algorithms

MDPI

Article

A Novel Approach to Predict the Asian Exchange Stock Market Index Using Artificial Intelligence

Rohit Salgotra [1,2,*], Harmanjeet Singh [3], Gurpreet Kaur [4], Supreet Singh [5], Pratap Singh [6] and Szymon Lukasik [1]

[1] Faculty of Physics and Applied Computer Science, University of Kraków, 30-059 Kraków, Poland; slukasik@agh.edu.pl
[2] Data Science Institute, University of Technology Sydney, 15 Broadway, Ultimo 2007, Australia
[3] Chitkara University Institute of Engineering and Technology, Chitkara University, Chandigarh 140401, India; singh.harmanjeet@chitkara.edu.in
[4] Computer Science & Engineering Department, Punjabi University, Patiala 147002, India; gurpreetpunj144@gmail.com
[5] School of Computer Science, UPES, Dehradun 248007, India; uic.supreet@gmail.com
[6] Department of Computer Science & Engineering, Guru Jambheshwar University of Science & Technology, Hisar 125001, India; pratap.malik@gjust.org
* Correspondence: r.03dec@gmail.com or rohits@agh.edu.pl

Abstract: This study uses real-world illustrations to explore the application of deep learning approaches to predict economic information. In this, we investigate the effect of deep learning model architecture and time-series data properties on prediction accuracy. We aim to evaluate the predictive power of several neural network models using a financial time-series dataset. These models include Convolutional RNNs, Convolutional LSTMs, Convolutional GRUs, Convolutional Bi-directional RNNs, Convolutional Bi-directional LSTMs, and Convolutional Bi-directional GRUs. Our main objective is to utilize deep learning techniques for simultaneous predictions on multivariable time-series datasets. We utilize the daily fluctuations of six Asian stock market indices from 1 April 2020 to 31 March 2024. This study's overarching goal is to evaluate deep learning models constructed using training data gathered during the early stages of the COVID-19 pandemic when the economy was hit hard. We find that the limitations prove that no single deep learning algorithm can reliably forecast financial data for every state. In addition, predictions obtained from solitary deep learning models are more precise when dealing with consistent time-series data. Nevertheless, the hybrid model performs better when analyzing time-series data with significant chaos.

Keywords: disruptive series; deep learning; economic forecasting; stock market; time series

check for updates

Citation: Salgotra, R.; Singh, H.; Kaur, G.; Singh, S.; Singh, P.; Lukasik, S. A Novel Approach to Predict the Asian Exchange Stock Market Index Using Artificial Intelligence. *Algorithms* **2024**, *17*, 457. https://doi.org/10.3390/a17100457

Academic Editor: Stefano Mariani

Received: 11 July 2024
Revised: 2 October 2024
Accepted: 4 October 2024
Published: 15 October 2024

Time-series data are among the most frequently gathered forms of information. This particular data type has the potential to be employed in the construction of a model that can accurately depict the actions of a dynamic system [1]. Many approaches exist to accomplish this, each of which has undergone comprehensive research and demonstrated practical utility [2,3]. A primary challenge encountered in time-series analysis, prevalent in numerous fields, including computing, statistics, and banking, is selecting the most accurate forecasting model from historical data.

Many methodologies and strategies have been devised to construct prediction models from time-series data to approximate their future motions as stated in the introductory paragraph (see, for instance, references [4,5]. Deep learning is among the most efficient techniques for predicting time series. Artificial neural networks are helpful in forecasting time series because the models can autonomously identify detailed patterns within the data and circumvent conventional statistical techniques. When predicting time series, RNNs and variations on them, such as LSTM networks, demonstrate excellent performance. Recurrent neural networks, or RNNs, are algorithms specifically built to analyze sequential data, including time series. These models do this by preserving a hidden state that contains all the observations made up to this point [6,7]. In training deep networks using sequential data, vanishing gradients may manifest. In response, LSTM networks, derivatives of RNNs, were devised [8]. Time-series data can be enhanced with the temporal feature extraction capabilities of CNNs, while long-range correlations between time steps can be discovered with transformer networks. One can use either of these methods to improve their time-series data. However, while most people agree that data attributes should guide model selection for time-lapse forecasting in deep learning, there are exceptions; the extensive evaluation of various models has sparked debates regarding the most suitable architecture or model for such data [9]. Although transformer models have shown great success in natural language processing (NLP) and other domains, they are not always the best choice for predicting stock exchange closing prices. Stock price data are time-series based, meaning they have a natural sequential structure, where past values strongly influence future ones [10,11].

Traditional models like ARIMA, LSTM, or GRU are explicitly designed to handle this sequential dependence. While transformers can process sequences, their parallelized approach could be more intuitive for handling long-term temporal dependencies than models specifically designed for time-series data. Further, market volatility makes stock price data noisy and often unpredictable. With their complex architecture and large number of parameters, transformers may be more prone to overfitting on these noisy data, especially if the dataset is not large or diverse enough [12,13]. It is crucial that time series exhibit periodicity. Seasonal decomposition and ARIMA are two techniques that may prove beneficial when the time series demonstrates a clear and uninterrupted periodic pattern. Complex models, such as those comprising multiple deep networks, may improve performance when the time series displays irregular patterns or cycles. Aspects that include the complexity, accessibility, and necessary interoperability of the data may affect the final determination [14]. Consequently, the model should align with the research's data and objectives. The fundamental goal of this endeavor is to improve the accuracy of financial time-series forecasts by applying deep learning models. Finding out which deep learning model excels in predicting financial time-series observations is the driving force for this study. This study will employ a multivariate hybrid deep learning model capable of handling multivariable input and predicting the close price of the stock index. Unlike univariate models, multivariate models yield higher accuracy predictions for dynamic systems that comprise multiple interconnected variables, such as financial systems [15] and capital market systems [16]. This is the primary rationale for selecting this methodology for the present study.

Furthermore, predictions will be made simultaneously on all variables for the deep learning models that will be assessed. One potential approach to improve the prediction accuracy is to employ a trained deep learning network that simultaneously generates predictions and records the relationship between variables [17]. A "multivariate" approach integrates the above-mentioned strategies into the deep learning model. Forecast accuracy may be affected by both the features of time-lapse data and the design of a deep neural network model; this is another research objective. A literature summary sets the stage for the study's description of the research methodologies used to compare and contrast the observed deep neural network models' performance.

To determine the most appropriate neural network technique for predicting the closing price of a stock market index from an OHLCAV (Open-High-Low-Close-Average-Volume) time-series dataset, which includes data points such as the stock's opening price, highest price, lowest price, closing price, adjusted closing price, and the volume of stock traded in a single day, Section 3 establishes the tests and analyses, and examines the evaluation outcomes. The study will be concluded in Section 4 through a comprehensive summary of its main findings, some concluding remarks, and a delineation of its long-term objectives and approaches for future research.

1. Literature Review

This study will discuss a substantial issue related to the research topic. The problem is examining the methodology employed to forecast the economy's condition. The primary challenge in time-series forecasting is to employ practical approaches that align with the characteristics of the data. Employing conventional statistical methods like regression analysis and ARIMA models may pose challenges because of the non-linear characteristics of financial data. Given the possibility of inaccurate results, it may be unwise to depend extensively on these approaches for forecasting the economy's future performance. Numerous scholars have dedicated significant time and effort over the past several years to employing sophisticated methods to predict the stock markets of foreign countries. Global stock exchange market datasets are the basis for numerous academic research endeavors.

Several advanced deep learning architectures and models have been developed in recent years as feasible solutions for forecasting time-series data and other applications. These models and architectures have been produced [18,19]. A frequently chosen architecture for deep learning applications that involve predicting time-series data is the Convolutional RNN and Convolutional LSTM systems. According to studies by [20], using LSTM to forecast the flow of financial data produces highly accurate results. The CNN is an important deep learning architecture that, along with the LSTM, can predict the future development of time-series data [21]. The main impetus behind the development of CNN was to enhance the efficiency of data feature extraction to facilitate classification [22]. In contrast, several studies have shown that CNNs can effectively make precise predictions on time-series data [23].

Several other deep learning models, such as ConvRNN [24], ConvLSTM [25], ConvGRU [26], Bi-directional ConvRNN [27], Bi-directional ConvLSTM [28], and Bi-directional ConvGRU [29] integrate LSTM and CNN characteristics. LSTM is designed to store data motion patterns, and CNN is designed to extract features. Integrating the two models is to take advantage of each model's qualities. As a consequence of this integration, the precision of time-series forecasts will subsequently improve. This study aligns with previous research carried out in the field of time-series prediction and uses the integrated deep learning approach. Ref. [30] also confirmed the effectiveness of this strategy in predicting air quality. Furthermore, previous studies have indicated that CNN and LSTM can accurately predict the onset of COVID-19 [31]. The researchers at [32] conducted a study to validate the effectiveness of various models for complex wind speed forecasting and specifically explored using mixed deep learning algorithms applied to wind speed data. A further study that used electrical load forecasting data [33] demonstrated similar findings, as it evaluated and empirically analyzed the most efficient deep learning models suitable for predicting short-term load.

This article evaluates six prominent deep learning models: the Convolutional RNN, Convolutional LSTM, Convolutional GRU, Bi-directional ConvRNN, Bi-directional ConvLSTM, and Bi-directional ConvGRU. These models can predict time series and may be applied to datasets with multiple variables. References to earlier research and the characteristics of the models served as inspiration for this choice. Figure 1 presents a comparative examination of the six deep learning models' input/output flow topologies.

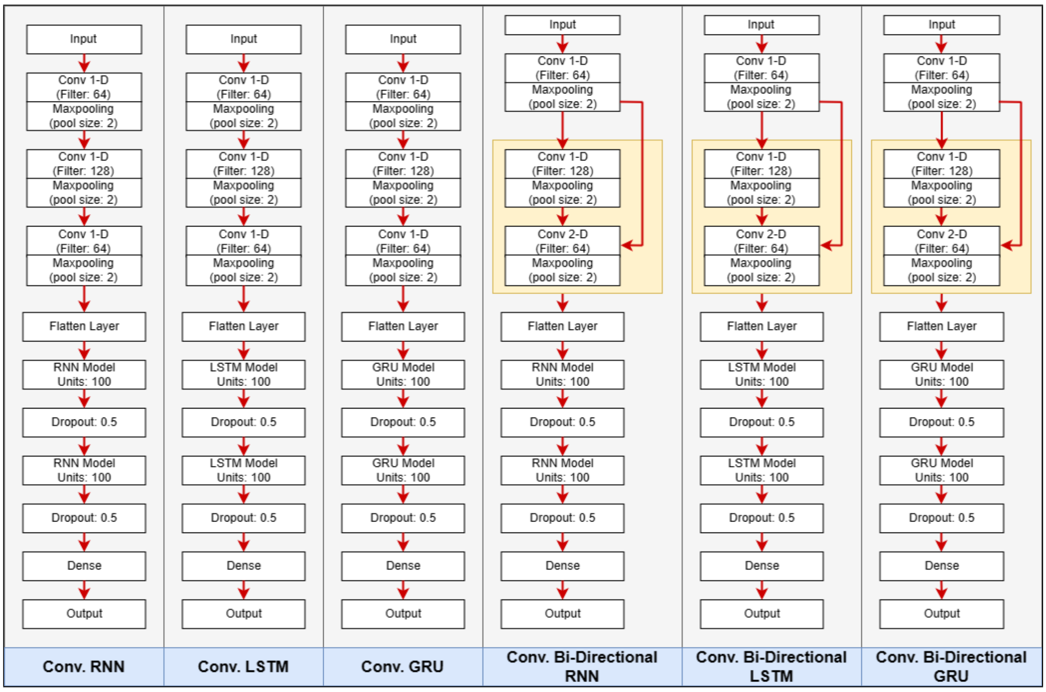

Input
Conv 1-D (Filter: 64) Maxpooling (pool size: 2)
Conv 1-D (Filter: 128) Maxpooling (pool size: 2)
Conv 1-D (Filter: 64) Maxpooling (pool size: 2)
Conv 2-D (Filter: 64) Maxpooling (pool size: 2)
Flatten Layer
RNN Model Units: 100
LSTM Model Units: 100
GRU Model Units: 100
Dropout: 0.5
Dense
Output
Conv. RNN
Conv. LSTM
Conv. GRU
Conv. Bi-Directional RNN
Conv. Bi-Directional LSTM
Conv. Bi-Directional GRU

Figure 1. Architecture of Conv. RNN, Conv. LSTM, Conv. GRU, Conv. BRNN, Conv. BLSTM, and Conv. BGRU.

1.1. Deep Neural Network

This part of the study summarizes the frameworks employed for deep learning, specifically Convolutional RNNs, Convolutional LSTM, and Convolutional GRUs, listed in that sequence.

1.1.1. Convolutional Neural Network

A certain kind of neural network, termed Convolutional CNNs, is distinguished by its grid-based topology and the application of mathematical convolution to analyze input data [34]. Multiple-layered Convolutional CNN is utilized in natural language processing applications to extract local features. These networks process the input features using linear filters through the convolution process [35].

1.1.2. Recurrent Neural Network (RNN)

An RNN, short for recurrent neural network, is a specialized neural network explicitly designed for sequence modeling. A directed graph is a representation that can be used to characterize the connections that exist between neurons in an RNN [36]. The RNN is particularly suitable for Natural Language Processing (NLP) applications because it can analyze input sequences using its internal state. The outputs of the RNN are generated by iteratively applying the identical function for every occurrence using the identical approach. Calculations that have been used before are employed to determine the performance. Within the RNN design, the input length is utilized to ascertain the size of each separate time step. The data given to the architecture at time step t are denoted as x_t, whereas the hidden state at time step t is denoted as s_t:

$$s_t = f(Ux^t + Ws^{t-1} + b) \tag{1}$$

Utilizing the hidden state from the previous time phase and the current input, as referenced in the study [37], it is possible to compute the current hidden state s_t using Equation (1) given in the previous sentence. The input and recurrent of the cell at time t are represented by the symbols x^t and s^{t-1}, respectively. The weights that are shared over time are denoted by the symbols U and W, while the bias is represented by b. The pointwise non-linear activation function is represented by the function f. Regrettably, the

RNN cannot identify the relationship between pertinent information when the time interval is extensive. Long Short-Term Memory is suggested by the study [38] as a solution to the "long-term dependency" issue.

1.1.3. Long Short-Term Memory (LSTM)

An LSTM is an architecture for deep neural networks that uses forget gates to solve the problems of vanishing gradients or bursting. Ref. [39] states that it is essential to note that, in contrast to standard recurrent neural network topologies, LSTM facilitates loss backpropagation over a restricted amount of time steps in numbers. A standard LSTM unit comprises three gates: an entry gate, an exit gate, and a gate that can be forgotten. The cell itself is also included in this unit. Through monitoring the gates' operation, the cell can determine which data should be stored and when information should be granted to other units. The LSTM transition is built on the foundation established by the subsequent equations [38]:

$$i_t = \sigma(W^i X_t + U^i h_{t-1} + b^i) \tag{2}$$

$$f_t = \sigma(W^f X_t + U^f h_{t-1} + b^f) \tag{3}$$

$$o_t = \sigma(W^o X_t + U^o h_{t-1} + b^o) \tag{4}$$

$$u_t = tanh(W^u X_t + U^u h_{t-1} + b^u) \tag{5}$$

$$c_t = i^{\circ}{}_t u_t + f^{\circ}{}_t c_{t-1} \tag{6}$$

$$h_t = o^{\circ}{}_t tanh(c_{t-1}) \tag{7}$$

The input vector to the LSTM unit is denoted as x_t, the activation vector for the forget gate is denoted as f_t, the activation vector for the input gate is denoted as i_t, and the activation vector for the output gate is denoted as o_t.

The cell state vector is denoted as c_t, whereas the hidden state vector is represented as h_t. In this model, the letter W represents the weight matrices, whereas the letter b represents the parameters of the bias vector [40]. Bi-LSTM employs bidirectional hidden layers to encompass both historical and future information. As a result, there is an enhanced acquisition of knowledge inside the network and a more robust and reliable exchange of information in both directions over time.

1.1.4. Gated Recurrent Unit (BGRU)

Nevertheless, an alternative artificial neural network (ANN), Gated Recurrent Unit (GRU), is a type of architecture based on recurrent neural networks (RNNs) as proposed by [41] in 2014. The LSTM architecture effectively maintains the weight of long-term memory, resolving the disappearing gradient problem in RNN. Nevertheless, LSTM requires extensive computations due to its intricate architecture. In most GRU architectures, an update gate is utilized instead of an LSTM. The modification is incorporated into the GRU design based on the subsequent equations [42]:

$$z_t = \sigma(U_z x_t + W_z h_{t-1}) \tag{8}$$

$$r_t = \sigma(U_r x_t + W_r h_{t-1}) \tag{9}$$

$$s_t = tanh(U_s x_t + W_s.(h^{\circ}{}_t h_{t-1})) \tag{10}$$

Let x_t represent the input vector, h_t represent the output vector, r_t represent the reset gate vector, z_t represent the update gate vector, and W, U, and b represent the parameter matrices and vectors, respectively.

2. Proposed Methodology

An extensive literature review is conducted to identify several effective deep learning architectures commonly employed for time-series data modeling and prediction. The subsequent phase involves collecting time-series data from actual-world situations after a diverse array of architectures has been acquired. The third stage of this study consists of pre-processing the collected dataset, which involves normalizing the values based on the various ranges of current stock indices values. This article proposes a hybrid architecture to predict the close price of the Asian Stock Exchange. The architecture incorporates the random and temporal fluctuations of the exchange rate and the vulnerability to external market dynamics. A two-stage feature extraction technique is employed on the dataset to streamline the architecture's creation process and improve its capacity to accurately predict outcomes beyond the data it was trained on. This is necessary because of the abundance of influential factors and the unpredictable nature of exchange rates. One application of Auto-Encode (AE) is to simplify complex elements involved in influencing. Due to the volatility of stock exchange rates, the Self-Organize Map (SOM) is employed to cluster related data from the training set into one category. The Convolutional Neural Network (CNN) addresses traditional neural networks' limitations in effectively extracting features from time-series data. Due to its capacity to accurately predict based on time-series data, it is a highly potent tool. Bi-directional Gated Recurrent Unit is recognized as a very accurate and effective technique for non-linear integration, as it excels in accurately considering the weight of each propagation to predict the close price of Asian stock indices as shown in Figure 2.

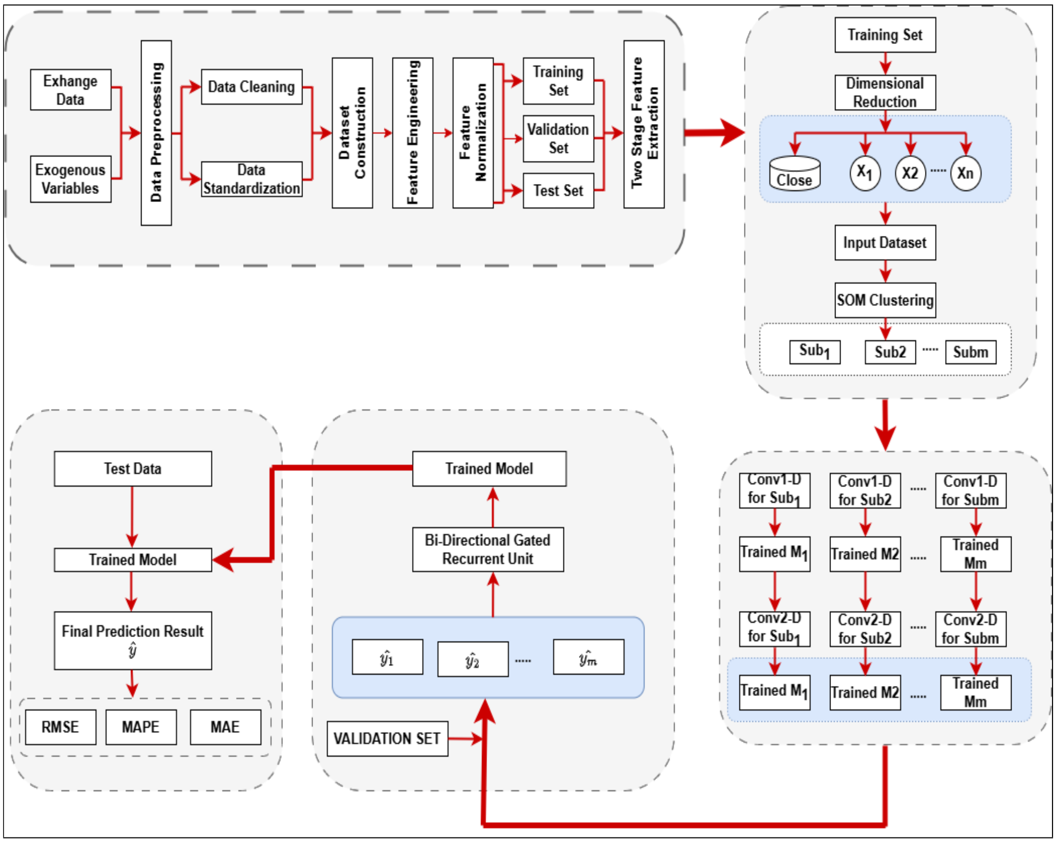


Figure 2. Proposed system architecture.

For this study, Yahoo Finance https://finance.yahoo.com/ (accessed on 1 April 2024) is the source of the time-series data obtained for each of the six Asian stock markets with corresponding tickers BSE (BSESN), Hongkong (HSI), Taiwan Index (TWII), Japan Index (JPXGY), Indonesia Index (JKSE), and Korea Index (KS11). The reason for choosing Yahoo Finance over other sources is its availability of an open-source library named "YFinance". Compared to many paid services, YFinance is free, making it an appealing option for researchers with limited resources. Further, YFinance seamlessly integrates with Python version 3.6 a popular language among data scientists and researchers. This allows users to easily combine it with other Python libraries, such as Pandas, NumPy, and Matplotlib, for data analysis and visualization.

The dataset contains six features: open price, close price, high price, low price, adjusted close price, and volume. During this investigation's third and final part, the dataset obtained is modified to correspond to the various percentage ranges of stock market index values. Pre-processing is the term that is most generally used to describe this method. In the following step, the dataset passes through the feature normalization and scaling, followed by the processed data being separated into two unique sets: training and testing. Before applying the models to time-series data that contain several variables, the subsequent stage is to construct and refine the deep learning structures that have been chosen. The training dataset is now fed into the series of deep learning models named the convolutional recurrent neural network (ConvRNN), Convolutional LSTM model, Convolutional GRU model, Convolutional Bi-directional RNN model, Convolutional Bi-directional LSTM model, and Convolutional Bi-directional GRU model. After completing the training phase of all six deep learning models, followed by a performance evaluation, the trained models are tested on the stock exchange's test data to predict the exchange's index value. The conclusion stage of this process involves identifying deep learning models and determining their structure to achieve the best level of prediction accuracy and lowest Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE).

The approach outlined in this study differs from the non-parallel prediction strategy, which involves creating individual forecasts for each variable of interest. Predictions created in parallel utilizing data significantly diverge from those generated in a non-parallel manner due to the intricate nature of the procedure. A multivariate parallel approach incorporates a complete model that receives multivariable inputs and generates multivariable outputs, in contrast to non-parallel prediction, which predicts each relevant variable individually. The prediction procedure is executed in parallel for all variables. Parallel prediction involves the execution of the prediction process for all variables simultaneously, while non-parallel prediction involves the execution of the process for each variable of concern individually. By integrating a well-trained model that establishes a correlation between variables into the prediction process, the prediction accuracy can be improved, thereby making prediction advantageous.

3. Dataset

As mentioned earlier, this study utilizes time-series data from the financial field. Based on results from the prior research [15,29] that showed the interdependence of these stock indices' movements, this study integrates data from six stock indices into a single multivariate dataset. Utilizing a multivariate modeling technique considers the impact and interrelation of various stock markets, leading to more accurate outcomes. Figure 3 shows the daily fluctuations of stock market indexes in India, while Figure 4 displays similar data from Hong Kong. Figure 5 presents the data from Taiwan, and Figure 6 illustrates the data from Indonesia. Figure 7 presents the data from Japan, and Figure 8 illustrates the data from Korea.

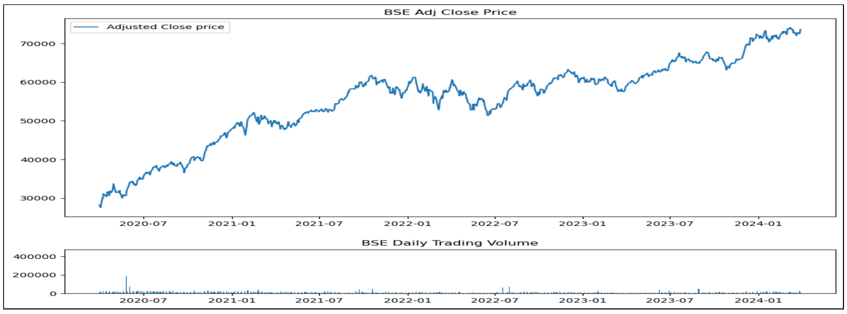


Figure 3. The BSE index from the first of April 2020 to the last day of March 2024.

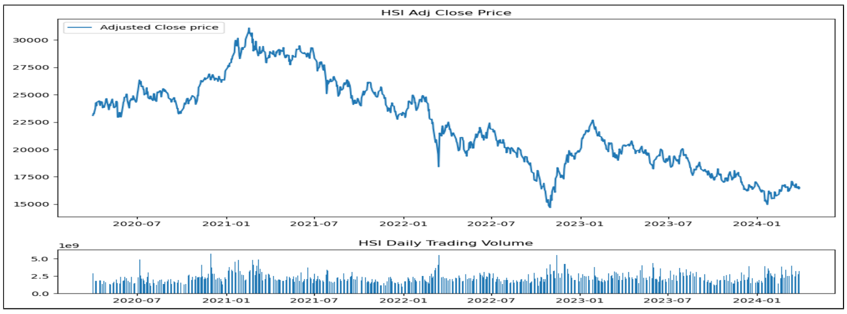


Figure 4. The HSI index from the first of April 2020 to the last day of March 2024.

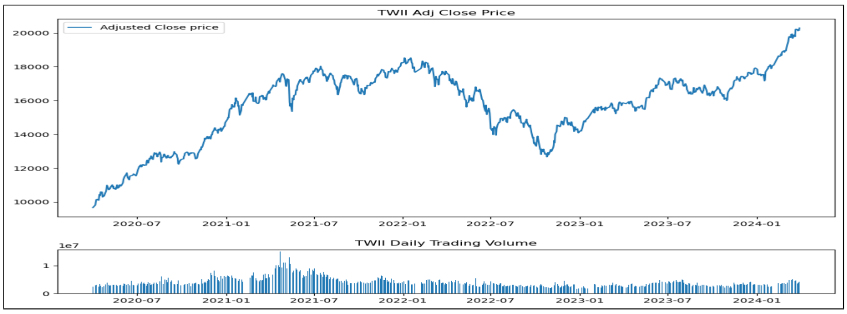


Figure 5. The Taiwan index from the first of April 2020 to the last day of March 2024.

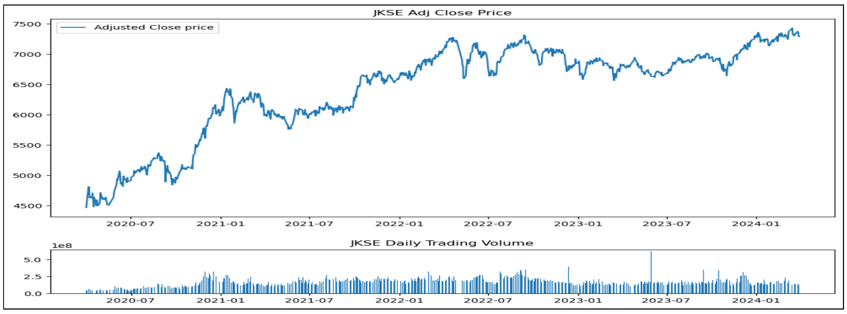


Figure 6. The JKSE index from the first of April 2020 to the last day of March 2024.

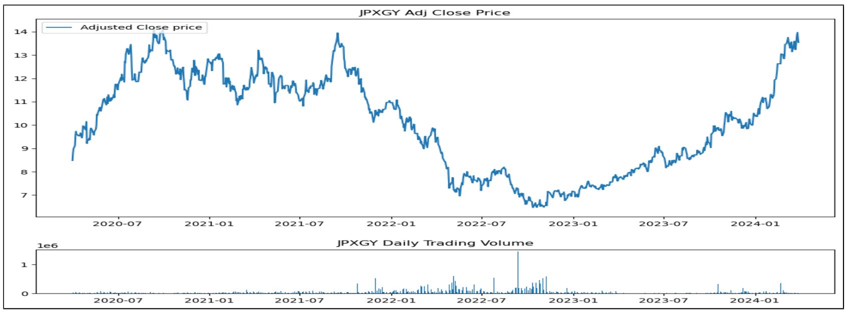


Figure 7. The JPXGY index from the first of April 2020 to the last day of March 2024.

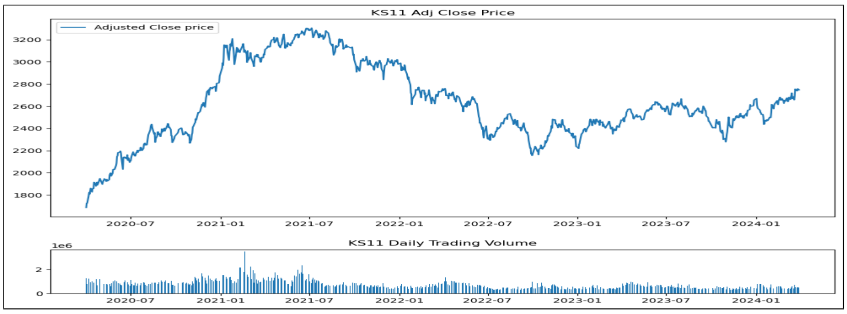


Figure 8. The KS11 index from the first of April 2020 to the last day of March 2024.

Information was gathered between 1 April 2020 and 31 March 2024. As a result of the COVID-19 pandemic in 2020, the financial market indexes went through a period of highly unpredictable volatility throughout the year, prompting us to select our era. The authors can employ comprehensive data from the whole duration of the COVID-19 pandemic, which had a profound influence on the global economy, to assess the effectiveness of the statistical deep neural network model. Figures 3–8 illustrate the financial data used in this experiment. They comprise roughly 770–800 data points, corresponding to the total number of trades from April 2020 to 31 March 2024 as mentioned in Table 1.

Table 1. Data points taken from six Asian stock exchanges.

Stock Exchange	Country	Start Date	End Date	Points	Indicators
BSE	India	1 April 2020	31 March 2024	992	OHLCAV
HSI	Hong Kong	1 April 2020	31 March 2024	986	OHLCAV
TWII	Taiwan	1 April 2020	31 March 2024	974	OHLCAV
JKSE	Indonesia	1 April 2020	31 March 2024	970	OHLCAV
JPXGY	Japan	1 April 2020	31 March 2024	1003	OHLCAV
KS11	Korea	1 April 2020	31 March 2024	986	OHLCAV

3.1. Training–Testing

For training purposes, 80% data points are used out of roughly 770–800 data points for different exchange markets that represent the daily progression of stock market index values from 1 April 2020 to 31 March 2024. Using this information and up to nineteen data points daily, forecasts regarding the stock market index value fluctuations during March 2024 are created. This study utilizes the Python artificial intelligence libraries such as PyTorch, Scikit-Learn, TensorFlow, and Keras to create six deep learning models: ConvRNN, ConvLSTM, ConvGRU, ConvBRNN, ConvBLSTM, and ConvBGRU. The experimental methods are conducted as an ancillary inquiry utilizing a personal computer with an Intel Core i7 CPU, 32 gigabytes of random access memory (RAM), and the Windows 11 operating system. At the same time, the exhaustive parameter settings for every single deep learning model utilized in this investigation are presented in Table 2. For Convolutional RNN, Convolutional LSTM, and Convolutional GRU, we use a single 1-D layer of 64 distinct convolutional filters with a maximum pool size of 2 followed by another 1-D CNN layer of 128 distinct convolutional filters with a pool size of 2 and another 1-D CNN layer of 64 distinct convolutional filters with a pool size of 2. For Convolutional Bi-directional RNN, Convolutional Bi-directional LSTM, and Convolutional Bi-directional GRU, we use a single 1-D layer of 64 distinct convolutional filters with a maximum pool size of 2 followed by another 1-D CNN layer of 128 distinct convolutional filters with a pool size of 2 and another 2-D CNN layer of 64 distinct convolutional filters with a pool size of 2. We use the same stride of 3 for every model to preserve more information in the pattern. We use a minimal learning rate of 1e-5 followed by the RELU activation function to minimize the error rate at every weight updation for each model. Further, every model has processed the training samples hundreds of times and updated their weights based on each sample's error. The training samples, forty in number, are used in the forward pass for Conv. RNN, Conv. LSTM, and Conv. GRU, whereas forty training samples are used in forward and backward passes for Conv. BRNN, Conv. BLSTM, and Conv. BGRU. The choices of parameter values in all the six deep learning models for predicting BSE stock prices [43] and in the studies [41,44] on how people moved around during the COVID-19 pandemic and how deep learning models were used to predict Asian stock market indices show how parameters are set up and structured.

Each of the six models is trained using a subset of the collected financial time-series dataset. The subsequent stage involves comparing the current values of each stock market index with their respective forecasts. It is determined that the Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE) are the most effective methods for determining the accuracy of each model's predictions in this investigation [45]. This study evaluates the deep learning models' performance using one-step prediction accuracy instead of multistep predictions. Pragmatic considerations drive the decision to use a one-step prediction technique. Due to the inclination of stock market investments to prioritize short-term goals, it is crucial to have more accurate projections for this specific time frame.

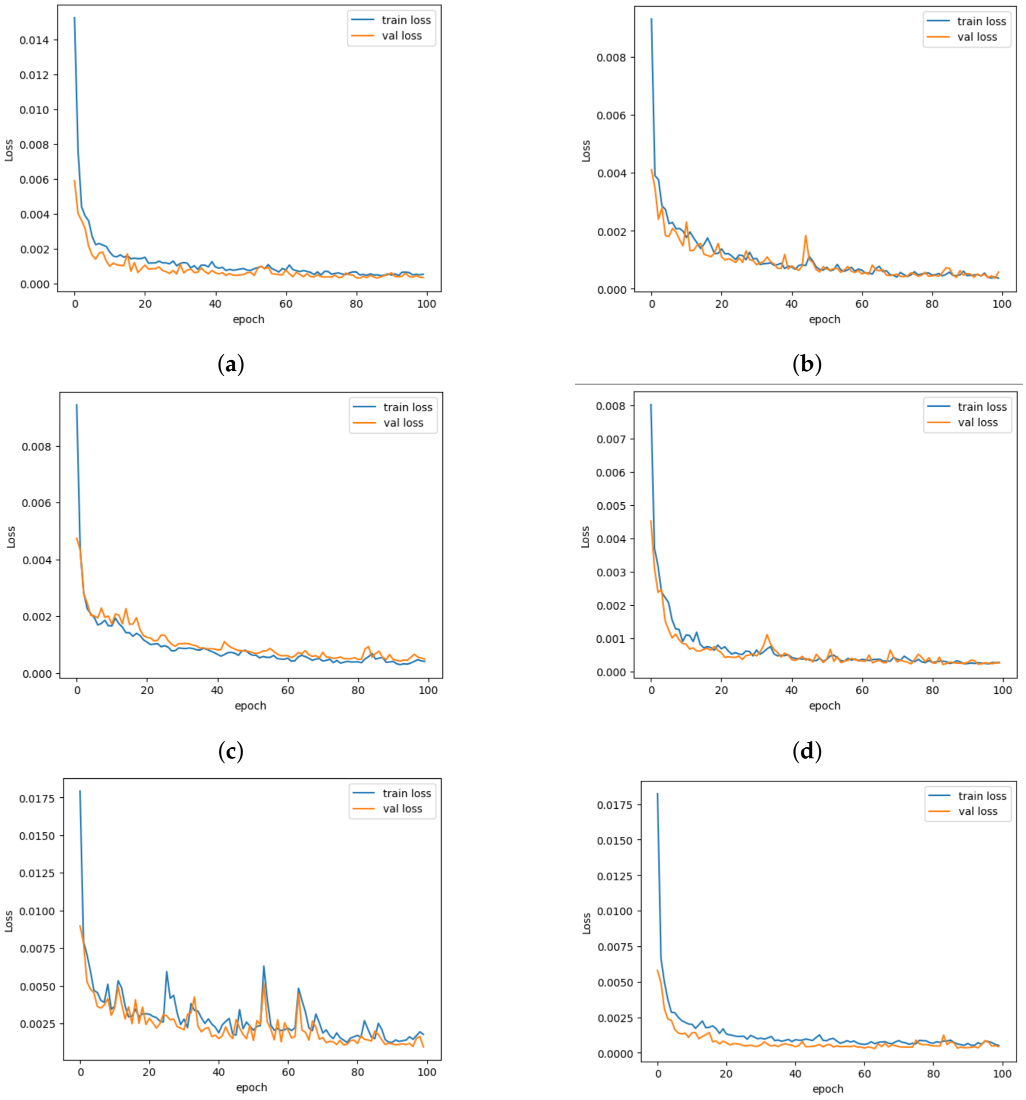
Table 2. Parameters for the ongoing research on the six-dimensional deep learning model.

Parameter	Conv. RNN	Conv. LSTM	Conv. GRU	Conv. BRNN	Conv. BLSTM	Conv. BGRU
Conv. layer filters	128	128	128	128	128	128
Conv. layer kernel size	3	3	3	3	3	3
Stride	2	2	2	2	2	2
Pooling	2	2	2	2	2	2
Learning Rate	1e-5	1e-5	1e-5	1e-5	1e-5	1e-5
Conv. Activation function	RELU	RELU	RELU	RELU	RELU	RELU
LSTM layer hidden unit	100	100	100	100	100	100
LSTM Activation Function	Linear	Linear	Linear	Linear	Linear	Linear
LSTM Optimizer	Adam	Adam	Adam	Adam	Adam	Adam
Loss function	MSE, MAE	MSE, MAE	MSE, MAE	MSE, MAE	MSE, MAE	MSE, MAE
Epochs	100	100	100	100	100	100
Dropout layer	0.5	0.5	0.5	0.5	0.5	0.5
Batch Size	40	40	40	40	40	40

3.2. Results and Discussion

The comparative analysis of training loss and value loss for all six Asian indices shown in Figure 9a–f is achieved by the Bi-Directional Gated Recurrent Unit (BGRU) and evaluated during the training phase. The training phase for all indices is executed with 100 epochs in BGRU, and it is observed that Figure 9a,c,f show almost the same loss at the 100th epoch. The decrease in loss during training is depicted for the BSE, HSI, TWII, JKSE, JPXCY, and KS11 exchange, respectively. The study employs data gathered between 1 April 2020 and 31 March 2024 to build the BGRU model. Upon further scrutiny of Figure 9b, it becomes apparent that the loss value tends to stabilize after the 80th iteration (epoch) when all deep learning models have an epoch set to 100 to complete the training as indicated in Table 2. In addition, Figure 9a,f demonstrate that the BGRU model achieves the fastest stability in reducing the loss and value for HSI and KS11 exchange, while the BGRU model achieves the slowest stability in the case of JKSE.

From Figure 10a–f, the predictions made by the BGRU model for all six Asian exchange markets stock markets on 31 March 2024 are shown. To summarize, the assessment of the six deep neural network models demonstrates the ability to accurately replicate the observed patterns of index value fluctuations in the six Asian stock exchanges. On the other hand, the expected yield curve shows a significant difference compared to the original value curve. Compared to the different exchange markets, the BGRU model produces the TWII index curve, which exhibits the most significant deviation from the original value curve in Figure 10c. Conversely, the BGRU model produces the BSE prediction curve that resembles the original value curve in Figure 10a. Figure 10c displays a variety of possibilities depicted on the Taiwan stock market chart. The prediction curve of TWII has the highest level of deviation from the original value on this graph compared to the curves of the other exchange markets. The BGRU model accurately creates the prediction curve for the KS11 as shown in Figure 10f. In addition, BGRU achieves the highest accuracy in predicting the curve for JKSE, while the BSE exchange remains an excellent performer for BGRU. These data show that none of the six Asian exchange markets under study have the highest level of prediction accuracy.



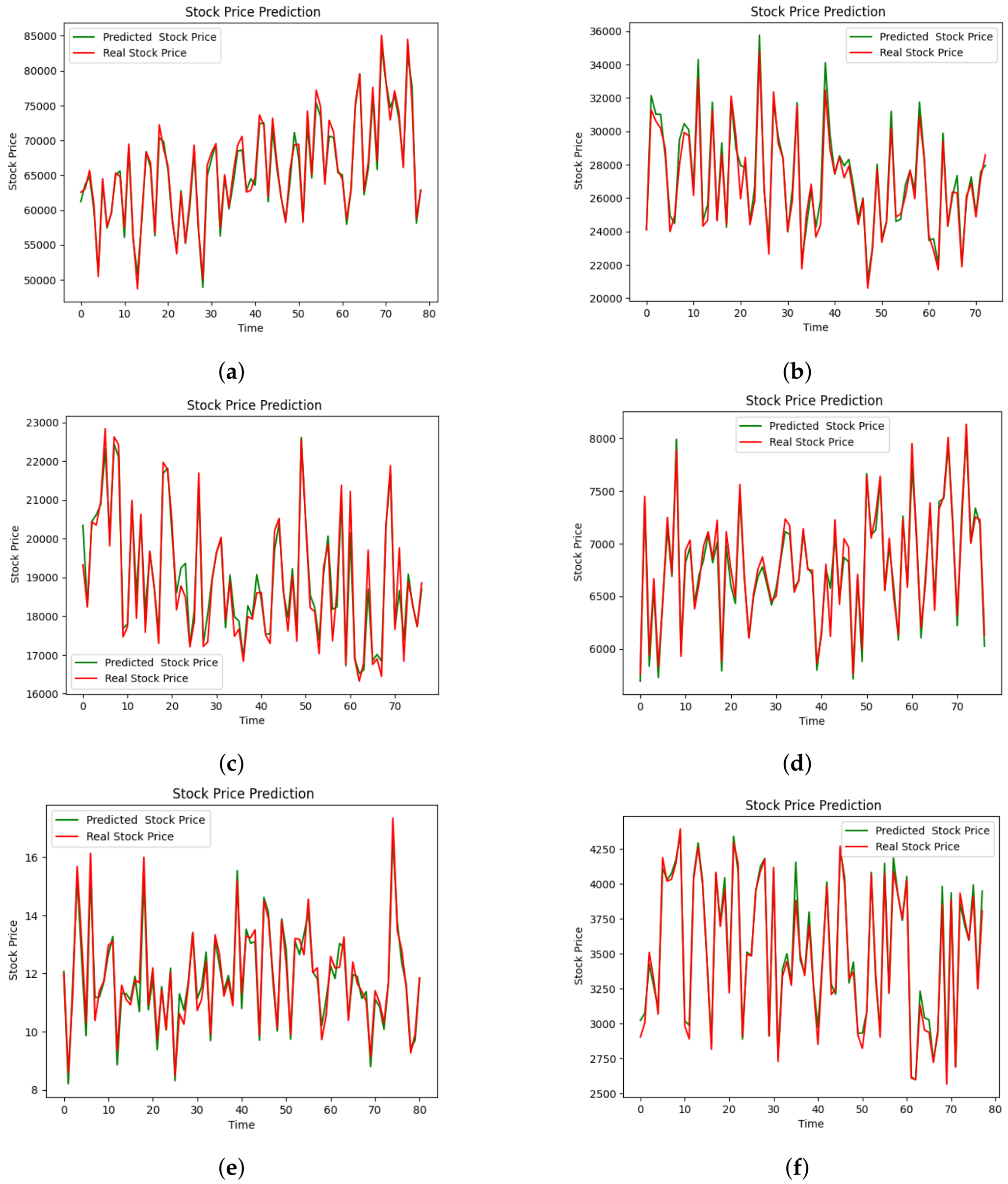
(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure 9. Training and value loss of all six Asian indices with BGRU model training. (**a**) Training and value loss for BSE index with BGRU Model; (**b**) training and value loss for HSI index with BGRU Model; (**c**) training and value loss for TWII index with BGRU Model; (**d**) training and value loss for JKSE index with BGRU Model; (**e**) training and value loss for JPXGY index with BGRU Model; (**f**) training and value loss for KS11 index with BGRU Model.

Further, the authors employ all six deep learning models to evaluate comparative loss and value on the BSE exchange market throughout the training phase. Figure 11 illustrates the comparative loss and value decrease accomplished in these models. The relative decreases in training loss are shown in Figure 11a–f for the Conv. RNN, Conv. LSTM, Conv. GRU, Conv. BRNN, Conv. BLSTM, and Conv. BGRU models that are used. The BSE exchange is created using data collected from 1 April 2020 to 31 March 2024, as stated in the paper. Table 2 demonstrates that all deep learning models have an epoch value of 100. Moreover, upon careful examination of Figure 11, it is evident that the loss value tends to reach a state of stability around the 91st iteration (epoch). Also, Figure 11b and f show that the Conv. LSTM and Conv. BGRU models have the fastest loss and value reduction, which shows that they are stable, especially when compared to the Conv. RNN and Conv. LSTM models. Conversely, Conv. RNN exhibits the slowest rate of achieving stability.



(**a**)

(**b**)

(**c**)

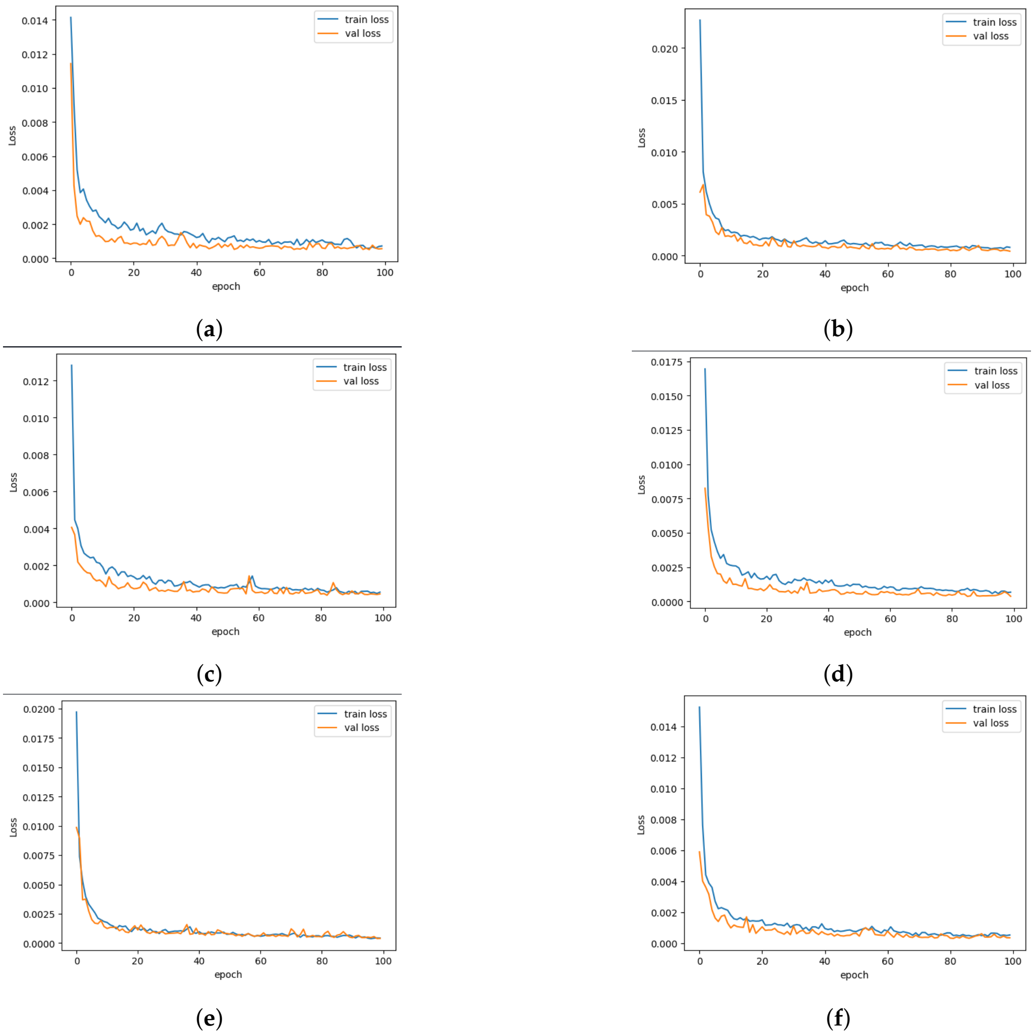
(**d**)

(**e**)

(**f**)

Figure 10. Actual and prediction values of all six Asian indices with BGRU model. (**a**) Actual and prediction values for BSE index with BGRU model; (**b**) actual and prediction values for HSI index with BGRU model; (**c**) actual and prediction values for TWII index with BGRU model; (**d**) actual and prediction values for JKSE index with BGRU model; (**e**) actual and prediction values for JPXGY index with BGRU model; (**f**) actual and prediction values for KS11 index with BGRU model.

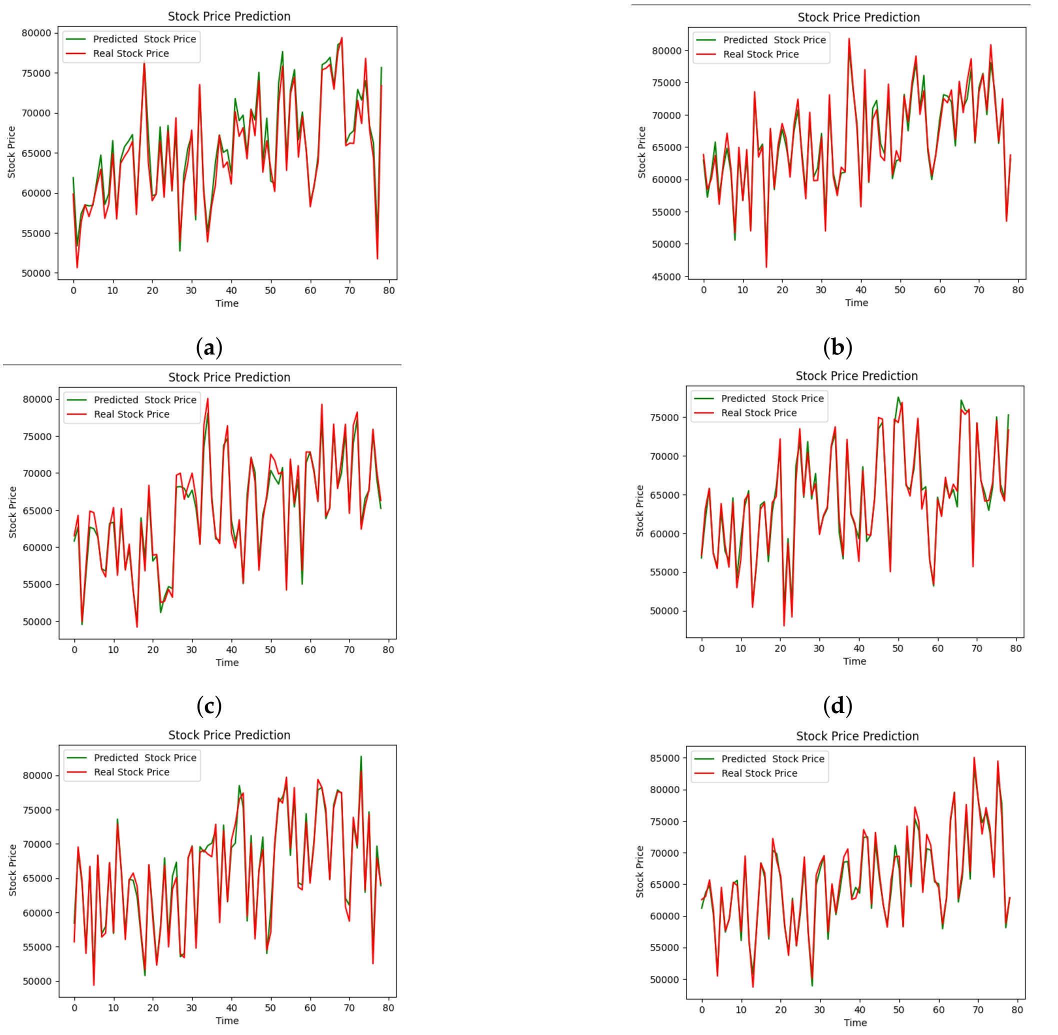
Further, the six assessed deep learning models successfully replicate the observed patterns of index value fluctuations in the six stock markets shown in Figure 12a–f. On the other hand, when compared to the initial value curve, the predicted yield curve demonstrates a considerable divergence. The Convolutional RNN model generates the BSE index curve that displays the most substantial deviation from the initial value curve compared to the additional models. The Convolutional BGRU model, on the other hand, generates the BSE prediction curve, which is comparable to the value curve that is first determined in Figure 12a. Figure 12 displays various possibilities on the BSE stock market chart. The prediction curve of Convolutional LSTM shows the most departure from the beginning value on this graph compared to the curves of the other models. The Convolutional Bi-directional RNN model accurately creates the prediction curve for the BSE as shown in Figure 12d. In addition, the Convolutional BGRU model has the highest level of accuracy in predicting the curve for BSE. In contrast, the Convolutional BLSTM model remains the top performer in performance for BSE. The statistics presented here demonstrate that none of the six investigated deep learning models achieve the highest accuracy regarding the generation of predictions.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 11. Training and Value Loss for BSE index with BGRU, Conv. LSTM, Conv. GRU, Conv. RNN, Conv. BLSTM, and Conv. BGRU models. (**a**) Training and value loss for BSE index with Conv. RNN; (**b**) training and value loss for BSE index with Conv. LSTM; (**c**) training and value loss for BSE index with Conv. GRU; (**d**) training and value loss for BSE index with Conv. BRNN; (**e**) training and value loss for BSE index with Conv. BLSTM; (**f**) training and value loss for BSE index with Conv. BGRU.

Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are the metrics that are utilized to measure the disparities that exist between the values that were anticipated and those that were observed. Tables 3 and 4 illustrate these distinctions in a graphical format. The fact that the error rates for RMSE and MAPE are comparable is illustrated by this, which shows that standalone models can deliver findings that are superior to those produced by combining models. Table 3, which can be located elsewhere, presents a comparison of the error rates that are observed during the training and testing stages of the process. During the training phase of the calculation, it is not unusual for the error rates to increase when dealing with time-series prediction. This is because of the nature of the problem. The fact that this particular case exists proves that this pattern does exist. The Convolutional LSTM algorithm has a value of 0.1404 in terms of the Root Mean Squared Error (RMSE), the Convolutional GRU algorithm has a value of 0.1582, the Convolutional RNN algorithm has a value of 0.1450, the Convolutional BLSTM algorithm has a value of 0.1494, the Convolutional BGRU algorithm has a value of 0.1205, and the Convolutional RNN algorithm has a value of 0.1545. The Convolutional BGRU and the Convolutional LSTM combinations are two models with the lowest total Root Mean Squared Error (RMSE) estimates. When simultaneously forecasting financial time-series data with many distinct components, the preliminary findings indicate that these two models perform much better than their competitors.



(**a**)

(**b**)

(**c**)

(**d**)

(**e**)

(**f**)

Figure 12. Actual and prediction values for BSE index with Conv. RNN, Conv. LSTM, Conv. BRNN, Conv. GRU, Conv. BLSTM and Conv. BGRU models. (**a**) Actual and prediction values for BSE index with Conv. RNN model; (**b**) actual and prediction values for BSE index with Conv. LSTM model; (**c**) actual and prediction values for BSE index with Conv. BRNN model; (**d**) actual and prediction values for BSE index with Conv. RNN; (**e**) actual and prediction values for BSE index with Conv. BLSTM; (**f**) actual and prediction values for BSE index with Conv. BGRU.

Table 3. Comparison of error for prediction of BSE, HSI, TWII, JKSE, JPXGY, and KS11 indexes.

	Root Mean Squared Error [1]					
	Conv. LSTM [2]		**Conv. GRU** [3]		**Conv. RNN** [4]	
Index	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
BSE	0.0132	0.0143	0.0227	0.0239	0.0147	0.0158
HSI	0.0263	0.0276	0.0251	0.0267	0.0179	0.0195
TWII	0.0197	0.0219	0.0213	0.0226	0.0189	0.0203
JKSE	0.0222	0.0236	0.0215	0.0233	0.0312	0.0336
JPXGY	0.0145	0.0154	0.0257	0.0269	0.0273	0.0294
KS11	0.0362	0.0376	0.0332	0.0348	0.0249	0.0264
	Mean Absolute Percentage Error [5]					
BSE	1.45%	1.87%	1.41%	1.76%	1.92%	1.64%
HSI	2.65%	3.16%	2.45%	2.89%	2.42%	3.12%
TWII	2.45%	1.55%	3.12%	4.27%	3.27%	2.64%
JKSE	1.73%	2.52%	2.54%	3.11%	2.84%	1.67%
JPXGY	3.61%	3.78%	3.63%	2.72%	2.77%	3.54%
KS11	2.55%	2.65%	3.22%	2.77%	4.26%	4.21%
	Mean Absolute Error [6]					
BSE	0.0093	0.0112	0.0191	0.0203	0.0132	0.0146
HSI	0.0226	0.0236	0.0227	0.0222	0.0156	0.0159
TWII	0.0146	0.0202	0.0200	0.0201	0.0135	0.0166
JKSE	0.0205	0.0219	0.0198	0.0209	0.0294	0.0309
JPXGY	0.0124	0.0127	0.0231	0.0225	0.0223	0.0261
KS11	0.0327	0.0341	0.0317	0.0314	0.0211	0.0231

[1] An indicator of how far off a regression model's predictions are from reality is the Root Mean Squared Error (RMSE). [2] One kind of recurrent neural network, called a Convolutional Long Short-Term Memory, uses convolutional design for transitions from one state to another and from input to input. [3] One subset of GRUs that includes the convolution operation is the Convolutional Gated Recurrent Unit, abbreviated as GRU. [4] A CNN is the first step in the construction of a convolutional recurrent neural network (CRNN), which is then followed by an RNN. [5] Forecasting system accuracy is measured by MAPE. To ascertain the accuracy, divide the real values by each period's average absolute percentage error. [6] The disparity is the gap between the actual and measured values.

Table 4. Comparison of error for the prediction of BSE, HSI, TWII, JKSE, JPXGY, and KS11 indexes.

	Root Mean Squared Error [1]					
	Conv. BLSTM [2]		**Conv. BGRU** [3]		**Conv. BRNN** [4]	
Index	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
BSE	0.0132	0.0143	0.0227	0.0239	0.0147	0.0158
HSI	0.0263	0.0276	0.0251	0.0267	0.0179	0.0195
TWII	0.0197	0.0219	0.0213	0.0226	0.0189	0.0203
JKSE	0.0222	0.0236	0.0215	0.0233	0.0312	0.0336
JPXGY	0.0145	0.0154	0.0257	0.0269	0.0273	0.0294
KS11	0.0362	0.0376	0.0332	0.0348	0.0249	0.0264
	Mean Absolute Percentage Error [5]					
BSE	1.45%	1.87%	1.41%	1.76%	1.92%	1.64%
HSI	2.65%	3.16%	2.45%	2.89%	2.42%	3.12%
TWII	2.45%	1.55%	3.12%	4.27%	3.27%	2.64%
JKSE	1.73%	2.52%	2.54%	3.11%	2.84%	1.67%
JPXGY	3.61%	3.78%	3.63%	2.72%	2.77%	3.54%
KS11	2.55%	2.65%	3.22%	2.77%	4.26%	4.21%

Table 4. *Cont.*

	Root Mean Squared Error [1]					
	Conv. BLSTM [2]		Conv. BGRU [3]		Conv. BRNN [4]	
Index	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
	Mean Absolute Error [6]					
BSE	0.0093	0.0112	0.0191	0.0203	0.0132	0.0146
HSI	0.0226	0.0236	0.0227	0.0222	0.0156	0.0159
TWII	0.0146	0.0202	0.0200	0.0201	0.0135	0.0166
JKSE	0.0205	0.0219	0.0198	0.0209	0.0294	0.0309
JPXGY	0.0124	0.0127	0.0231	0.0225	0.0223	0.0261
KS11	0.0327	0.0341	0.0317	0.0314	0.0211	0.0231

[1] An indicator of how far off a regression model's predictions are from reality is the Root Mean Squared Error (RMSE). [2] Bidirectional Long Short-Term Memory (BiLSTM) layer in a sequence model. To process data both forward and backward, the model employs two LSTM layers. [3] Two Gated Recurrent Units (GRUs) in a BiGRU process sequential data. Two receptors receive information backwards and forward. [4] In Bi-directional Recurrent Neural Networks (BRNNs), two hidden layers produce opposite outputs. Generative deep learning allows the output layer to learn from past and future data. [5] Forecasting system accuracy is measured by MAPE. To ascertain the accuracy, divide real values by each period's average absolute percentage error. [6] The disparity is the gap between the actual and measured values.

Further, Table 5 presents the comparative analysis of outcomes of the suggested hybrid model, where layers of each of the six models are included and tested on the BSE stock exchange and other recent frameworks on RMSE, MAPE, and MAE evaluation parameters. The proposed hybrid model achieves an RMSE of 0.0122 while tested on the BSE index, and that is the least in comparison with other indices due to the large corpus available in BSE. Further, the hybrid model achieves a MAPE of 1.31% and an MAE of 0.0082 on the BSE index's test data.

Based on the study conducted by other researchers, it has been consistently shown that hybrid models tend to outperform individual models across various types of datasets. While the difference between hybrid models may not be statistically significant, it has been discovered that using them produces better results in predicting the stock exchange close price than the traditional approach of deep learning models. Despite no significant and statistically meaningful difference between the hybrid models, this phenomenon continues.

Table 5. Result comparison of the proposed framework on six Asian Stock Indices and also with recently developed frameworks by other researchers.

	Root Mean Squared Error					
Ref.	**BSE**	**HSI**	**TWII**	**JKSE**	**JPXGY**	**KS11**
Proposed Framework	0.0122	0.0131	0.0125	0.0133	0.0156	0.0191
[35]	0.0143	0.0152	0.0152	0.0163	0.0184	0.0202
[31]	0.0139	0.0177	0.0192	0.0158	0.0213	0.0199
[5]	0.0147	0.0163	0.0179	0.0163	0.0198	0.0238
	Mean Absolute Percentage Error					
Proposed Framework	1.31%	1.54%	1.28%	1.66%	1.59%	1.48%
[35]	1.38%	1.41%	1.34%	1.78%	2.13%	1.68%
[31]	1.34%	1.42%	1.23%	1.79%	1.64%	1.56%
[5]	1.35%	1.49%	1.29%	1.82%	1.83%	1.60%

Table 5. *Cont.*

	Root Mean Squared Error					
Ref.	**BSE**	**HSI**	**TWII**	**JKSE**	**JPXGY**	**KS11**
	Mean Absolute Error					
Proposed Framework	0.0082	0.0084	0.0092	0.0091	0.0109	0.0111
[35]	0.0085	0.0091	0.0094	0.0090	0.00103	0.0114
[31]	0.0093	0.0102	0.0111	0.0099	0.0113	0.0124
[5]	0.0092	0.0091	0.0099	0.0114	0.0126	0.0121

4. Conclusions

Empirical studies and research findings indicate that no deep learning model can provide precise predictions for all potential states within the framework of time-series data. When using several models, such as Convolutional RNN, Convolutional LSTM, Convolutional GRU, Convolutional BRNN, Convolutional BLSTM, and Convolutional BGRU, to predict the movement of time-series data, it is evident that the performance varies for each dataset. The study also found a strong association between the level of instability in time-lapse information and the ideal architecture of the deep neural networks model for precise forecasting. A single deep learning model with consistent time-series data produces precise predictions compared to many models that work with unorganized data. Even though the framework suggested in this article can accurately forecast stock indices in a close process, numerous obstacles still need to be overcome in stock forecasting. Initially, the stock data utilized in this paper are exclusively from Asian markets and do not accurately predict stock indices in the Middle East, Africa, or Europe. Secondly, enhancing the parameters for tuning CNN and BGRU is necessary. The framework is posited to achieve superior outcomes among the stock indices' intensive parameter adjustments. Furthermore, we believe that numerous hybrid models still need to be addressed in this article, which can potentially improve the error in predicting the close price of stock indices, even though the Convolutional BGRU hybrid approach is employed in this paper. Nevertheless, the Convolutional BGRU-equipped stock indices prediction approach proposed in this article can assist investors in making accurate judgments in the stock exchange to a certain extent.

Additional research is underway to create an independent deep learning system that can identify the most appropriate model depending on the unpredictable characteristics of the input data. To validate the conclusions of this research and create a deep learning model for predicting financial time series, it is crucial to acquire extensive and varied financial data from other businesses.

Author Contributions: Conceptualization, R.S. and S.L.; methodology, R.S. and H.S.; software, G.K. and S.S.; validation, S.S. and P.S.; writing—original draft preparation, R.S., S.S. and P.S.; writing—review and editing, H.S., G.K., P.S. and S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lanbouri, Z.; Achchab, S. Stock market prediction on high frequency data using long-short term memory. *Procedia Comput. Sci.* **2020**, *175*, 603–608. [CrossRef]
2. Liu, B.; Wu, Q.; Cao, Q. An improved Elman network for stock price prediction service. *Secur. Commun. Netw.* **2020**, *2020*, 8824430. [CrossRef]
3. Singh, H.; Malhotra, M. Stock market and securities index prediction using artificial intelligence: A systematic review. *Multidiscip. Rev.* **2024**, *7*, 2024060. [CrossRef]

4. Salgotra, R.; Gandomi, M.; Gandomi, A. Time series analysis and forecast of the COVID-19 pandemic in India using genetic programming. *Chaos Solitons Fractals* **2020**, *138*, 109945. [CrossRef] [PubMed]
5. Singh, H.; Malhotra, M. A Novel Approach for Predict Stock Prices in Fusion of Investor's Sentiment and Stock Data. In Proceedings of the IEEE 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 1–3 November 2023; pp. 1163–1169.
6. Singh, H.; Malhotra, M. A time series analysis-based stock price prediction framework using artificial intelligence. In Proceedings of the Springer 2024 International Conference on Artificial Intelligence of Things, Chandigarh, India, 30–31 March 2023; pp. 280–289.
7. Singh, H.; Malhotra, M.; Singh, S.; Sharma, P.; Prabha, C. Design and Development of Artificial Intelligence Framework to Forecast the Security Index Direction and Value in Fusion with Sentiment Analysis of Financial News. *SN Comput. Sci.* **2024**, *5*, 787. [CrossRef]
8. Singh, H.; Malhotra, M. A novel approach of stock price direction and price prediction based on investor's sentiment. *SN Comput. Sci.* **2023**, *4*, 823. [CrossRef]
9. Yang, C.; Zhai, J.; Tao, G. Deep learning for price movement prediction using convolutional neural network and long short-term memory. *Math. Probl. Eng.* **2020**, *2020*, 2746845. [CrossRef]
10. Singh, H.; Sharma, P.; Prabha, C.; Singh, S. Ensemble Learning with an Adversarial Hypergraph Model and a Convolutional Neural Network to Forecast Stock Price Variations. *Ing. Des Syst. D'Inf.* **2024**, *29*, 1151. [CrossRef]
11. Singh, H.; Rani, A.; Chugh, A.; Singh, S. Utilising Sentiment Analysis to Predict Market Movements in Currency Exchange Rates. In Proceedings of the IEEE 2024 3rd International Conference for Innovation in Technology (INOCON), Banglore, India, 1–3 March 2024; pp. 1–7.
12. Zhou, Q.; Zhou, C.; Wang, X. Stock prediction based on bidirectional gated recurrent unit with convolutional neural network and feature selection. *PLoS ONE* **2021**, *17*, e0262501. [CrossRef]
13. Singh, H.; Sharma, C.; Attri, V.; Singh, S. Time Series Forecast with Stock's Price Candlestick Patterns and Sequence Similarities. In Proceeding of the IEEE International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 5–7 March 2024; pp. 1–6.
14. Kaur, M.; Joshi, K.; Singh, H. An efficient approach for sentiment analysis using data mining algorithms. In Proceedings of the IEEE 2022 International Conference on Computing, Communication and Power Technology, Visakhapatnam, India, 7–8 January 2022; pp. 81–87.
15. Xu, Y.; Chhim, L.; Zheng, B.; Nojima, Y. Stacked Deep learning structure with bidirectional long-short term memory for stock market prediction. In Proceedings of the Springer 2020 International Conference on Neural Computing for Advanced Applications, Shenzhen, China, 3–5 July 2020; pp. 447–460.
16. Singh, H.; Malhotra, M. Artificial intelligence-based hybrid models for prediction of stock prices. In Proceedings of the IEEE 2023 2nd International Conference for Innovation in Technology (INOCON), Bangalore, India, 3–5 March 2023; pp. 1–6.
17. Salgotra, R.; Singh, S.; Singh, U.; Kundu, K.; Gandomi, A.H. An adaptive version of differential evolution for solving cec2014, cec 2017 and cec 2022 test suites. In Proceedings of the IEEE 2022 Symposium Series on Computational Intelligence (SSCI), Bangalore, India, 4–7 December 2022; pp. 1644–1649.
18. Lim, S.; Kim, M.J.; Ahn, C.W. A genetic algorithm (GA) approach to the portfolio design based on market movements and asset valuations. *IEEE Access* **2020**, *8*, 140234–140249. [CrossRef]
19. Sidhu, N.K.; Gupta, S.; Sharma, S.; Singh, M. *Navigating Tomorrow: The Uncharted Frontier of Robo-Advisors in Wealth Management*; IGI Global: Hershey, PA, USA, 2024; pp. 162–174.
20. Chawla, J.; Walia, N.K. Artificial intelligence based techniques in respiratory healthcare services: A review. In Proceedings of the IEEE 2022 3rd International Conference on Computing, Analytics and Networks (ICAN), Rajpura, Punjab, India, 18–19 November 2022; pp. 1–4.
21. Hasan, A.; Kalıpsız, O.; Akyoku¸s, S. Modeling traders' behavior with deep learning and machine learning methods: Evidence from bist 100 index. *Complexity* **2020**, *8*, 1–16. [CrossRef]
22. Yan, H.; Ouyang, H. Financial time series prediction based on deep learning. *Wirel. Pers. Commun.* **2018**, *102*, 683–700. [CrossRef]
23. Pant, P.; Mishra, K.K.; Mohan, A. Algorithmic Approaches to Financial Technology: Forecasting, Trading, and Forecasting. In *Artificial Intelligence and Machine Learning-Powered Smart Finance*; IGI Global: Hershey, PA, USA, 2024; pp. 50–81.
24. Chacon, H.D.; Kesici, E.; Najafirad, P. Improving financial time series prediction accuracy using ensemble empirical mode decomposition and recurrent neural networks. *IEEE Access* **2020**, *8*, 117133–117145. [CrossRef]
25. Jaswal, G.S.; Sasan, T.K.; Kaur, J. Early stage emphysema detection in chest X-ray images: A machine learning based approach. In Proceedings of the IEEE 2023 World Conference on Communication & Computing (WCONF), Raipur, India, 14–16 July 2023; pp. 1–6.
26. Bukhari, A.H.; Raja, M.A.Z.; Sulaiman, M.; Islam, S.; Shoaib, M.; Kumam, P. Fractional neurosequential arfima-lstm for financial market forecasting. *IEEE Access* **2020**, *8*, 71326–71338. [CrossRef]
27. Cheng, W.X.; Suganthan, P.N.; Qiu, X.; Katuwal, R. Classification of stock market trends with confidence-based selective predictions. In Proceedings of the Springer: Swarm, Evolutionary, and Memetic Computing and Fuzzy and Neural Computing: 7th International Conference, SEMCCO 2019, and 5th International Conference, FANCCO 2019, Maribor, Slovenia, 10–12 July 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 93–104.

28. Zhang, J.; Shao, Y.-H.; Huang, L.-W.; Teng, J.-Y.; Zhao, Y.-T.; Yang, Z.-K.; Li, X.-Y. Can the exchange rate be used to predict the shanghai composite index? *IEEE Access* **2019**, *8*, 2188–2199. [CrossRef]
29. Nayak, S.C.; Sanjeev Kumar Dash, C.; Behera, A.K.; Dehuri, S. Improving stock market prediction through linear combiners of predictive models. In *Advances in Intelligent Systems and Computing*; Springer: Singapore, 2019.
30. Li, X.; Peng, L.; Yao, X.; Cui, S.; Hu, Y.; You, C.; Chi, T. Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environ. Pollut.* **2017**, *231*, 997–1004. [CrossRef]
31. Li, X.; Wu, X. Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. In Proceedings of the IEEE 2015 International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 19–24 April 2015; pp. 4520–4524.
32. Singh, H.; Shukla, A.K. An analysis of Indian election outcomes using machine learning. In Proceedings of the IEEE 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 10–11 December 2021; pp. 297–306.
33. Jishag, A.; Athira, A.; Shailaja, M.; Thara, S. Predicting the stock market behavior using historic data analysis and news sentiment analysis in R. In Proceedings of the Springer 2020 First International Conference on Sustainable Technologies for Computational Intelligence, Jaipur, India, 29–30 March 2019; pp. 717–728.
34. Wang, J.; Sun, T.; Liu, B.; Cao, Y.; Wang, D. Financial markets prediction with deep learning. In Proceedings of the IEEE 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Al Ain, United Arab Emirates, 17–18 November 2020; pp. 4520–4524.
35. Alonso-Monsalve, S.; Su´arez-Cetrulo, A.L.; Cervantes, A.; Quintana, D. Convolution on neural networks for high-frequency trend prediction of cryptocurrency exchange rates using technical indicators. *Expert. Syst. Appl.* **2020**, *149*, 113250.
36. Fazeli, A.; Houghten, S. Deep learning for the prediction of stock market trends. In Proceedings of the IEEE 2019 International Conference on Big Data, Los Angeles, CA, USA, 9–12 December 2019; pp. 5513–5521.
37. JuHyok, U.; Lu, P.; Kim, C.; Ryu, U.; Pak, K. A new lstm based reversal point prediction method using upward/downward reversal point feature sets. *Chaos Solitons Fractals* **2020**, *132*, 109559.
38. Weytjens, H.; De Weerdt, J. Process outcome prediction: Cnn vs. lstm (with attention) In *Business Process Management Workshops: BPM 2020 International Workshops*; Springer: Seville, Spain, 2020; pp. 321–333.
39. Lim, B.; Zohren, S. Time-series forecasting with deep learning: A survey. *Philos. Trans. R. Soc. A* **2021**, *379*, 20200209. [CrossRef]
40. Sultana, F.; Sufian, A.; Dutta, P. Advancements in image classification using convolutional neural network. In Proceedings of the IEEE 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 22–23 November 2018; pp. 122–129.
41. Widiputra, H.; Mailangkay, A.; Gautama, E. Multivariate cnn-lstm model for multiple parallel financial time-series prediction. *Complexity* **2021**, *2021*, 9903518. [CrossRef]
42. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
43. Rojas-Barahona, L.M. Deep learning for sentiment analysis. *Lang. Linguist. Compass* **2016**, *10*, 701–719. [CrossRef]
44. Widiputra, H. GA-optimized multivariate CNN-LSTM model for predicting multi-channel mobility in the COVID-19 pandemic. *Emerg. Sci. J.* **2021**, *5*, 619–635. [CrossRef]
45. Baghel, Y.; Jindal, H.; Chawla, J. Early diagnosis of emphysema using convolutional neural networks. In Proceedings of the IEEE 2023 World Conference on Communication & Computing (WCONF), Raipur, India, 14–16 July 2023; pp. 1–5.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.