BOOST STOCK FORECASTING ACCURACY USING THE MODIFIED FIREFLY ALGORITHM AND MULTICHANNEL CONVOLUTIONAL NEURAL NETWORK

NILESH B. KORADE[1], DR. MOHD. ZUBER[2]

[1]Research Scholar, Department of Computer Science and Engineering, Madhyanchal Professional University, Ratibad, Bhopal, Madhya Pradesh-462044, India.

[2]Associate Professor, Department of Computer Science and Engineering, Madhyanchal Professional University, Ratibad, Bhopal, Madhya Pradesh-462044, India.

[1*]nilesh.korade.ml@gmail.com, [2]mzmkhanugc@gmail.com

ABSTRACT

The number of stock investors is growing every day, so providing more accurate predictions for trend and stock value is essential to earning money. Neural networks have been used in different research studies to predict future movements in stock prices. In comparison to other methodologies, convolutional neural networks (CNN) perform impressively in stock price forecasting. A parameter whose value is used to regulate the learning process is known as a hyperparameter. One of the main issues in CNN is selecting the set of appropriate parameters to train it on a stock dataset that gives correct results. Due to the stochastic nature of a neural network, which uses randomness during learning, there are variations in the performance of CNN on the same dataset. In this study, we proposed a modified firefly algorithm (MFA) approach to identify the best configurations for CNN architectures, and a multi-channel convolutional neural network (MCNN) is used to improve CNN accuracy in stock forecasting. The MFA controls the random movement of fireflies by updating the randomization parameter, and if there is no local best solution in the neighborhood, the firefly will move towards the best solution instead of moving randomly. The outcome of the comparison between the proposed method and existing methods demonstrates that it has the potential to identify the best CNN design parameters in fewer iterations and forecast stock value more accurately, which is closer to actual value. The MFA-MCNN performs better than several state-of-the-art optimization techniques and offers higher accuracy. A smaller number of iterations is needed by the MFA-MCNN to determine the ideal parameter for the CNN architecture. The randomization parameter has a high value in the early iterations, forcing MFA to find the best solution; but, as the iteration progresses, the randomization parameter's value decreases, providing greater convergence.

Keywords: *Convolutional Neural Networks, Forecasting, Firefly Algorithm, Optimization, Stock.*

1. INTRODUCTION

The interest of the investor in trading stocks in the market increases when the exact value of the stock is predicted. Many different things, including employment, businesses, technology, and the economy, are significantly impacted by the stock market [1]. The types of trading include delivery trading, intraday trading, and option trading. In delivery trading, stocks you purchase through delivery trades are added to your demat account. They stay in your possession until you decide to sell them, which could be in a few days to a few years. Buying and selling shares on the stock exchange on the same day is known as intraday or day trading [2]. An option is a contract that grants the buyer the right, but not the necessity, to purchase the underlying asset at a certain price on or before a specific date (in the case of a call option) or sell it (in the case of a put option) [3]. Due to the uncertainty, it is difficult and challenging to predict the correct stock value. Researchers working on deep learning (DL) have recently started using DL methods to address this issue. Since there are more stock market traders now than ever before, intelligent decision systems are gaining attention in the financial markets. The CNN is a deep learning technique that produces promising results, identifies changing stock price trends, and has high prediction accuracy [4]. Choosing the optimal parameter for CNN architecture is one of the challenges we face while training it on stock datasets. There is variation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

in accuracy, and CNN always produces different predictions even when the same dataset and parameters are used each time for training [5]. Therefore, the best methodology is required to quickly choose the ideal parameters and improve CNN's forecasting powers.

The CNN model requires different parameters such as convolution layer filters, kernel size, pool size for the max pooling layer, activation, batch size, epoch, etc. The goal of the optimization process is to choose the best option out of all possible options in order to decrease or increase the cost of the objective function without affecting training effectiveness [6]. Multiple studies have used iterative optimization strategies to solve this issue by automatically determining the best parameters for CNN architectures that improve CNN performance. The best meta-heuristic techniques presently used for CNN hyperparameter optimization include the random search (RS) algorithm [7], particle swarm optimization (PSO) [8], bat algorithm [9], bee algorithm [10], etc. In other studies, CNN architectures are optimized using the Firefly Algorithm (FA) [11, 12], which gives promising outcomes.

Less-bright fireflies in the FA algorithm are attracted to more-bright fireflies with the optimal solutions; otherwise, they would travel randomly. The objective function, which must be maximized or minimized by choosing appropriate values from a range of feasible values for the variables, determines the brightness of a firefly. The weaknesses of the FA are its inability to memorize or its tendency to become trapped in local optima [13]. The primary contribution of the research is the implementation of a hybrid method where the Multi-Channel Convolutional Neural Network (MCNN) is used to improve CNN's performance in stock forecasting and the Modified Firefly Algorithm (MFA) is used to find the best configurations for CNN architectures. The proposed MFA-MCNN method identifies the best CNN architecture parameter in fewer iterations with less error and employs a multichannel neural network to improve forecasting accuracy by combining predictions from multiple CNN. The randomization parameter in the proposed technique decreases linearly with each iteration rather than remaining constant. The MFA algorithm regulates the random moving behavior of fireflies, where the fireflies would be attracted towards the best solution rather than moving randomly if there was no local best solution in the neighborhood. The MFA algorithm overcomes the drawbacks by increasing the algorithm's potential for exploitation and exploration, decreasing its randomness, and increasing the collective movement of the fireflies [14]. Due to the neural network's stochastic nature, which uses randomness during learning, the performance of CNN may vary. The performance of CNN can be improved by using multiple CNNs running in parallel, and the output of each CNN is merged by MCNN.

2. LITERATURE SURVEY

When it comes to evaluating the financial markets, there are two primary viewpoints: technical analysis and fundamental analysis. In order to forecast future stock prices, fundamental analysis looks at the financial standing of the company, its employees, board of directors, balance sheets, income reports, etc. Investors frequently examine the return on equity ratio, debt/equity ratio, market capitalization, price/sales ratio, earnings per share, price/earnings ratio, return on assets ratio, and other ratios in fundamental research. The technical analyst uses charts that show past market prices and technical indicators to try and forecast the direction of the stock market. Moving average, exponential moving average, simple moving average, relative strength index, convergence/divergence rules, and on-balance volume are some of the technical indicators used in technical analysis. Stock forecasting is impossible when done manually; an automated method is required. CNN is more accurate and can identify both upward and downward stock trends compared to other deep learning techniques [15].

The selection of hyperparameters in a DL model has a significant impact on the performance of the model. While training CNN, choosing the best parameters is essential because it affects both learning time and accuracy [16]. The values of the hyperparameter vary with the dataset, so it is difficult to identify a perfect set of hyperparameters in a reasonable time period. In random search, combinations of the hyperparameters are chosen randomly using probability to find the optimal solution for the model [17]. The weighted random search approach, which combines probabilistic greedy heuristic strategy with random search, finds the global optimum faster than random search [18]. Swarm intelligence (SI) is an artificial intelligence (AI) technique that focuses on collective behaviors to address challenging optimization issues and gives promising results [19]. PSO is a population-based stochastic optimization technique based on social behaviors in flocks of birds, where each bird in the swarm constantly modifies its search pattern based on its own and other members' learning experiences [20].

Journal of Theoretical and Applied Information Technology
15th April 2023. Vol.101. No 7

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Each particle in PSO has a specific velocity and position, which stand for the moving parameters and attributes of a solution. A candidate solution is fed into a model using an objective function, which evaluates it against the training dataset and produces a fitness value that represents the particle's quality. The best solutions that each particle individually discovers and the best solution discovered by all particles, respectively, are maintained by PSO as pbest and gbest [21]. A metaheuristic algorithm called the Firefly algorithm is used to solve optimization problems and is intended to mimic the social behavior of fireflies [22]. The FA generates a random population by assigning positions and calculating the brilliance of fireflies using an objective function. The less brilliant fireflies will attract more brilliant fireflies by updating their position [23]. Numerous studies have demonstrated that the FA outperforms other meta-heuristic algorithms in terms of performance [24, 25].

The type of training dataset determines which neural network (NN) should be used, such as an RNN, CNN, or ANN. Multichannel input enables us to build a single neural network by merging different types of essential neural networks, dividing distinct computational flows, and merging them into one output. The use of multichannel CNN makes it possible to combine different findings from separate CNNs into a single, more accurate output [26]. The output of each CNN is merged by MCNN, which runs multiple convolution layers parallelly and improves the performance of CNN [27, 28]. We can configure multiple CNNs using MCNN with varying filter sizes, pool sizes, epochs, activation, batch size, etc. Significant performance gains are observed when comparing the MCNN method to other DL algorithms [29].

3. MATERIALS AND METHODS

3.1 Data Description

State Bank of India (SBI) historical stock data was collected from Yahoo Finance between January 1, 2003, and October 31, 2022, using Panda's data reader [30]. Table 1 summarizes different features and their values for the State Bank of India stock dataset. Investors can use closing prices as useful benchmarks to evaluate how stock values have changed over time. Even in the 24-hour trading period, every stock and other security has a closing price, which is the last price at which it trades on any given day during regular market hours.

Table 1: SBI Stock Dataset

Date	High	Low	Open	Close	Volume	Adj. Close
2003-01-01	27.36	26.70	27.36	26.85	17053263	20.60
2003-01-02	27.58	26.73	26.89	27.19	38374255	20.87
2003-01-03	27.81	26.91	27.74	27.00	21238047	20.72
2003-01-06	26.98	26.23	26.98	26.31	17650251	20.20
2003-01-07	26.66	26.14	26.35	26.37	17585684	20.24
….	….	….	….	….	….	….
2022-10-24	572.50	565.90	567.00	570.50	3692065	570.50
2022-10-25	586.20	568.00	572.50	578.55	19899324	578.55
2022-10-27	585.00	577.60	583.95	579.65	13613533	579.65
2022-10-28	582.65	567.00	579.95	570.75	10043644	570.75
2022-10-31	577.45	568.40	574.95	573.80	9894639	573.80

We selected the closing price of a stock as the target variable as it is conventional to analyze profit or loss estimation using the closing price of a stock on a specific day. Figure 1 depicts the target variable's plot against time.

Figure 1: SBI Closing Price

To transform features on the same scale, a dataset was normalized using MinMaxScalar. The formula for MinMaxScalar is shown in Equation 1.

$$X_{i\_New} = \frac{X_i - X_{Min}}{X_{Max} - X_{Min}} \quad (1)$$

3.2 CNN

A convolutional neural network (CNN) is a type of neural network specifically created for visual data, such as videos or photos, but it also performs well with non-image data, particularly in text classification or natural language processing (NLP). The use of CNN to address issues with time-series forecasting has gained more attention from researchers [31]. There is a need to convert one-dimensional time series data into an input image matrix structure in order to use CNN for stock forecasting efficiently. Figure 2 shows the architecture for CNN.

Journal of Theoretical and Applied Information Technology
15th April 2023. Vol.101. No 7

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

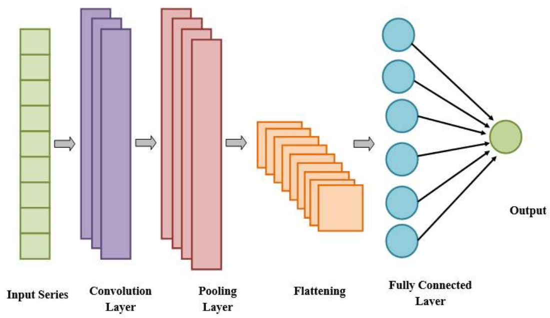

Figure 2: CNN Architecture

The convolution layer, which is the first layer of CNN, performs convolution operations on the data by receiving input signals and applying filters over them. The dot product of the filter matrix and the input matrix produces the convolution layer's final output. Padding and stride operations can be used to adjust the width and height of output vectors. The padding increases the processing region by adding pixels to an input matrix during the convolution process, whereas the stride determines filter movement over the input matrix. The output matrix size can be calculated using Equation 2.

$$\text{Output Matrix Dimension (d)} = \left(\left(\frac{I - F + 2P}{S}\right) + 1\right) \quad (2)$$

where 'I' represents the dimensions of the input matrix, 'F' represents the dimensions of the filter matrix, 'S' represents stride, and 'P' represents padding.

The pooling layer multiplies the resultant matrix from the convolution layer and pooling matrix, which boosts computational power by reducing the dimension of data. A dropout layer can be used, in which a few neurons are eliminated from the NN during the training phase, resulting in a smaller model to avoid overfitting. A matrix is transformed into a one-dimensional series by a flattening operation and then given as input to the fully connected layer. The final few layers that define the outcome are the fully connected layers. The following operation is carried out by an output layer and a fully connected layer [32].

$$O = \text{activation}\left((I\ .K) + B\right) \quad (3)$$

Where O, I, K, and B represent the output, input vector, weight matrix, and bias vector generated by the layer, respectively. The activation function, also called the transfer function, is used to produce the node's output. The tanh, sigmoid or logistic, relu, etc., are the most popular activation functions used in NN [33].

$$sigmoid\ function\ g(z) = \frac{1}{(1 + e^{-z})} \quad (4)$$

$$tanh\ function\ g(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \quad (5)$$

$$relu\ function\ g(z) = max(0\ , z) \quad (6)$$

3.3 Multichannel CNN

Multiple parallel CNN that read the input separately with various kernel sizes can be used to build the multichannel convolutional neural network (MCNN). Each channel is comprised of the convolution, max pooling, flattening, dense layers, etc. described in Figure 3. To create the final output using an average layer, the mean of the output from each channel is calculated [34].

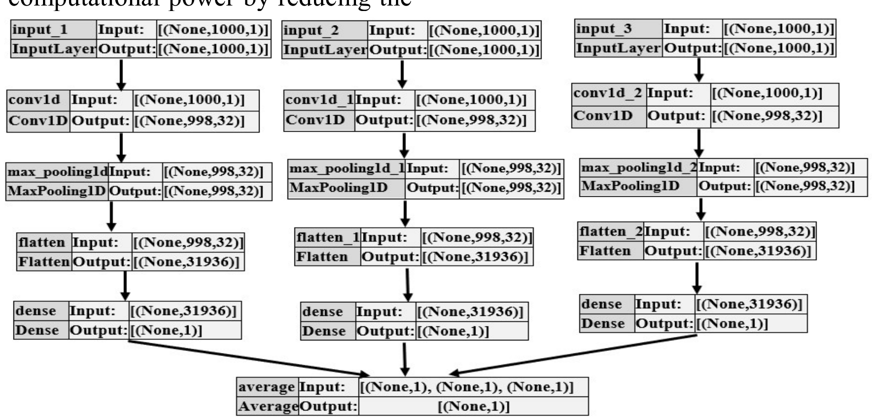

Figure 3: Multichannel Convolutional Neural Network

3.4 Standard Firefly Algorithm

The goal of the optimization method is to find the optimal set of inputs for your NN by training it on different parameters without impacting training performance. An optimization problem includes three components: the set of potential solution values from which to select the variable's value; the function to be optimized; and the optimization rule. The Firefly optimization algorithm was developed by Xin-Shi Yang and is motivated by the mating behavior of fireflies. The foundation of the firefly algorithm is the idealization of the flashing behavior of fireflies. The rules for the Firefly optimization algorithm are listed below.

1. A firefly's brilliance is calculated using an objective function.
2. Brightness and attractiveness are directly proportional to each other, and as the distance between them grows, both brilliance and attractiveness will be reduced.
3. All fireflies are unisex, and any firefly that is less bright is attracted to the brighter firefly; otherwise, it will move randomly [35].

The Firefly Algorithm's processing stages are described below [36].

Inputs: objective function f(X); variable boundary; population size (n); number of dimensions (d); number of iterations (max_iter); attraction (β0); absorption coefficient (ϒ); randomization parameter (α).

Output: the best solution

1. Calculate the random position X for each firefly in all dimensions using the following equation:

$$X_i = \text{lower\_bound} + \text{random\_number} * (\text{upper\_bound} - \text{lower\_bound}) \quad (7)$$

2. Using the objective function $f(X_i)$, where i = 1..n, calculate the cost of each firefly.
3. For it=1 to max_iter
4. For ff_i= 1 to n
5. For ff_j= 1 to n
6. If $f(X_{ffi})>f(X_{ffj})$
7. Calculate the new position and cost, and then move the firefly ff_i toward the firefly ff_j.
8. Otherwise, Firefly ff_i will move at random.
9. End if
10. End for ff_j
11. End for ff_i
12. Determine the best position that has the lowest cost.
13. End for it

The cost is returned by an objective function that determines brightness; as cost increases, fireflies become less brilliant, and vice versa. Using the following equation, a firefly i, which is less brilliant, will be attracted to a firefly j, which is more brilliant.

$$X_i^{it+1} = X_i^{it} + \beta_0 e^{-\Upsilon r_{ij}^2} * \left(X_j^{it} - X_i^{it}\right) + \alpha_{it}\, \epsilon_i^{it} \quad (8)$$

A firefly 'i' will migrate according to the equation above when it encounters a firefly 'j' that is brighter. The X_i^{it+1} is the updated position, and X_i^{it} is the current position of the i[th] firefly [37]. The recommended values used in most of the implementations for attraction (β_0) and randomization parameters (α_{it}) are $\beta_0 = 1$ and α_{it} = [0, 1], respectively. The random number ϵ_i derived using normal or gaussian distribution and the light absorption coefficient (γ), theoretically, varies from 0 to ∞, but in most of the implementations it ranges from 0.01 to 100 [38]. The cartesian or Euclidean distance (r) between two fireflies, i and j, can be calculated using Equation 9 [39].

$$distance\ (r_{i,j}) = \sqrt{\sum_{k=1}^{d} \left(X_{i,k} - X_{j,k}\right)^2} \quad (9)$$

3.5 Modified Firefly Algorithm

Each time the MFA iterates, the randomization parameter that controls how randomly a firefly moves is calculated using Equation 10.

$$\alpha = \frac{0.5 * it}{\max\_iter} \quad (10)$$

Initial iterations have a high value for the randomization parameter, which forces MFA to focus on exploration. Larger α offered better global searching ability in the early stages, whereas smaller α offered better convergence in the later stages. If there is no local best solution in the neighborhood, the firefly will move towards the best solution instead of moving randomly using Equation 11.

$$X_i^{it+1} = X_i^{it} + \beta_0 e^{-\Upsilon r_{i\,gbest}^2} * \left(X_{gbest}^{it} - X_i^{it}\right) + \alpha_{it}\, \epsilon_i^{it} \quad (11)$$

The distance (r) between firefly i and the global best firefly is determined by Equation 12.

$$distance\ (r_{i,gbest}) = \sqrt{\sum_{k=1}^{d} \left(X_{i,k} - X_{gbest,k}\right)^2} \quad (12)$$

The proposed MFA achieves fast convergence by reducing algorithmic randomness. As a result, the suggested approach influences how fireflies progress toward an optimal solution and decreases the likelihood that they will become stuck in a local optimum. The flowchart for the proposed MFA-MCNN is shown in Figure 4.

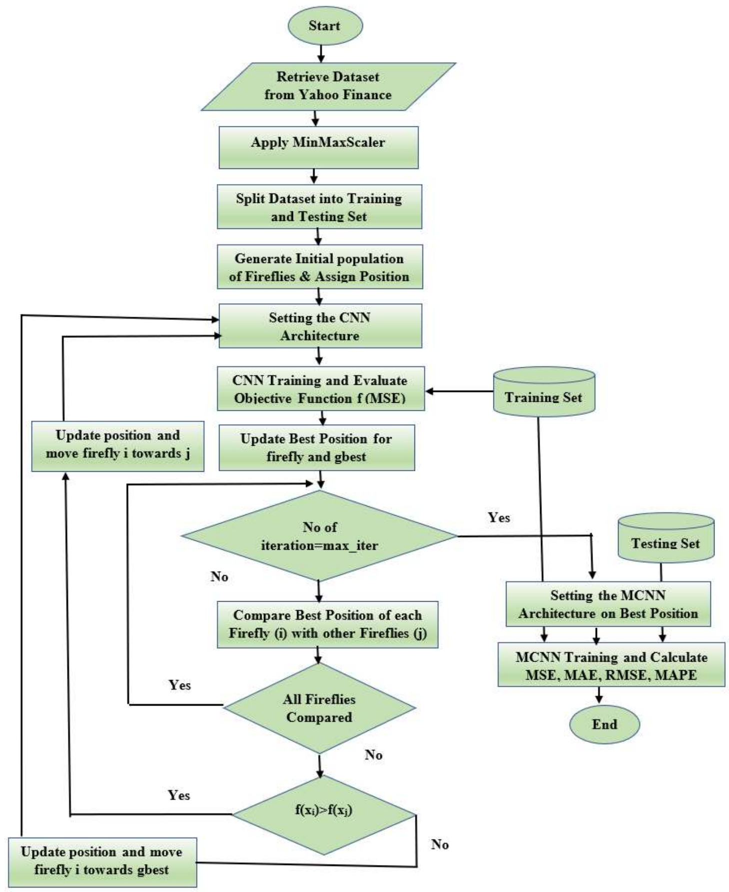

Figure 4: Flowchart of Proposed Methodology

4. RESULT AND DISCUSSION

The training set and testing set of the SBI stock dataset were collected from Yahoo Finance between January 1, 2003, and October 31, 2022. The training set contains 4902 records for the time period from January 1, 2003, to September 30, 2022, and the testing set contains 19 records for the time period from October 1, 2022, to October 31, 2022. The objective function takes the training set and calculates the MSE in MFA and MCNN trained on the return parameter. On training and testing datasets, MFA-MCNN performance is evaluated with FA-CNN, PSO-CNN, and RS-CNN using comparative metrics like MSE, MAE, RMSE, and MAPE. The formula for evaluation metrics is as below [40]:

$$Mean\ Square\ Error = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2 \quad (13)$$

$$Mean\ Absolute\ Error\ \frac{1}{N}\sum_{i=1}^{N}\left|Y_i - \hat{Y}_i\right| \quad (14)$$

$$Root\ Mean\ Square\ Error = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2} \quad (15)$$

$$\begin{matrix}Mean\ Absolute\\ Percentage\ Error\end{matrix} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|Y_i - \hat{Y}_i\right|}{Y_i} * 100\% \quad (16)$$

Time and accuracy in CNN are influenced by the selections made for parameters such as convolution layer filters, kernel size, pool size, batch size, etc. Tables 2 and 3 present the hyperparameter boundary values and a summary of the control parameters for MFA-MCNN, FA-CNN, and PSO-CNN.

The outcome demonstrates that MFA-MCNN finds the optimal parameter in the initial iteration and MSE converges in later iterations, whereas PSO-CNN and FA-CNN required extra iterations to find the optimal parameter. Because the RS-CNN chooses the best parameter at random and compares the results to the previous ones, the possibility of selecting the best parameter is reduced due to randomization.

Table 2: SBI Stock Dataset Upper and Lower Bounds for the Hyperparameters

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Hyperparameter	Lower Bound	Upper Bound	Step Size
Convolution Layer Filters	16	128	16
Kernel Size	1	5	1
Pool Size	1	5	1
Batch Size	32	256	32

Table 3: Summary of the Control Parameters

Control Parameter	PSO-CNN	Control Parameter	FA-CNN	MFA-MCNN
Inertia Weight (W)	0.5	Attraction Parameter (β_0)	0.97	0.97
Correlation Factors (C_1, C_2)	0.5, 0.5	Randomization Parameter (α_t)	1	1-0
Random Number (r_1, r_2)	0≤1	Absorption Coefficient (γ)	0.01	0.01
Maximum Iteration Number	5	Maximum Iteration Number	5	5
Number of Particles	10	Number of Fireflies	10	10

Different optimization techniques are used to train the CNN on parameter return, and the value of the respective evaluation metrics is calculated. Tables 4 compare the training and testing performances of RS-CNN, PSO-CNN, FA-CNN, and MFA-MCNN on the SBI stock dataset. The maximum iteration is set to 5, and number of epochs while training CNN is set to 10.

Table 4: Values of Evaluation Metrics

	Training Performance				Testing Performance			
Evaluation Metrics	RS-CNN	PSO-CNN	FA-CNN	MFA-MCNN	RS-CNN	PSO-CNN	FA-CNN	MFA-MCNN
MSE	314.9	194.5	179.9	120.9	3576.8	1302.3	838.1	281.3
MAE	14.3	10.6	10.3	8.4	58.9	34.8	27.2	14.8
RMSE	17.7	13.9	13.4	10.9	59.8	36.0	28.9	16.7
MAPE	6.2	4.5	4.5	3.7	10.7	6.4	5.0	2.6

The result shows that the value of evaluation metrics is significantly reduced by MFA-MCNN, whereas RS-CNN has the highest value and PSO-CNN and FA-CNN are comparatively equal but greater than MFA-MCNN.

The MFA-MCNN models are tested against state-of-the-art techniques for forecasting the subsequent day's stock price and stock prices for the October month. Figure 5 summarizes the forecasting performance for each technique on the training datasets. By averaging multiple CNN outputs, the MFA-MCNN improves forecasting accuracy on training and testing datasets, bringing predicted output closer to actual.

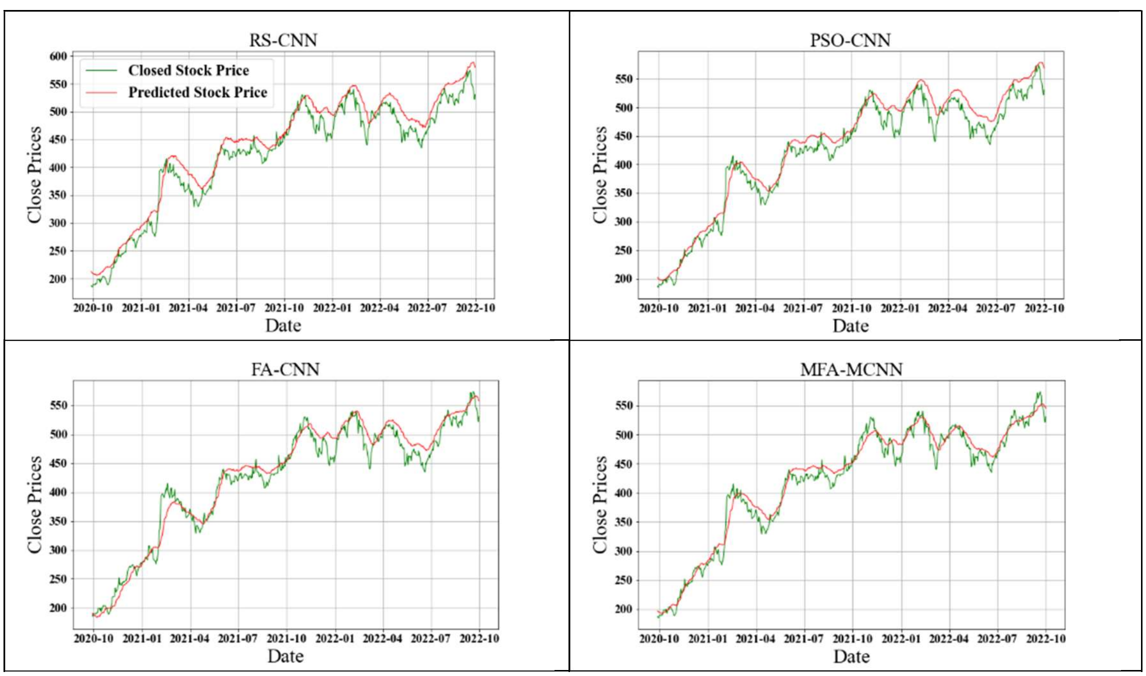

Figure 4: Performance of Stock Price Forecasting for the Next Day

The results demonstrate that MFA-MCNN can correctly identify the next day's stock price as compared to other approaches with less error. Because the MCNN calculates the mean of all projected output by individual CNNs, it is more accurate than other methods for the training dataset. When training a CNN on a stock dataset, the RS-CNN creates higher errors since a parameter is selected at random. The PSO-CNN and FA-CNN are giving some promising results, but not as good as compared to the MFA-MCNN.

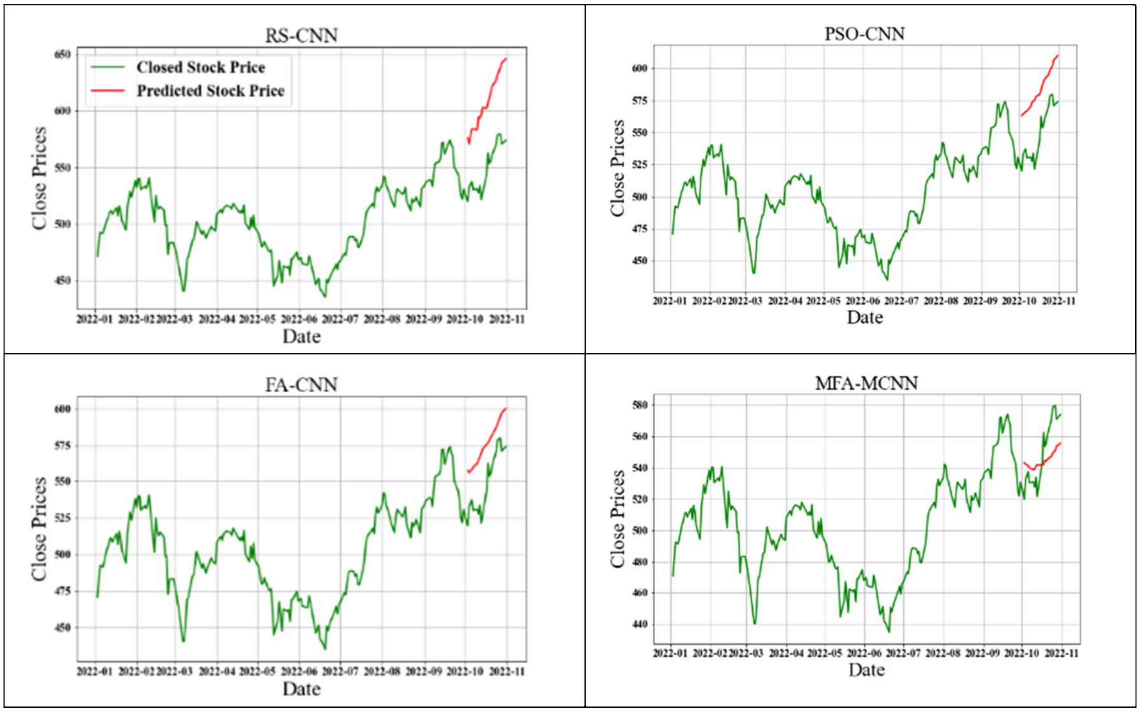

Figure 4: Performance of Stock Price Forecasting for the October Month

The outcome demonstrates that MFA-MCNN finds the optimal parameter in the initial iteration and MSE converges in later iterations, whereas PSO and FA require extra iterations to find the optimal parameter. The results demonstrate that for long-term prediction, the MFA-MCNN can correctly identify trends for stock price as compared to other approaches where actual and anticipated values are closer to each other.

5. CONCLUSION

In this research, we have proposed an MFA-MCNN model to optimize CNN architecture parameters and strengthen CNN forecasting accuracy. The experimental findings demonstrate that the MFA-MCNN delivers higher accuracy and outperforms several state-of-the-art optimization techniques. The MFA-MCNN requires fewer iterations to determine the ideal parameter for CNN architecture. Initial iterations have a high value for the randomization parameter, which forces MFA to identify an optimal solution, whereas a smaller value for the randomization parameter offered better convergence in the later stages. The predicted stock price is closer to the actual price due to averaging multiple CNN outputs. MFA-MCNN can estimate stock prices appropriately and provide relevant information for investors to make a substantial profit. Future enhancements will concentrate on using more data, including predictions based on news stories, and applying more advanced optimization algorithms.

REFERENCES:

[1] D. Shah, H. Isah, and F. Zulkernine, "Stock Market Analysis: A Review and Taxonomy of Prediction Techniques," International Journal of Financial Studies, vol. 7, no.26, 2019, doi :10.3390/ijfs7020026.

[2] H. Liu, P. Ji, and J. Jin, "Intra-Day Trading System Design Based on the Integrated Model of Wavelet De-Noise and Genetic Programming," Entropy, vol. 18, 2016, doi: 10.3390/e18120435.

[3] K. Li, "The effect of option trading," FinancInnov, vol. 7, no. 1, 2021, doi: 10.1186/s40854-021-00279-5.

[4] L. Sayavong, Z. Wu, and S. Chalita, "Research on Stock Price Prediction Method Based on Convolutional Neural Network," 2019

International Conference on Virtual Reality and Intelligent Systems (ICVRIS), pp. 173-176, 2019, doi: 10.1109/ICVRIS.2019.00050.

[5] G. Habib, and S. Qureshi, "Optimization and acceleration of convolutional neural networks: A survey," Journal of King Saud University – Computer and Information Sciences, vol. 34, no. 7, pp. 4244-4268, 2022, doi: 10.1016/j.jksuci.2020.10.004.

[6] B. Qolomany, M. Maabreh, A. Al-Fuqaha, A. Gupta and D. Benhaddou, "Parameters optimization of deep learning models using Particle swarm optimization," 13th International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 1285-1290, 2017, doi: 10.1109/IWCMC.2017.7986470.

[7] J. Bergstra, and Y. Bengio, "Random Search for Hyper-Parameter Optimization, Journal of Machine Learning Research," vol. 13, pp. 281-305, 2012.

[8] J. Chen, J. Jiang, X. Guo and L. Tan, "A self-Adaptive CNN with PSO for bearing fault diagnosis," Systems Science & Control Engineering, vol. 9, no. 1, pp.11-22, 2021, doi: 10.1080/21642583.2020.1860153.

[9] T. Hussein, M. Frikha, S. Ahmed, and J. Rahebi, "BA-CNN: Bat Algorithm-Based Convolutional Neural Network Algorithm for Ambulance Vehicle Routing in Smart Cities," Mobile Information Systems, pp. 1-14, 2022, doi:10.1155/2022/7339647.

[10] M. Ahmad, A. A. Ikram, R. Lela, I. Wahid, and R. Ulla, "Honey bee algorithm–based efficient cluster formation and optimization scheme in mobile ad hoc networks," International Journal of Distributed Sensor Networks, vol. 13, no. 6, 2017, doi:10.1177/1550147717716815.

[11] N. Bacanin, T. Bezdan, K. Venkatachalam, and F. Al Turjman. "Optimized convolutional neural network by firefly algorithm for magnetic resonance image classification of glioma brain tumor grade," Journal of Real-Time Image Processing, vol. 18, pp. 1085–1098, 2021, doi: 10.1007/s11554-021-01106-x.

[12] D. Sánchez, P. Melin, J. Carpio and H. Puga, "A firefly algorithm for modular granular neural networks optimization applied to iris recognition," 2016 International Joint Conference on Neural Networks (IJCNN), pp. 139-144, 2016, doi: 10.1109/IJCNN.2016.7727191.

[13] M. Gupta, and D. Gupta, "A New Modified Firefly Algorithm," International Journal of Recent Contributions from Engineering, Science & IT, vol. 4, no. 2, pp.18-23, 2016, doi: 10.3991/ijes.v4i2.5879.

[14] L. Wang, and Z. Meng, "Multichannel Two-Dimensional Convolutional Neural Network Based on Interactive Features and Group Strategy for Chinese Sentiment Analysis," Sensors, vol. 22, no. 3, 2022, doi: 10.3390/s22030714.

[15] N. Isaac, A. Adebayo, and W. Benjamin, "A systematic review of fundamental and technical analysis of stock market predictions," Artificial Intelligence Review, vol. 53, 2020, doi:10.1007/s10462-019-09754-z.

[16] A. B. Hidri, "Optimization for training CNN deep models based on swarm intelligence," 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), pp. 284-289, 2018, doi: 10.1109/ASET.2018.8379872.

[17] J. Bergstra, and Y. Bengio, "Random Search for Hyper-Parameter Optimization," Journal of Machine Learning Research, vol. 13, pp. 281-305, 2012.

[18] S. Peddapati, and K. K. P. Kruthiventi, "A New Random Search Algorithm: Multiple Solution Vector Approach," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, India, pp. 187-190, 2016, doi: 10.1109/IACC.2016.43.

[19] Y. Cai, and A. Sharma, "Swarm Intelligence Optimization: An Exploration and Application of Machine Learning Technology," Journal of Intelligent Systems, vol. 30, no. 1, pp. 460-469, 2021, doi: 10.1515/jisys-2020-0084.

[20] Y. Wang, H. Zhang, and G. Zhang, "cPSO-CNN: An efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks," Swarm and Evolutionary Computation, vol. 49, pp. 114-123, 2019, doi :10.1016/j.swevo.2019.06.002.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, pp. 1942-1948, 1995, doi: 10.1109/ICNN.1995.488968.

[22] A. Abdulhadi, "Application of the Firefly Algorithm for Optimal Production and Demand Forecasting at Selected Industrial Plant," Open Journal of Business and Management, vol. 08, pp. 2451-2459, 2022, doi:10.4236/ojbm.2020.86151.

[23] R. Aswanandini, C. Deepa, "Hyper-Heuristic Firefly Algorithm Based Convolutional Neural Networks for Big Data Cyber Security," Indian Journal of Science and Technology vol.14, no. 38, pp. 2934-2945, 2021, doi:10.17485/IJST/v14i38.1401.

[24] J. Rajeshwari, and M. Sughasiny, "Dermatology disease prediction based on firefly optimization of ANFIS classifier," AIMS Electronics and Electrical Engineering, vol. 6, no. 1, pp. 61-80, 2021, doi: 10.3934/electreng.2022005.

[25] T. Tiwari, and M. A. Saraswat, "new firefly algorithm-based superpixel clustering method for vehicle segmentation," Soft Computing, pp.1-14., 2022, doi: 10.1007/s00500-022-07206-5.

[26] R. Dinter, C. Catal, and B. Tekinerdogan, "A Multi-Channel Convolutional Neural Network approach to automate the citation screening process," Applied Soft Computing, vol. 112, 2021, doi: 10.1016/j.asoc.2021.107765.

[27] J. Hu, Y. Kuang, B. Liao, L. Cao, S. Dong, and P. Li, "A Multichannel 2D Convolutional Neural Network Model for Task-Evoked fMRI Data Classification," Computational Intelligence and Neuroscience, 2019, doi: 10.1155/2019/5065214.

[28] S. Chakraborty, S. Aich, M. Joo, M. Sain, and H. Kim, "A Multichannel Convolutional Neural Network Architecture for the Detection of the State of Mind Using Physiological Signals from Wearable Devices," Journal of Healthcare Engineering, vol. 2019, doi: 10.1155/2019/5397814.

[29] N. Kesav, and M.G. Jibukumar, "Multi-Channel CNN based image classification using SKIP connection and MSVM," International Journal of Computers and Applications, vol. 44, no. 10, pp.981-990, 2022, doi: 10.1080/1206212X.2022.2047443.

[30] Yahoo Finance - Business Finance Stock Market News, Available: https://in.finance.yahoo.com, [Accessed on November 01,2022]

[31] R. Chauhan, H. Kaur, and B. Alankar, "Air Quality Forecast using Convolutional Neural Network for Sustainable Development in Urban Environments," Sustainable Cities and Society, vol. 75, 2021, doi: 10.1016/j.scs.2021.103239.

[32] L. Alzubaidi, J. Zhang, A. J. Humaidi, Y. Duan, O. Al Shamma, J. Santamaría, M. A. Fadhel, M. Al Amidie, and L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," Journal of Big Data, vol. 8, no. 53, 2021, doi: 10.1186/s40537-021-00444-8.

[33] G. Lin, W. Shen, "Research on convolutional neural network based on improved Relu piecewise activation function," Procedia Computer Science, vol. 131, pp. 977-984, 2018, doi: 10.1016/j.procs.2018.04.239.

[34] L. Lu, C. Zhang, K. Cao, T. Deng and Q. Yang, "A Multichannel CNN-GRU Model for Human Activity Recognition," IEEE Access, vol. 10, pp. 66797-66810, 2022, doi: 10.1109/ACCESS.2022.3185112.

[35] S. Agarwal, A. P. Singh and N. Anand, "Evaluation performance study of Firefly algorithm, particle swarm optimization and artificial bee colony algorithm for non-linear mathematical optimization functions," 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), pp. 1-8, 2013, doi: 10.1109/ICCCNT.2013.6726474.

[36] W. Dai, L. Liang, and B. Zhang, "Firefly Optimization Algorithm for the Prediction of Uplift due to High-Pressure Jet Grouting", Advances in Civil Engineering, vol. 2020, doi: 10.1155/2020/8833784.

[37] X. S. Yang, "Firefly Algorithms for Multimodal Optimization," O. Watanabe and T. Zeugmann (Eds.): SAGA 2009, Lecture Notes in Computer Science, pp. 169–178, 2009, doi: 10.1007/978-3-642-04944-6_14.

[38] L. Liberti, C. Lavor, N. Maculan, and A. Mucherino, "Euclidean Distance Geometry and Applications" SIAM Review, vol. 56, 2012 doi: 10.1137/120875909.

[39] S. K. Pal, C.S Rai, and A. P. Singh, "Comparative Study of Firefly Algorithm and Particle Swarm Optimization for Noisy NonLinear Optimization Problems", I.J. Intelligent Systems and Applications, vol. 10, pp. 50-57, 2012, doi: 10.5815/ijisa.2012.10.06.

[40] A. Botchkarev, "Evaluating performance of regression machine learning models using multiple error metrics in Azure Machine Learning Studio," SSRN Electronic Journal, 2018, doi: 10.2139/ssrn.3177507.