.

It helps in predicting future behavior based on past behavior taking the lagged past p terms. Moving Average is said to be a time series model where the output of the model is linearly dependent on the present and the varying past values of an imperfectly predictable term. Both Autoregressive and Moving Average works on a stationary time series. The basic laws of stationarity of a time series state that the behavior of the process should not change over time. The flow diagram of the ARIMA model is described in Fig. 1.

As the ARIMA model works over stationary time series, the model has to be differenced based on d value and so, differencing a non-stationary time series model makes the time series data stationary. The number of iterations required to make non-stationary time series as stationary is denoted by d. So, the ARIMA model can be expressed as

$$\hat{x}_t = \alpha_0 + \alpha_1 x_{t-1} + \ldots + \alpha_p x_{t-p} - \theta_1 e_{t-1} - \ldots - \theta_q e_{t-q} \quad (1)$$

Here, $\hat{x}_t$ represents the actual value, e_t represent the error term at time t, α_i and θ_j are the coefficients. For finding the values of p, d, and q, we used Akaike Information Criterion (AIC) and for finding the minimum AIC value in an optimized way from various combinations of p, d, q values, we used particle swarm optimization (PSO) that results in the best fit for forecasting accuracy. PSO optimizes the parameters of ARIMA by iteratively executing the best fit function and gives the optimized values for p, d, and q.

3.3 Long Short Term Memory

It comes under the category of Recurrent Neural Networks (RNN). This algorithm helps to classify, cluster and can do predictions about the data that may be a time series data. LSTMs are efficient in capturing long term data and are well designed to keep away the problem called long-term dependency.
LSTM can be owned for univariate time series prediction problems such as stock market prediction. In general, LSTM is built up of a cell, input gate, output gate and the forget gate. The cell is used as a buffer which keeps remembers the values over an arbitrary period and remaining units (3 gates) control the flow of data in an out to the cell. The flow diagram of the LSTM is shown in Fig. 2. The following equation represents the LSTM gates

$$I_t = \sigma\left(W_i\left[H_{t-1}, X_t\right] + B_i\right) \tag{2}$$

$$f_t = \sigma\left(W_f\left[H_{t-1}, X_t\right] + B_f\right) \tag{3}$$

$$o_t = \sigma\left(W_o\left[H_{t-1}, X_t\right] + B_o\right) \tag{4}$$

Where I_t is the input gate, f_t is the forget gate, o_t is the output gate, σ is the sigmoid function, H_{t-1} is the previous LSTM block output at time t-1, X_t is the present input, W_x represents weights and B_x is the biases for gate(x).

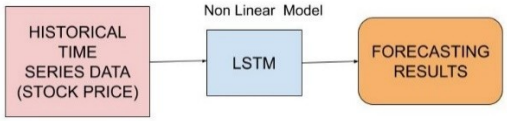

Fig. 2. Flow Diagram of LSTM model.

3.4 Deep Belief Network

DBN is a sort of deep neural network which is built up of multiple RBMs (Restricted Boltzmann Machines). Each RBM consists of 2 layers, one is the visible layer and the other one is the hidden layer. The visible layer of first RBM takes input from the time series data and the hidden layer is computed based on weights, biases, and input of the visible layer. The hidden layer works as a feature extractor. In the next RBM, the output of the first hidden layer is given as input to the visible layer and repeatedly this process is executed for all the RBMs present in the DBN network. The mathematical equations for DBN are as follows

$$P(m \vee n) = \frac{P(m,n)}{P(n)} \tag{5}$$

$$P(n \vee m) = \frac{P(n,m)}{P(m)} \tag{6}$$

$$P(m,n) = P(n,m) = \frac{1}{2} e^{-E(n,m)} \tag{7}$$

$$Z = \sum_{n,m} e^{-E(n,m)} \tag{8}$$

Where m and n are the hidden units and the visible unit respectively. Here, $P(m \vee n)$ is the conditional probability and similarly we can calculate $P(n \vee m)$. $P(m,n)$ or $P(n,m)$ is the joint distribution and $P(n)$, $P(m)$ are the marginal distribution. $E(n,m)$ is the joint configuration energy of the visible and the hidden units. There are are two steps to train the DBN model. Former one is the pre-training step and the later is fine-tuning. In pre-training, each RBM is trained and in fine-tuning using ANN the parameters of combined RBMs are adjusted.

3.5 ARIMA-DBN Hybrid model

There may be various components present in a time series such that its features are not extracted by the single models. In time series analysis there are two components present, one is the linear and second one is the non-linear component. If we use only a single model, it can predict the time series based on linear or maybe a non-linear component but there is hidden information in the form of non-linear components or linear components (output residuals of a single model). For modeling the linear pattern in the time series we use the ARIMA model and for modeling the non-linear pattern in the time series model, we used deep neural network model as Deep Belief Network.

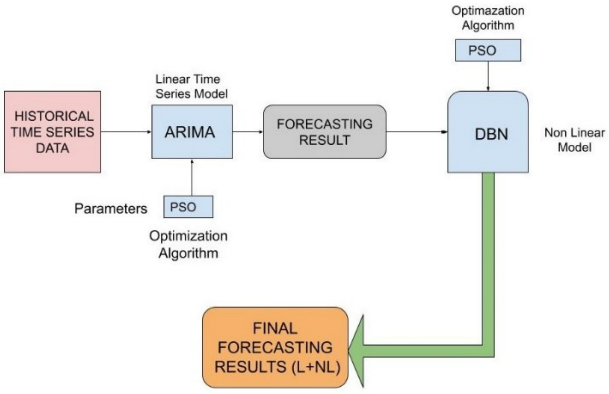

Fig. 3. Flow Diagram of ARIMA-DBN model.

So, here comes the series hybrid model where the time series is given input to the ARIMA model so it can handle the linear pattern of the time series and gives the non-linear residual output which is given further input to Deep Belief Network. So, this an advantage to forecasting accuracy as both the linear and non-linear components are handled by the hybrid model which was a disadvantage to the single models. So, on combining both the modelâ€™s prediction, the series hybrid model accuracy of the time series prediction is improved. Flow diagram of series hybrid ARIMA-DBN model is shown in Fig. 3. Here also for searching the best-fit parameters of the ARIMA model, we have used the PSO technique in which it helps in finding the minimum AIC value in an optimized way from different combinations of p, d and q values.

We computed the number of hidden units present in each RBM using particle swarm optimization to speed up the training process. The objective function for PSO is to minimize the mean squared error to give better accuracy.

3.6 ARIMA-LSTM Hybrid model

In this hybrid series model, ARIMA is used to solve the linear pattern present in the time series and LSTM is used to solve the non-linear pattern. So, time series is given input to the ARIMA model which can handle the linear component of the time series and gives the non-linear residual output which is given further input to LSTM. Now, LSTM handles the non-linear pattern and the output prediction is merged with the estimations of the ARIMA model to give the final output of the series hybrid ARIMA-LSTM model. The flow diagram of the series hybrid ARIMA-LSTM model is shown in Fig. 4.

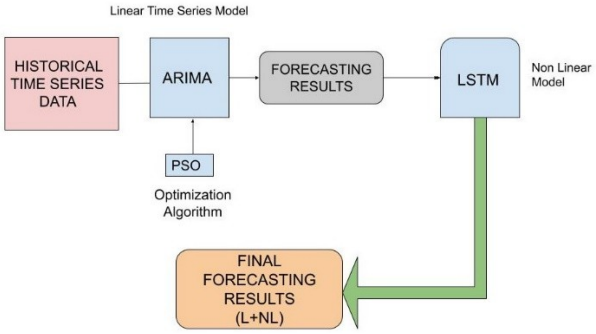

Fig. 4. Flow Diagram of ARIMA-LSTM model.

4 Experiment Analysis and Results

Datasets

For analysis of the models, two Indian benchmark datasets are taken using the opening price of Infosys Limited (INFY.NS) and the opening price of Mahindra &

Mahindra Limited (M&M.NS). The information about the datasets, the procedure of models is explained in the next subsections.

4.1 Infosys Limited (INFY.NS)

Infosys Limited (INFY.NS) dataset contains the stock opening prices from 8th March 2010 to 6th March 2020 and has a total of 2473 day values, i.e. data values are collected each day. We have used 80% of the dataset for training purposes and the rest for testing purposes. The plot of Infosys Limited (INFY.NS) dataset is shown in Fig. 5.

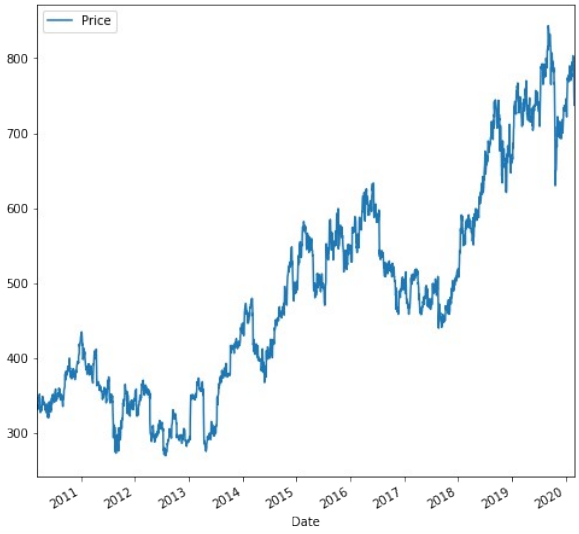

Fig. 5. Infosys Limited (INFY.NS) plot.

The Arima Model. With the help of Particle Swarm Optimization, we optimized the parameters of the ARIMA model to get the minimum AIC and found the best-fitted model as ARIMA(2,1,0). The predicted test values of the input time series for the ARIMA model is visualized in Fig. 6.

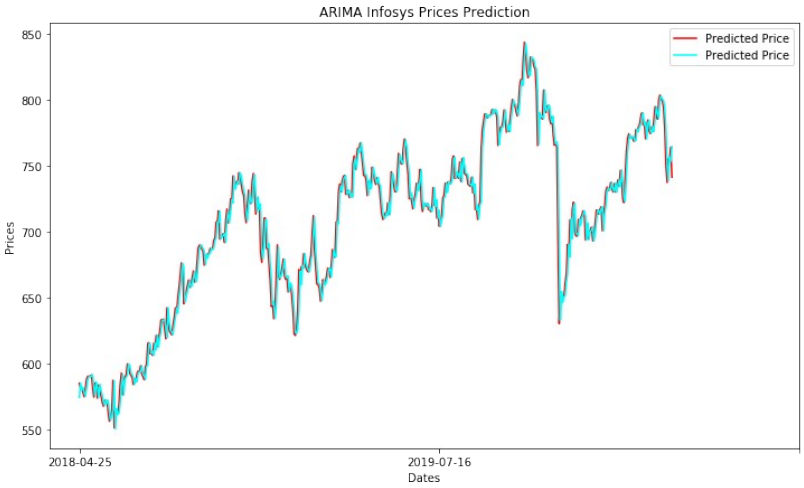

Fig. 6. Estimated values of ARIMA model for Infosys Limited (INFY.NS) (Test Dataset).

The LSTM Model The input layer is specified by Input shape, with one hidden LSTM layer and one Dense output layer. We defined an LSTM model in which an input layer expects one or more values and window size is taken as 50. The predicted test values of the input time series by the LSTM model is shown in Fig. 7.

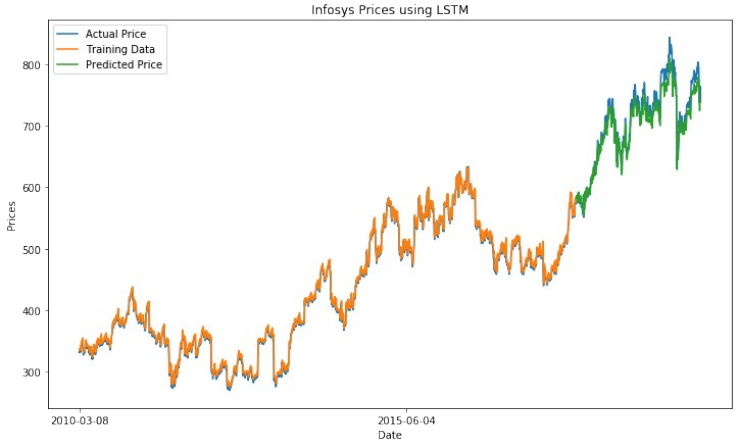

Fig. 7. Estimated values of LSTM model for Infosys Limited (INFY.NS) (Test Dataset).

The Series Hybrid ARIMA-LSTM model The parameters of the ARIMA model are optimized by Particle Swarm Optimization having minimized AIC value and gives the best fit as ARIMA(2,1,0). The predicted output of the ARIMA model is shown in Fig. 6. The residuals of the ARIMA model is now given input to LSTM. The predicted test values of the input time series by series hybrid ARIMA-LSTM model is shown in Fig. 8.

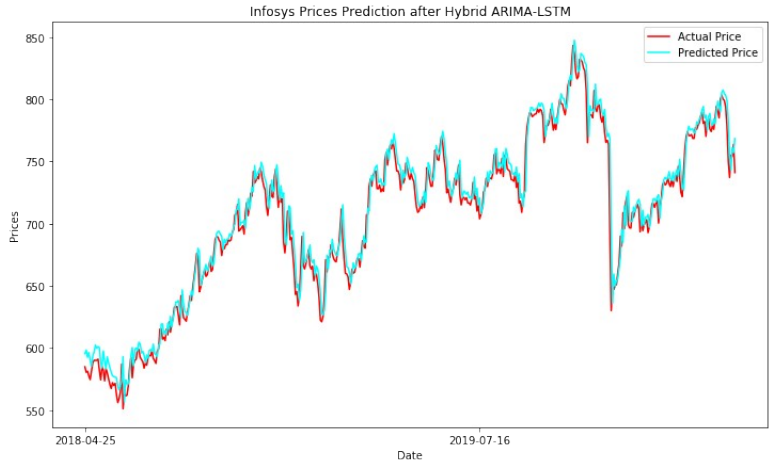

Fig. 8. Estimated values of hybrid ARIMA-LSTM model for Infosys Limited (INFY.NS) (Test Dataset).

The Series Hybrid ARIMA-DBN Model On optimizing the ARIMA parameters using PSO, the best fit model is ARIMA(2,1,0) with minimized AIC. The predicted output of the ARIMA model is shown in Fig. 6 and is now given input to DBN. Using PSO we found 512 hidden units present in one RBM layer. The predicted test values of the input time series by series hybrid ARIMA-DBN model are shown in Fig. 9.

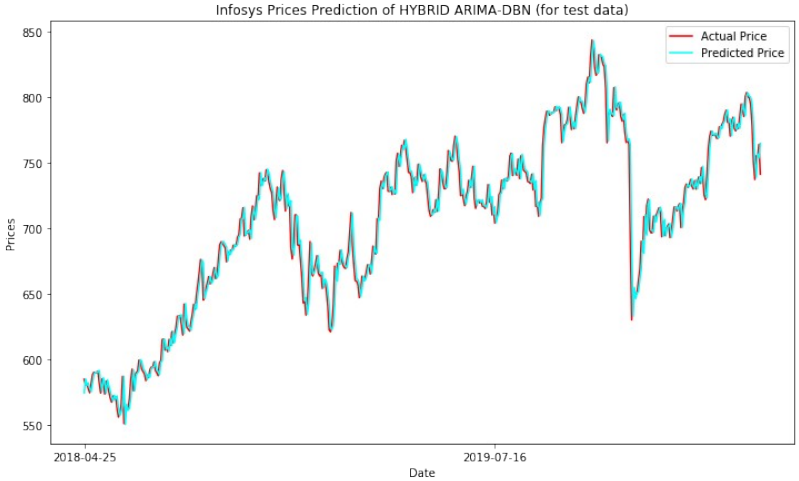

Fig. 9. Estimated values of hybrid ARIMA-DBN model for Infosys Limited (INFY.NS) (Test Dataset).

4.2 Mahindra & Mahindra Limited (M&M.NS)

M&M.NS dataset contains the stock opening prices from 8th March 2010 to 6th March 2020 and has a total of 2473 day values, i.e data values are collected on each

day. The plot of the M&M.NS dataset is shown in Fig. 10. We have used 80% of the dataset for training purposes and the rest for testing purposes.

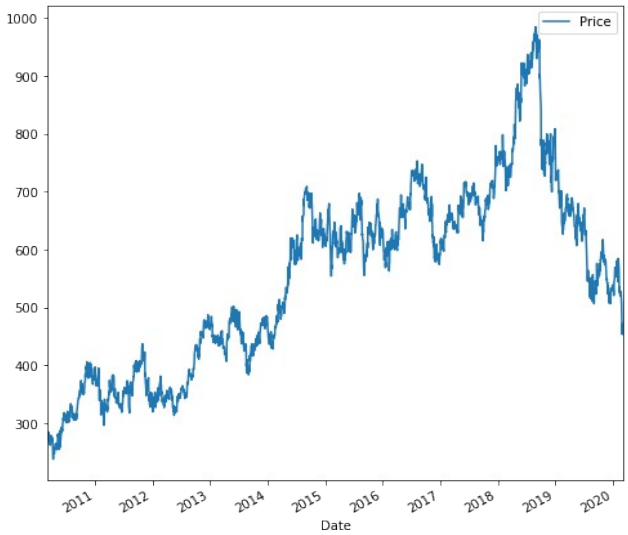

Fig. 10. Mahindra & Mahindra Limited.

The ARIMA model With the help of Particle Swarm Optimization, we optimized the parameters of the ARIMA model to get the minimum AIC and found the best-fitted model as ARIMA(3,1,2). The predicted test values of the input time series model are shown in Fig. 11.

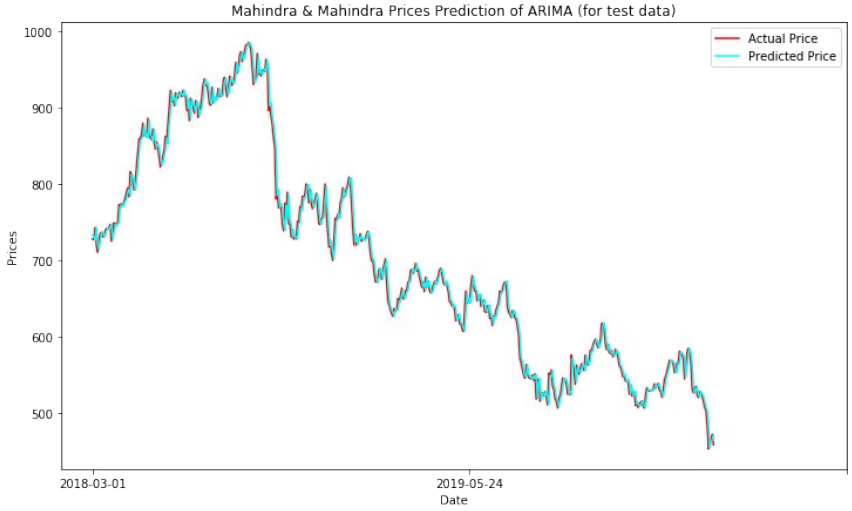

Fig. 11. Estimated values of ARIMA model for Mahindra & Mahindra Limited (M&M.NS) (Test Dataset).

The LSTM Model The input layer is specified by Input shape, with one hidden LSTM layer, one dense output layer and window size is taken as 50. The predicted test values of the input time series by LSTM model is visualized in Fig. 12.

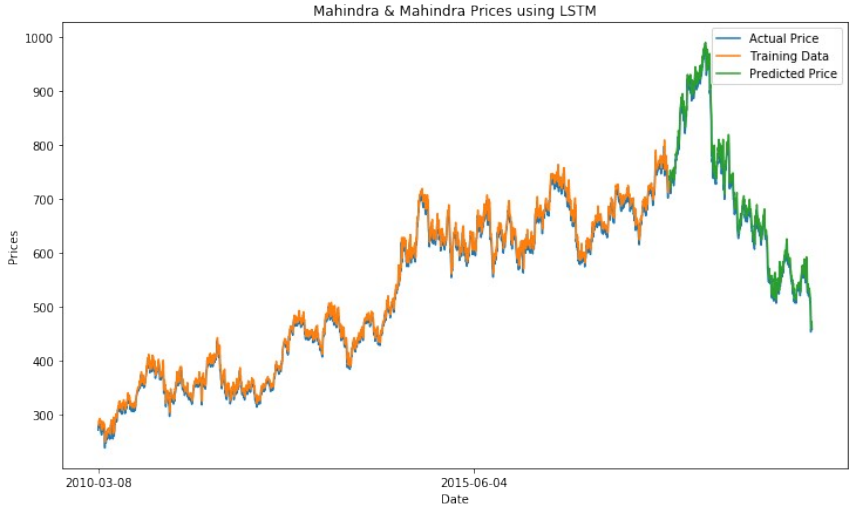

Fig. 12. Estimated values of LSTM model for Mahindra & Mahindra Limited (M&M.NS) (Test Dataset).

The Series Hybrid ARIMA-LSTM Model The parameters of the ARIMA model are optimized by Particle Swarm Optimization having minimized AIC value and gives the best fit as ARIMA(3,1,2). The predicted output of the ARIMA model is shown in Fig. 11. The ARIMA model giving the output residuals is now given input to LSTM. The predicted test values of the input time series by series hybrid ARIMA-LSTM model are shown in Fig. 13.

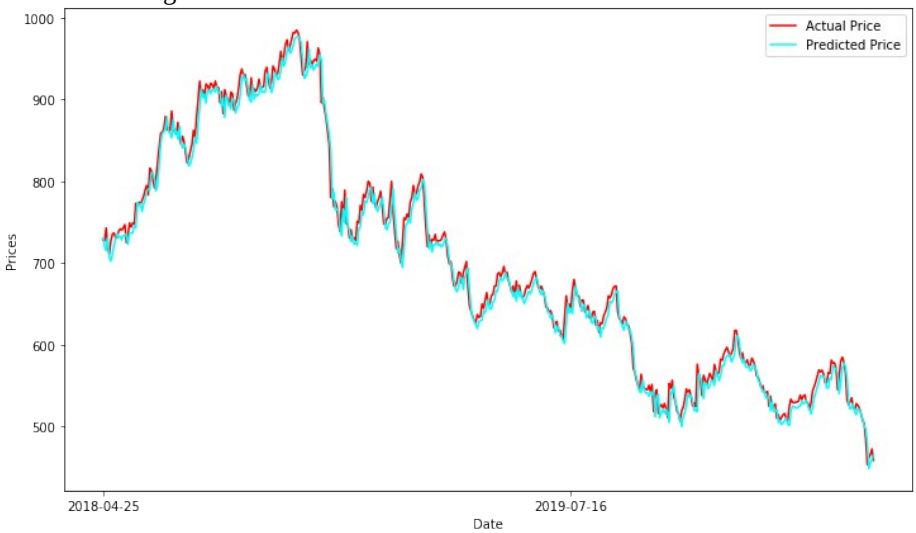

Fig. 13. Estimated values of hybrid ARIMA-LSTM model for Mahindra & Mahindra Limited (M&M.NS) (Test Dataset).

The Series Hybrid ARIMA-DBN Model On optimizing the ARIMA parameters using PSO, the best fit model is ARIMA(3,1,2) with minimized AIC. The predicted output of the ARIMA model is shown in Fig. 11 and its non-linear residual is now given input to DBN. Using PSO we found 512 hidden units in one RBM. The predicted test values of the input time series by series hybrid ARIMA-DBN model is shown in Fig. 14.

4.3 Comparison of forecasting results

All four models are compared based on various performance indicators including Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Root Mean Squared Logarithmic Error (RMSLE), Mean Absolute Percentage Error (SAME) and Symmetric Mean Absolute Percentage Error (SMAPE).

$$MSE = \frac{1}{M}\sum_{1}^{M}\left(z_i - \hat{z}\right) \tag{9}$$

$$MAE = \frac{1}{M}\sum_{1}^{M}\left|z_i - \hat{z}\right| \tag{10}$$

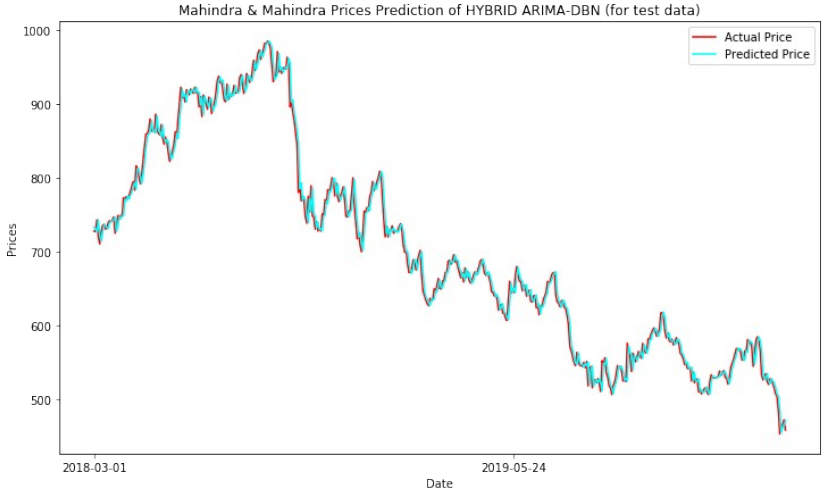

Fig. 14. Estimated values of hybrid ARIMA-DBN model for Mahindra & Mahindra Limited (M&M.NS) (Test Dataset).

$$RMSE = \sqrt{\sum_{1}^{M} ¿¿¿} \tag{11}$$

$$RMSLE = \sqrt{\frac{1}{M}\sum_{1}^{N} ¿¿} \tag{12}$$

$$MAPE = \frac{100\,\%}{M}\sum_{1}^{N}\left|\frac{(z_i - \hat{z}_i)}{z_i}\right| \tag{13}$$

$$SMAPE = \frac{100\,\%}{M}\sum_{1}^{M}\frac{\left|z_i - \hat{z}_i\right|}{\frac{(|z_i| - |\hat{z}_i|)}{2}} \tag{14}$$

Where z_i is the actual value, $\hat{z}_i$ is the estimated value and M is the number of samples.

Results for Infosys Limited (INFY.NS) Comparing the models with various performance indicators, ARIMA-DBN gives the least MSE of 117.176, MAE of 7.668 and least RMSLE of 0.015 among all models shows an improvement as compared to ARIMA and LSTM. Table 1 shows the various performance metrics in Infosys Limited (INFY.NS) for all the models.

Results for Mahindra & Mahindra Limited (M&M.NS) Comparing the models with various performance indicators, ARIMA-DBN gives the least MSE of 175.261, MAE of 9.896 and RMSLE of 0.019 among all models. Among all the models ARIMA-DBN gives better performance results as compared to other models as most commonly MSE is less than others. Table 2 shows the various performance metrics in Mahindra Limited (M&M.NS) for all the models.

Table 1**.** Performance evaluation for Infosys Limited (INFY).

Performance Indicator	**Models**			
	ARIMA	**LSTM**	**ARIMA-LSTM**	**ARIMA-DBN**
MSE	117.191	270.334	139.759	**117.176**
MAE	7.670	13.316	8.678	**7.668**
RMSE	92.644	**16.442**	92.189	92.644
RMSLE	0.015	0.022	0.017	**0.015**
MAPE	0.107	**0.018**	0.107	0.107
SMAPE	10.655	**1.851**	10.572	10.654

Table 2**.** Performance evaluation for Mahindra & Mahindra Limited (M&M.NS).

Performance Indicator	Models			
	ARIMA	**LSTM**	**ARIMA-LSTM**	**ARIMA-DBN**
MSE	175.278	265.064	204.015	**175.261**
MAE	9.897	12.620	10.975	**9.896**
RMSE	196.696	**16.281**	196.943	196.696
RMSLE	0.019	0.024	0.021	**0.019**
MAPE	0.232	**0.019**	0.231	0.232
SMAPE	22.383	**1.829**	22.519	22.383

Table 3**.** Results compared with existing work.

Authors	Performance Metrics (MSE)
M.M Dube, et al. [10]	307.10
G. Peter Zhang [11]	186.827
Proposed Work	
Infosys	117.176
Mahindra & Mahindra	175.261

We observed our MSE for both the dataset is less than as compared to the existing work done by the author as mentioned in Table 3.

5 Conclusion and Future Work

By performing experiments, we can say that the series hybrid ARIMA-DBN gives the best accuracy on average. As PSO is used an optimization technique helps the hybrid models reaching their efficient goals. Based on Mean Squared Error, the series hybrid ARIMA-DBN model shows the least among all the other models. On average, we can say that hybrid models are finer than single models over various performance metrics. We have seen that hybrid models are capable of giving fine quality results and so it can give a competition to other existing models.
In the future, we will analyze the behavior of other hybrid models with optimizations, such that there may be more accurate results in the field of time series domain.

References

1. Biju R Mohan, G Ram Mohana Reddy.: A Hybrid ARIMA-ANN Model for Resource Usage Prediction, International Journal of Pure and Applied Mathematics, Volume 119 No. 12, (2018).
2. Mehdi Khashei, Zahra Hajirahimi.: A comparative study of series arima/mlp hybrid models for stock price forecasting, Communication in Statistics – Simulation and Computation, (2018).

3. Mengjiao Qin, Zhihang Li, Zhenhong Du.: Red tide time series forecasting by combining ARIMA and deep belief network, Knowledge-Based Systems, (2017).
4. Iyan E. Mulia, Harold Tay K.: Roopsekhar Pavel Tkalich, Hybrid ANN-GA model for predicting turbidity and chlorophyll-aconcentrations, Journal of Hydro-environment Research, (2013).
5. Ina Khandelwal, Ratnadip Adhikari, Ghanshyam Verma.: Time Series Forecasting using Hybrid ARIMA and ANN Models based on DWT Decomposition, International Conference on Intelligent Computing, Communication & Convergence (2015).
6. Li Wang1, Haofei Zou, Jia Su, Ling Li and Sohail Chaudhry.: An ARIMA-ANN Hybrid Model for Time Series Forecasting, Systems Research and Behavioral Science, (2013).
7. P. Pai and C. Lin.: A hybrid arima and support vector machines model in stock price forecasting, Omega, vol. 33, no. 6, pp. 497–505, (2005).
8. L. Yu and Y. Zhang.: Evolutionary fuzzy neural networks for hybrid financial prediction, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 35, no. 2, pp. 244–249, (2005).
9. N. Gradojevic.: Non-linear, hybrid exchange rate modeling and trading profitability in the foreign exchange market, Journal of Economic Dynamics and Control, vol. 31, no. 2, pp. 557–574, (2007)
10. M.M Dube, K. O. Awodele, O. Olayiwola and K.O. Akpeji.: Short Term Load Forecasting Using ARIMA, ANN and Hybrid ANN-DWT, SAUPEC, (2017).
11. G. Peter Zhang.:Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing, Volume 50, Pages 159-175, (2003).
12. Nagaraj Naik, Biju R Mohan.: Study of Stock Return Predictions Using Recurrent Neural Networks with LSTM, International Conference on Engineering Applications of Neural Networks, (2019).