



# SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting

Warda M. Shaban<sup>1</sup> · Eman Ashraf<sup>2</sup> · Ahmed Elsaid Slama<sup>3</sup>

Received: 19 April 2023 / Accepted: 20 October 2023 / Published online: 17 November 2023  
© The Author(s) 2023

## Abstract

As the economy has grown rapidly in recent years, more and more people have begun putting their money into the stock market. Thus, predicting trends in the stock market is regarded as a crucial endeavor, and one that has proven to be more fruitful than others. Profitable investments will result in rising stock prices. Investors face significant difficulties making stock market-related predictions due to the lack of movement and noise in the data. In this paper, a new system for predicting stock market prices is introduced, namely stock market prediction based on deep learning (SMP-DL). SMP-DL splits into two stages, which are (i) data preprocessing (DP) and (ii) stock price's prediction (SP<sup>2</sup>). In the first stage, data are preprocessed to obtain cleaned ones through several stages which are detect and reject missing value, feature selection, and data normalization. Then, in the second stage (e.g., SP<sup>2</sup>), the cleaned data will pass through the used predicted model. In SP<sup>2</sup>, long short-term memory (LSTM) combined with bidirectional gated recurrent unit (BiGRU) to predict the closing price of stock market. The obtained results showed that the proposed system perform well when compared to other existing methods. As RMSE, MSE, MAE, and  $R^2$  values are 0.2883, 0.0831, 0.2099, and 0.9948. Moreover, the proposed method was applied using different datasets and it performs well.

**Keywords** Stock prediction · Deep learning · LSTM · Artificial intelligence

## 1 Introduction

It's no secret that the financial market is fascinating to students and scholars of all stripes because it's a dynamic and ever-changing environment where they can practice and hone their skills as traders, analysts, and other professionals. While different investors may approach the market from different angles, for example, by studying

market behavior, identifying influential factors, trading stocks, forecasting the direction of the market, making asset recommendations for portfolio management, etc., a lack of financial literacy and understanding of basic economic principles can have a significant impact on investment returns [1]. In the world, there are numerous stock exchanges that make up the stock market, or equity market. Due to the law of supply and demand, investors and the general public buy and sell shares whose prices are constantly fluctuating [2, 3]. Owning stock or shares in a company gives you ownership in it. A share is sought after at the lowest cost by buyers, while a share is sought after at the highest cost by sellers.

Predicting future events and profiting from them in a risk-free manner is possible, despite the fact that the future itself is unknown and unpredictable. One possibility is using machine learning (ML) and artificial intelligence (AI) to make stock market predictions [4–6]. Despite the volatility of the stock market, it is possible and wise to use AI for forecasting purposes before making any purchases. Artificial neural networks (ANNs), which have been the

✉ Warda M. Shaban  
warda\_mohammed@nilehi.edu.eg

Eman Ashraf  
eman.ashraf@deltauniv.edu.eg

<sup>1</sup> Department of Communication and Electronics Engineering, Nile Higher Institute for Engineering and Technology, Mansoura, Egypt

<sup>2</sup> Department of Electronics and Communications Engineering, Faculty of Engineering, Delta University for Science and Technology, Belqas, Egypt

<sup>3</sup> Leader of AI Candle Team, Nile Higher Institute for Engineering and Technology, Mansoura, Egypt

subject of extensive study, have found applications in fields as varied as pattern recognition, financial securities, and signal processing. Its benefits in terms of regression and classification in stock market forecasting are also well-known. Traditional ANN algorithms may also make inaccurate stock market predictions due to the problem's initial weight easily falling into the local optimal [5].

Financial applications have found that deep learning (DL) works particularly well because it can process massive amounts of data, such as historical market conditions or stock price movements, to identify patterns and determine actions to take based on those conditions [7]. DL is a subset of ML and AI that employs a suite of algorithms to model abstract ideas at multiple depths of representation. It's possible to learn in a supervised, semi-supervised, or unsupervised setting. DL's primary benefits lie in its ability to automatically learn features, multilayer learning of features, precision in results, generalization power, and the ability to recognize new data [4]. Recent studies suggest that researchers are trying to apply DL to the problem of stock prediction. Since speech is a time series data and stock data are also a time series, the idea that this method can be applied to the stock market emerged after its successful application in the speech domain [8].

The main contribution of this paper is summarized in introducing a new system based on DL to predict the stock price. The proposed known as stock market prediction based on deep leaning (SMP-DL). The proposed SMP-DL splits into two parts which are data preprocessing (DP) and stock price's prediction ( $SP^2$ ). In DP, several processes are performed to clean the data which are detect and reject missing value, feature selection, and data normalization. Actually, DP is an important and critical step, because it controls the prediction process. While the data are cleaned and preprocessed, it is ready for the second part of the proposed system (e.g.,  $SP^2$ ). In  $SP^2$ , a new method that combine between long short-term memory (LSTM) and bidirectional gated recurrent unit (BiGRU) was proposed to predict the closing price of stock market. Actually, for time series prediction, LSTM is a popular DL technique in recurrent neural network (RNN). In spite of its effectiveness, LSTM is easy to overfit, and LSTM is sensitive to different random weight initializations and requires more memory to train. On the contrary, BiGRU uses less training parameter and therefore uses less memory and executes fast. Consequently, the proposed prediction model that combines between LSTM model and BiGRU can predict the closing price of stock market in the next day accurately.

The rest of the paper is organized as follows: the second section presents stock market prediction (SMP) applicability. The third section discusses ethics of using SMP-DL. The fourth section presents the previous efforts about stock market prediction methods. The fifth section presents the

proposed SMP-DL. In the sixth section, the experimental results are presented. In the seventh section, conclusions are discussed.

### 1.1 Research approach

In light of this, the following research questions (RQ) are attempted to be answered in this work.

RQ1: What kinds of ML algorithms are used to forecast the stock market?

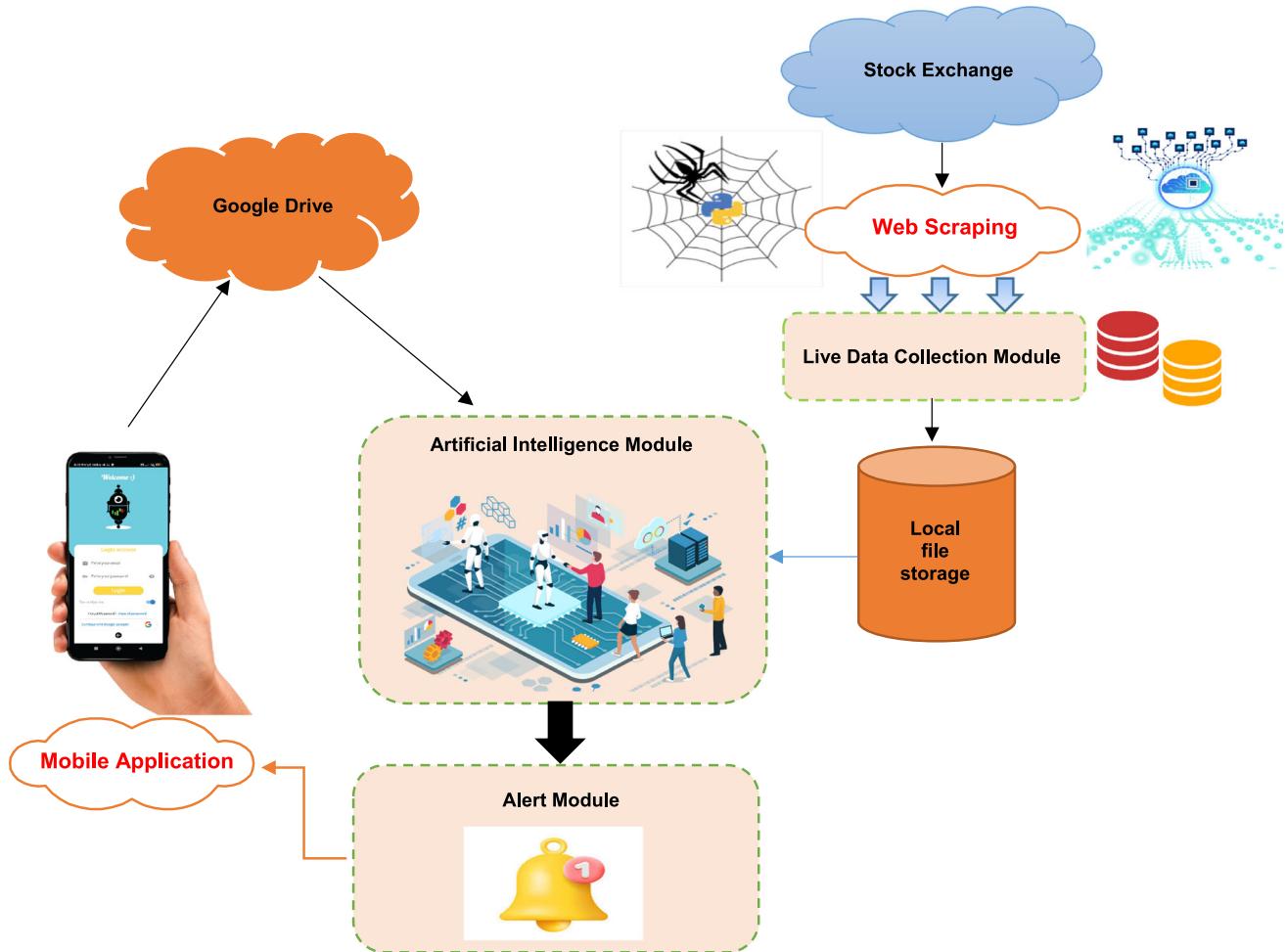
RQ2: Is any hybrid approach of ML model has been used for stock market forecasting or not?

RQ3: To what extent selecting the suitable method is more effective in predicting stock market prices?

## 2 SMP-DL applicability for stock market prediction

Predicting the movement of stock closing prices is a classic problem at the interface of computer science and finance. Actually, there are different ways to predict stock market price using AI [9]. However, there are many problems plaguing these systems such as the accuracy is not effective enough, and the complexity of the systems. In this paper, a new system based on DL is proposed to predict the closing price of stock market. Also, in this paper, a mobile application based on SMP system was designed for anyone to have it on their mobile. The design application called smart trading platform (STP). The main aim of this application is to enable anyone to speculate in the stock exchange in a simple and secure way. As shown in Fig. 1, it mainly contains four modules: live data collection module, URL management module, AI module, and alert module.

In fact, we have provided a real solution to the investor in the real market of the stock exchange to reduce the risk ratio, limit the loss of the money and achieve the highest accuracy. To clarify the applicability of the proposed SMP, if anyone want to speculate in the stock exchange, the designed application (e.g., STP) is faster, cheaper and secure way. In this paper, the proposed system can be used to predict the closing price of the stock market based on DL. Based on that, web scraping (WS) can be used to update the desired company data periodically. The advantages of the designed application are, (i) safe as it is based on efficient DL system which allow to predict the stock price, (ii) cheap as anyone can download the application without any charge, and (iii) SMP is robust, easy to use, versatile, quick, and well-suited to practical situations.



**Fig. 1** The architecture of STP

### 3 Ethics of using SMP-DL

In today's technologically advanced world, where AI is increasingly common, it is crucial to make sure that it is created and used ethically. Transparency, fairness, and algorithmic ethics are all necessary for ethical AI. AI systems must be transparent in order to be responsible and reliable. It speaks to an AI system's capacity to communicate its decision-making procedures in a way that humans can comprehend and interpret. This is particularly important in high-stakes fields like healthcare, finance, and criminal justice, where AI systems' decisions can have a big impact on people's lives and wellbeing. It is therefore essential to make sure that AI is developed and used in these fields in an ethical and responsible manner.

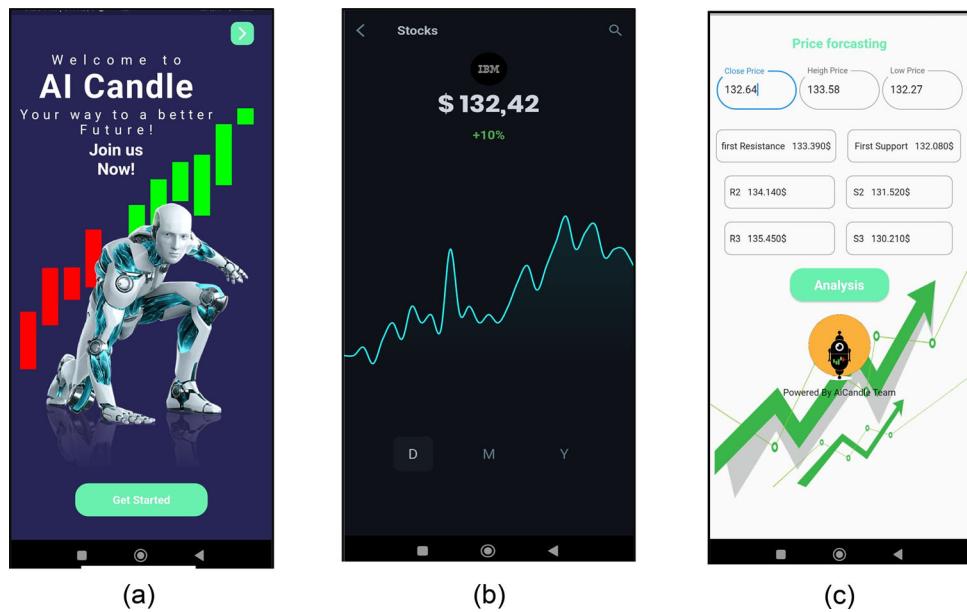
Trading entails controlling risk. Implementing a stop-loss strategy is a crucial aspect of trading. A stop-loss order can automatically sell your stocks if their value falls below a predetermined limit that you have set for them. This strategy enables you to trade with confidence while

minimizing your potential losses. In your trading career, using charts on a web trading platform can be very beneficial. You can effectively track its fluctuations by opening the stock's chart on your mobile device and keeping a close eye on it. This gives you the ability to time your buying and selling well, which may result in significant profits throughout the day.

In our system, until now, there has been no miss prediction. The reason is that our application is designed to perform a financial analysis of the market in addition to AI model. Thus, in our system there are two important decision support factors. Figure 2 shows an example of what the application looks like. However, there are some rules that have to be followed to use the system for successful trading. These rules are:

- (i) Investigate: before purchasing any security, be sure to investigate it and weigh the risks and potential rewards.

**Fig. 2** An example of what the application looks like, **a** the interface of our application, **b** stock line chart, and **c** financial analysis



- (ii) Make a trading strategy: to maintain discipline and steer clear of impulsive trades, establish specific objectives and a trading strategy.
- (iii) Manage your risk: to reduce potential losses, use risk management techniques like stop-loss orders.
- (iv) Keeping emotions in control: emotions can cloud judgment and lead to poor investment decisions. Stay focused and disciplined.
- (v) Stay informed: stay up-to-date with market news and trends to make informed decisions about your investments.

#### 4 Literature review

Because of the importance of stock price prediction, many researchers have invested time and work to studying it. In this section, first and second research questions will be answered. Later, researchers tried using nonlinear models for prediction and developed machine learning techniques like neural networks and support vector machine (SVM), and various other techniques which they successfully used to predict stock price time series [10–14]. A thorough review of the suggested methodologies, including calculation methods, ML algorithms, and significant features is also provided in [12–14] by the authors. The process of choosing studies to conduct is guided by research questions. In order to locate the ML methods and their associated data for forecasting the stock market, these studies were selected. ANN techniques are frequently used to produce precise stock market predictions. The most advanced method of stock market forecasting still has

serious shortcomings after much research. This study supports the notion that stock price forecasting is an integrated process and that it is possible to forecast the stock market more accurately by employing different factors.

In [15], DL algorithms for stock price prediction are implemented. The models proposed are LSTM and convolutional neural network (CNN) with the hill climbing (HC) heuristic. Although the LSTM provides a higher total profit, the results reveal that in the investing simulation, the dual deep net leads to a special part of that profit. When comparing the 25 companies in the survey, CNN has a greater profit in terms of earned market share than its lost. While HC-LSTM has a larger annual return yield, the sharpness ratio is larger due to the benefits of HC-CNN.

Also, adaptive neuro-fuzzy inference system (ANFIS), support vector machine (SVM), and artificial bee colony (ABC) have been introduced in [16]. For example, technical indicators are optimized using ABC to produce the most reliable predictions possible. Following that, ANFIS was utilized to forecast long-term price fluctuations in the stocks. Finally, in order to reduce the predicting errors of the present system, SVM was utilized to construct a relationship between the stock market and a technical indicator. The effectiveness of ABC-ANFIS-SVM as a tool for forecasting stock prices has been demonstrated by computational findings.

A deep learning stock market prediction (DLSMP) has been proposed, as shown in [17]. Stock market group forecasts were the main focus of the proposed strategy. Based on a historical record of ten years' data collection, decision tree (DT), bagging, random forest (RF), adaptive boosting (Adaboost), gradient boosting, and extreme gradient boosting (XGBoost), as well as ANN, RNN, and

LSTM, were used to accomplish this objective. According to experimental findings in [17], LSTM is much more precise. As described in [18], an innovative neural network approach (INNA) has been developed to provide more precise stock market forecasting.

The suggested method makes use of deep LSTM and embedded layer. A stock prediction system (SPS) has also been suggested in [19]. The suggested SPS calls for three steps: (1) technical indicators and news sentiments are used to represent numerical price data in technical analysis and textual news articles in sentiment analysis, respectively, (2) creating a layered DL model to understand the sequential data seen in market snapshot series created by technical indicators and news comments, and (3) creating a fully connected CNN to predict asset values. The results collected demonstrate that the suggested strategy outperforms the baselines for both measures for both the validation and testing phase.

Additionally, A new strategy for predicting stock market closing prices in the following day has been proposed in [20]. CNNs, bidirectional LSTM, and attention mechanism (AM) are the proposed methods (CNN-BiLSTM-AM). CNN-BiLSTM-AM is divided into three stages: CNN was used to extract characteristics from entered data. The obtained attribute data was then learned and forecasted with BiLSTM. Finally, AM can be used to capture the impact of the feature changes of the time series data at varying moments on the prediction outputs. In addition, the authors of [21] use numerous ML models to forecast stock market movements using Spark MLLib and PySpark. Linear regression (LR), decision tree (DT), random forest (RF), and generalized LR are some of the models available. The obtained findings show that generalized linear regression is a more effective model.

As presented in [22], improved particle swam optimization (IPSO) and LSTM, a revolutionary model that has been developed to estimate stock price, are presented. In reality, IPSO was utilized to set the LSTM hyperparameter. The suggested model outperforms support vector regression, LSTM, and PSO-LSTM on the Australian stock market index. The pricing of stocks at the ending of the day was predicted by the authors of [23] using 1D DenseNet and an autoencoder. Lower levels of correlation between the calculated stock technical indicators (STIs) were obtained after they were initially passed into an autoencoder for dimensionality reduction. Both of the STIs and the Yahoo finance data were loaded into the 1D DenseNet. The 1D DenseNet's softmax layer received input from the output features to predict closing stock prices over a range of time horizons.

As shown in [24], a novel forecasting technique based on fuzzy time series and induced ordered weighted averages (IOWA) and weighted averages (WA) has been developed. By establishing a new IOWAWA layer in neural networks to handle difficult nonlinear prediction for a big data set, it first contributes to theory. Using the approach to predict nonlinear financial price data is the second novelty. The results show that the suggested IOWAWA performs well when compared to sixteen other current methods. Additionally, the application of ML and AI in predictive modeling for the financial markets has also been introduced, as illustrated in [25]. The authors of this paper discuss the benefits and drawbacks of utilizing ML to forecast the stock market, in addition to some of the benefits and obstacles of doing so.

As illustrated in [26] a novel framework for recurrent neural network (RNN), long short-term memory (LSTM), gated recurrent unit (GRU), and bi-directional long short-term memory (BiLSTM) models have been proposed. These models are used to predict short and long horizon time series forecasting. The obtained results have been shown that BiLSTM performs well compared to RNN, LSTM, and GRU. As presented in [27], Several ML and DL algorithms have been introduced to predict stock market prices. These algorithms can produce a solid prediction with fewer chances of error. These algorithms are: ANN or deep feed-forward neural network (DFNN) and convolutional neural network (CNN). According to the experimental findings, the CNN model had an accuracy of 98.92% compared to the ANN model with an accuracy of 97.66%.

Also included in [28] is a convolutional extreme learning machine model with kernel support (CKELM). For predicting the stock market, CKELM was suggested. In reality, the CKELM model enhances feature extraction and data categorization by including convolutional and subsampling layers to the KELM's hidden layer. The convolutional layer and the subsampling layer do not employ the gradient technique to modify their parameters as some designs fared better with random weights. The experimental results demonstrate that the proposed model performs admirably, with an accuracy of approximately 98.3%. In the above literature, the proposed stock price prediction techniques advantages and disadvantages are summarized in Table 1. All these previous proposed methods used various techniques to predict stock price in future.

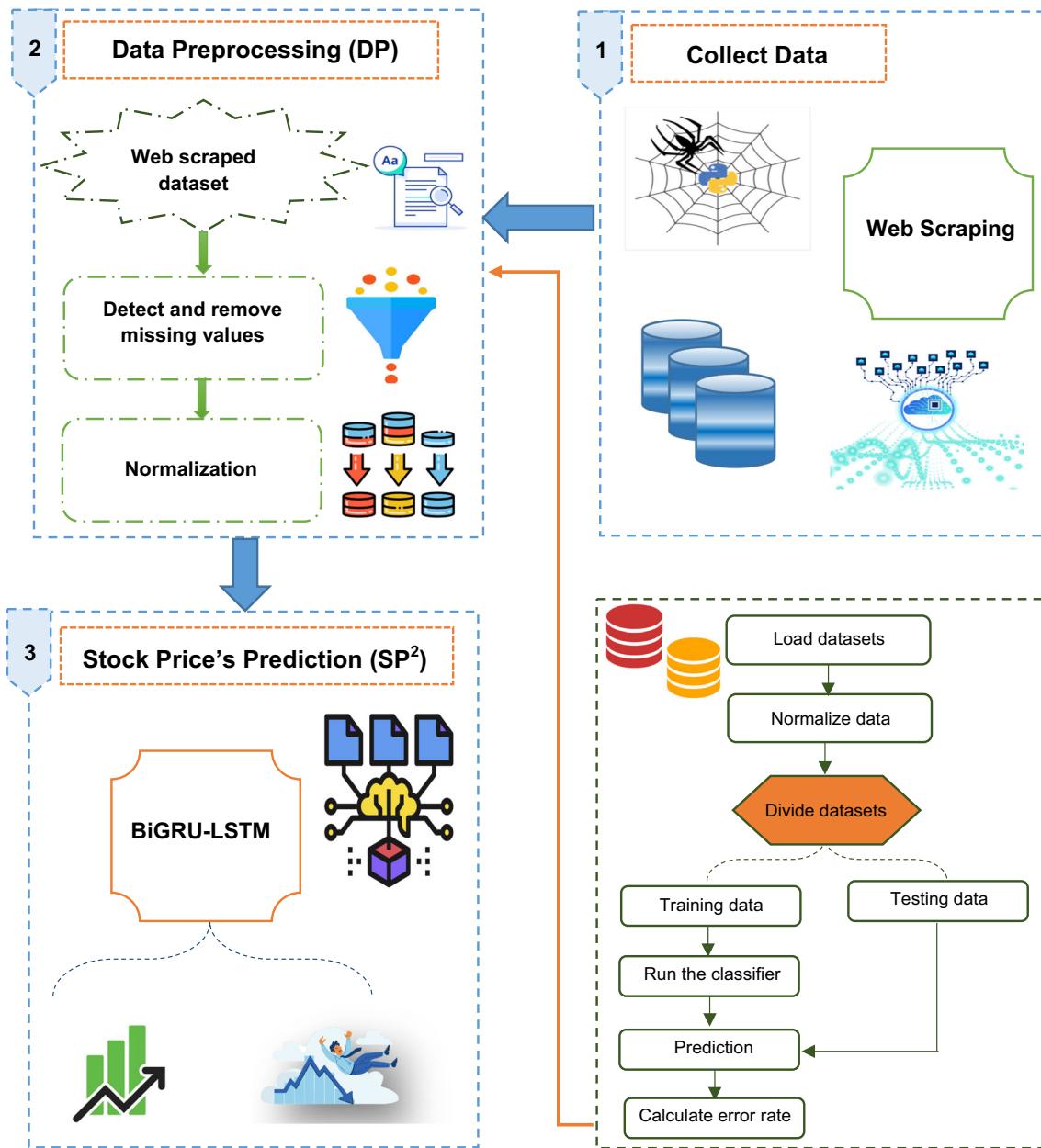
**Table 1** A brief comparison of stock price prediction techniques

Author	Year	Description	Merits	Demerits
Sedighi, et al., [16]	2019	Adaptive neuro-fuzzy inference system, support vector machine, and artificial bee colony (ANFIS-SVM-ABC) was proposed to predict stock prices. Actually, ANFIS was used to forecast long-term price fluctuations in the stocks, ABC was used to optimize technical indicators and to construct a relationship between the stock market and a technical indicator, SVM was used	The advantages of ANFIS-SVM-ABC: Improves precision in computation. Can reach to the most optimal technical indicators	SVM required large training time. ANFIS-SVM-ABC suffers from time complexity
Nabipour et al., [17]	2020	A deep learning stock market prediction (DLSMP) was presented using different DL methods such as decision tree (DT), bagging, random forest (RF), adaptive boosting (Adaboost), gradient boosting, and extreme gradient boosting (XGBoost), in addition to ANN, RNN, and LSTM	LSTM has a powerful ability to predict stock market prices with the lowest error compared to other used DL methods	DLSMP suffers from time complexity
Lu, et al., [20]	2021	A CNN-BiLSTM-AM method was proposed for predicting stock price. At first, to extract characteristics from entered data, CNN was used, while BiLSTM was used to learn and forecast the obtained attribute data. Finally, AM can be used to capture the impact of time series data feature changes at different points on prediction outputs	The proposed CNN-BiLSTM-AM gives the optimal performance to predict stock prices. CNN-BiLSTM-AM is better suited for stock price prediction and provides a reliable way for investors to invest	AM in deep learning suffers from: Difficulty to train the problem of exposure bias
Ji, et al., [22]	2021	A new improved particle swarm optimization with long short-term memory (IPSO-LSTM) was introduced for predicting stock indices. Actually, IPSO was used to optimize LSTM parameters	IPSO has the ability to find better hyperparameters for the LSTM network	IPSO suffers from early convergence in the primary stages
Albahli, et al., [23]	2022	A new DenseNet model with an autoencoder (AEI-DNET). This model uses 1D DenseNet and an autoencoder to predict closing stock prices based on ten years of Yahoo Finance data from ten well-known stocks such as AAPL, MSFT, GOOG, TSLA, etc.	Computationally efficient. processing or prediction time is low	The proposed model doesn't evaluate on an unseen database to show its generalization ability better
Bhambu [26]	2023	A comparative study was presented using A novel framework for RNN, GRU, LSTM, and BiLSTM to predict short and long horizon time series forecasting using five time series datasets	BiLSTM method outperforms other deep neural networks when the hyperparameters are properly tuned	Regarding the disadvantages of BiLSTM compared to other used models, it is a much slower model and requires more time for training
S. Mukherjee, et al., [27]	2023	Two approaches to predict stock market indices and stock prices was proposed which are deep feed-forward neural network (DFNN) and convolutional neural network (CNN)	CNN outperforms DFNN as it requires less training data and less time	For DFNN to give the optimal accuracy, it requires many training data and epochs. Also, it suffers from overfitting problem
V. Agarwal, et al., [28]	2023	A convolutional extreme learning machine model with kernel support (CKELM) was proposed. CKELM was used to forecast stock market using social media data. Actually, CKELM was used to enhance feature extraction as well as data categorization	The proposed CKELM provides the optimal accuracy	More complex

## 5 The proposed stock market prediction based on deep learning (SMP-DL)

Modern technologies have made the stock market more appealing to investors, and forecasting can help with accurate market prediction. The investment and trading of market information directly affects the forecast of market patterns. The stock market forecasting tools can be used to monitor, forecast, and control the market as well as to help with decision-making [29, 30]. To put it another way, stock market-based prediction methodologies significantly contribute to bringing together new stakeholders and existing

investors. Investors are able to make wise choices thanks to accurate stock market forecasts. Consequently, in this paper, a new system has been proposed to predict the stock market closing price using DL algorithms. The proposed system has been known as stock market prediction based on deep learning (SMP-DL) which splits into two parts which are (i) data preprocessing (DP) and (ii) stock price's prediction ( $SP^2$ ) as shown in Fig. 3. In the next subsection, each part of the proposed SMP-DL will be discussed briefly.



**Fig. 3** The proposed SMP-DL

## 5.1 Data preprocessing (DP)

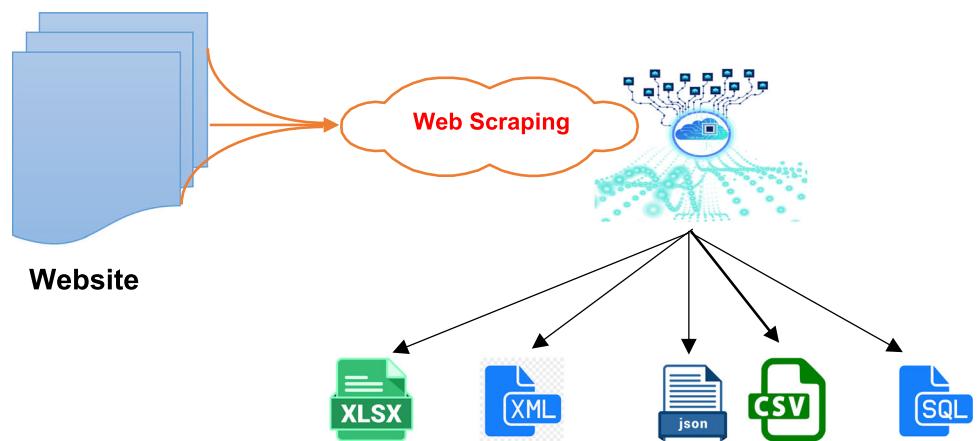
Data preprocessing (DP) is a critical step in our system because it effects on the next stage. Thus, it is an important stage in order to improve the prediction's performance. Consequently, it consists of several process. Firstly, data are collected, updated and stored using web scraping (WS). WS is the process of using bots to extract content and data from a website. It was used to update data [31]. The collected data can be stored in different formats as shown in Fig. 4. WS is done for a set period of time from ' $M$ ' different sources to collect ' $N$ ' sets of raw data. Algorithm 1 illustrates WS process.

Then the second stage is DP which can be performed through several process. At first after collecting data, incomplete data should be detected and removed. In fact, there are no missing values in our data. Then, normalization process is performed when the amount of data is large. The main aim of performing the normalization process is to limit the preprocessed data to a specific range in order to remove the negative effects brought on by the single sample of data [32]. So, normalization is an effective method for scaling the data so that they fall into a specific interval [33]. Normalization speeds up and improves the accuracy of gradient descent. In this work, when scaling data between particular ranges, min–max normalization is frequently used. This is done by applying a linear transformation to the initial data. Suppose that the lower value is represented as  $y_{\min}$  and the higher value is represented as  $y_{\max}$ . Then, the normalized value  $Y_{\text{norm}}$  is assigned a value between  $[y_{\min}, y_{\max}]$  using Eq. (1).

$$Y_{\text{norm}} = \frac{y - y_{\min}}{y_{\max} - y_{\min}} \quad (1)$$

where  $Y_{\text{norm}}$  is the normalized value and  $y$  is the old value.  $y_{\min}$  and  $y_{\max}$  are the minimum and maximum values, respectively.

**Fig. 4** Web scraping process



## Algorithm 1 WS process

### Inputs:

$M = \text{number of different data sources}$

### Output:

$DS = \text{Web scraped data stored in .csv file for } M$

### Steps:

- 1: Begin
- 2: For different data sources
- 3:     | Select Number of sources,  $M$
- 4: End For
- 5: Choose the used platform // Python for GOOGLE and APPLE
- 6: Set up scraping platform by importing required packages for each source
- 7: Select keyword for GOOGLE and APPLE
- 8: Specify time range
- 9: Apply scraping by calling library functions
- 10: End

## 5.2 Stock Price's prediction ( $SP^2$ )

In the financial literature, price prediction techniques are divided into four groups: technical analytical approaches, numerical analysis of data collected, forecasting using time series, and ML [34]. By identifying new patterns in historical data, ML tries to realize the linear and nonlinear models that are present in these and determine the underlying function from which data are created. A subset of ML known as DL was first introduced in 2005, and since 2012, it has been given substantial consideration. The learning of attribute and information processing in model layers forms the basis of DL. Deep neural network models include CNNs, RNNs, deep belief networks, and autoencoder

neural networks. In fact, timed data fit well with RNNs. It suffers from the vanishing gradient problem [35].

The LSTM network model was proposed by Schmidhuber et al. in 1997 [36]. Vanishing gradient and long-term memory problems during training processing typically affect RNN models. A network model known as LSTM was created to address the long-term memory concerns with vanishing gradient problems in RNN [37, 38]. The model uses a gate control mechanism to control the information flow and carefully evaluates how much entering information must be preserved for each time step. The input gate, output gate, and forgetting gate make up the three control gates that make up the central LSTM unit, as seen in Fig. 5.

The forget gate defines what information should be remembered as opposed to being forgotten. The sigmoid function gets data from the current input  $x_t$  and the hidden state  $h_{t-1}$ . Sigmoid generates values ranging from 0 to 1. It comes to the conclusion that the old output's portion is required (by giving the output closer to 1). This value of  $f(t)$  will then be used by the cell for point-by-point multiplication using Eq. (2) [39]:

$$S_t = \sigma((W_S * h_{t-1}) + (W_S * x_t) + b_S) \quad (2)$$

where  $W_S$  is the weight and  $b_S$  is the bias factor. The input gate then performs the following operations to update the cell status. To begin, the second sigmoid function is passed two arguments: the current state  $x_t$  and the previously hidden state  $h_{t-1}$ . Transformed values range from 0 (important) to 1 (not-important). The tanh function will then receive the same data from the hidden state and current state. The tanh function will then get input from both the hidden and present states. To control the network, the tanh operator will construct a vector ( $C(t)$ ) containing every possible value between  $-1$  and  $1$ . The activation functions' output values are generated for multiplication on a point-by-point basis using Eq. (3) [39]:

$$I_t = \sigma((W_I * h_{t-1}) + (W_I * x_t) + b_I) \quad (3)$$

where  $I_t$  is the result of the input gate,  $W_I$  is the weight, and  $h_{t-1}$  is the value of the previous state. Moreover  $x_t$  is the input of the current state and  $b_I$  is the bias.

The next step is to make a decision and save the data from the new state in the cell state. The forget vector  $f$  doubles the previous cell state  $C_{t-1}$ . If the result is 0, values will be erased from the cell state. The network then executes point-by-point addition on the output value of the input vector  $i_t$ , updating the cell state and constructing a new cell state  $C_t$  using the following Eqs. (4) & (5) [39]:

$$\bar{C}_t = \tanh((W_I * h_{t-1}) + (W_I * x_t) + b_I) \quad (4)$$

$$C_t = C_{t-1} * S_t + I_t + \bar{C}_t \quad (5)$$

Finally, the output gate finds the next hidden state using the following equations [39]:

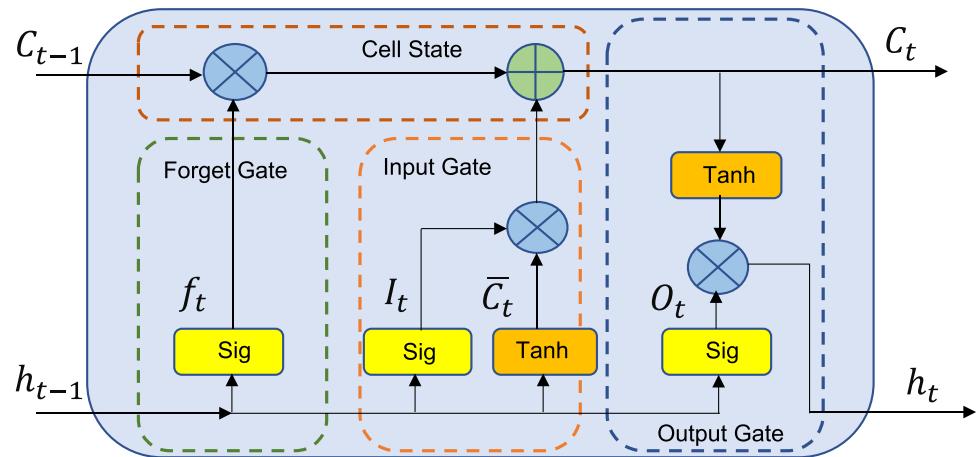
$$O_t = \sigma((W_{Og} * h_{t-1}) + (W_{Og} * x_t) + b_{Og}) \quad (6)$$

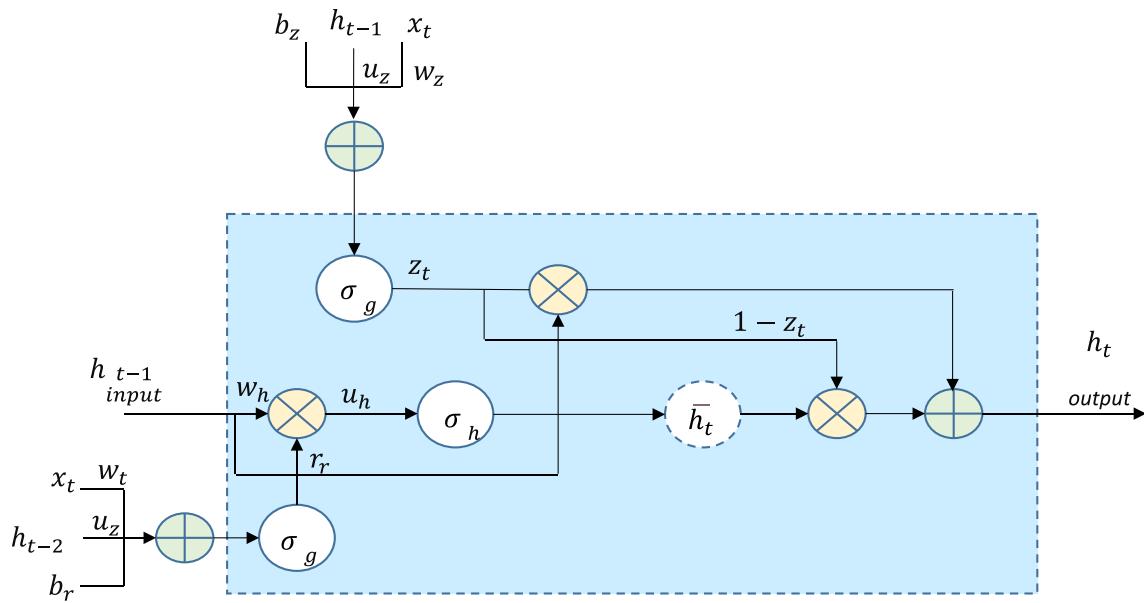
$$O_{og} = O_t * \tanh(C_t) \quad (7)$$

where  $O_t$  is the current output value,  $W_{Og}$  is the weight, and  $b_{Og}$  is the bias. Additionally,  $C_t$  represented the cell state. In this work, gated recurrent unit (GRU) was combined with LSTM to predict the closing price of stock market. A lot like an LSTM, GRU is a more recent generation of recurrent neural networks with fewer parameters. The hidden state was used by GRUs to transmit information instead of the cell state. The update gate and the reset gate are the only gates it has. Figure 6 shows the architecture of GRU.

As shown in Fig. 6, GRU model consists of two-gate function which are reset gate and update gate. Actually, reset gate was used to decide what needs to be taken out of the cell's internal state before it moves on to the following time step. While update gate was used to determined how much of the state from the previous time step should be utilized in the current time step. In fact, reset gate and

**Fig. 5** LSTM architecture





**Fig. 6** GRU structure

update gate are used to deal with the vanishing gradient problem. In many use cases, GRUs typically have a slight advantage over LSTMs, particularly when GRU cells are a little simpler than LSTM cells. The mathematical representation of the behavior of these two gates in the GRU model for output  $h_t$  in cell state  $c_t$  is donated through (8–11) [40]:

$$\text{UG}_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (8)$$

$$\text{RG}_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (9)$$

$$\bar{h}_t = \sigma(W_h x_t + (\text{RG}_t * U_h h_{t-1}) + b_h) \quad (10)$$

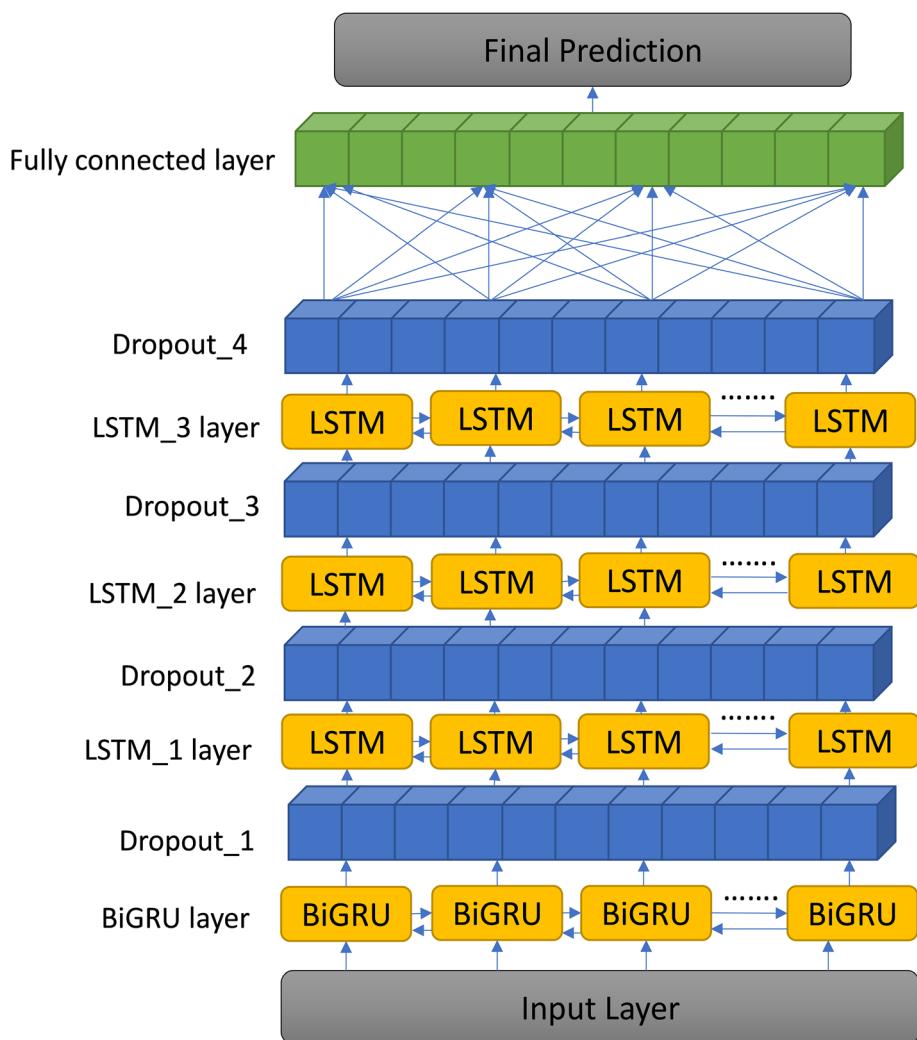
$$h_t = (\text{UG}_t * h_{t-1}) + (1 - z_t) * \bar{h}_t \quad (11)$$

where  $\text{UG}_t$  represented the update gate and  $\text{RG}_t$  is the reset gate.  $W_z$ ,  $W_r$ ,  $W_h$ ,  $U_z$ ,  $U_r$  and  $U_h$  are the applied weights.  $b_z$ ,  $b_r$ , and  $b_h$  are the corresponding bias. Then, in an attempt to answer the third research question to what extent selecting the suitable method is more effective in predicting stock market prices. In this work, a novel hybrid method that combines between LSTM and BiGRU was proposed to improve the overall accuracy of the proposed system. Actually, LSTM can overcome the issue of exploding and vanishing gradients caused by long input sequences. GRU uses less memory and is faster than LSTM, however, LSTM is more accurate when using datasets with longer sequences. Consequently, the combination of BiGRU and LSTM has higher accuracy than the BiGRU and LSTM models themselves. The reason is that, these two deep learning models are excellent for making predictions about time series. It has two main benefits, namely the most reliable performance is determined using a variety of input scenarios. In order to make accurate

predictions, (i) the model embeds a time layer to eliminate the time series decomposition process, and (ii) strengthens its high-level temporal representation. In this subsection BiGRU-LSTM which integrates between BiGRU and LSTM is introduced to quickly and accurately predict stock market price in future as shown in Fig. 7. Basically, BiGRU is a sequential processing model composed of two GRUs, one for talking the input forward and the other for talking the input backward.

Indeed, our proposed hybrid model is made up of five layers, the first of which contains a BiGRU with 100 hidden units and the second of which contains an LSTM with 100 hidden units. LSTM layering with 50 hidden neurons make up the third and fourth layers. The last layer is thick layers with 1 hidden neuron. We trained this model on 10- and 30-min interval data that we had before processed from the initial 1-min interval data. The inner structure of the suggested prediction model is depicted in Fig. 7. The BiGRU layer receives all of the dataset's attributes as input at first. BiGRU is the first hidden layer we employ. Every BiGRU neuron collects data and generates a weighted value along the way. Following that, the LSTM layer, our second hidden layer, collects input from the BiGRU layer. The connection from the BiGRU layer to the LSTM layer is used to generate a weighted value once more. The final hidden layer, the dense layer, gets data in a similar fashion. A weighted value is generated by LSTMs and dense layer. The dense layer was used in producing the output. Data are subsequently transmitted to the fourth hidden layer's output neuron, which generates the relevant weight. Actually, between each layer we have added a dropout layer. A small dropout value of 20%–50% is recommended to start with

**Fig. 7** The proposed prediction model

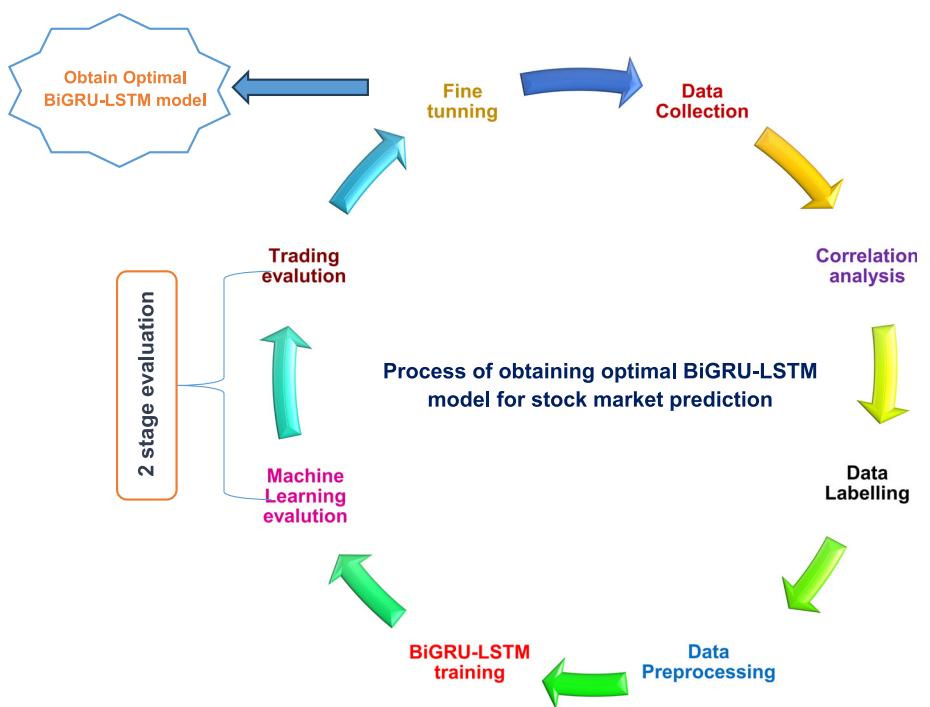


[41]. Actually, dropout is like selective blindness in that too much of it will your network under-learn while too less will lead to overfitting. In this paper, we use a dropout rate is set to 20% as this value provides high performance, reduces overfitting and improves generalization error. Consequently, using dropout value of 20% means that one out of five inputs will be randomly excluded from each refresh cycle. The error function is then determined by comparing the output to the original value. When the weighted values reach the minimum point of the cost function, they are then saved for use in future predictions and updated based on the difference between the actual value and the predicted value. The system's performance is evaluated after the future predictions for the next 10 and 30 min are completed based on the weighted values that have been saved. Figure 8 shows the process of obtaining optimal BiGRU-LSTM model for stock market prediction.

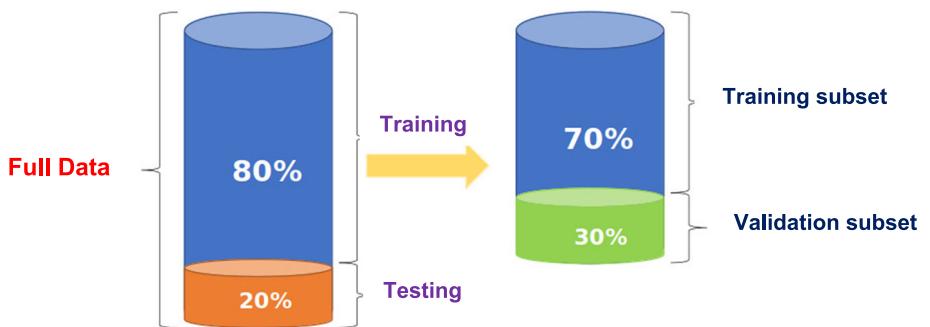
## 6 Experimental results

Here in this section, the proposed SMP-DL will be evaluated. The proposed SMP-DL consists of two phases which are DP and SP<sup>2</sup>. In DP, several stages were performed which are, collect data using WS process, detect and remove null or missing values in datasets, and normalize the data to be in specified scale consequently, data are ready for the next phase (i.e., SP<sup>2</sup>). In SP<sup>2</sup>, a novel prediction method has been proposed which combine between LSTM and BiGRU. In the proposed LSTM-BiGRU, the first layer after input will be BiGRU layer to speed up the process and to improve the prediction accuracy of stock market. The proposed system has been implemented using data that were collected from [42]. To evaluate the proposed system, dataset was divided into 80% to train the proposed stock model and 20% for testing to measure the prediction effect of the proposed model as shown in Fig. 9. Additionally, Python3 version '3.9.12' was used as programming language to run the experiments for this paper.

**Fig. 8** Process of obtaining optimal BiGRU-LSTM model for stock market prediction



**Fig. 9** The mechanism of dividing the used datasets



In this virtual environment, the following packages were installed which are Numpy 1.21.5, Pandas 1.4.2, Matplotlib 3.5.1, Seaborn 0.11.2, yfinance 3.9.12, Sklearn 1.0.2, Tensorflow 2.10.0, and Keras 13 2.10.0-tf. The program code was simulated using GPU with 8 GB RAM, and CPU: Intel i7-8565U (4) @ 1.800 GHz. Table 2 shows the proposed prediction model summary.

### 6.1 Time series data description

We gathered information for International Business Machines (IBM) stock from the New York Stock Exchange (NYSE), the world's largest stock exchange. We collected 216,833 stocks in total. The data were taken from numerous industries and referenced from 25 March 2020 to 6 April 2022 by using raw data for each trading minute of the industry indexes; each item of data includes: opening price,

**Table 2** The proposed prediction model summary

Layer (type)	Output shape	Parameters
Bi GRU	(None, 1,200)	61,800
dropout (Dropout)	(None, 1,200)	0
LSTM_1	(None, 1,100)	120,400
dropout_1 (Dropout)	(None, 1,100)	0
LSTM_2	(None, 1,50)	30,200
dropout_2 (Dropout)	(None, 1,50)	0
LSTM_3	(None, 50)	20,200
dropout_3 (Dropout)	(None, 50)	0
Dense (Dense)	(None, 1)	51
Total parameters:	232,651	
Trainable parameters:	232,651	
Non-trainable parameters:	0	

highest price, lowest price, closing price, and volume. NumPy, Pandas, Matplotlib, Seaborn, Plotly, and Scikit-learn Python libraries are utilized in this study for data visualization and implementation of proposed algorithms. The data set specification is shown in Table 3.

A data preprocessing method called normalization is used to convert the values of features in a dataset to a standard scale. This is carried out to render data processing and modeling easier, as well as to lessen how differing scales affect how accurate ML models are. As seen in Fig. 10, the pricing data are then normalized for quick convergence and the learning process.

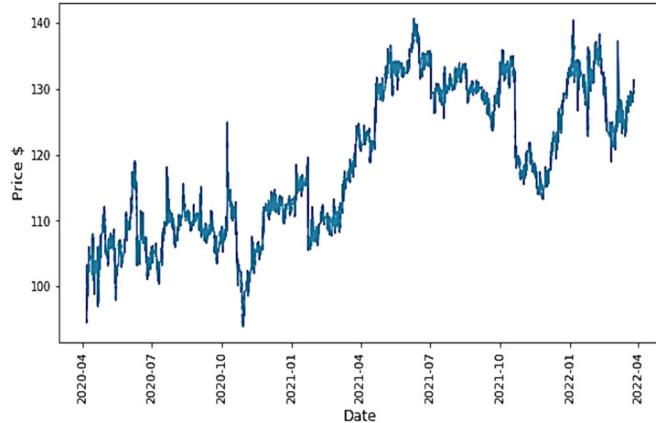
Several businesses use statistical analysis to arrange gathered data and forecast upcoming trends using that data. Despite the fact that there are many alternatives available to organizations regarding how to handle their big data, statistical analysis provides a technique to both assess the data as a whole and split it into individual samples. The statistical analysis of the employed datasets is also included in Table 4.

With a box plot visualization, you can look at how the data are distributed. There is a box plot for each attribute component. The boxplot of the used datasets is displayed in Fig. 11.

**Table 3** The specification of the used datasets

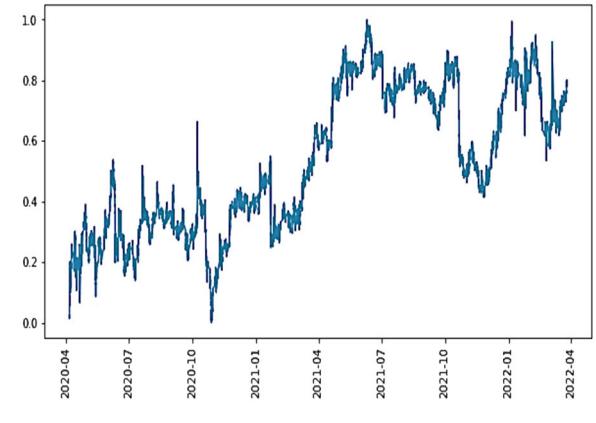
Feature name	Description	Type
Date	The date of the stock price	Date
Open	A given day's opening price	Number
High	High price on the specified day	Number
Low	Low price on the specified day	Number
Close	The day's closing price	Number
Volume	Is the total number of shares or contracts that buyers and sellers have exchanged	Number

std=11.032586998608828 , mean=118.56867830467517



(a)

std=0.23592095046636719 , mean=0.5276258445926093

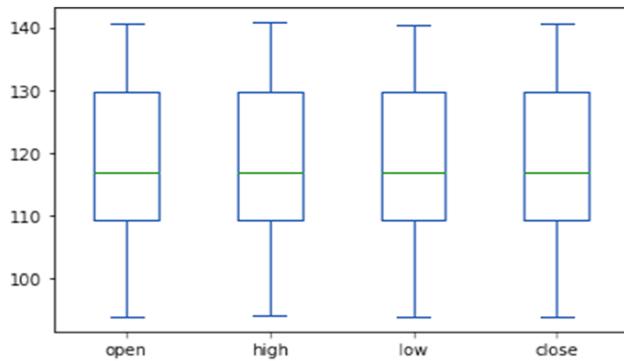


(b)

**Fig. 10** IBM data, **a** before and **b** after normalization process

**Table 4** The statistical analysis of the used datasets

	Count	Mean	Std	Min	25%	50%	75%	Max
Open	216,883.0	119.0	11.0	94.0	109.0	117.0	130.0	141.0
High	216,883.0	119.0	11.0	94.0	109.0	117.0	130.0	141.0
Low	216,883.0	119.0	11.0	94.0	109.0	117.0	130.0	140.0
Close	216,883.0	119.0	11.0	94.0	109.0	117.0	130.0	141.0
Volume	216,883.0	11,230.0	25,509.0	0.0	3896.0	6786.0	11,751.0	2,110,106.0



**Fig. 11** The boxplot of features in the used data

Correlation is a scaled version of covariance in statistics that determines whether variables are connected positively or negatively. Correlation is an important topic in technical stock market analysis because it allows us to hypothesize on the mechanics of price movements. Figure 12 depicts the relationship between all features and its graph.

## 6.2 Technical indicators of stock market

A popular statistical method for determining the trend in a stock is the technical indicator [43]. Market analysts and investors can use a moving average as a technical indicator to pinpoint a trend's direction. We calculate the mean by adding together all the data points related to financial security for a given time period and dividing the result by the total number of data points. The “moving” in its name refers to the fact that it is continually updated with the most recent pricing information. By analyzing an asset’s price changes, analysts can assess support and resistance using a moving average. A security’s prior price action or movement is reflected by the moving average. The information is then used by analysts or investors to forecast the price direction of the asset. It is referred to as a lag indicator since it uses the price movement of the underlying asset to generate signals or reveal trends. In order to recognize the upward trend in the price of the stock, we took into account three technical moving average indicators. Figure 13 illustrates how technical indicators are calculated for 24 h, 30 days, and 500 days using the formula used in [44].

## 6.3 Model evaluation

To evaluate the proposed model, different evaluation metrics are used to represent the difference between the predicted and actual value. These metrics are mean absolute error (MAE), mean-squared error (MSE), root mean-squared error (RMSE) and  $R^2$ -squared ( $R^2$ ). These metrics can be calculated by using the following equations:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (12)$$

$$\text{MSE} = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (13)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (14)$$

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y}_i)^2} \quad (15)$$

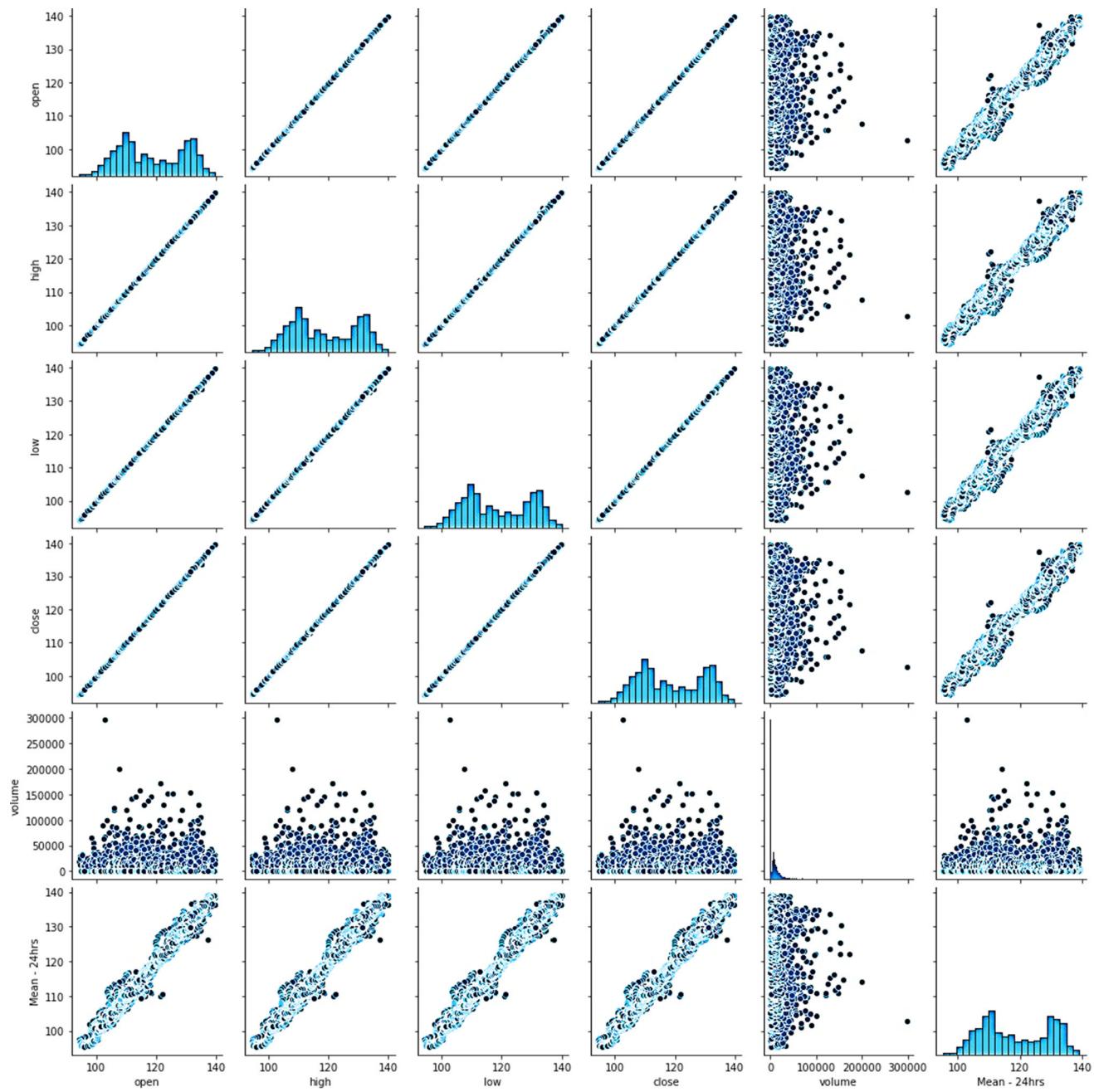
where  $y_i$  represents the actual value, and  $\hat{y}_i$  represents the predicted value. Additionally,  $\bar{y}_i$  denotes the mean value of  $y$ , and  $n$  is the total number of samples.

## 6.4 Testing the proposed SMP-DL

Using an IBM dataset, we evaluate and explain the effectiveness of the suggested system in this section. The outcomes are displayed in Fig. 14a, c. The proposed model’s tests and training loss are shown in Fig. 14a. Figure 14b, c also depicts the suggested MSE and MAE models for training and testing. A typical method for evaluating forecast error in the analysis of time series is the MAE, which provides the mean of the absolute difference between predicted data and target output [45]. Larger prediction errors are better noted by MSE, while MAE does not punish them. The average squared error between the values expected and observed is measured using MSE. The better prediction is represented by a lower MSE value. In the field of stock market forecasting, these indicators have frequently been applied to resolve regression issues and assess forecast performance. Figure 14 illustrates how effectively the suggested model operates, with donations of MAE of 0.2099 and MSE of 0.0831. As a consequence, based on the results, the proposed model achieves the lowest error rate. As a result, it is useful for forecasting IBM’s stock market closing price.

## 6.5 Comparison of the proposed BiGRU-LSTM with the state of the art

In this subsection, to prove the effectiveness of the proposed BiGRU-LSTM, it is compared against some of recent stock market prediction methods using IBM dataset. These methods are ABC-ANFIS-SVM [16], DLSMP [17], CNN-BiLSTM-AM [20], PSO-LSTM [22], BiLSTM [26], CNN [27], and CKELM [28] as shown in Table 1. Results are presented in Fig. 15a → h. As shown in Fig. 15a → h, among the seven prediction methods, the order of degree of the dashed line to fit the true value to the predicted value is the proposed PSO-LSTM, DLSMP, ABC-ANFIS-SVM,

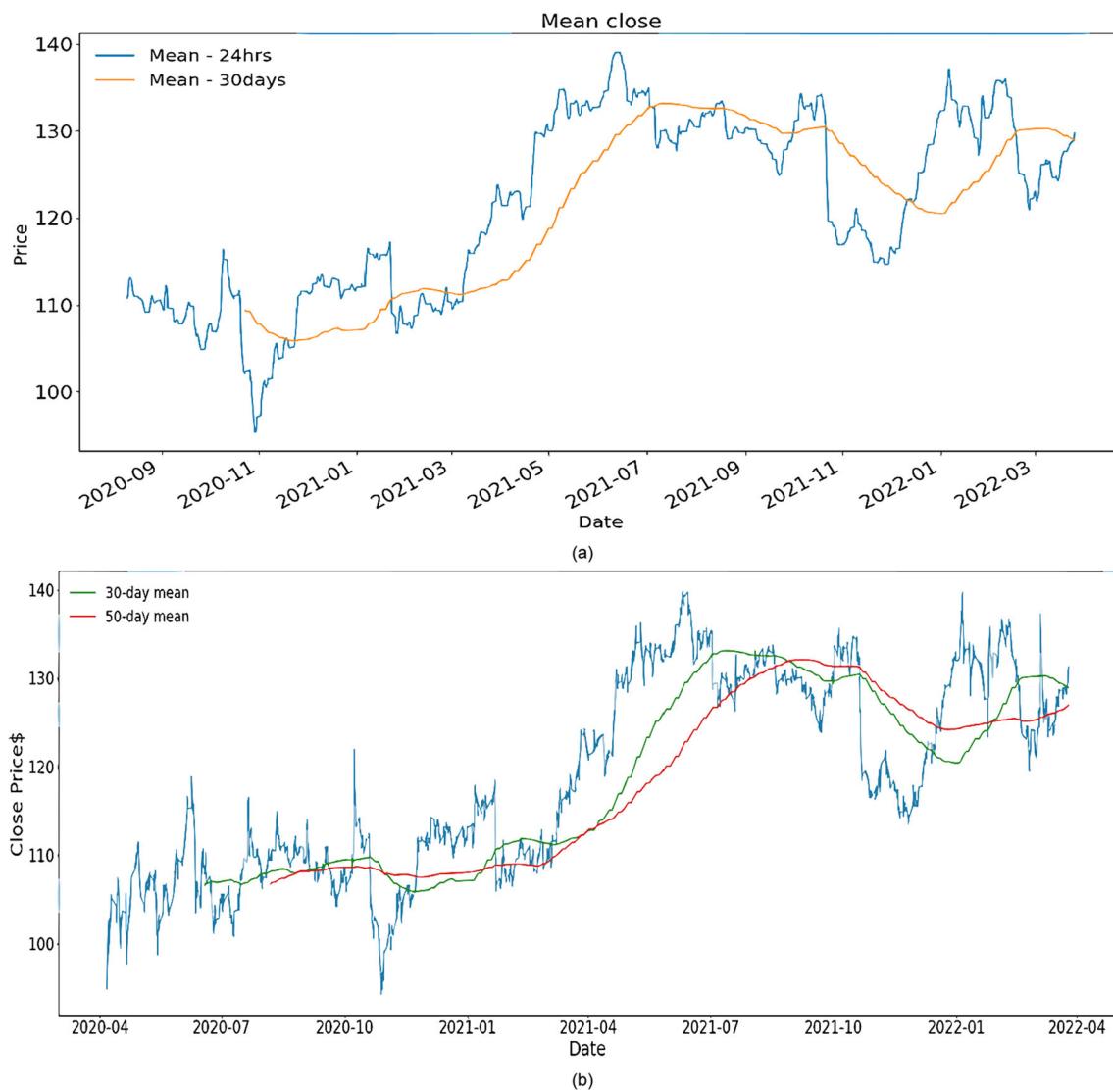


**Fig. 12** Examining the correlation with the features

CNN-BiLSTM-AM, BiLSTM, CNN, CKELM, and BiGRU-LSTM from low to high. While MLP has the lowest broken line degree of fitting, CNN-BiLSTM-AM has the highest broken line degree of fitting between real value and predicted value, which is almost entirely coincident. The evaluation error indices of each method can be calculated based on the predicted value of each method and the real value, and the comparison outcomes of the eight methods are displayed in Table 5.

As illustrated in Table 5, RMSE and MAE of PSO-LSTM are the highest and  $R^2$  is the lowest. On the contrary,

the RMSE and MAE of BiGRU-LSTM are the lowest and  $R^2$  is the highest and very closer to 1. As shown in Table 5, the order of the prediction performance is PSO-LSTM, DLSMP, ABC-ANFIS-SVM, CNN-BiLSTM-AM, BiLSTM, CNN, CKELM, and BiGRU-LSTM from low to high. Comparing BiLSTM with PSO-LSTM, its RMSE, MAE is less, while its  $R^2$  value is more. BiLSTM RMSE value is 0.3348 compared to 0.3987 for PSO-LSTM and its  $R^2$  value is 0.9649 compared to 0.9400 for PSO-LSTM. Compared with BiLSTM, CNN-BiLSTM-AM increases RMSE from 0.3348 to 0.3487, MAE from 0.3564 to

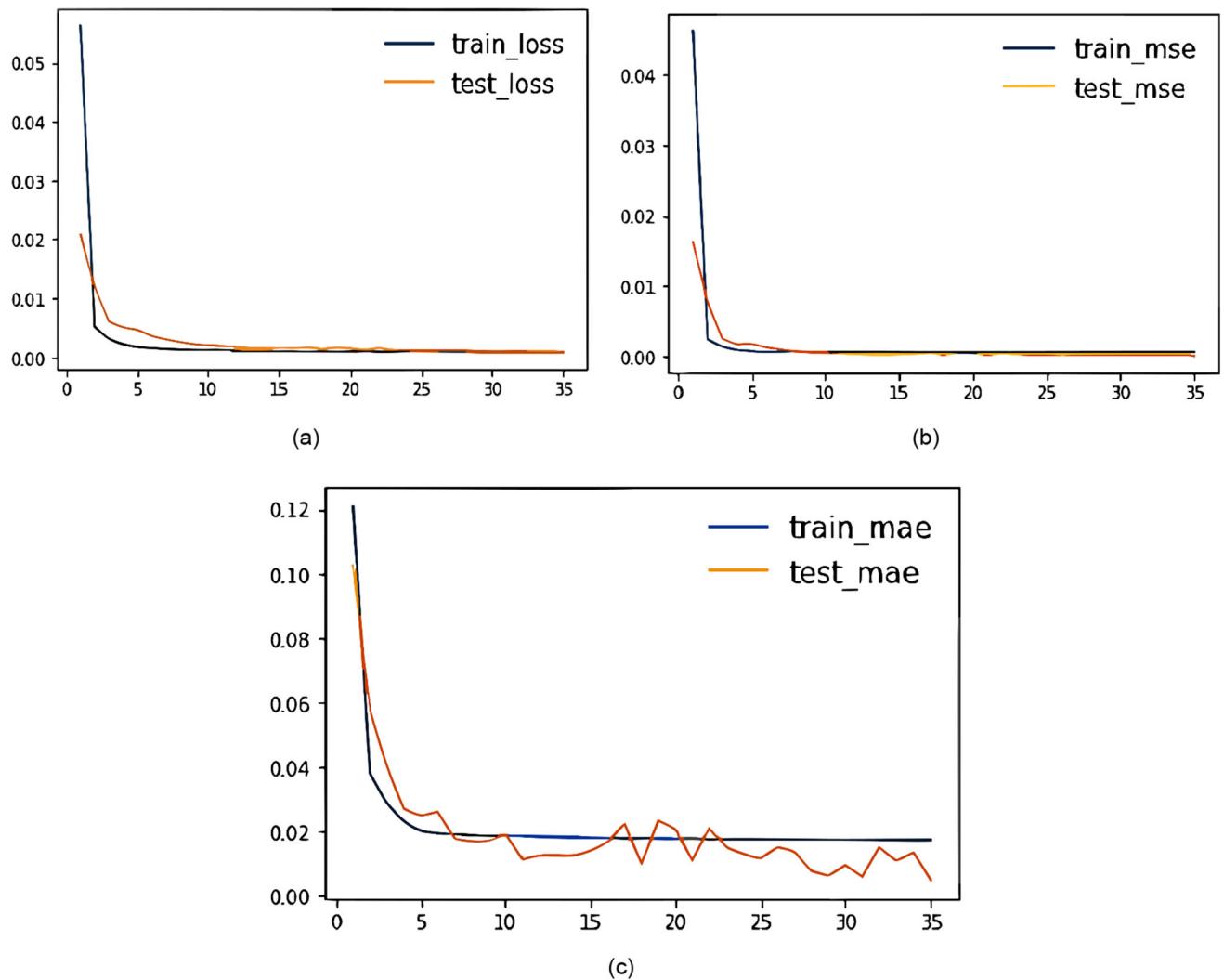


**Fig. 13** The moving average of IBM through **a** 24 h and 30 days **b** 30 days and 50 days

0.3591, and decreases  $R^2$  from 0.9646 to 0.9590. When BiGRU is introduced to LSTM, its predictive accuracy improves. MAE decreases by 0.1708, RMSE by 0.1104, and  $R^2$  increases by 0.0548 than of using PSO with LSTM. Thus, as demonstrated in Table 5, the proposed BiGRU-LSTM outperforms the other methods in terms of numbers of epochs as it is 40 compared to the others as the numbers of epochs of them are 140, 120, 150, 110, 100, 130, and 70 for ABC-ANFIS-SVM, DLSMP, CNN-BiLSTM-AM, PSO-LSTM, BiLSTM, CNN, AND CKELM at the same order. Finally, Table 5 demonstrates the effectiveness of the suggested method for predicting the stock market price in future and serve as a guide for investors to choose the best investments with less testing computational time value of 2.5 Sec.

## 6.6 Testing the proposed stock market prediction using different datasets

Using various datasets, the effectiveness of the suggested model will be evaluated in this section. The IBM dataset is used to show how accurate the suggested model's predictions are. The actual price and the expected price are roughly equal, as seen in Fig. 16a. Additionally, the suggested model will be assessed using datasets from AAPL [46] and GOOG [47]. Whereas the GOOG dataset offers historical data for Alphabet Inc. and is available on a daily basis with USD currency, the AAPL dataset comprises Apple's stock data for the previous ten years (from 2010 to the present). Figure 16b, c presents the results obtained, and it is clear from these figures that the proposed model performs better when applying various datasets to forecast closing prices. As a result, based on the data, the proposed



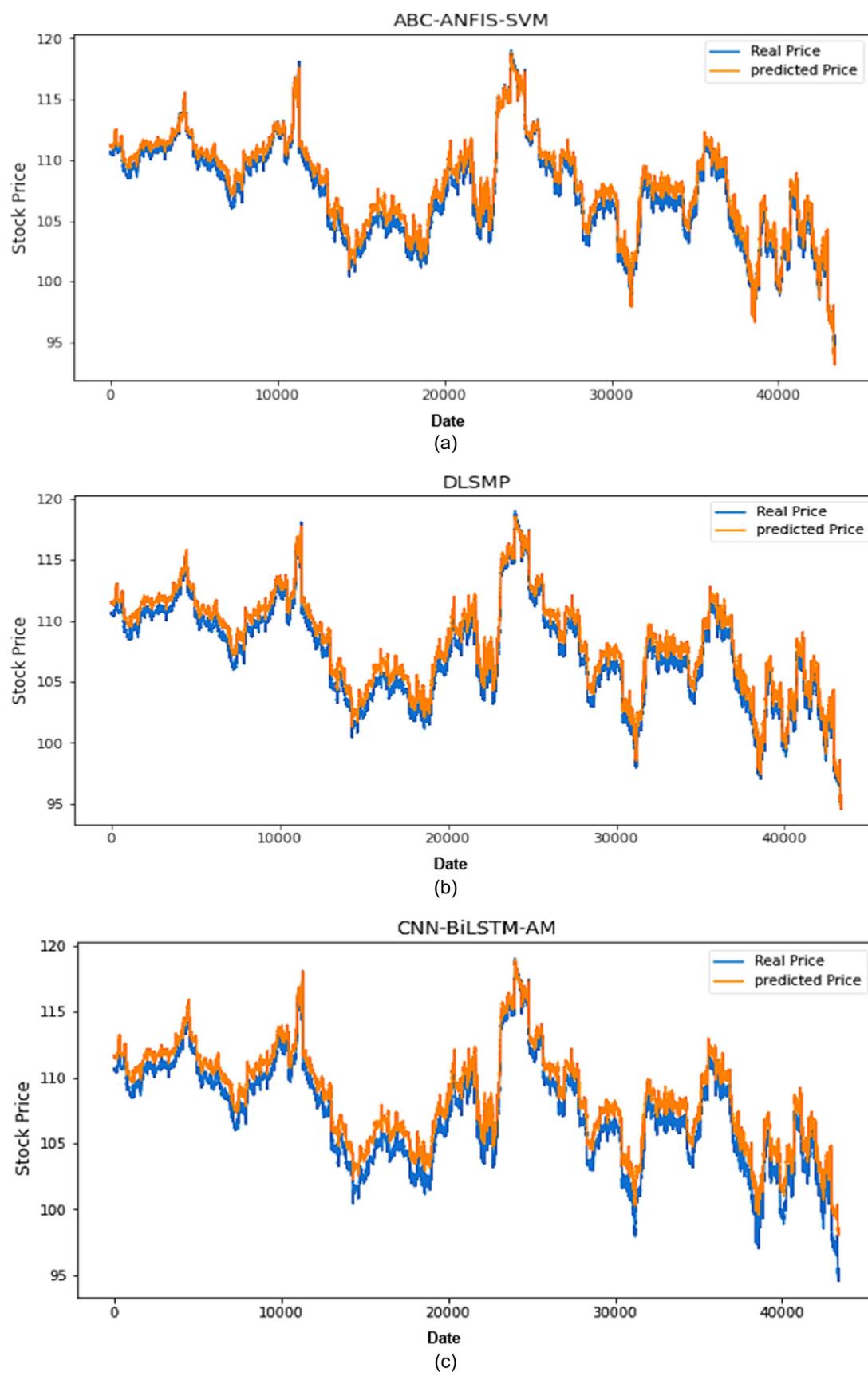
**Fig. 14** The proposed model, **a** Training and testing loss **b** Training and testing MSE **c** Training and testing MAE

model performs well in forecasting the stock market's closing price.

### 6.7 Comparison with different datasets and different models

This subsection applies the suggested model using various datasets. Also, we compared the proposed system with many models that are current in the field to demonstrate its validity. LSTM and GRU-LSTM models are employed. The outcomes are displayed in Fig. 17a → c. The date and closing price of the dataset utilized are displayed on the x-axis and the y-axis, respectively, in figures (a) through (c). The suggested model is contrasted with the most popular models utilizing the IBM, AAPL, and GOOG datasets, as

illustrated in Fig. 17a → c. The results collected demonstrate that the suggested model is superior to LSTM and GRU-LSTM in predicting the closing price of IBM stock. In addition, because LSTM enables the learning of even more parameters, as illustrated in Fig. 17a → c, it can be utilized to forecast the stock market. This makes it the most effective method for making forecasts, particularly when your data exhibits a longer-term trend, as in stock market analysis. Because we combined LSTM with BiGRU, which can enhance the performance of LSTM models that cannot capture both forward and reverse sequence information and reduce the complexity, the proposed model in this paper is the one that comes the closest to representing the actual value of the closing price.



◀ Fig. 15 Comparison of recent stock market prediction methods predicted value with real value: **a** ABC-ANFIS-SVM, **b** DLSMP, **c** CNN-BiLSTM-AM, **d** PSO-LSTM, **e** BiLSTM, **f** CNN, **g** CKELM DLSMP, and **h** proposed BiGRU-LSTM

## 6.8 Time complexity of the proposed model

Since we employed a hybrid BiGRU-LSTM model, we will now go through the time complexity of each layer individually. First, a BiGRU layer's time complexity is impacted by input sequence length, hidden unit count, and network depth. A single GRU cell has a time complexity of  $O(N^2)$ , where  $N$  is the total number of hidden units. This is because the hidden state, along with the update and reset gates, all take up an area of size  $N * N$ . BiGRU performs both forward and backward processing of the input sequence. If we replace  $N$  with the length of the input sequence,  $T$ , we find that the time complexity of a single-layer BiGRU is  $O(2N^2 T)$ .  $O(L2N^2*T)$ , where  $L$  is the number of layers, is the time complexity for multilayer BiGRU. The input length has no effect on the network's storage needs with LSTM, and the network's time complexity per weight is  $O(1)$  at each time step [29]. Since there are  $w$  weights in an LSTM, the complexity of each time step is  $O(w)$ . The proposed BiGRU-LSTM layer complexity per time step is thus equal to the product of the BiGRU layer's complexity and the LSTM layer's complexity.

## 7 Conclusions

In this study, a new system based on DL is proposed to predict the closing price of the stock market. In addition, in this paper, an application was designed for anyone to use on their mobile device which is called smart trading platform (STP). Actually, STP is to allow anyone to invest in the stock market. Through STP anyone can buy and sell a variety of securities including stocks, futures, options, bonds, commodities, and currencies. A computer, laptop,

or mobile device with internet access is all you need to start trading on an online trading platform. You can virtually invest in stocks from anywhere, including your office or home. Also, we have shown the effectiveness of a hybrid model that combines BiGRU and LSTM to forecast the stock market closing price. This study was conducted on IBM, Google, and Apple. As a proof of concept, we predicted the price 10 and 30 min in advance of the actual time. From the historical data website, we downloaded a dataset with 1 min intervals. Then data were added to a BiGRU model, which created a weighted value before sending the information to the LSTM network. The dense layer receives the new weight value that the LSTM determined and processed it. The dense layer's output, which generated the overall model output, is delivered to the output layer. In the output layer, weighted values are optimized to bring down the value of the loss function after the system output is compared to the outcome that actually occurred. The experimental findings demonstrated that the BiGRU-LSTM hybrid model outperformed two of the most common and dependable time series analyzers: LSTM and GRU, in terms of predicting the closing price of the stock market. The main advantages of the proposed application which is based on DL model are: (i) real-time trading, one of the main benefits of using STP is the ability to execute trades in real time as anyone can easily check the stock's current price on the online trading platform rather than phoning your broker for quotes. Within a few seconds, it is possible to check stock prices, place an order, and complete the trade. (ii) Cost effective, our platform is very inexpensive as you pay less in brokerage and other fees, which is not the case with traditional investing. (iii) Instant access to market data, our platform provides access to technical charts and investment tools that provide comprehensive research insights and statistics to traders. This helps traders make informed investment decisions to increase their returns. Moreover, it saves time and significantly reduces risks. Finally, (iv) our platform provides the ability to receive real-time notifications.

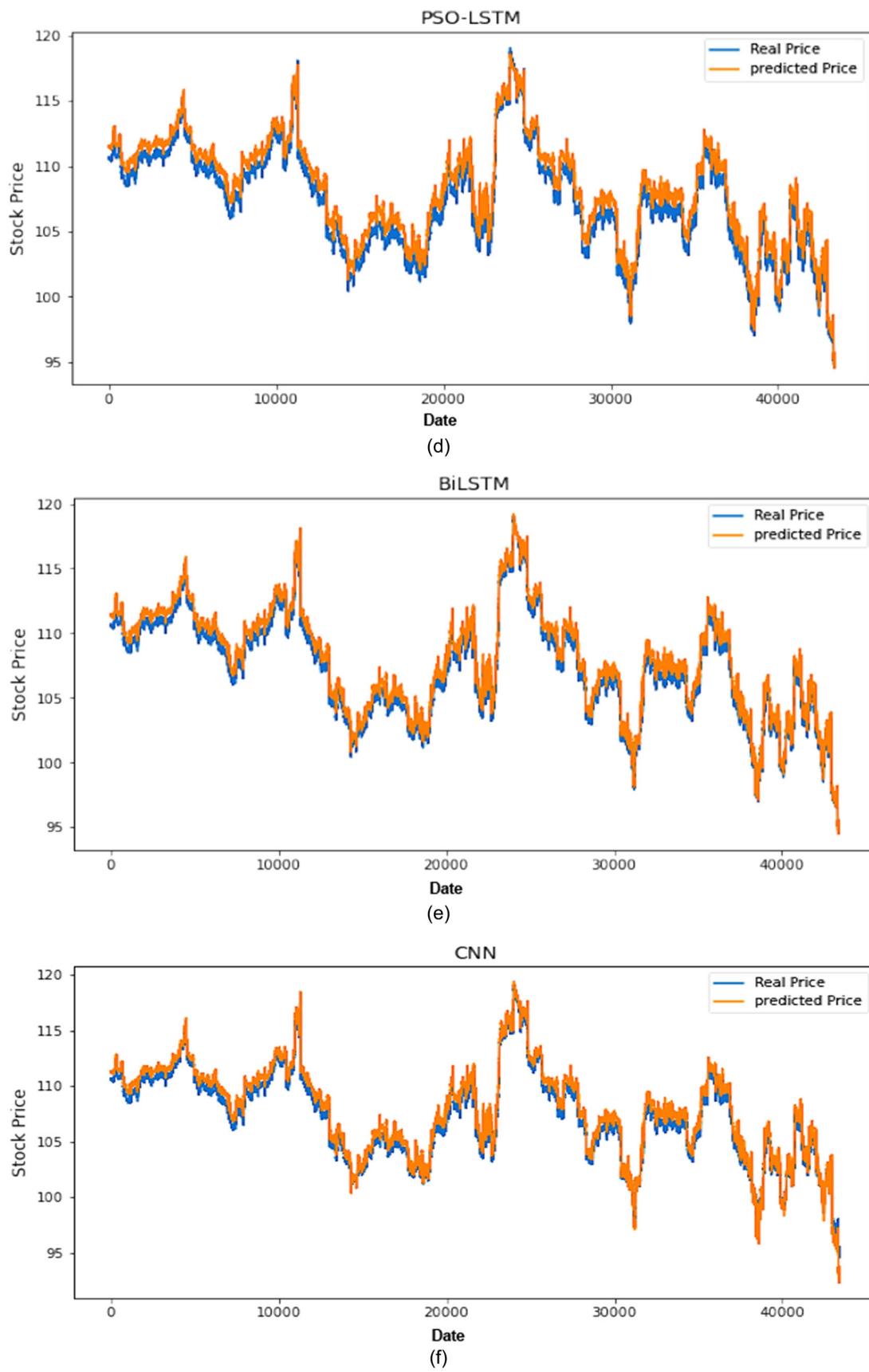
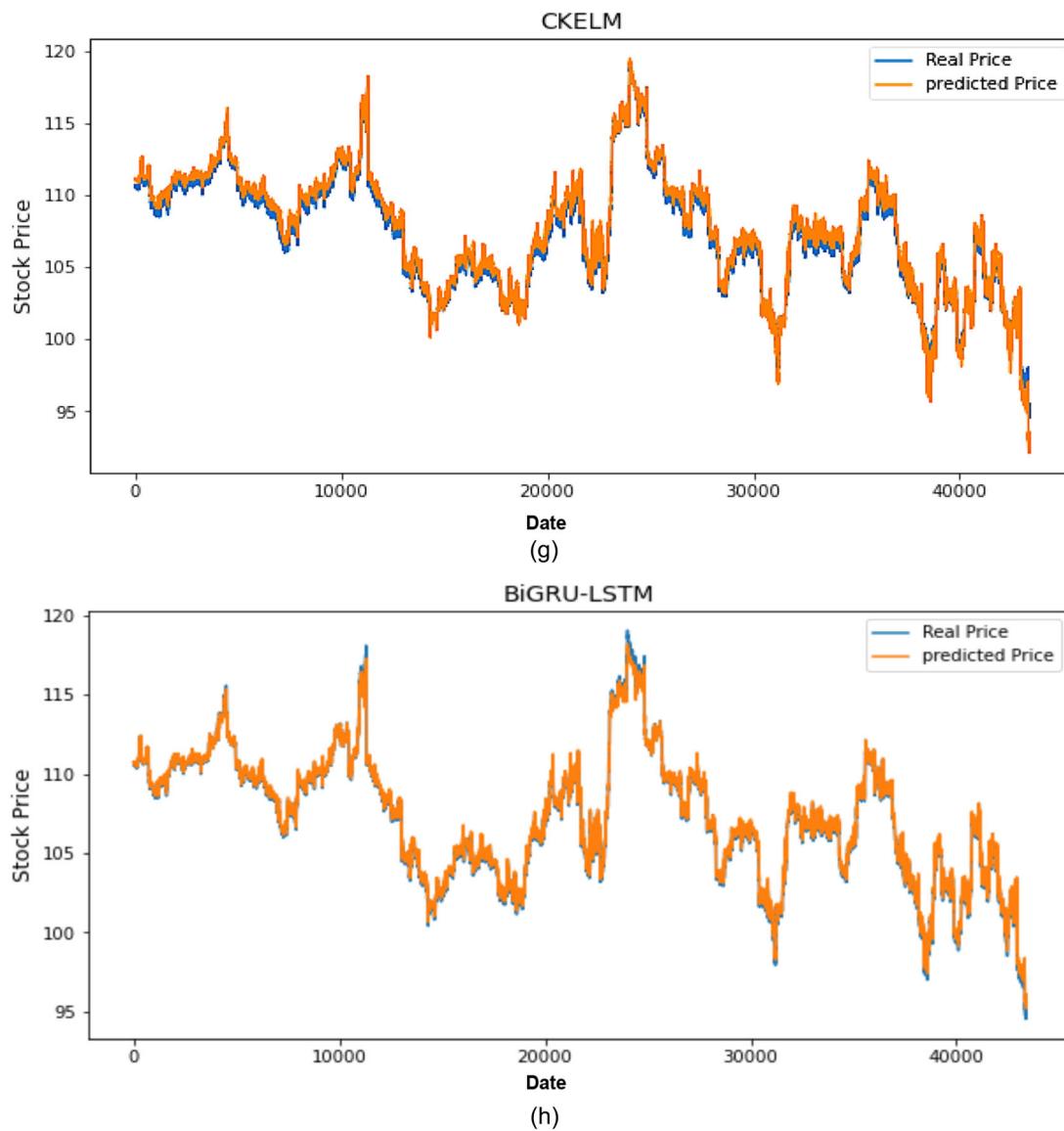
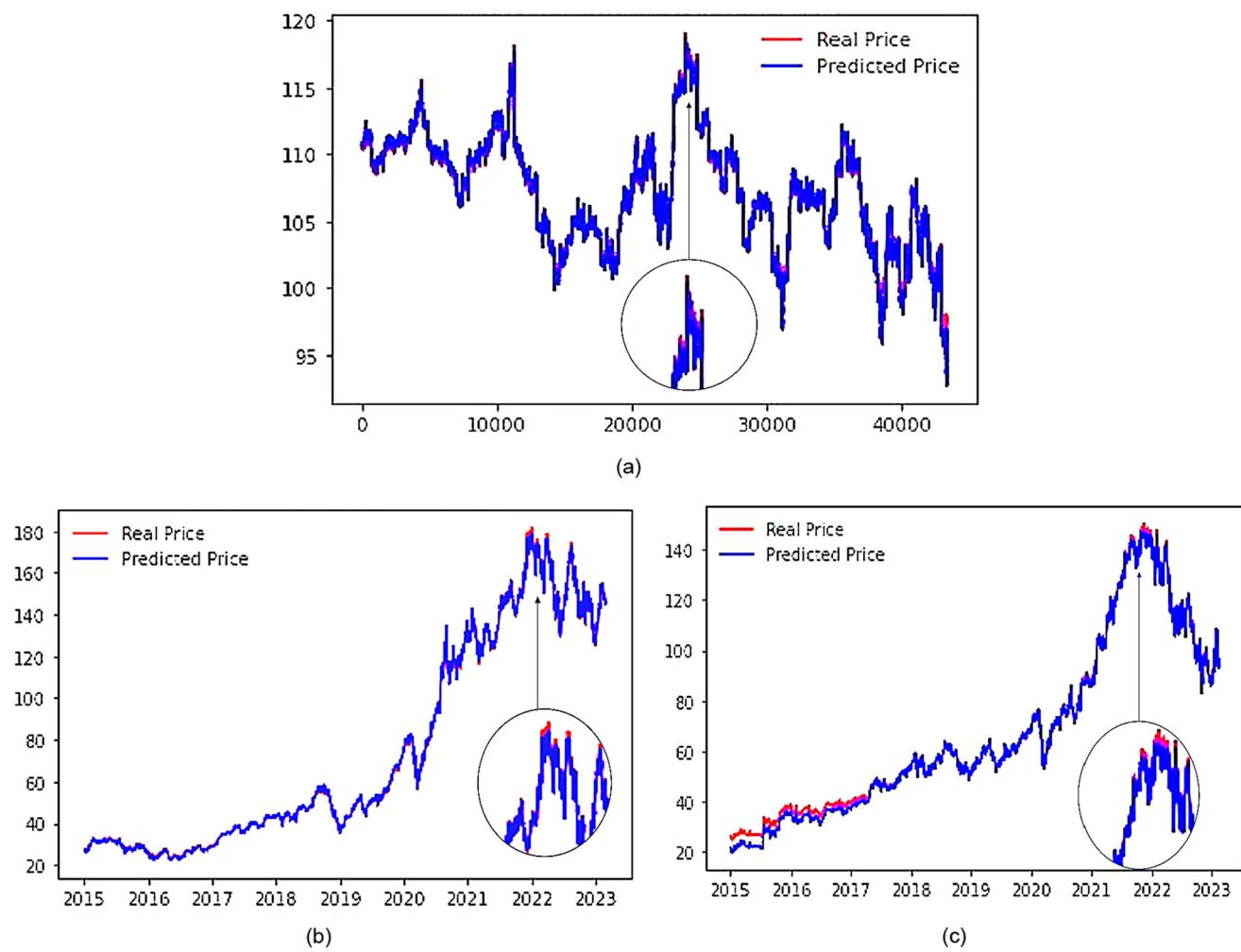


Fig. 15 continued

**Fig. 15** continued

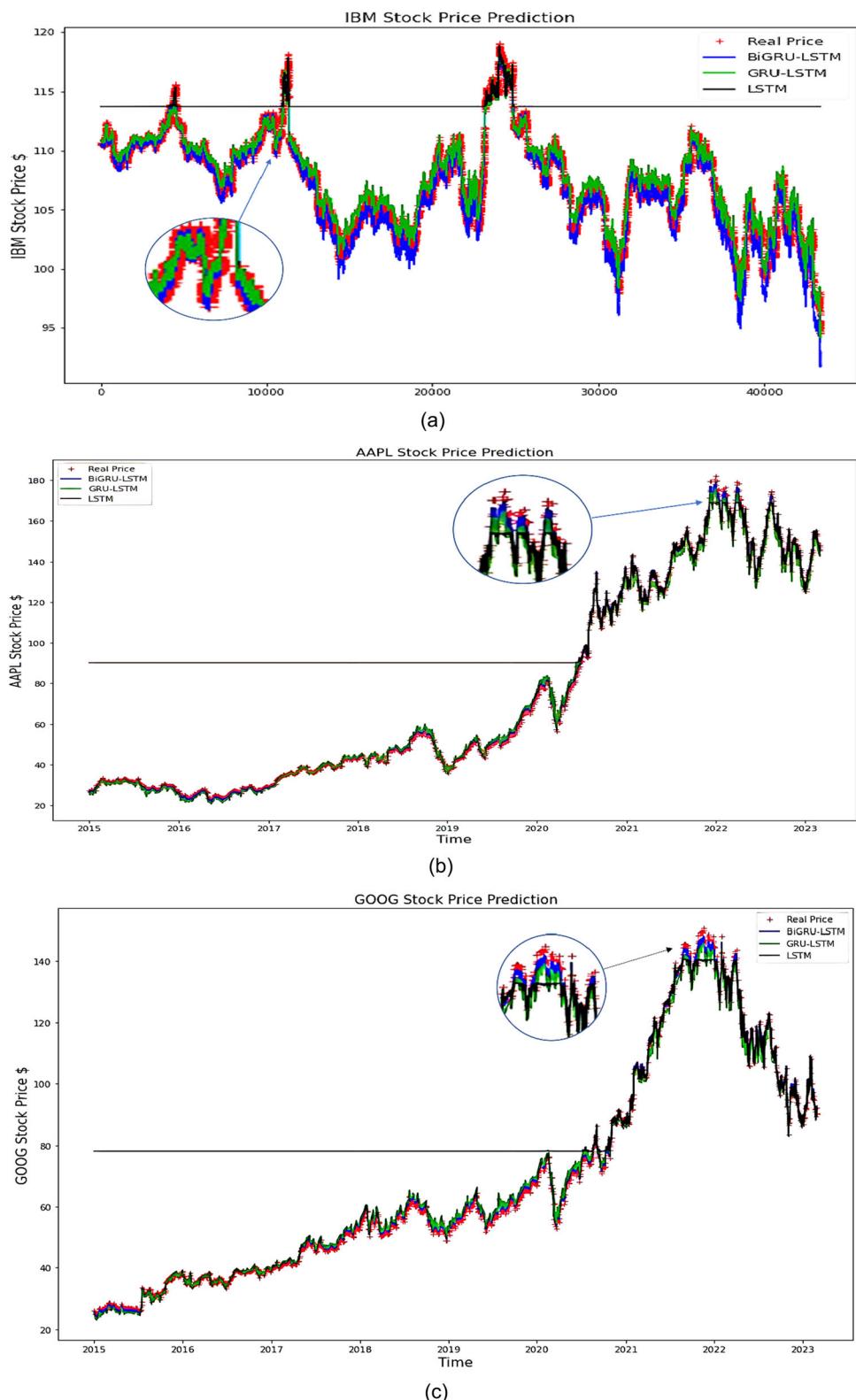
**Table 5** Comparison of our proposed model with the other state-of-the-art methods in terms of RMSE, MAE,  $R^2$ , computational time and the number of epochs

Prediction method	RMSE	MAE	$R^2$	Computational time (S)		Epochs
				Training	Testing	
ABC-ANFIS-SVM	0.3638	0.3652	0.9458	1730.5	6.5	140
DLSMP	0.3941	0.3785	0.9403	1124	7.5	120
CNN-BiLSTM-AM	0.3487	0.3591	0.9590	1650	8.6	150
PSO-LSTM	0.3987	0.3807	0.9400	1800	9	110
BiLSTM	0.3348	0.3564	0.9646	1250	4.5	100
CNN	0.3979	0.2755	0.9694	1500	6	130
CKELM	0.3169	0.2195	0.9804	1750	5	70
Proposed BiGRU-LSTM	0.2883	0.2099	0.9948	1500	2.5	40



**Fig. 16** Real prices vs predicted daily closing prices using proposed system with, **a** IBM **b** AAPL **c** GOOG, datasets

**Fig. 17** The prediction results of, **a** IBM Stock **b** Apple Stock **c** Google Stock, under different models



**Funding** Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

**Data availability** The dataset is available at: <https://www.kaggle.com/code/eslamreda0101/ibm-stock-analysis-lstm/data>

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest. “This paper does not contain any studies with human participants or animals performed by any of the authors.”

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Vijh M, Chandola D, Tikkwal VA, Kumar A (2020) Stock closing price prediction using machine learning techniques. *Procedia Comput Sci* 167:599–606
- Khan W, Ghazanfar M, M Azam et al (2022) Stock market prediction using machine learning classifiers and social media, news. *J Ambient Intell Human Comput*, Springer 13:3433–3456. <https://doi.org/10.1007/s12652-020-01839-w>
- Nabipour M, Nayyeri P, Jabani H, Shahab S, Mosavi A (2020) Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis. *IEEE Access* 8:150199–150212. <https://doi.org/10.1109/ACCESS.2020.3015966>
- Nikou M, Mansourfar G, Bagherzadeh J (2019) Stock price prediction using deep learning algorithm and its comparison with machine learning algorithms. *Intell Syst Account, Finan Manag* 26(4):164–174
- Sharma DK, Hota HS, Brown K, Handa R (2022) Integration of genetic algorithm with artificial neural network for stock market forecasting. *Int J Syst Assur Eng Manag* 13(Suppl 2):828–841. <https://doi.org/10.1007/s13198-021-01209-5>
- Htun HH, Biehl M, Petkov N (2023) Survey of feature selection and extraction techniques for stock market prediction. *Financ Innov* 9(1):26
- Jiang W (2021) Applications of deep learning in stock market prediction: recent progress. *Expert Syst Appl* 184:115537
- Singh R, Srivastava S (2017) Stock prediction using deep learning. *Multimed Tools Appl* 76:18569–18584. <https://doi.org/10.1007/s11042-016-41Payal>
- Soni P, Tewari Y, Krishnan D (2022) Machine Learning approaches in stock price prediction: a systematic review. In: *Journal of Physics: Conference Series* (Vol 2161, No. 1, p. 012065). IOP Publishing
- Jamous R, ALRahhal H, El-Darieby M (2021) A new ann-particle swarm optimization with center of gravity (ann-psocog) prediction model for the stock market under the effect of covid-19. *Scientific Programming*, 2021:1–17. <https://www.hindawi.com/journals/sp/2021/6656150/>
- Thakkar A, Chaudhari K (2021) Fusion in stock market prediction: a decade survey on the necessity, recent developments, and potential future directions. *Inf Fusion* 65:95–107
- Kumar D, Sarangi PK, Verma R (2022) A systematic review of stock market prediction using machine learning and statistical techniques. *Mater Today: Proceed* 49:3187–3191
- Mintarya LN, Halim JN, Angie C, Achmad S, Kurniawan A (2023) Machine learning approaches in stock market prediction: a systematic literature review. *Procedia Comput Sci* 216:96–102
- Krishnapriya CA, James A (2023) A survey on stock market prediction techniques. In: *2023 International Conference on Power, Instrumentation, Control and Computing (PICC)* (pp 1–6). IEEE
- Stoean C, Paja W, Stoean R, Sandita A (2019) Deep architectures for long-term stock price prediction with a heuristic-based strategy for trading simulations. *PLoS ONE* 14(10):e0223593
- Sedighi M, Jahangirnia H, Gharakhani M, Farahani Fard S (2019) A novel hybrid model for stock price forecasting based on metaheuristics and support vector machine. *Data* 4(2):75
- Nabipour M, Nayyeri P, Jabani H et al (2020) Deep learning for stock market prediction, entropy. *Multidiscip Digit Publish Inst (MDPI)* 22(8):1–23
- Pang X, Zhou Y, Wang P, Lin W, Chang V (2020) An innovative neural network approach for stock market prediction. *J Supercomput* 76:2098–2118
- Li X, Wu P, Wang W (2021) Incorporating stock prices and news sentiments for stock market prediction: a case of Hong Kong. *Image Process Manag Elsevier* 57(5):1–19
- Lu W, Li J, Wang J, Qin L (2021) A CNN-BiLSTM-AM method for stock price prediction. *Neural Comput Appl* 33:4741–4753
- Awan M, Shafry M, Nobaneh H, Munawar A et al (2021) Social media and stock market prediction: a big data approach. *Comput, Mater Contin* 67(2):2569–2583
- Ji Y, Liew AWC, Yang L (2021) A novel improved particle swarm optimization with long-short term memory hybrid model for stock indices forecast. *Ieee Access* 9:23660–23671
- Albahli S, Nazir T, Mehmood A, Irtaza A, Alkhalfah A, Albattah W (2022) AEI-DNET: a novel densenet model with an autoencoder for the stock market predictions using stock technical indicators. *Electronics* 11(4):611
- Hussain W, Merigó JM, Raza MR (2022) Predictive intelligence using ANFIS-induced OWA for complex stock market prediction. *Int J Intell Syst* 37(8):4586–4611
- Chhajer P, Shah M, Kshirsagar A (2022) The applications of artificial neural networks, support vector machines, and long-short term memory for stock market prediction. *Dec Anal J* 2:100015
- Bhambu A (2023) Stock Market prediction using deep learning techniques for short and long horizon. In: *Soft Computing for Problem Solving: Proceedings of the SocProsS 2022* (pp 121–135). Singapore: Springer Nature Singapore
- Sonkavde G, Dharrao DS, Bongale AM, Deokate ST, Doreswamy D, Bhat SK (2023) Forecasting stock market prices using machine learning and deep learning models: a systematic review, performance analysis and discussion of implications. *Int J Financ Stud* 11(3):94
- Agarwal V, Kumar PR, Shankar S, Praveena S, Dubey V, Chauhan A (2023) A deep convolutional kernel neural network based approach for stock market prediction using social media data. In: *2023 7th International Conference on Intelligent*

- Computing and Control Systems (ICICCS) (pp 78–82). IEEE. <https://doi.org/10.1109/ICICCS56967.2023.10142522>
- 29. Qiu Y, Song Z, Chen Z (2022) Short-term stock trends prediction based on sentiment analysis and machine learning. *Soft Comput* 26(5):2209–2224. <https://doi.org/10.1007/s00500-021-06602-7>
  - 30. Sahu AK, Gupta PK, Dohare AK, Singh AK, Mishra A, Rao A, Jha S (2023). Stock market prediction using machine learning. In: 2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN) (pp. 512–515). IEEE. <https://doi.org/10.1109/CICTN57981.2023.10140750>
  - 31. Das N, Sadhukhan B, Chatterjee T, Chakrabarti S (2022) Effect of public sentiment on stock market movement prediction during the COVID-19 outbreak. *Soc Netw Anal Min* 12(1):92
  - 32. Wu JMT, Li Z, Srivastava G, Tasi MH, Lin JCW (2021) A graph-based convolutional neural network stock price prediction with leading indicators. *Softw Practic Exp* 51(3):628–644
  - 33. Aldhyani TH, Alzahrani A (2022) Framework for predicting and modeling stock market prices based on deep learning algorithms. *Electronics* 11(19):3149
  - 34. Kumbure MM, Lohrmann C, Luukka P, Porras J (2022) Machine learning techniques and data for stock market forecasting: a literature review. *Expert Syst Appl* 197:116659. <https://doi.org/10.1016/j.eswa.2022.116659>
  - 35. Liu Q, Tao Z, Tse Y, Wang C (2022) Stock market prediction with deep learning: the case of China. *Finance Res Lett* 46:102209. <https://doi.org/10.1016/j.frl.2021.102209>
  - 36. Hochreiter S, Schmidhuber J (1997) Long short-term memory recognition. *Neural Comput* 9(8):1735–1780
  - 37. Chen Y, Fang R, Liang T, Sha Z, Li S, Yi Y, Song H (2021) Stock price forecast based on CNN-BiLSTM-ECA model. *Sci Program* 2021:1–20
  - 38. Yadav A, Jha CK, Sharan A (2020) Optimizing LSTM for time series prediction in Indian stock market. *Procedia Comput Sci* 167:2091–2100. <https://doi.org/10.1016/j.procs.2020.03.257>
  - 39. Zaheer S, Anjum N, Hussain S et al (2023) A multi parameter forecasting for stock time series data using LSTM and deep learning model mathematics. *Multidiscip Digit Publish Inst (MDPI)* 11(3):1–24
  - 40. Gupta U, Bhattacharjee V, Bishnu PS (2022) StockNet—GRU based stock index prediction. *Expert Syst Appl* 207:117986. <https://doi.org/10.1016/j.eswa.2022.117986>
  - 41. Bathla G, Rani R, Aggarwal H (2023) Stocks of year 2020: prediction of high variations in stock prices using LSTM. *Multimed Tools Applic* 82(7):9727–9743. <https://doi.org/10.1007/s11042-022-12390-5>
  - 42. Kaggle (2022) IBM Stock Analysis (LSTM), <https://www.kaggle.com/code/eslamreda0101/ibm-stock-analysis-lstm/data>, Last access 1 Jan 2023
  - 43. Manickamahesh N (2021) A study on technical indicators for prediction of select indices listed on NSE. *Turkish J Comput Math Educ (TURCOMAT)* 12(11):5730–5736
  - 44. Srivinay MBC, Kabadi MG, Naik N (2022) A hybrid stock price prediction model based on PRE and deep neural network. *Data* 7(5):51
  - 45. Hyndman RJ, Koehler AB (2006) Another look at measures of forecast accuracy. *Int J Forecast* 22(4):679–688
  - 46. Yahoo Finance (2022) Apple Inc. (AAPL), <https://finance.yahoo.com/quote/AAPL/history?p=AAPL>, Last access 1 Mar 2023
  - 47. Yahoo Finance (2022) Alphabet Inc. (GOOG), <https://finance.yahoo.com/quote/GOOG/history?p=GOOG>, Last access 1 Mar 2023

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”). Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

[onlineservice@springernature.com](mailto:onlineservice@springernature.com)