Engineering Applications of Artificial Intelligence 129 (2024) 107617

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Forecasting multistep daily stock prices for long-term investment decisions: A study of deep learning models on global indices

Mohit Beniwal [a,*], Archana Singh [a], Nand Kumar [b]

[a] Delhi School of Management, Delhi Technological University, Delhi, 110042, India
[b] Department of Humanities, Delhi Technological University, Delhi, 110042, India

ARTICLE INFO

Keywords:
Deep learning
Bi-LSTM
DNN
RNN
CNN
GRU
Forcasting
Index
LSTM
Long-term
Prediction
Machine learning
Artificial intelligence
Technical analysis
Fundamental analysis

ABSTRACT

Deep machine learning algorithms play an important role in facilitating the development of predictive models for the stock market. However, most studies focus on predicting next-day stock prices or movements, limiting the usability of the predictive model for investors. This study extensively explores the ability of deep learning models to predict out-of-sample the daily prices of global stock indices over a long term, up to a year. The performance of six models, including Deep Neural Network (DNN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network (CNN), are compared using Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE). The models predict the long-term daily prices of five global stock indices, namely the Nifty, the Dow Jones Industrial Average (DJIA), the DAX performance index (DAX), the Nikkei 225 (NI225), and the Shanghai Stock Exchange composite Index (SSE). The results confirm the superiority of LSTM for predicting long-term daily prices. The Bi-LSTM does not improve the result of LSTM but performs better than other algorithms. CNN overfits the training data and poorly forecasts the long-term stock prices of global indices on the testing data. This research demonstrates the potential of deep learning models for long-term stock price forecasting, offering valuable insights for investors. Additionally, the patterns of predicted daily prices can be helpful in building trading and risk management decision systems.

1. Introduction

The stock market is a complex and dynamic system, and predicting its future prices is a challenging task (Beniwal et al., 2023a). Although, the Efficient Market Hypothesis (Fama, 1970) and Random Walk Hypothesis (Fama, 1995) suggest that it is futile to predict the stock market. However, accurate stock market predictions are crucial for making informed investment decisions, reducing risk, and maximizing returns. Traditional financial models rely on fundamental and technical analysis to forecast future prices. Fundamental analysis involves analyzing financial statements and economic indicators. On the other hand, technical analysis relies on analyzing historical prices to predict future prices. In addition to these traditional methods, statistical and econometric methods such as Auto-Regressive Integrated Moving Average (ARIMA), Seasonal Auto-Regressive Integrated Moving Average (SARIMA), Vector Auto-Regression (VAR), etc. have also been used to forecast stock market time series. However, these linear methods may not capture the complex and non-linear relationships between variables that affect stock prices (Shahi et al., 2020). Recently, machine learning algorithms have emerged as powerful tools for stock market analysis, providing a more objective and data-driven approach to predicting stock prices. These algorithms can extract hidden patterns that are difficult to detect by other methods.

Various machine learning algorithms have been used in stock market analysis, such as Support Vector Machines (SVM), Decision Trees (DT), Random Forests (RF), and Naive Bayes (NB). These algorithms have demonstrated varying degrees of success in predicting stock prices. The accuracy of predictions is subject to a margin of error, which is influenced by the choice of algorithm employed (Nikou et al., 2019). However, neural networks have emerged as robust and effective machine learning algorithms that can efficiently handle noisy and nonlinear data to forecast time series (Yu and Yan, 2020). Deep learning has become a popular approach for analyzing stock market data among neural networks due to its superior performance in prediction tasks. Deep learning enables the creation of computational models that consist of multiple layers of processing, which can learn to represent data with varying

* Corresponding author.
E-mail address: mohitbeniwal@dtu.ac.in (M. Beniwal).

https://doi.org/10.1016/j.engappai.2023.107617
Received 4 June 2023; Received in revised form 25 September 2023; Accepted 24 November 2023
Available online 4 December 2023
0952-1976/

degrees of abstraction (Lecun et al., 2015). There is a growing trend among asset management companies and investment banks to allocate more research funds toward the development of artificial intelligence techniques, particularly in deep learning (Jianga). Deep learning models have shown better performance than linear and machine learning models in stock market prediction tasks, owing to their ability to effectively handle large volumes of data and identify complex nonlinear relationships between input features and prediction targets (Jiangb).

The application of deep learning models has significantly impacted the field of finance, particularly in predicting stock prices. One advantage of deep learning is its ability to automatically extract features from raw data (Wu et al., 2022), (Liang et al., 2017), eliminating the need for feature engineering and improving forecasting accuracy. Deep learning models comprise multiple layers of interconnected neurons, each layer responsible for extracting higher-level features from the input data. Numerous deep-learning architectures have been created to address diverse problems and the inherent structure of datasets (Bhandari et al., 2022). In this paper, we explore some frequently used deep learning models such as Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), Gated Recurrent Units (GRU), and Convolutional Neural Networks (CNN). DNN includes dense hidden layers with a hierarchical topology (Thakkar and Chaudhari). DNNs can learn multiple levels of features from raw input data using multiple layers of nonlinear transformations. An RNN is a specialized type of Artificial Neural Network (ANN) that can handle sequential inputs by incorporating internal feedback connections between neurons (Kumar and Haider, 2021). LSTM, a specific type of RNN, is a neural network architecture that can retain memory. It is well-suited for processing and predicting significant events with longer intervals and time delays within time series data (Lin et al., 2021). However, LSTM can only learn from past information (Alkhatib et al.). Bi-LSTM, a variant of LSTM, can learn from past and future information because it has two hidden layers with opposite directions connected with the same output (Alkhatib et al.)– (Houssein et al., 2022). Like LSTM, GRU can handle sequence data while simplifying the complex computations involved in LSTM (Zhang et al., 2023). While CNN is inspired by computer vision, it can be designed for financial data (Shah et al.), (Hoseinzade and Haratizadeh, 2019).

In stock market prediction, most of the research has focused on predicting the next day's price (Rouf et al., 2021), using iterative methods to predict prices for the entire test data. This approach has occasionally achieved high accuracy but is not always useful for traders seeking long-term predictions. For long-term prediction, machine learning algorithms need to provide multi-output predictions. However, structuring machine learning algorithms for multi-output long-term predictions can be tedious and sometimes impractical (Beniwal et al., 2023b). To address this limitation, in this study, we propose a novel approach for stock market prediction that leverages the time dependency of stock prices. Instead of predicting only the next day's price, we train machine learning algorithms to learn the patterns of price fluctuations in relation to time. By exploiting the time dependency of prices, we aim to predict long-term prices with higher accuracy and precision. To test the robustness of the approach, we experiment with the ML algorithms on stock indices of the top five economies in terms of Gross Domestic Product (GDP) ("Countries by GDP), namely Nifty, the Dow Jones Industrial Average (DJIA), DAX performance index (DAX), Nikkei 225 (NI225), Shanghai Stock Exchange composite index (SSE). Predicting long-term daily stock prices using deep learning models presents multiple challenges compared to short-term predictions. Here are some insights into these challenges:

1. **Temporal Dependency:** Long-term predictions require understanding and modeling more extended sequences of historical data. Deep learning models must capture temporal dependencies that span over extended periods, which can be more challenging than short-term predictions, where the focus is on immediate past patterns.
2. **Data Complexity:** Long-term predictions involve Managing and Processing complex data that can be computationally intensive.
3. **Volatility and Trends:** Long-term stock prices are influenced by both short-term fluctuations and long-term trends. Deep learning models may not capture day-to-day volatility influenced by short-term events.
4. **Overfitting and Underfitting:** With a more extended training period, there's a risk of overfitting or underfitting. Preventing overfitting and underfitting is crucial for accurate long-term predictions.
5. **Feature Engineering**: While deep learning models excel at feature extraction, features such as the P/E ratio and other fundamental indicators are not available on a daily frequency. Hence, they are difficult to include in this study.

The study contributes to bridging the gap between short-term prediction models and the practical needs of investors looking for robust and informed long-term predictions. The proposed approach adds important value to predicting long-term stock prices using deep learning. Further, it can provide traders and investors with more accurate and reliable long-term predictions. This can help them make informed investment decisions and reduce risk. The proposed approach can also be used to develop advanced trading strategies, such as trend-following or mean-reverting. Overall, this study contributes to the ongoing research efforts to improve the accuracy and reliability of stock market predictions using advanced deep learning algorithms. Hence, this paper contributes in the following ways:

1. **Temporal Dependency**: Most predictive studies in the stock market emphasize next-day prices. In contrast, this study presents an approach to exploit price patterns in relation to time dependency to forecast long-term stock prices of global indices.
2. **Models Comparison**: Extensive experiments are conducted on the top five GDPs to evaluate the predicting robustness of LSTM and other deep learning algorithms in the long term.
3. **Evaluation of Deep Learning Models**: The research addresses multiple questions regarding the ability to forecast long-term stock prices of global indices, such as whether RNN performs better than DNN, whether it is better to use Bi-LSTM instead of LSTM, whether GRU outperforms LSTM, and which of DNN, RNN, LSTM, Bi-LSTM, GRU, and CNN is the least suitable.
4. **Practical Utility**: The approach helps investors gauge the market outlook for the long term and make informed decisions.
5. **Inspiration for Future Research**: The study may inspire other researchers to develop long-term trading and risk management systems.

The remainder of the paper is organized as follows: Section 2 presents the literature on stock prediction using deep learning. Section 3 describes the deep learning models and an overview of the prediction algorithm. Section 4 discusses the methodologies to explain the experimental design, and Section 5 presents the results of experiments conducted on global indices, followed by a discussion of the findings. Finally, Section 6 concludes the study, discusses its limitations, and suggests directions for future research.

2. Literature review

2.1. Deep learning in stock prediction

DNNs are feedforward neural networks that process input data in one direction, from the input layer through one or more hidden layers to the output layer. Singh et al. (Singh and Srivastava, 2017) aimed to demonstrate that deep learning can enhance the accuracy of stock market forecasting. The study compared the performance of (2D)2PCA + Deep Neural Network (DNN) with 2-Directional 2-Dimensional Principal Component Analysis (2D)2PCA + Radial Basis Function Neural Network (RBFNN) and found that the proposed method outperformed the RBFNN method, with an improved accuracy of 4.8% for Hit Rate with a window size of 20. The proposed model was also compared with Recurrent Neural Network (RNN) and showed an improved accuracy of 15.6% for Hit Rate. Additionally, the correlation coefficient between actual and predicted return for DNN was 17.1% more than RBFNN and 43.4% better than RNN. Kanwal et al. (2022) proposed a hybrid deep learning (DL) model for timely and efficient prediction of stock prices. The proposed model, BiCuDNNLSTM-1dCNN, combined Bidirectional Cuda Deep Neural Network Long Short-Term Memory and a one-dimensional Convolutional Neural Network. The model was compared with other hybrid DL-based and state-of-the-art models using five stock price datasets. The results indicated that the proposed hybrid model was accurate and reliable for supporting informed investment decisions.

RNNs are designed to handle sequential data with temporal dependencies, such as time series data or natural language processing tasks. Lui et al. (Liu et al., 2020) proposed two attention-based RNN models, DSTP-RNN and DSTP-RNN-II, for long-term and multivariate time series prediction. These models outperformed state-of-the-art methods and provided insights for further exploration of attention-based methods in time series prediction. Ranjan and Mahadani (Ranjan et al., 2022) compared LSTM, bi-directional LSTM, and RNN models with univariate and multivariate features to predict stock prices. The study found that the recurrent neural network approach had the highest accuracy with univariate and multivariate features. The performance was evaluated using the root mean square error (RMSE) and mean square error (MSE) criteria. Naik et al. (Naik and Mohan, 2019) proposed an RNN with recurrent dropout to avoid overfitting and used stock returns based on closing prices as input to the model. Data was collected from NSE India, and the proposed RNN with LSTM outperformed a feed-forward artificial neural network regarding error minimization.

RNNs suffer from vanishing and exploding gradients, making it difficult to train the model effectively. LSTMs overcome the vanishing gradient problem in RNNs. Gülmez (2023) developed a deep LSTM network with the ARO model (LSTM-ARO) to predict stock prices using DJIA index stocks. Four other models, one ANN and three LSTM, including one optimized by Genetic Algorithm (GA), were compared with LSTM-ARO using MSE, MAE, MAPE, and R2 evaluation criteria. The results indicate that LSTM-ARO outperformed the other models. Budiharto (Budiharto) experimented with LSTM and found it to be a reliable predictor for short-term data with an accuracy of 94.57%. Using a shorter training period of 1 year with high epochs produced better results than using three-year training data. Rather (2021) implemented a new regression scheme on an LSTM-based deep neural network to construct a predicted portfolio. The author conducted a large set of experiments using stock data of NIFTY-50 obtained from the National Stock Exchange of India. The results indicated that the proposed model outperformed various standard predictive and portfolio optimization models.

Bi-LSTM has an additional LSTM layer that processes the input data in the reverse order. Lee et al. (2022) proposed and applied an attention-based BiLSTM (AttBiLSTM) model to trading strategy design. The model is evaluated with various technical indicators (TIs), including a stochastic oscillator, RSI, BIAS, W%R, and MACD. Two trading strategies suitable for deep neural networks (DNNs) are also proposed and verified for effectiveness. The study introduces five well-known TIs and demonstrates the highest accuracy of 68.83% in predicting stock trends. Additionally, exporting the probability of the deep model to the trading strategy is introduced, resulting in the highest return on investment of 42.74% on the back test of TPE0050. GRU has fewer parameters to train compared to LSTM. Hamayel and Owda (2021) proposed and compared three recurrent neural networks (RNN) models for predicting cryptocurrency prices: LSTM, bi-LSTM, and GRU. The models were evaluated using mean absolute percentage error (MAPE). Results showed that the GRU model outperformed LSTM and bi-LSTM for all three cryptocurrencies (Bitcoin, Ethereum, and Litecoin), with the lowest MAPE percentages. The bi-LSTM model had the highest prediction error. Overall, the proposed models showed accurate predictions of cryptocurrency prices.

CNN has been given less emphasis on forecasting stock prices. Generally, CNN is used for computer vision, but recently, it has been applied to stock time series forecasting. Khodaee et al. (2022) used a hybrid model consisting of a Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) was developed to forecast Turning Points (TPs) in stock prices. The model first classified each day in the time series as a TP or Ordinary Point (OP) and used a balancing approach to have a balanced number of TPs and OPs. Technical indicators were then converted into 2D images to consider their relationship, and the Fuzzy C-Means algorithm was applied to segment the inputs and aid training efficiency. A classification hybrid CNN-LSTM-ResNet model was proposed to forecast TPs and OPs, and augmentation techniques, including Residual Networks (ResNet), were employed. The proposed model outperformed other benchmarks with an average accuracy of 60.19% in Dow-30 and 63.62% in ETFs, achieving a profit of up to three times in Dow-30 and up to four times more than the Buy and Hold strategy in ETFs.

2.2. Research gaps

Most research in time series stock forecasting has primarily focused on predicting prices for the next day, leaving a gap in the literature regarding the prediction of long-term prices (Nazareth and Reddy). This short-term emphasis on prediction has created this gap. In this study, the authors addressed this gap by predicting long-term prices. Additionally, most studies have experimented with a single index. However, this study used the daily historical price of the top five global economies' indices and predicted the daily price for the next year at once, providing investors and traders with a long-term market outlook to make informed investment decisions and improve risk management. Furthermore, unlike other studies, this study exhaustively experimented with six deep learning models, namely DNN, RNN, LSTM, Bi-LSTM, GRU, and CNN, to forecast long-term stock prices. This extensive research differentiates this study from other studies and addresses all these gaps in the literature.

3. Deep learning models

3.1. Deep Neural Network (DNN)

ANNs were proposed in the 1940s as the simplest model to mimic how human brains process information and learn from it. However, ANN learning becomes challenging if data increases in size (Awad and Khanna, 2015). Multi-layer Perceptron (MLP) is a variant of feedforward ANN and is the foundation of DNN (Sarker, 2021). This study calls a fully connected MLP having more than or equal to two hidden layers a DNN. DNNs are highly scalable and can handle large and complex datasets with ease. Hinton et al. (2006) first proposed DNNs in 2006. In a DNN, there are three distinct layer types, starting with the input layer, then two or more hidden layers, and concluding with an output layer. The input layer receives and processes the input data, which is then transformed through the subsequent hidden layers via activation functions. Each hidden layer comprises multiple neurons, where the output of each neuron is fully connected to all neurons in the next layer, creating a dense layer. Lastly, the output layer produces the network's final output, which can either be a single value or a vector of values. Fig. 1 shows the architecture of a typical DNN.

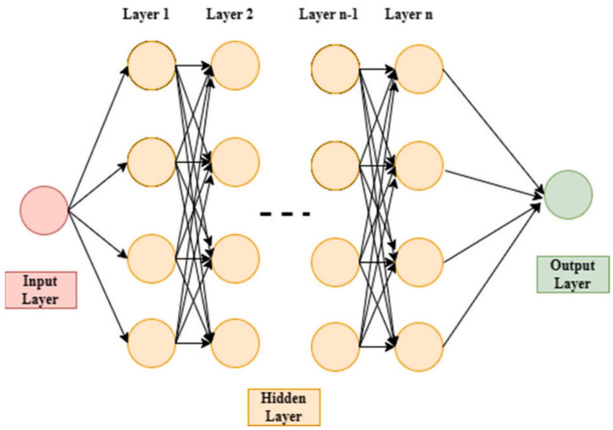

Fig. 1. DNN architecture.

3.2. Recurrent neural network (RNN)

DNNs are feedforward neural networks where information only flows in one direction, from the input layer through the hidden layers to the output layer. Feedforward neural networks encounter difficulties when processing temporally related data, such as time series and managing varying sequences (Tsantekidis et al., 2022). In contrast, RNNs have a feedback mechanism where information can flow back into the network. Hence, RNNs can use a data sequence to build a kind of memory that helps them understand the current and past data in that sequence. Elman (1990) first introduced simple RNN (Zhang and Man, 1998). This paper uses SimpleRNN from TensorFlow, which is fully connected. A fully connected RNN layer means that each neuron is connected to every neuron in the next layer. Fig. 2 shows the architecture of simple RNN.

3.3. Long Short-Term Memory (LSTM)

The problem of the vanishing gradient is a common issue in traditional RNNs. The vanishing gradient prevents learning long-term dependencies (Rehmer and Kroll, 2020), (Hochreiter and Schmidhuber, 1997). To address this issue, Hochreiter and Schmidhuber (1997) introduced LSTM, a type of RNN that overcomes the vanishing gradient problem. LSTMs can handle long-term dependencies much better than RNNs, as they have a more sophisticated memory mechanism that allows them to remember or forget information over time selectively. LSTM has three gates: forget gate, input gate, and output gate. The input gate manages the flow of information into the cell, the forget gate determines which information should be discarded from the cell's previous state, and the output gate manages the release of information from the cell (Wang et al., 2019). Fig. 3 shows the LSTM unit.

3.4. Bi-directional Long Short-Term Memory (Bi-LSTM)

Bidirectional RNN (Bi-RNN) was proposed by Schuster and Paliwal in 1997 (Schuster and Paliwal, 1997). Later, Graves and Schmidhuber proposed Bi-LSTM in 2005 (Graves and Schmidhuber, 2005). LSTMs utilize past information to make predictions. However, LSTMs do not consider future information in their predictions, which may limit their ability to capture certain patterns in the data. Bi-LSTM can obtain not only past context information but also future context information (Wang et al.). LSTMs utilize past information to make predictions. However, LSTMs do not consider future information in their predictions, which may limit their ability to capture certain patterns in the data. Fig. 4 shows the architecture of Bi-LSTM.

3.5. Gated Recurrent Unit (GRU)

GRU was proposed by Kyunghyun Cho in 2014 (Cho et al., 2014). A GRU is a subtype of RNN that shares some similarities with LSTM. However, GRU is characterized by only two gates: a reset gate and an update gate. Like LSTM, GRU also largely tackles the vanishing gradient problem (Shen et al., 2018). Further, due to its simpler structure, GRU trains faster (Samsani et al.) and is computationally more efficient (Xia et al., 2022) than LSTM. There are two gates in the GRU: the reset gate and the update gate. The reset gate is crucial in determining how much the previous hidden state should be discarded, and the new input should be included in the current hidden state. The update gate, conversely, determines the extent to which the previous hidden state should be retained and how much of the new input should be added to the hidden state. Fig. 5 shows the architecture of the GRU unit.

3.6. Convolution neural network (CNN)

CNN in its primitive form was first introduced by Yann LeCun (LeCun et al., 1998), (Bhatt et al., 2021) inspired by the work of Kunihiko Fukushima (1980). However, the focus on CNN increased after introducing a deep convolution neural network, also known as Alexnet

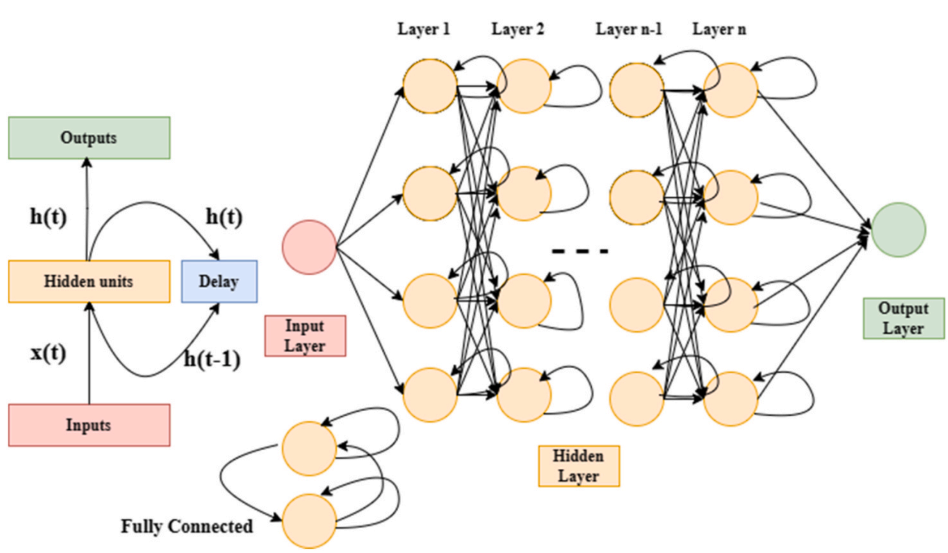

Fig. 2. RNN architecture.

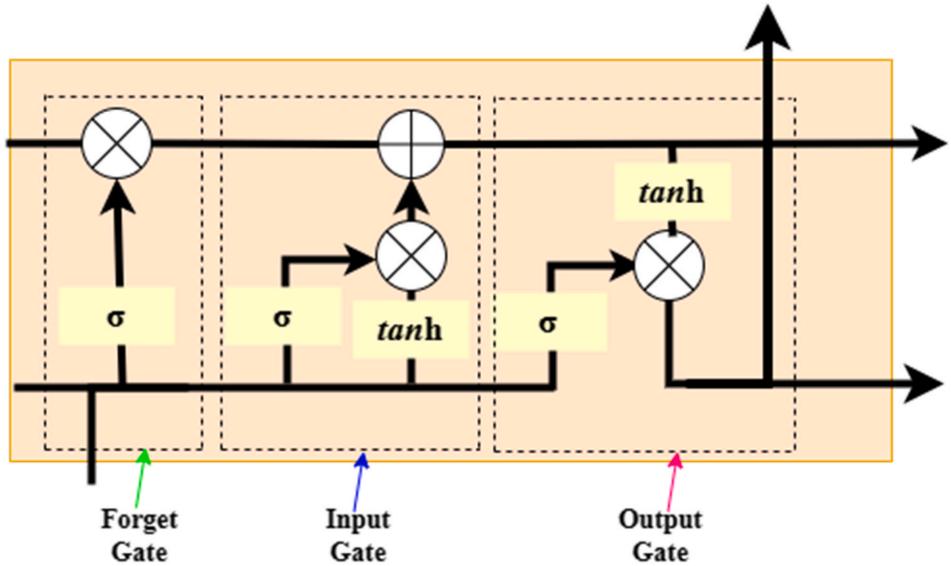

Fig. 3. LSTM unit.

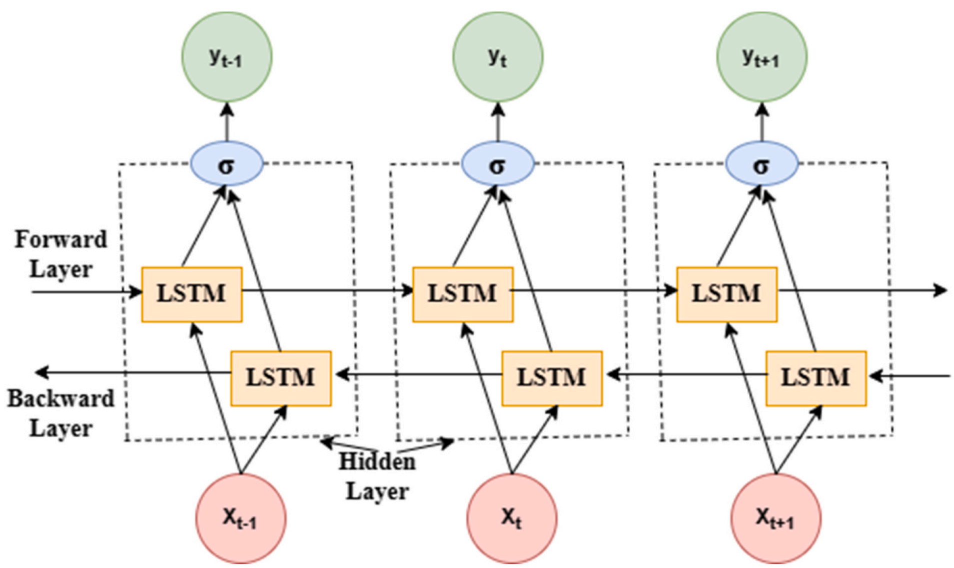

Fig. 4. Bi-LSTM architecture.

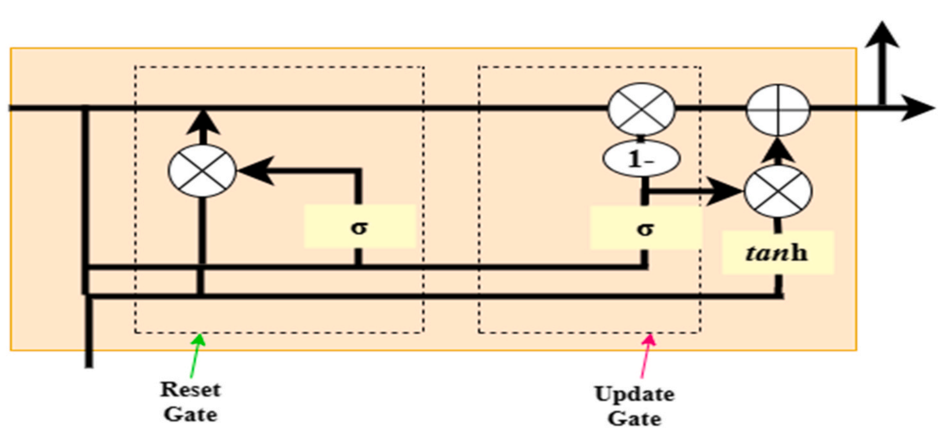

Fig. 5. GRU unit.

(Hinton et al., 2012), (Krizhevsky et al., 2017).

Fig. 6 shows the architecture of a one-dimensional CNN. In 1D-CNN, the convolution kernel is an array of one dimension. As shown in the figure, a typical CNN architecture comprises several layers, including the input, convolutional, pooling, fully connected, and output layers. Although CNN is used for image recognition, it also can process time series data (Wang et al., 2021).

4. Methodologies

4.1. Prediction method

In the study, all deep learning models carry the same configuration, so the comparison can be unbiased. Fig. 7 shows the deep learning layers architecture. The first layer of each model is the input layer. Generally, data is normalized to feed in the model (Ghaderzadeh et al., 2022)–(Gheisari et al., 2023). The dates are converted into an integer from 1 to n. Along with integer dates, this layer also has an input of a transformed array of scaled close prices using the min-max scaler formula in Eq. 1

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

Assuming X is the feature matrix, X_{min} and X_{max} correspond to its minimum and maximum values, respectively. The input layer gives output to the model's first layer. The first layer of the model is the corresponding fully connected neuron cell such as DNN, RNN, LSTM, Bi-LSTM, GRU, or CNN. There are 256 fully connected neuron units. This layer has a ReLU activation function. Deep learning models sometimes can overfit the training data. Hence, it is important to design a prediction method that tackles the problem of overfitting the training data. A dropout layer is added after the first layer with 256 units to handle the overfitting in training data. The dropout rate is kept at 20% in the dropout layer for all models.

A second layer of deep learning units is added to make the learning deep. The second layer also has 256 units. This layer also has a ReLU activation function. A second dropout layer is again stacked to the deep learning layer with the same 20% dropout rate and 256 units. Finally, a dense layer with a single neuron is added with a sigmoid activation function. The data is input into the models with a batch size of 32. The models are compiled and trained using Adam optimizer and mean squared error as a loss function. For all models, 100 epochs are used to train them. The output from the model is inversed, transformed, and compared with the original prices. The final evaluation of the models is done based on RMSE and MAPE. The parameters of the algo are given in Table 1.

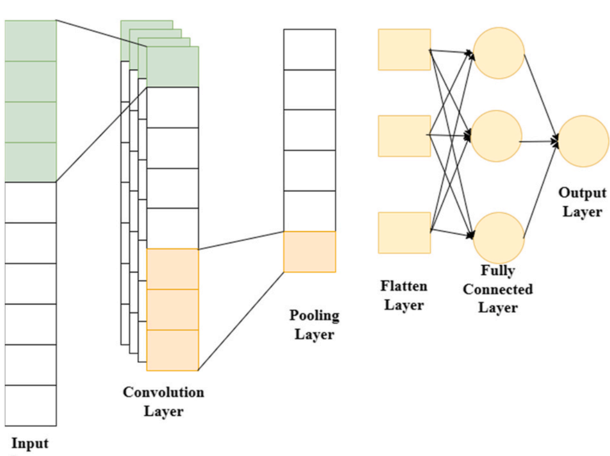

Fig. 6. 1D-CNN architecture.

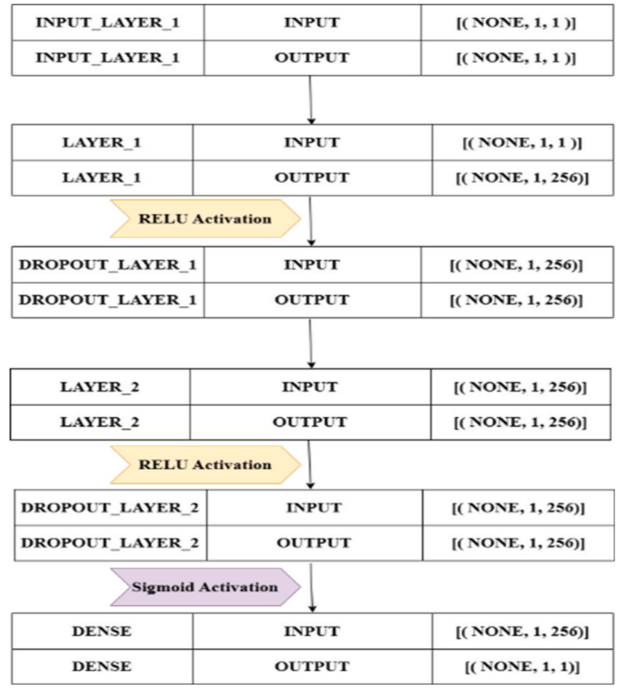

Fig. 7. Deep learning model layers architecture.

Table 1
Parameters.

Parameter	Value
Loss	mse
Optimizer	adam
Metrics	MeanSquaredError
Epochs	100
Batch Size	32

4.2. The experimental setup

In this research, an investigation is conducted using five prominent global indices, which are the Nifty, the Dow Jones Industrial Average (DJIA), the DAX performance Index (DAX), the Nikkei 225 (NI225), and the Shanghai Stock Exchange composite index (SSE). These indices belong to India, the USA, Germany, Japan, and China. The countries are the top five economies of the world. As the stock market is reflected in a country's economy, these indices from the diverse economies can help test the robustness of the deep learning model. The selection of the top five GDPs was based on the economic significance, the diversity of economies, and the global impact of these economies. Using multiple indices tests the robustness of the deep learning models across different markets. This comparative approach enables the study to identify trends, patterns, and variations in the deep learning model effectiveness across diverse economic landscapes. The Data is collected from Yahoo Finance for around ten years, from January 1st, 2013, to February 28th, 2023. Such a long period helps the model train on different market phases, such as bull, bear, and stagnant. The last year's data is kept for testing purposes, while the rest is utilized for data training. Upon completion of training, the trained models make predictions on future prices using dates from the testing data. The prices obtained from the testing phase are inverse-transformed and compared to the predicted values. The efficacy of the models is evaluated using RMSE on both training and testing data. MAPE is used to compare the results obtained from different indices because the value of indices is different from market to market. Fig. 8 shows the overall experimental design.

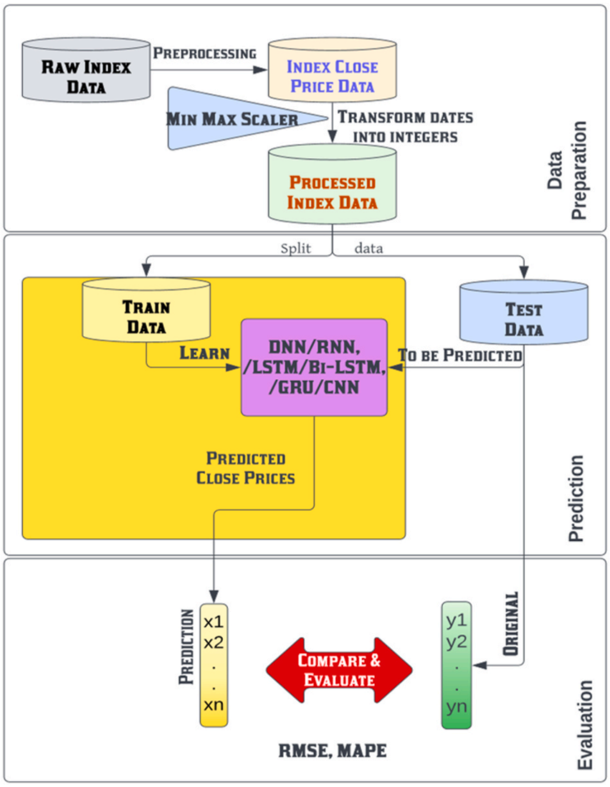

Fig. 8. Experimental design.

5. Results and discussion

5.1. Nifty

Table 2 shows the results of all models on the Nifty index. In the training phase, CNN performed the best, and LSTM performed the worst. Further, GRU is the second best in terms of RMSE and MAPE. However, in the testing phase, LSTM performed the best, and CNN was the second worst in performance. This indicates that fitting well in the training phase does not guarantee better performance in the testing phase. Bi-LSTM also performed better than LSTM in the training phase but not in the testing phase. Fig. 9 shows visually the price pattern and prediction in the training and testing phase. The gray line separates the training and testing periods.

5.2. The Dow Jones Industrial Average (DJIA)

Table 3 shows the results of all models on the DJIA index. The performance of CNN in the training phase is the best in the DJIA index, similar to that of CNN on Nifty. DNN is the second best in terms of training RMSE and training MAPE. GRU performed worst in the training period, and LSTM was the second worst. However, the scenario changes in the testing phase. The LSTM model again performed the best, while the CNN performed the worst. Fig. 10 shows the performance of deep learning models during the training and testing period. The MAPE values are higher in DJIA compared to the Nifty index.

Table 2
Nifty.

Model	Train RMSE	Test RMSE	Train MAPE	Test MAPE
DNN	503.34	1255.32	3.52	6.2
RNN	510.74	1353.29	3.56	6.8
LSTM	581.37	881.38	4.51	3.95
Bi-LSTM	576.12	1064.63	4.46	4.96
GRU	506.26	1224.16	3.6	5.98
CNN	499.28	1326.14	3.49	6.64

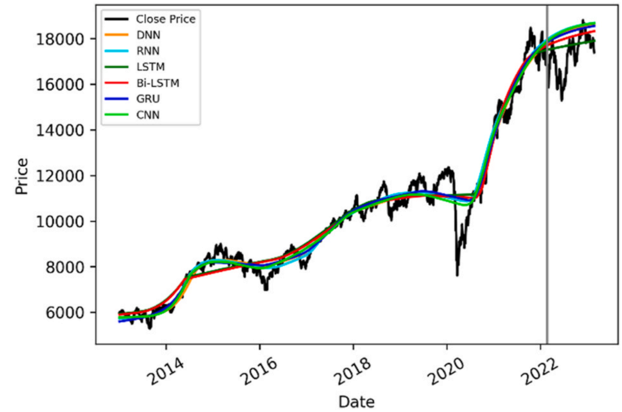

Fig. 9. Nifty predictions.

Table 3
DJIA.

Model	Train RMSE	Test RMSE	Train MAPE	Test MAPE
DNN	957.45	4027.82	2.76	11.67
RNN	1011.53	3953.16	3.26	11.42
LSTM	1041.41	3736.37	3.1	10.71
Bi-LSTM	991.63	3928.81	2.82	11.34
GRU	1085.22	3942.12	3.43	11.39
CNN	937.4	4145.69	2.69	12.05

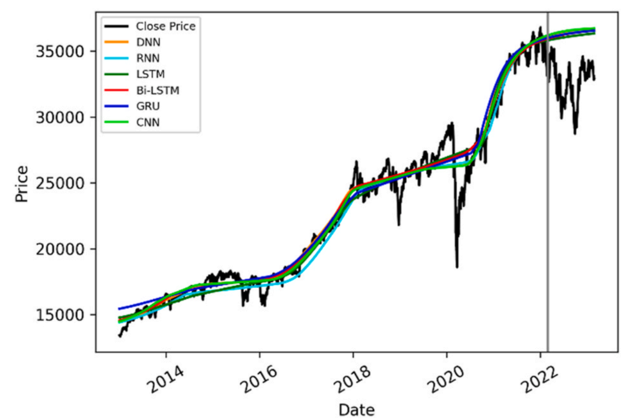

Fig. 10. DJIA predictions.

5.3. DAX performance index (DAX)

Table 4 shows the performance of deep learning models on the DAX performance index. Notably, predictions show big RMSE and MAPE values. This could be attributed to a sharp fall during the predicted period. In contrast to the performance of CNN on the Nifty and DJIA, the

Table 4
DAX.

Model	Train RMSE	Test RMSE	Train MAPE	Test MAPE
DNN	596.34	2453.73	3.71	17.06
RNN	601.07	2479.58	3.67	17.26
LSTM	684.68	2246.89	4.53	15.48
Bi-LSTM	681.42	2473.32	4.52	17.21
GRU	679.04	2414.38	4.45	16.76
CNN	598.2	2421.44	3.74	16.82

performance on CNN is not the best during the training phase. The DNN model performed slightly better than the CNN model. Following a similar pattern in the previous two indices, the LSTM model performed the worst regarding the training RMSE and MAPE. However, the LSTM is the best-performing model during testing in DAX. The MAPE values are higher in DAX compared to both Nifty and DJIA. Fig. 11 visualizes the pattern of prices both during the training and testing phase.

5.4. Nikkei 225 (NI225)

Table 5 reports the results of the six models on Nikkei 225. The results are similar to previous experiments. The CNN model shows superiority during the training phase, and the GRU model outperforms other models during the testing period. The MAPE values are smaller than DJIA and DAX but bigger than Nifty. Fig. 12 shows the chart of the original close price and fit and prediction prices during the testing and training period, respectively.

5.5. Shanghai Stock Exchange composite index (SSE)

Table 6 shows the performance of the models on the SSE index. RNN fits the best on training data of the SSE index, and LSTM fits the worst. The Bi-LSTM models performed the best on testing data. The LSTM model is in second place. The CNN models again performed poorly on the testing data. Fig. 13 visualizes the results of all models. It can be seen that the CNN predicted prices are farthest from testing close prices.

5.6. Discussion

In all models, the CNN model tends to overfit the training prices and underfit the testing prices. In contrast, the LSTM model tends to underfit the training price but performs the best on the testing data. Table 7 shows the consolidated result. To compare the models between the five indices, RMSE cannot be used as the scales of close prices are different for each index. Hence, MAPE is appropriate for comparison. Except for LSTM, the Bi-LSTM is superior to all other models. This can be associated with the fact that LSTM and Bi-LSTM have a lot of similarities in architecture. The performance of GRU and DNN is the same on testing data. However, DNN fits better on the training data compared to the GRU. GRU also has a gated mechanism like LSTM and Bi-LSTM, but it cannot outperform these models to predict long-term prices (see Table 8).

Table 7 shows the best and worst performers during the training and testing phase. RNN is the second worst performed, possibly due to a vanishing gradient problem. CNN performed the poorest compared to other models, and this might be because CNN is designed to work with computer vision and images. It can be used for time series analysis, but with similar deep learning architecture, it does not predict long-term prices well.

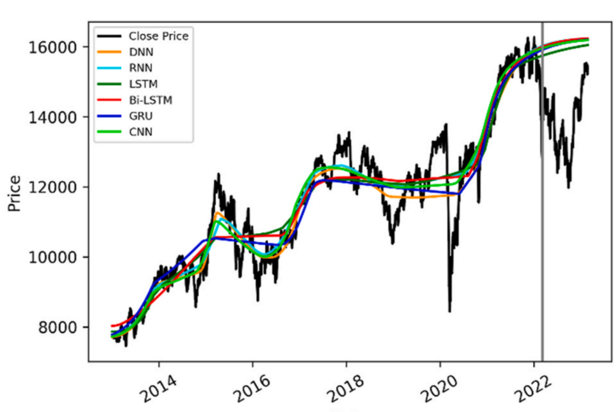

Fig. 11. DAX predictions.

Table 5
Nikkei 225.

Model	Train RMSE	Test RMSE	Train MAPE	Test MAPE
DNN	1016.41	3013.85	3.91	10.8
RNN	996.18	2932.33	3.89	10.49
LSTM	1421.02	3080.97	5.88	11.06
Bi-LSTM	1259.34	2908.46	5.14	10.4
GRU	1058.77	2610.49	4.23	9.25
CNN	990.39	2989.09	3.8	10.71

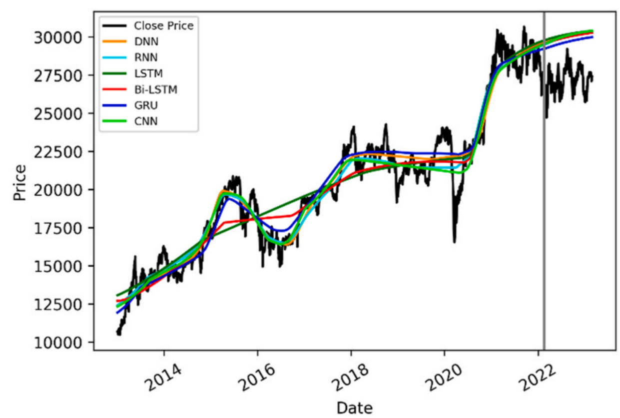

Fig. 12. NI225 predictions.

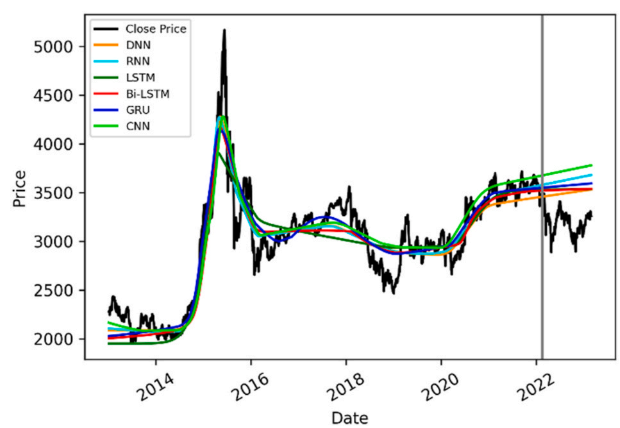

Fig. 13. NI225 predictions.

Table 6
SSE.

Model	Train RMSE	Test RMSE	Train MAPE	Test MAPE
DNN	188.75	324.18	4.8	9.5
RNN	183.04	452.4	4.49	13.7
LSTM	234.99	357.77	5.8	10.68
Bi-LSTM	198.46	354.77	4.89	10.57
GRU	184.49	395.5	4.49	11.89
CNN	189.66	546.76	4.71	16.77

5.7. Individual model assessment

Deep Neural Network: DNNs are characterized by their fully connected layers and may not perform the best in long-term stock price prediction because they lack temporal memory like RNN, LSTM, or Bi-LSTM. The historical stock prices may hold future information for

Table 7
Consolidated.

Model	Train MAPE	Test MAPE
DNN	3.74	11.05
RNN	3.77	11.93
LSTM	4.76	10.38
Bi-LSTM	4.37	10.90
GRU	4.04	11.05
CNN	3.69	12.60

Table 8
Overview of models' results.

	Training		Testing	
	The Best	The Worst	The Best	The Worst
Nifty	CNN	LSTM	LSTM	RNN
DJIA	CNN	GRU	LSTM	CNN
DAX	CNN	LSTM	LSTM	RNN
NI225	CNN	LSTM	GRU	LSTM
SSE	RNN	LSTM	Bi-LSTM	CNN

understanding patterns and trends, which DNNs might struggle to capture effectively.

Recurrent Neural Network: RNNs, although capable of modeling sequential data, are prone to the vanishing gradient problem. This issue can hinder their ability to capture long-term dependencies in stock price data, resulting in suboptimal performance in long-term forecasting compared to LSTM or Bi-LSTM.

Long Short-Term Memory: The performance of LSTM was superior to other models. The gating mechanism of LSTM helps mitigate the vanishing gradient problem, ensuring that the LSTM cell retains its past information to forecast future prices. LSTM's excellent performance can be attributed to its inherent ability to capture temporal dependencies within the data. LSTM's superior performance indicates that stock prices are influenced by their historical values, and LSTM's capacity to retain and utilize this historical context contributes to its accurate predictions.

Bidirectional Long Short-Term Memory: The relative lack of improvement seen with Bi-LSTM suggests that, in financial time series forecasting, future outputs backpropagation may not impact the accuracy of predictions. Bidirectional data processing may be more beneficial in scenarios where future values have substantial influence over current predictions, which is not necessarily true in stock markets.

Gated Recurrent Unit: GRU, while training faster than LSTM, may not necessarily outperform LSTM in long-term stock price prediction. The two gates mechanism, although useful in terms of speed of processing, might not fully compensate for the vanishing gradient problem and the need to capture long-term dependencies like LSTM.

One-Dimensional Convolutional Neural Network: The comparatively poor performance of 1D-CNN can be attributed to its design, which was originally tailored for image analysis and not for time series analysis. While it can capture local patterns in the dataset, it may not effectively model the intricate temporal relationships in financial time series data. Further, the presence of noise in financial time series needs a forget mechanism which is absent in CNN.

Choosing the best model for long-term stock market prediction depends on the dataset's specific characteristics. In the stock market, temporal sequence holds information about trends and patterns useful for future prediction. LSTM's success in forecasting long-term stock prices can be attributed to its specialized architecture designed to excel in modeling temporal sequences, making it a valuable tool for investors seeking accurate long-term forecasts in the stock market.

5.8. Managerial insights

Based on the analysis, the study found that the LSTM model is superior to other models in predicting long-term stock prices for the five indices considered. The LSTM model may perform poorly on training data compared to other models. Still, it performed well on the testing data, indicating that it is an appropriate model to predict long-term stock prices of global indices. However, the CNN model tends to overfit the training data and underfit the testing data, making it unsuitable for predicting long-term stock prices. Furthermore, the study found that the performance of the models varies across the different indices, indicating that the stock market trends in different regions are not identical. Additionally, the study highlights the importance of selecting the appropriate evaluation metrics when comparing the performance of the models. In this case, MAPE was used because it accounts for the difference in scale among the indices. Overall, the findings of this study could be useful for investors and traders who rely on stock price predictions to make informed investment and risk mitigation decisions. The LSTM model, in particular, could be a valuable tool for predicting long-term stock prices, providing a competitive edge in the stock market.

6. Conclusion

Most studies focus on predicting next-day stock prices or movements, limiting the usability of the predictive model for investors. Long-term stock price forecasting is more useful to investors, but the literature has a big gap. This study is a landmark study to address this gap using deep learning models. we extensively explore the ability of deep learning models to predict the daily prices of global stock indices over a long term, up to a year. The performance of six models, including DNN, RNN, LSTM, Bi-LSTM, GRU, and CNN, are compared using RMSE and MAPE. The models predict the long-term daily prices of five global stock indices: the Nifty, DJIA, DAX, NI225, and SSE.

The LSTM performed the best on Nifty, DJIA, and DAX. Overall, the MAPE on all indices demonstrates that LSTM is most appropriate to predict long-term stock prices. The Bi-LSTM also performed better than other models except LSTM. However, the input processing by Bi-LSTM in reverse order does not improve the prediction over LSTM. Hence, Bi-LSTM may not improve performance forecasting of stock prices. CNN overfits the training data and performs poorly on testing data and, hence, least appropriate to predict long-term stock prices. Similarly, RNN also performed poorly on testing data. Literature suggests that GRU is computationally efficient, but its forecasting ability is inferior to LSTM, Bi-LSTM, and DNN based on the training and testing MAPE. This study contributes to long-term stock price prediction by leveraging time-dependent patterns and allows other researchers to develop trading and risk management systems.

The study can be further applied to different data frequencies, such as 1-min, 5-min, 15-min, etc. Additionally, the patterns of predicted daily prices can help build trading and risk management decision systems. The researcher can further optimize the number of neurons, activation function, layers, and epochs using evolutionary algorithms such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), etc. Other models, such as Random Forest, XGBoost, Support Vector Regression, etc., can also be explored. The performance can further be improved using hybrid models, which may combine two or more machine-learning models.

One of the study's limitations is that the architecture of deep learning models is selected manually. Hence, automated Machine Learning (AutoML) might improve results by optimizing the hyperparameters. Apart from dropout, other regularization techniques, such as forward validation and moving window training, could also have enhanced the results. Further, the study has only experimented on stock indices, not individual stocks. Indices tend to go upward in the long term, whereas individual stocks might also go downward in the long term. Hence, it will be interesting to explore the models on individual stocks.

Compliance with ethical standards

This article does not contain any studies with human participants or animals performed by any of the authors.

CRediT authorship contribution statement

Mohit Beniwal: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Archana Singh:** Writing – review & editing, Supervision. **Nand Kumar:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Alkhatib, K., Khazaleh, H., Alkhazaleh, H.A., Alsoud, A.R., Abualigah, L., 2022. A new stock price forecasting method using active deep learning approach. Journal of Open Innovation: Technology, Market, and Complexity 8 (2). https://doi.org/10.3390/joitmc8020096.

Awad, M., Khanna, R., 2015. Deep neural networks. In: Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers. Apress.

Beniwal, M., Singh, A., Kumar, N., 2023a. Predicting next quarter nifty 50 price using genetic algorithm and support vector regression. In: 2023 2nd International Conference on Applied Artificial Intelligence and Computing. ICAAIC), pp. 631–635.

Beniwal, M., Singh, A., Kumar, N., 2023b. Forecasting long-term stock prices of global indices: a forward-validating Genetic Algorithm optimization approach for Support Vector Regression. Appl. Soft Comput. 110566.

Bhandari, H.N., Rimal, B., Pokhrel, N.R., Rimal, R., Dahal, K.R., Khatri, R.K.C., 2022. Predicting stock market index using LSTM. Machine Learning with Applications 9, 100320. https://doi.org/10.1016/j.mlwa.2022.100320.

Bhatt, D., et al., 2021. CNN variants for computer vision: history, architecture, application, challenges and future scope. MDPI Electronics 10 (20). https://doi.org/10.3390/electronics10202470.

Budiharto, W., 2021. Data science approach to stock prices forecasting in Indonesia during Covid-19 using Long Short-Term Memory (LSTM). J Big Data 8 (1). https://doi.org/10.1186/s40537-021-00430-0.

Cho, K., et al., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 [Online]. Available: http://arxiv.org/abs/1406.1078.

Elman, J.L., 1990. Finding structure in time. Cognit. Sci. 14 (2), 179–211. https://doi.org/10.1207/s15516709cog1402_1.

Fama, E.F., 1970. Efficient capital markets: a review of theory and empirical work. J. Finance 25 (2), 383–417 [Online]. Available:

Fama, E.F., 1995. Random walks in stock market prices. J. Finance 75–80.

Fukushima, K., 1980. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. 36, 202.

Ghaderzadeh, M., Aria, M., Hosseini, A., Asadi, F., Bashash, D., Abolghasemi, H., 2022. A fast and efficient CNN model for B-ALL diagnosis and its subtypes classification using peripheral blood smear images. Int. J. Intell. Syst. 37 (8), 5113–5133.

Gheisari, M., et al., 2023. Deep Learning: Applications, Architectures, Models, Tools, and Frameworks: A Comprehensive Survey. CAAI Trans Intell Technol.

Graves, A., Schmidhuber, J., 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. In: Neural Networks. Jul, pp. 602–610. https://doi.org/10.1016/j.neunet.2005.06.042.

Gülmez, B., 2023. Stock price prediction with optimized deep LSTM network with Artificial Rabbits Optimization Algorithm. Expert Syst. Appl., 120346 https://doi.org/10.1016/j.eswa.2023.120346.

Hamayel, M.J., Owda, A.Y., 2021. A novel cryptocurrency price prediction model using GRU, LSTM and bi-LSTM machine learning algorithms. AIComput Appl 2 (4), 477–496. https://doi.org/10.3390/ai2040030.

Hinton, G.E., Osindero, S., Teh, Y.-W., 2006. A fast learning algorithm for deep belief nets. Neural Comput. 18 (7), 1527–1554.

Hinton, G.E., Krizhevsky, A., Sutskever, I., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25, 1, 1106–1114.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8).

Hoseinzade, E., Haratizadeh, S., 2019. CNNpred: CNN-based stock market prediction using a diverse set of variables. Expert Syst. Appl. 129, 273–285. https://doi.org/10.1016/j.eswa.2019.03.029.

Houssein, E.H., Dirar, M., Abualigah, L., Mohamed, W.M., 2022. An efficient equilibrium optimizer with support vector regression for stock market prediction. Neural Comput. Appl. 34 (4), 3165–3200. https://doi.org/10.1007/s00521-021-06580-9.

W. Jiang, "Applications of Deep Learning in Stock Market Prediction: Recent Progress," *Expert Systems With Applications*, vol. vol. 184. Elsevier Ltd, Dec. 01, 2021. doi: 10.1016/j.eswa.2021.115537.

W. Jiang, "Applications of Deep Learning in Stock Market Prediction: Recent Progress," *Expert Systems With Applications*, vol. vol. 184. Elsevier Ltd, Dec. 01, 2021. doi: 10.1016/j.eswa.2021.115537.

Kanwal, A., Lau, M.F., Ng, S.P.H., Sim, K.Y., Chandrasekaran, S., 2022. BiCuDNNLSTM-1dCNN — a hybrid deep learning-based predictive model for stock price prediction. Expert Syst. Appl. 202 https://doi.org/10.1016/j.eswa.2022.117123.

Khodaee, P., Esfahanipour, A., Mehtari Taheri, H., 2022. Forecasting turning points in stock price by applying a novel hybrid CNN-LSTM-ResNet model fed by 2D segmented images. Eng. Appl. Artif. Intell. 116 https://doi.org/10.1016/j.engappai.2022.105464.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM 60 (6), 84–90. https://doi.org/10.1145/3065386.

Kumar, K., Haider, M.T.U., 2021. Enhanced prediction of intra-day stock market using metaheuristic optimization on RNN–LSTM network. New Generat. Comput. 39 (1), 231–272. https://doi.org/10.1007/s00354-020-00104-0.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2323. https://doi.org/10.1109/5.726791.

Lecun, Y., Bengio, Y., Hinton, G., 2015, 436–444, May 27. Deep Learning," *Nature*, vol. 521. Nature Publishing Group, p. 7553. https://doi.org/10.1038/nature14539.

Lee, M.C., Chang, J.W., Yeh, S.C., Chia, T.L., Liao, J.S., Chen, X.M., 2022. Applying attention-based BiLSTM and technical indicators in the design and performance analysis of stock trading strategies. Neural Comput. Appl. 34 (16), 13267–13279. https://doi.org/10.1007/s00521-021-06828-4.

Liang, H., Sun, X., Sun, Y., Gao, Y., 2017. Text feature extraction based on deep learning: a review. Dec. 01, 2017. In: Eurasip Journal on Wireless Communications and Networking, vol. 1. Springer International Publishing. https://doi.org/10.1186/s13638-017-0993-1.

Lin, Y., Yan, Y., Xu, J., Liao, Y., Ma, F., 2021. Forecasting stock index price using the CEEMDAN-LSTM model. N. Am. J. Econ. Finance 57. https://doi.org/10.1016/j.najef.2021.101421.

Liu, Y., Gong, C., Yang, L., Chen, Y., 2020. DSTP-RNN: a dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction. Expert Syst. Appl. 143 https://doi.org/10.1016/j.eswa.2019.113082.

Naik, N., Mohan, B.R., 2019. Study of stock return predictions using recurrent neural networks with LSTM. In: Communications in Computer and Information Science. Springer Verlag, pp. 453–459. https://doi.org/10.1007/978-3-030-20257-6_39.

N. Nazareth and Y. Y. R. Reddy, "Financial applications of machine learning: a literature review," Expert Systems with Applications, vol. vol. 219. Elsevier Ltd, Jun.01, 2023. doi: 10.1016/j.eswa.2023.119640.

Nikou, M., Mansourfar, G., Bagherzadeh, J., 2019. Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms. Intell. Syst. Account. Finance Manag. 26 (4), 164–174. https://doi.org/10.1002/isaf.1459.

Ranjan, A., Mahadani, Kumar, A., 2022. Stock price prediction using deep learning-based univariate and multivariate LSTM and RNN. In: Sikdar Biplab, R.A., Prasad, Maity (Eds.), Proceedings of the 3rd International Conference on Communication, Devices and Computing. Springer Singapore, Singapore, pp. 95–103.

Rather, A.M., 2021. LSTM-Based deep learning model for stock prediction and predictive optimization model. EURO Journal on Decision Processes 9 (Jan). https://doi.org/10.1016/j.ejdp.2021.100001.

Rehmer, A., Kroll, A., 2020. On the vanishing and exploding gradient problem in gated recurrent units. In: IFAC-PapersOnLine. Elsevier B.V., pp. 1243–1248. https://doi.org/10.1016/j.ifacol.2020.12.1342

Rouf, N., et al., 2021. Stock market prediction using machine learning techniques: a decade survey on methodologies, recent developments, and future directions. Electronics 10 (21). https://doi.org/10.3390/electronics10212717. MDPI, Nov. 01.

S. S. Samsani, H. Mutahira, and M. S. Muhammad, "Memory-based crowd-aware robot navigation using deep reinforcement learning," Complex and Intelligent Systems, Apr. 2022, doi: 10.1007/s40747-022-00906-3.

Sarker, I.H., 2021. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. Springer, Nov. 01 SN Comput. Sci. 2 (6). https://doi.org/10.1007/s42979-021-00815-1.

Schuster, M., Paliwal, K.K., 1997. Bidirectional recurrent neural networks. IEEE Trans. Signal Process. 45 (11).

J. Shah, D. Vaidya, and M. Shah, "A comprehensive review on multiple hybrid deep learning approaches for stock prediction," Intelligent Systems with Applications, vol. vol. 16. Elsevier B.V., Nov. 01, 2022. doi: 10.1016/j.iswa.2022.200111.

Shahi, T.B., Shrestha, A., Neupane, A., Guo, W., 2020. Stock price forecasting with deep learning: a comparative study. Mathematics 8 (9). https://doi.org/10.3390/math8091441.

Shen, G., Tan, Q., Zhang, H., Zeng, P., Xu, J., 2018. Deep learning with gated recurrent unit networks for financial sequence predictions. In: Procedia Computer Science. Elsevier B.V., pp. 895–903. https://doi.org/10.1016/j.procs.2018.04.298

Singh, R., Srivastava, S., 2017. Stock prediction using deep learning. Multimed. Tool. Appl. 76 (18), 18569–18584. https://doi.org/10.1007/s11042-016-4159-7.

A. Thakkar and K. Chaudhari, "A comprehensive survey on deep neural networks for stock market: the need, challenges, and future directions," Expert Systems with Applications, vol. vol. 177. Elsevier Ltd, Sep. 01, 2021. doi: 10.1016/j.eswa.2021.114800.

Tsantekidis, A., Passalis, N., Tefas, A., 2022. Chapter 5 - recurrent neural networks. In: Iosifidis, A., Tefas, A. (Eds.), Deep Learning for Robot Perception and Cognition. Academic Press, pp. 101–115. https://doi.org/10.1016/B978-0-32-385787-1.00010-5.

Y. Wang, Q. Chen, M. Ding, and J. Li, "High precision dimensional measurement with convolutional neural network and bi-directional long short-term memory (LSTM)," Sensors, vol. 19, no. 23, Dec. 2019, doi: 10.3390/s19235302.

Wang, S., Wang, X., Wang, S., Wang, D., 2019. Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting. Int. J. Electr. Power Energy Syst. 109, 470–479. https://doi.org/10.1016/j.ijepes.2019.02.022.

Wang, H., Wang, J., Cao, L., Li, Y., Sun, Q., Wang, J., 2021. A stock closing price prediction model based on CNN-BiSLSTM. Complexity 2021. https://doi.org/10.1155/2021/5360828.

Wu, D., Wang, X., Wu, S., 2022. Jointly modeling transfer learning of industrial chain information and deep learning for stock prediction. Expert Syst. Appl. 191 https://doi.org/10.1016/j.eswa.2021.116257.

Xia, H., Huang, K., Liu, Y., 2022. Unexpected interest recommender system with graph neural network. Complex and Intelligent Systems. https://doi.org/10.1007/s40747-022-00849-9.

Yu, P., Yan, X., 2020. Stock price prediction based on deep neural networks. Neural Comput. Appl. 32 (6), 1609–1628. https://doi.org/10.1007/s00521-019-04212-x.

Zhang, J., Man, K.-F., 1998. Time series prediction using RNN in multi-dimension embedding phase space. In: SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218), pp. 1868–1873.

Zhang, S., Luo, J., Wang, S., Liu, F., 2023. Oil price forecasting: a hybrid GRU neural network based on decomposition–reconstruction methods. Expert Syst. Appl. 218 https://doi.org/10.1016/j.eswa.2023.119617.

Countries by GDP," https://www.populationu.com/gen/countries-by-gdp.