Integrating deep learning and optimization algorithms to forecast real-time stock prices for intraday traders

Nilesh B. Korade[1], Mahendra B. Salunke[2], Amol A. Bhosle[3], Dhanashri Joshi[1], Kavita Patil[4], Sunil M. Sangve[5], Rushali A. Deshmukh[1], Aparna S. Patil[1]

[1]Department of Computer Engineering, JSPM's Rajarshi Shahu College of Engineering, Pune, India
[2]Department of Computer Engineering, PCET's, Pimpri Chinchwad College of Engineering and Research, Pune, India
[3]Department of Computer Science and Engineering, MIT Art, Design and Technology University, Pune, India
[4]Department of Computer Science and Business Systems, JSPM's Rajarshi Shahu College of Engineering, Pune, India
[5]Department of Artificial Intelligence and Data Science, Vishwakarma Institute of Technology, Pune, India

Article Info

Article history:

Received Jun 10, 2024
Revised Dec 8, 2024
Accepted Dec 19, 2024

Keywords:

Convolutional neural networks
Forecasting
Intraday
Long short-term memory
Optimization
Particle swarm optimization
Stock

ABSTRACT

The number of stock investors is steadily increasing due to factors such as the availability of high-speed internet, smart trading platforms, lower trading commissions, and the perception that trading is an effective way of earning extra income to enhance financial stability. Accurate forecasting is crucial to earning profits in the stock market, as it allows traders to anticipate price changes and make strategic investments. The traders must skillfully negotiate short-term market changes to maximize gains and minimize losses, as intraday profit mostly depends on the timing of buy and sell decisions. In the presented work, we provide minute-by-minute forecasts that assist intraday traders in making the best decisions on when to buy and sell, consequently maximizing profits on each trade they make. We have implemented a one-dimensional convolutional neural network and bidirectional long-short-term memory (1DCNN-BiLSTM) optimized with particle swarm optimizer (PSO) to forecast the value of stocks for each minute using real-time data extracted from Yahoo Finance. The proposed method is evaluated against state-of-the-art technology, and the results demonstrate its strong potential to accurately forecast the opening price, stock movement, and price for the next timeframe. This provides valuable insights for intraday traders to make informed buy or sell decisions.

Corresponding Author:

Nilesh B. Korade
Department of Computer Engineering, JSPM's Rajarshi Shahu College of Engineering
Pune, India
Email: nilesh.korade.ml@gmail.com

1. INTRODUCTION

The number of traders is growing as it's easier for them to make investments in the financial markets from anywhere due to the accessibility of sophisticated online trading platforms. The proliferation of instructional resources and equipment has enabled a greater number of people to comprehend and get involved in trading [1]. With the potential for substantial benefits and flexible participation, trading is considered an additional source of income for individuals. Traders often get involved with three primary trading options. The purchase and sale of publicly traded company shares on stock exchanges with the objective of making gains by way of dividends and stock price growth is known as equity trading [2]. When you buy stocks through equity trades, your Demat account is credited with those stocks. They stay in your possession until you make the choice to offer them for sale, which may take a few days to several years. Trading options involve the buying and selling of contracts that grant the right, but not the commitment, to purchase or sell a specific asset at a fixed price within a given window of time [3]. The authority to buy an asset at a fixed price is offered by a call option, and the authority to sell it is offered by a put option. To profit from transient price swings, intraday traders buy and sell stocks on the same trading day. In addition to technical research, traders implement strategies such as momentum trading and scalping to detect opportunities. Intraday trading is risky due to fluctuations in the market, but you can reduce your potential loss with the use of mechanisms like set stop-loss orders [4]. There are a few obstacles faced by intraday traders.

a. When a price hits a certain level in intraday trading and stays there for a time, traders may find it difficult to decide whether to sell or wait. Selling could result in a loss of potential gains if the price rises further, while holding can lead to losses if it falls [5].
b. It is challenging to pinpoint the lowest and highest price points in intraday trading, which makes it problematic for traders to successfully place buy and sell orders within the idle price window [6].
c. It is challenging to say with certainty whether the stock price will rise or fall during the present-day trading session at the opening time of the market while trading intraday [7].

Several types of machine learning (ML) and deep learning (DL) algorithms have been implemented by numerous researchers to anticipate stock prices [8]. Most research on stock forecasting concentrates on long-term stock forecasting, with very limited attention given to intraday forecasting with shorter timeframes. However, these intraday prediction strategies often lack accuracy, highlighting the need for a more accurate forecasting framework to improve their reliability. The innovative computational methods enable traders to analyze large volumes of market data, identify trends, and offer forecasts for future price changes. By applying these innovative techniques, researchers desire to boost the accuracy of their forecasts, which will ultimately lead to an enhanced understanding of market dynamics and more intelligent investing decisions [9]. In order to assist intraday traders in picking the most appropriate moments to execute buy and sell orders, this study focuses on minute-level stock price prediction. The presented research utilizes minute-level SBI stock data for June gathered from Yahoo Finance. The collected stock data is preprocessed and used to train a one-dimensional convolutional neural network and bidirectional long-short-term memory (1DCNN-BiLSTM) model. The model predicts the stock price for a 15-minute interval, after which new data is retrieved from Yahoo Finance to forecast the next 15-minute price, and this process continues iteratively. particle swarm optimizer (PSO) optimization has been applied for estimating the values for several 1DCNN-BiLSTM parameters that influence forecasting accuracy. The proposed approach is assessed against existing methods using regression metrics, trend forecasting capability, and swing identification. The research findings reveal that PSO selects the optimal parameters in fewer iterations, and the 1DCNN-BiLSTM accurately forecasts the stock's opening price, trend, and value compared to other methods.

2. LITERATURE SURVEY

As the stock market is so complicated and continuously changing direction, it is challenging to predict its movements. Implementing an extra trees classifier (ETC) model optimized for small-term stock return estimation, Pagliaro introduces a novel approach to stock return prediction. The model is trained using technical indicators, and the target is the percentage change in closing prices for 120 enterprises across different industries after 10 trading days. The indicators used to predict the direction of the stock price are the medium, upper, and lower Bollinger bands, referred to as BOLL M, BOLL U, and BOLL L; average true range (ATR); average directional index (ADX); fractal weighted moving average (FWMA); volume-weighted moving average (VWMA); Chande forecast oscillator (CFO); and Schaff trend cycle (STC). evaluating ETC, which was trained on decision trees, against bagging, nu-support vector machine (SVM), K-neighbors, XGBoost (XGB), and light gradient boosting machine (LGBM) classifiers, the results indicate that ETC outperformed the others, achieving an accuracy of 86.1% [10]. The stock market has a huge influence on a variety of things, such as jobs, businesses, technology, and the economy. Singh *et al.* [11] presented a framework to estimate the stock price using the live market's real-time stream, which relied on two learning strategies: incremental learning, which updates the model every time it receives a new instance of the stock from the live stream, and offline-online learning, which retrains the model after each trading session to ensure it takes into account the most recent data complexities. The examination uses real-time data streams from the National Stock Exchange (NSE) and NASDAQ to forecast stock prices for the next fifteen minutes, assisting intra-day traders in making decisions. Incremental linear regression has been implemented for incremental models, whereas long short-term memory (LSTM) and convolutional neural network (CNN) variations have been implemented in forecasting through offline-online models. B-LSTM performed better than other variations of LSTM, including bidirectional LSTM, vanilla LSTM, stacked LSTM, and CNN, when compared on regression metrics.

As there are numerous elements that influence the stock market and create an unpredictable environment for investors, choosing a stock for a short-term investment carries a significant risk of failure.

Haryono *et al.* [12] present a methodology to assess the performance of graph convolutional networks (GCN) layers, LSTM, gated recurrent units (GRU), and CNN in Indonesian stock forecasting. The 2,588,451 rows of stock data from 727 companies on the Indonesia stock exchange (IDX) are included in the dataset. The outcome reveals that the architecture constructed using the LSTM layer yields the greatest results when compared in terms of performance and validates the best outcome. Opening and closing prices, trading volume, and market indicators are a few of the characteristics that define stock data, which is time-series in nature, highly volatile, and multidimensional. Using LSTM, Kumar and Gandhmal [13] constructed an intelligent system for predicting the Indian stock market's movements. Technical indicators such as WILLR, ROCR, MOM, RSI, and ROCR. Have been calculated using the stock data for Infosys and Zensar that has been collected from the NSE and BSE. The LSTM is trained using technical indicators, and an optimization algorithm is applied to obtain the optimal value of the LSTM hyperparameters. The forecasts' outcomes guarantee that their findings will assist traders and stock market specialists in correctly choosing whether to purchase or sell a specific share.

As the economy has continued to grow significantly in recent years, an increasing number of people have begun making investments in the stock market. A DL-based approach that forecasts stock prices by combining LSTM and BI-GRU has been introduced by Shaban *et al*. [14]. The NYSE data, which was gathered for every minute between March 2020 and April 2022, was normalized first, and BiGRU-LSTM was trained using that data. The results indicate that the suggested model yields R2 values of 0.9948, RMSE of 0.2883, MSE of 0.0831, and MAE of 0.2099. The suggested method has been assessed with various datasets and the latest techniques, and the results demonstrate that it delivers better forecasting results in terms of trend and closing price estimation. [15] Jin makes use of machine intelligence for investigating sentiment-driven stock prediction in China's financial market, particularly focusing on the banking industry. The study aims to reveal how digital sentiment influences stock price movements by integrating historical stock data from key Chinese banks with sentiment indices from Baidu. The seven prestigious banks listed on the Shanghai Stock Exchange with daily stock prices from August 2022 to August 2023 from Yahoo Finance are used in the study. Prior to constructing a final sentiment-integrated LSTM (SB-LSTM) model on the SSE dataset, grid search was used to determine the ideal value for the hyperparameter. The observations primarily demonstrate that the SB-LSTM model outperforms a normal LSTM model in terms of prediction performance.

Improving profit and avoiding risk are crucial for a strong economy, which is the intent of every stock market investment. An intelligent soft computing framework based on effective feature optimization and hybrid detection techniques is presented by Venkateswararao and Reddy [16] to forecast the direction of stock movement. An improved Ebola optimization (IEO) is implemented in the investigation to prepare the data and eliminate noise and irregular patterns. The chaotic farmland fertility (CFF) algorithm is used to select hyperparameters and effectively increases the speed of convergence for high-dimensional optimization problems. A hybrid spiking-quantum neural network (H-SQNN) framework is presented to forecast stock movement direction, ensuring that false prediction rates are decreased. Based on the observation, the hybrid SQNN framework achieves predicting accuracy of 95.2% for the U.S. stock, 94.32% for the Australian stock, and 93.56% for China's wind economy datasets, respectively. As financial markets are the foundation of every nation's economy, precisely correct stock market forecasting is crucial to helping investors maximize their investment returns as well as governments. The ultimate objective of Ali *et al.* [17] research is to discover an innovative approach for accurately predicting the KSE-100 index's daily closing values. As stated by the authors, the demonstrated hybrid Akima-EMD and the LSTM technique are quite successful in making predictions with nonstationary and nonlinear data. The hybrid Akima-EMD-LSTM model has been suggested as an effective model for the prediction of non-stationary and nonlinear complex financial time series data, as it outperforms all other models considered in the present research when compared to a single LSTM and other ensemble models like SVM, RF, and DT.

3. METHOD

In the proposed approach presented in Figure 1, we collect live data from Yahoo Finance. This data is preprocessed and used to train several models, including 1DCNN, LSTM, BiLSTM, and 1DCNN-BiLSTM. Regression metrics evaluate each model's performance, and the model that provides the best values and accurately identifies trends is selected for forecasting. Parameters for the models are optimized using techniques such as random search, PSO, ant colony, and Firefly. These optimization algorithms are assessed based on the number of iterations required and the accuracy of the models trained with those parameters. The final forecasting model is built using the selected model and optimization techniques. Every 15 minutes, fresh stock data is collected, and the chosen model is rebuilt with parameters identified by the optimization algorithm. The trained model predicts stock values for the next 15 minutes to assist intraday traders and so on.

Google Colab serves as the primary platform for this research's environmental setup, offering a cloud-based environment with GPU support for effective model training and execution. The *pandas_datareader* library is used to retrieve stock data from Yahoo Finance, ensuring easy access to reliable financial datasets. In order to standardize inputs and efficiently access model correctness, the sklearn library is used for data preprocessing and performance evaluation. TensorFlow and Keras are used to build deep learning models, offering a strong foundation for implementing complex architectures into practice. Matplotlib and Seaborn are used to visualize data trends and outcomes, allowing for unambiguous and insightful graphical representations throughout the analysis.

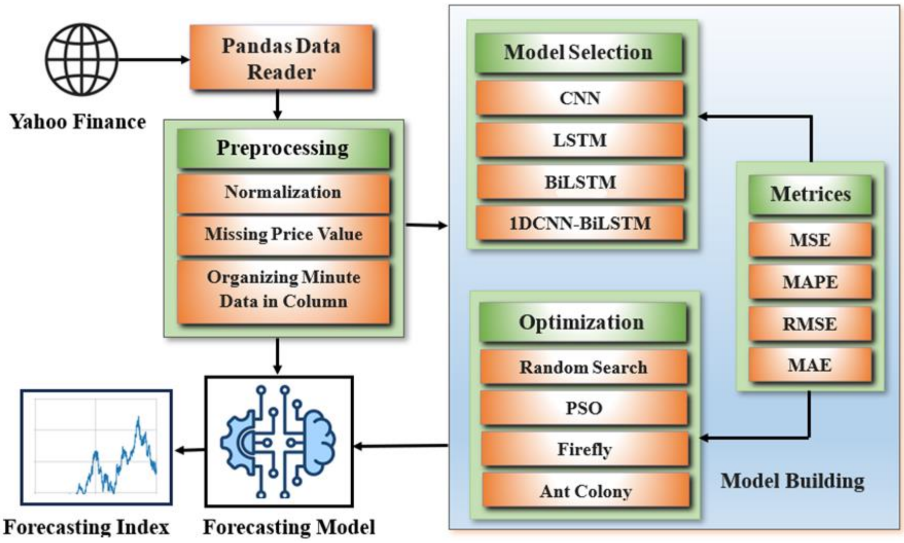


Figure 1. Proposed stock forecasting approach

3.1. Dataset

Panda's data reader had been applied to extract minute-by-minute historical stock data from Yahoo Finance of the State Bank of India's (SBI) June 2024 month [18]. Basic preprocessing was conducted to standardize the data, including normalization. Missing values were replaced with the preceding value, as it was observed that Yahoo Finance does not record the stock price if it remains the same for two consecutive minutes. The data is organized minute by minute in the column displayed in Table 1.

Table 1. SBI minute-wise opening price dataset

Date/Time	9:15:00	9:16:00	9:17:00	9:18:00	9:19:00	9:20:00	9:21:00	…	15:29:00
3/6/2024	867.04	866.34	875.29	867	869.70	865.29	862.84	…	909
4/6/2024	880.40	864.04	874.20	879.40	878.54	874.5	874.29		779.95
5/6/2024	790	781.15	784.5	781.54	776.79	773.75	765.09		787.09
6/6/2024	796.79	808.09	808.95	810	814.34	811	807.90		817.54
7/6/2024	815	813.04	814.29	812.54	812.59	813.84	815.29		829.70
…	…	…	…	…	…	…	…	…	…

3.2. Convolutional neural network

Convolutional neural network's popularity has been increasing as a result of its effectiveness in dealing with a variety of time-series forecasting challenges as well as its potential to discover patterns and trends in data to increase predicted accuracy [19]. As compared to conventional fully connected networks, 1DCNNs require fewer parameters for processing one-dimensional data, such as time-series data, and are capable of autonomously extracting relevant features from raw data. Figure 2 illustrates the architecture of the CNN.

In 1D-CNN, the convolutional layers perform convolution operations along a single dimension and identify local patterns and features by scrolling over the data with filters termed kernels [20]. 1D-CNN controls the size of an output matrix by adding padding, which specifies the number of pixels added to an input matrix during the convolution process. The stride, or the number of displaced pixels, affects how the filter convolves across the input matrix. The pooling layers preserve the most important features while reducing the computational complexity and dimensionality of the input. By periodically setting the input units to zero at a random frequency at each training step, the dropout layer helps reduce overfitting. A one-dimensional array produced by flattening the output matrix is fed into the feed-forward neural network so that it can be processed further [21].

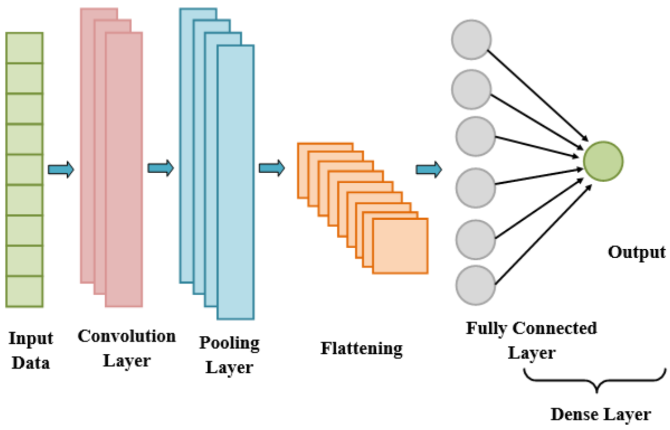


Figure 2. CNN architecture

3.3. Long short-term memory

Long short-term memory (LSTM), an upgraded version of RNN that can capture data from previous stages and use it for future predictions, can overcome the vanishing gradient problem and exploding gradient problem, two major challenges in RNN. The memory line-based LSTM has proven to be exceedingly successful in forecasting scenarios with lengthy data sets. A single processing unit that can store information over time is called an LSTM cell, and it is found within an LSTM layer [22]. Three gates can add or remove information from the cell state, which is represented by the straight line at the top of Figure 3.

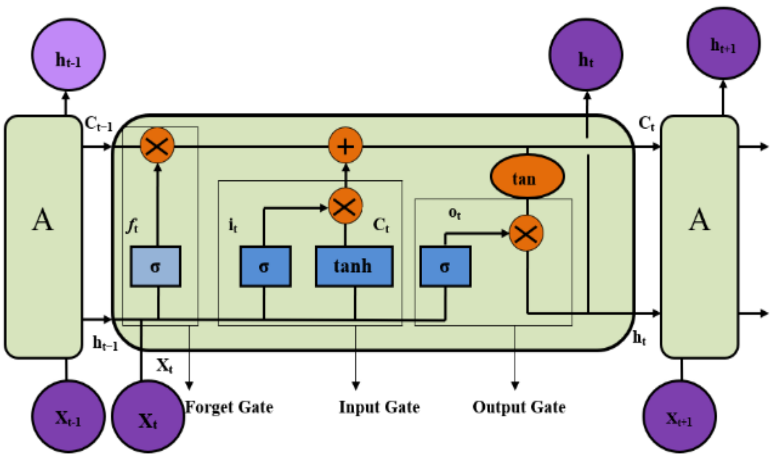


Figure 3. LSTM Architecture

What data should be retained and erased from the cell state is determined by the sigmoid layer, also known as the forget gate layer. The data from the current input x_t and data from a previous hidden state h_{t-1} are merged and run through the sigmoid function, which yields values between 0 and 1, where "discard" is indicated by values nearer 0, and "keep" is indicated by values closer to 1.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f \tag{1}$$

Selecting the new data that will be stored in the cell state includes two phases. The values we will update are determined by the sigmoid layer (σ), and a tanh layer generates a vector with new potential values ($\hat{C}_t$). Equation (4) is applied in order to update the cell state from the old C_{t-1} to the new C_t.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i \quad (2)$$

$$\hat{C}_t = \tanh\,(W_c[h_{t-1}, x_t] + b_c \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \quad (4)$$

The output gate's sigmoid layer determines which portions of the cell state to return as output. The tanh layer transforms the cell state from 1 to -1, and output is produced by multiplying the sigmoid layer's output, by the tanh layer's output represented in (5) and (6) [23].

$$O_t = \sigma(W_0[h_{t-1}, x_t] + b_0 \quad (5)$$

$$h_t = O_t * \tanh(C_t) \quad (6)$$

3.4. Particle swarm optimizer

Particle swarm optimizer (PSO) is a computational technique that aims to improve a candidate solution iteratively with respect to a certain quality parameter. Every particle in the swarm that the program maintains is a potential solution [24]. The best-known positions of the individual particles, as well as those of their neighbors or the entire swarm, all have an impact on the velocity at which particles travel across the search space. Every particle in the swarm has its position updated so that it will move in the direction of the particle with the best position. Every particle keeps track of two things: "gbest," the best solution found by all particles, and "pbest," the best solution found by each particle independently, in order to update its position and velocity in each iteration [25]. As the goal is to lower forecasting error, the MSE is utilized as the measure, and the objective function's purpose is to minimize the MSE value. An overview of the phases of processing needed to apply the PSO is presented below [26], [27].

Output: The lowest MSE is obtained from the most effective parameter values.
Inputs: $f(X)$ objective function; (l, u) variable boundary; (n) population size; (d) number of dimensions; (m_{itr}) number of iterations; (W) inertia weight (W); (C_1, C_2) correlation factors; (r_1, r_2) random number.
Level 1: Compute arbitrary velocity 'v' and position 'p' in all directions through the following formula stated in (7) and (8).

$$Position\ of\ particle_i(x_{i)} = \mathrm{l} + \mathrm{r} * (\mathrm{u} - \mathrm{l}) \quad (7)$$

$$Initial\ Velocity\ for\ Particle_i(v_i) = l + r * (u - l) \quad (8)$$

Level 2: Set particle best position $(pbest)$=initial position (x_i) and global best $(gbest)$=best position among all particles having the lowest MSE.
Level 3: For iteration 1 to m_{iter}.
Level 4: For each particle 1 to n, calculate the new position and velocity using (9) and (10).

$$X_i^{iter+1} = X_i^{iter} + V_i^{iter+1} \quad (9)$$

$$V_i^{iter+1} = WV_i^{iter} + C_1 r_1^{iter}\left(P_{iter}^b - X_{iter}\right) + C_2 r_2^{iter}(P^q - X_{iter}) \quad (10)$$

where V_i^t refers to the velocity of the i^th particle in iteration t, X_i^t refers to the position of a particle, V_i^{t+1} is the newly calculated velocity, and X_i^{t+1} refers to the newly computed position. The P_t^b is the particle's own best position in iteration t, and P^q is the global best position of all participants.
End of for loop
Update $gbest$ set $gbes$t=best position from all particle having lower MSE in iteration t.
End of iteration

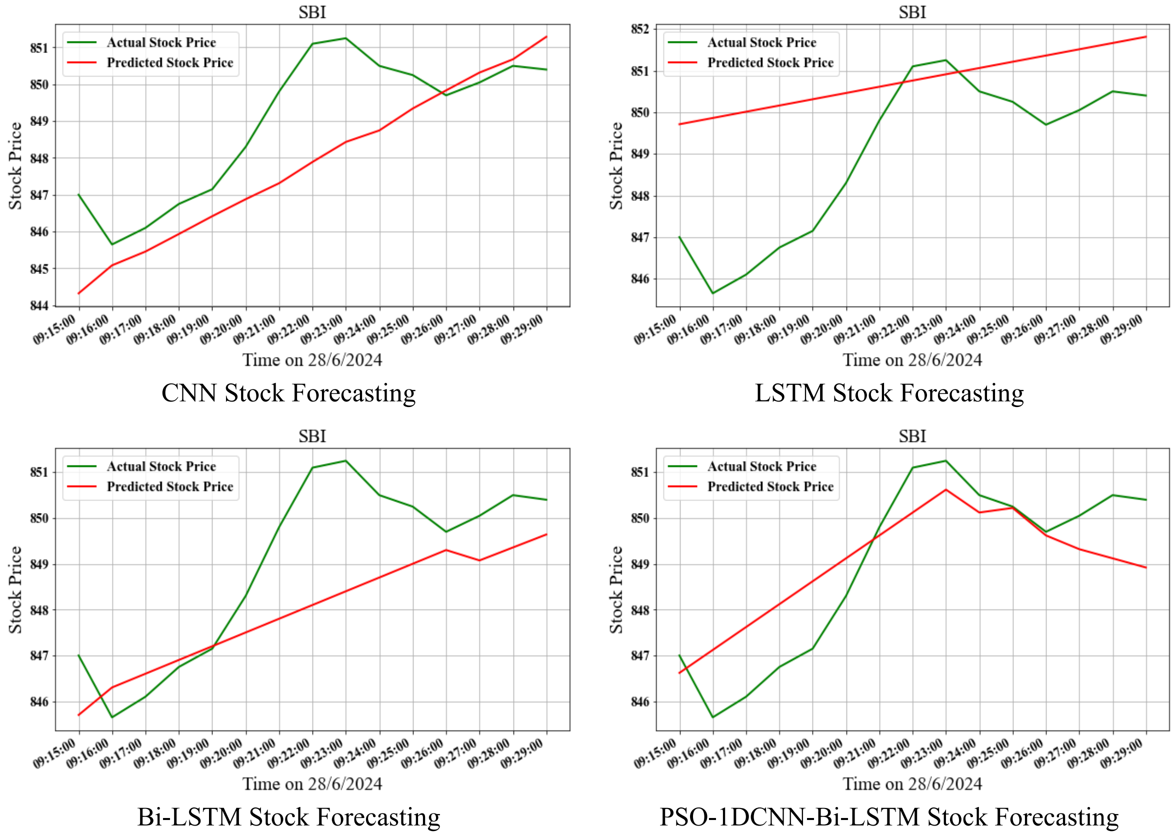
4. RESULT AND DISCUSSION

The 1DCNN, LSTM, Bi-LSTM, and proposed 1DCNN-BiLSTM using PSO were trained using the June month stock minute-wise data. The prediction interval is set to 15 minutes. After each 15-minute period, new live data will be retrieved from Yahoo to forecast the subsequent 15 minutes of stock movement, and so on. Intraday traders can improve their trading decisions and risk management by using the trend to determine market direction, momentum, and prospective patterns, while the opening price provides initial market sentiment and important points of reference. Table 2 shows the opening price predictions from different models, the trend, and the accuracy metrics for predicting the next 15-minute interval.

Figure 4 presents the stock value for 15 minutes on June 28, 2024, from 9:15 AM to 9:30 AM, as estimated using different approaches. The result shows that the proposed approach forecasts a nearby accurate opening price, the trend movement, and stock values for a 15-minute interval. When the stock market opens at 9:15 am on a working day, the opening price may significantly differ from the previous day's closing price, potentially leading to a substantial rise or fall in prices in a short period of time after the market opens. If you can predict the nearby opening price and trend for a short period after the market opens, you can make smart short-term investments and potentially earn substantial profits.

Table 2. Comparison of SBI stock forecasting: opening price, trend, and regression metrics

Model	Opening Price	Trend	MSE	MAE	RMSE	MAPE
1DCNN	844.32	Uptrend	2.690	1.300	1.640	0.153
LSTM	849.71	Uptrend	5.135	1.884	2.266	0.222
Bi-LSTM	845.70	Uptrend	2.12	1.174	1.458	0.138
1DCNN-Bi-LSTM	846.62	Uptrend (9:15 to 9:23) Downtrend (9:23 onwards)	1.033	0.859	1.016	0.101



CNN Stock Forecasting LSTM Stock Forecasting

Bi-LSTM Stock Forecasting PSO-1DCNN-Bi-LSTM Stock Forecasting

Figure 4. Actual and forecasted SBI trend, market opening price, and stock price for 15-minute intervals

The results indicate that the presented approach predicts the opening price more accurately compared to other methods. The trend forecasted by the presented approach indicates a rise in price until 9:23 am, followed by a subsequent decline that other methods were unable to forecast. The 1DCNN-BiLSTM model provides highly accurate stock forecasts with minimal errors. During model training, the values of parameters have a substantial impact on the model's performance, highlighting the essential need to identify optimal parameter values. We determined optimal values for filters, kernel size, pool size, number of units for LSTM, batch size, and epochs using random search [28], Firefly [29], ant colony [30], and PSO [31] optimization techniques. The results indicate that PSO outperformed other methods, achieving lower regression metric values and identifying optimal parameters in a single iteration. Predicting whether the stock price will rise or fall during the current intraday trading session is inherently challenging. The 1DCNN-LSTM accurately predicts price movements, enabling intraday traders to make informed decisions regarding buying or selling.

5. CONCLUSION

A lot of people think that trading is an effective way to increase their income and improve their financial security. Predicting price movements and making intelligent investments have become essential for making money in the stock market, and both require accurate forecasting. The research focuses on intraday traders who buy and sell stocks within the same day. In intraday trading, knowing the next movement of a stock allows traders to make buy or sell decisions more easily. The study implemented a 1DCNN-BiLSTM model optimized with PSO, loading stock data from Yahoo Finance at 15-minute intervals. Forecasting is performed at 15-minute intervals, and the results indicate that the proposed method accurately identifies trends and opening prices, with lower regression metric values compared to other strategies. PSO achieves higher accuracy with fewer iterations in identifying optimal parameters. The proposed approach offers valuable insights to intraday traders, enabling them to make informed decisions on whether to sell, buy, or hold. The study can be further extended by exploring additional algorithms and optimization techniques. The application of 1DCNN-LSTM is a significant improvement for intraday traders' toolbox as it offers excellent capabilities for better stock price forecasting as well as strategic insight into market trends and movements.

REFERENCES

[1] A. Yenireddy, M. S. Narayana, K. V. B. Ganesh, G. P. Kumar, and M. Venkateswarlu, "Stock market index prediction based on market trend using LSTM," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 35, no. 3, pp. 1601–1609, 2024, doi: 10.11591/ijeecs.v35.i3.pp1601-1609.

[2] S. Sakphoowadon, N. Wisitpongphan, and C. Haruechaiyasak, "Predicting stock price movement using effective thai financial probabilistic lexicon (ThaiFinLex)," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 4313–4324, 2021, doi: 10.11591/ijece.v11i5.pp4313-4324.

[3] N. B. Korade and M. Zuber, "Stock price forecasting using convolutional neural networks and optimization techniques," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 11, pp. 378–385, 2022, doi: 10.14569/IJACSA.2022.0131142.

[4] K. N. Myint and M. Khaing, "Time series forecasting system for stock market data," *IEEE International Conference on Control and Automation, ICCA*, vol. 2023-Febru, pp. 56–61, 2023, doi: 10.1109/ICCA51723.2023.10181945.

[5] N. B. Korade and M. Zuber, "Forecasting stock price using time-series analysis and deep learning techniques," *Lecture Notes in Electrical Engineering*, vol. 1146 LNEE, pp. 403–421, 2024, doi: 10.1007/978-981-97-0037-0_31.

[6] N. Rouf *et al.*, "Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions," *Electronics (Switzerland)*, vol. 10, no. 21, 2021, doi: 10.3390/electronics10212717.

[7] P. Chhajer, M. Shah, and A. Kshirsagar, "The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction," *Decision Analytics Journal*, vol. 2, Mar. 2022, doi: 10.1016/j.dajour.2021.100015.

[8] N. B. Korade and M. Zuber, "Stock forecasting using multichannel CNN and firefly algorithm," *Cognitive Science and Technology*, vol. Part F1466, pp. 447–458, 2023, doi: 10.1007/978-981-99-2742-5_46.

[9] A. Maiti and P. Shetty D, "Indian stock market prediction using deep learning," in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2020, vol. 2020-Novem, pp. 1215–1220. doi: 10.1109/TENCON50793.2020.9293712.

[10] A. Pagliaro, "Forecasting significant stock market price changes using machine learning: extra trees classifier leads," *Electronics (Switzerland)*, vol. 12, no. 21, 2023, doi: 10.3390/electronics12214551.

[11] T. Singh, R. Kalra, S. Mishra, Satakshi, and M. Kumar, "An efficient real-time stock prediction exploiting incremental learning and deep learning," *Evolving Systems*, vol. 14, no. 6, pp. 919–937, 2023, doi: 10.1007/s12530-022-09481-x.

[12] A. T. Haryono, R. Sarno, and K. R. Sungkono, "Stock price forecasting in Indonesia stock exchange using deep learning: A comparative study," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 1, pp. 861–869, Feb. 2024, doi: 10.11591/ijece.v14i1.pp861-869.

[13] K. Kumar and D. P. Gandhmal, "An intelligent indian stock market forecasting system using LSTM deep learning," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 21, no. 2, pp. 1082–1089, 2020, doi: 10.11591/ijeecs.v21.i2.pp1082-1089.

[14] W. M. Shaban, E. Ashraf, and A. E. Slama, "SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting," *Neural Computing and Applications*, vol. 36, no. 4, pp. 1849–1873, 2024, doi: 10.1007/s00521-023-09179-4.

[15] S. Jin, "Sentiment-driven forecasting LSTM neural networks for stock prediction-case of China bank sector," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 11, pp. 1–7, 2023, doi: 10.14569/IJACSA.2023.0141101.

[16] K. Venkateswararao and B. V. R. Reddy, "An intelligent soft computing framework for forecasting stock movement direction in international stock markets," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 9, pp. 3634–3652, 2023.

[17] M. Ali, D. M. Khan, H. M. Alshanbari, and A. A.-A. H. El-Bagoury, "Prediction of complex stock market data using an improved hybrid EMD-LSTM model," *Applied Sciences*, vol. 13, no. 3, Jan. 2023, doi: 10.3390/app13031429.

[18] Yahoo, “Yahoo!” https://in.finance.yahoo.com (accessed Jul. 01, 2024).

[19] N. B. Korade, M. B. Salunke, A. A. Bhosle, P. B. Kumbharkar, G. G. Asalkar, and R. G. Khedkar, “Strengthening sentence similarity identification through OpenAI embeddings and deep learning,” *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 4, pp. 821–829, 2024, doi: 10.14569/IJACSA.2024.0150485.

[20] W. Ma, C. Gou, and Y. Hou, “Research on adaptive 1DCNN network intrusion detection technology based on BSGM mixed sampling,” *Sensors*, vol. 23, no. 13, p. 6206, Jul. 2023, doi: 10.3390/s23136206.

[21] N. B. Korade and M. Zuber, “Boost stock forecasting accuracy using the modified firefly algorithm and multichannel convolutional neural network,” *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 7, pp. 2668–2677, 2023.

[22] S. Sang and L. Li, “A novel variant of LSTM stock prediction method incorporating attention mechanism,” *Mathematics*, vol. 12, no. 7, 2024, doi: 10.3390/math12070945.

[23] S. Zaheer *et al.*, “A multi-parameter forecasting for stock time series data using LSTM and deep learning model,” *Mathematics*, vol. 11, no. 3, Jan. 2023, doi: 10.3390/math11030590.

[24] A. G. Gad, “Particle swarm optimization algorithm and its applications: a systematic review,” *Archives of Computational Methods in Engineering*, vol. 29, no. 5, pp. 2531–2561, 2022, doi: 10.1007/s11831-021-09694-4.

[25] N. B. Korade, M. B. Salunke, A. A. Bhosle, G. G. Asalkar, B. Lal, and P. B. Kumbharkar, “Elevating intelligent voice assistant chatbots with natural language processing, and OpenAI technologies,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 37, no. 1, pp. 507–517, 2025, doi: 10.11591/ijeecs.v37.i1.pp507-517.

[26] J. C. Bansal, “Particle swarm optimization,” *Studies in Computational Intelligence*, vol. 779, pp. 11–23, 2019, doi: 10.1007/978-3-319-91341-4_2.

[27] W. Dongshu, T. Dapei, and L. Lei, “Particle swarm optimization algorithm: an overview,” *Soft Computing*, vol. 22, pp. 387–408, 2018.

[28] Bergstra J and Bengio Y, “Random search for hyper-parameter optimization,” *Journal of machine learning research*, vol. 13, no. 2, pp. 281–305, 2012.

[29] 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, no. 4, pp. 1085–1098, 2021, doi: 10.1007/s11554-021-01106-x.

[30] F. Abbas, M. Yasmin, M. Fayyaz, M. A. Elaziz, S. Lu, and A. A. Abd El-Latif, “Gender classification using proposed CNN-based model and ant colony optimization,” *Mathematics*, vol. 9, no. 19, 2021, doi: 10.3390/math9192499.

[31] 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.

BIOGRAPHIES OF AUTHORS

Nilesh B. Korade serves as an assistant professor at JSPM’s Rajarshi Shahu College of Engineering, Pune, India. He is a data science professional and holds a Ph.D. in computer science and engineering. His research interests encompass data science, data analytics, time series analysis, machine learning, natural language processing, and deep learning. He has contributed to numerous consultancy projects, filed several Indian patents, and obtained copyrights for his innovative ideas. He can be contacted at email: nilesh.korade.ml@gmail.com.

Mahendra B. Salunke is an assistant professor at PCET’s Pimpri Chinchwad College of Engineering and Research, Pune, India. He is a data science professional with a PhD in computer engineering. His research areas are data science, IoT, and microprocessors. He worked as Pune University's former department head and chairman of the microprocessor subject. He has worked on several consultancy projects, filed several Indian patents, and obtained copyrights for his creative ideas. He can be contacted at email: mahendra.salunke@pccoer.in.

Amol A. Bhosle is an associate professor at the School of Computing, MIT Art, Design, and Technology University, Pune, India. He is a data science professional with a PhD in computer science and engineering. His research areas are data science, data analysis, time series, machine learning, NLP, and deep learning. He has worked on several consultancy projects, filed several Indian patents, and obtained copyrights for his creative ideas. He can be contacted at email: amolabhosle@gmail.com.

Dhanashri Joshi is an assistant professor at JSPM's Rajarshi Shahu College of Engineering, Pune, India. She is a data science professional with a master in computer engineering. Her research areas are data science, cyber security, and blockchain technology. She can be contacted at email: jdhanashrim@gmail.com.

Kavita Patil is an assistant professor at JSPM's Rajarshi Shahu College of Engineering, Pune, India. She is a data science professional with a Master in Computer Engineering. Her research areas are data science, cyber security, and blockchain technology. She can be contacted at email: kavitapatil.06@gmail.com

Sunil M. Sangve is a professor, at Department of Artificial Intelligence and Data Science at Vishwakarma Institute of Technology, Pune, India. He is a data science professional with a PhD in computer science and engineering. His research areas are data science, data analysis, time series, machine learning, NLP, and deep learning. He has worked on several consultancy projects, filed several Indian patents, obtained copyrights for his creative ideas, and guided several research scholars. He is a life member of Indian Society for Technical Education, member of IAENG (HK) and ISRD. He can be contacted at email: sunil.sangve@vit.edu.

Rushali A. Deshmukh is an associate professor at JSPM's Rajarshi Shahu College of Engineering, Pune, India. She is a data science professional with a Ph.D. in computer science and engineering. Her research areas are data science, data analysis, time series, machine learning, NLP, and deep learning. She has worked on several consultancy projects, filed several Indian patents, obtained copyrights for her creative ideas, and guided several research scholars. She can be contacted at email: rushalideshmukh78@gmail.com.

Aparna S. Patil is an assistant professor at JSPM's Rajarshi Shahu College of Engineering, Pune, India. She is a data science professional with a master in computer engineering. Her research areas are data science, cyber security, and blockchain technology. She can be contacted at email: aparnapatil16@gmail.com.