Original Research

Designing an Algorithmic Trading System, Based on Deep Learning (Case Study: Tehran Stock Exchange)

Paria Soleimani[1*]. Behzad Soleimani[1]. Mina Bagheriyan[1]. Erfan Taati[1]

Received: 29 Oct 2023/ Accepted: 07 Nov 2024/ Published online: 06 Dec 2024

*Corresponding Author, p_soleimani@azad.ac.ir

1- Department of Industrial Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran.

Abstract
With the development of computer systems in recent years, transactions in financial markets have been made available for investors. Artificial intelligence (AI) based models have also used in the financial markets due to the development of information systems and their ability to store and retrieve the large volumes of financial data. As a result, lots of research has been done using artificial intelligence methods to design algorithmic trading systems. In this regard, deep learning, one of the newest subfields of artificial intelligence, has also attracted attentions. This research presents a new approach for modeling the buying and selling process in the stock market based on deep learning and LSTM and CNN methods. In the proposed method, the forecast of the future value of stock indicators obtained using the LSTM algorithm is used as the input features of a CNN network. The CNN network as a classification model provides the buy/sell signal for the algorithmic trading system. In addition, the EC-FS model has been used to determine the most appropriate input indicators for the classification model. The proposed model has been evaluated on the Tehran Stock Exchange market and five selected stocks. The results of this model have been compared with other models such as LSTM-MLP, RNN-CNN and RNN-MLP. As a result, it can be concluded that the LSTM algorithm performs better for forecasting the indicators of selected stocks. According to this study, deep learning models are more efficient than the surface neural network models, demonstrating higher performance in predicting time series indicators and determining buy/sell signals. In four out of five tested stocks, the combined LSTM-CNN method had significantly higher accuracy than the other mixed methods. It is a general statement that the collaborative LSTM-CNN method is more effective than other methods at training the buying and selling process. This study can help the stock market of Iran participants for designing the most effective trading strategy.

Keywords - Deep Learning; LSTM-CNN method; Stock Market; Tehran Stock Exchange; Trading System

INTRODUCTION

Algorithmic trading is a method of executing orders using pre-programmed automated trading instructions. This type of trading was developed to take advantage of speed and processing capability of computers. Todays, retail traders, investment banks, pension funds and large financial institutions use algorithmic trading to achieve the most efficient trading decisions. In 2016, a study revealed that more than 80% of transactions in the Forex market are done through algorithmic trades instead of human ones.[۱] A signaling algorithm is an algorithm that generates buy or sell signals based on hidden market rules. Inputs of a signaling algorithm can include types of asset prices, returns, trading indicators, and even fundamental indicators such as ratios of financial statements or macroeconomic indicators. Signaling algorithms generally work by relying on forecast/classification methods. In other words, in these algorithms, a set of inputs is mapped to a two-way output (buying /selling position) or three-way output (buying /selling /holding position). These algorithms are designed based on the predictive methods. In recent years, forecast/classification models based on deep learning have shown impressive performance compared to the traditional models in various applications. Convolutional neural networks(CNN) are the class of deep neural networks commonly used for image or speech analysis in machine learning. In this regard, there are some studies. The paper [۲] utilized a convolutional neural network (CNN) with 15 technical analysis indicators and various parameter values ranging from 6 to 20 days as inputs and three signals of buy, sell, or hold for stocks as the outputs. The indicators used as input features do not account for the prediction of time series paths. One of the most critical factors in making stock trading decisions is considering future trends. Hence, time series forecasting models should be incorporated to provide a projection of the future state of the asset Another research [۳] has demonstrated the ability of trading algorithm to predict future trends. The daily signaling approach is used to evaluate the proposed method at high frequencies. Its performance is evaluated for more extended periods. This study presents a new LSTM-CNN signaling model and evaluates it on the Tehran Stock Exchange. This paper is structured as follows: in the second part, the review of the research literature is examined. In the third part, the proposed method of this research is presented and the evaluation method is explained. In the fourth section, the details of a case study is examined. The results are presented in the fifth section and the conclusion is discussed in the sixth part.

LITERATURE REVIEW

Recently, more research has been done on modeling of algorithmic transactions using artificial intelligence methods. For example, in reference [۴], indicator features have transformed into image features to determine the optimal policy for buying, selling or holding based on learning a convolutional neural network(CNN). For a stock market, the proposed model is implemented at a daily frequency. In reference[۴], an ensemble method is presented based on the algorithm of grammatical evolution. It determines the final rule by combining the different time windows. In addition, supervised learning, including recurrent neural network models, classification methods, pattern recognition and reinforcement learning as the methods of designing algorithmic transactions have gained popularity in recent years. The study [۵] has used the Q reinforcement learning method to determine the most appropriate parameters of the indicator. The proposed model has been evaluated in a stock market and for daily frequencies. In reseach [۶], a two-step approach is used to design the reinforcement learning model. In this research, at first, the ranges of stock trends have been determined with OHLCV data. The decision variable in this model is buying/selling and holding. The proposed model has been evaluated at daily frequency. In reference [۷] similar to research [۸]has been used a combination of reinforcement learning model and deep model for asset return forecast. The difference between these two articles is in utilizing the LSTM method to predict asset returns. For evaluating of the proposed model, it has been implemented for stock and future markets at one-minute frequencies and compared with the Deep A3C model. In reference [۹] similar to research [۱۰] , authors forecasted the asset return as a buy, sell or hold signal. This research uses a deep learning model of binary long-short-term memory based on random sampling (random sampling bi-LSTM). The model has been evaluated in a two-way crypto currency market with one-minute frequencies. In study [۱۱] , a deep-learning model of short-long term bagging ensembles was used to predict the return of currency pairs at a daily frequency.

Reference [۱۲] used a deep neural network with recurrent units by considering of risk minimization. Inputs of the model are typical indicators. Model evaluation is done in a two-way stock market and daily frequency. Reference [۱۳] uses the deep Q network model, OHLCV data and current trading position as the state variable. In reference [۱۴], the reinforcement learning model in combination with the convolutional neural network model, has been used to estimate the state-action function. The features considered in this model include OHLCV data. In researh [۱۵], a reinforcement learning model based on investor sentiment analysis is presented. In this model, the status variables include the features that extracted from social networks such as Twitter and Reuters. In references [۱۶] and [۱۷] , the A3C deep learning method has been used with price data and typical indicators for crypto currency markets. According to [۱۸], a deep reinforcement learning model is used to estimate the state-action functions of a convolutional neural network by considering the feature selection layers. The proposed model considers OHLCV data, stock indicators and asset correlation with other assets as the state variables and buy, sell, and hold actions as the action variables. In reference [۱۹], a control optimization approach, similar to reinforcement learning models, has been used to determine the optimal policy based on the final price state variable.

In an algorithmic trading system to achieve long-term profits, feature extraction and trading strategy design are two fundamental challenges. Recearch [٢٠] has presented a deep learning model that provides a combination of CNN and two-way LSTM units. Comparing the proposed model to individual models has shown 9% improvement in forecasting performance. In reference [٢١], a new hybrid forecasting approach based on Variable Mode Decomposition (VMD), Iterated Cumulative Sum of Squares (ICSS) and Bi-way Gated Regression Unit (BiGRU) method is proposed for algorithmic trading. Experimental results show that this hybrid forecasting approach can significantly improve forecasting performance. In reference [٢٢], they have proposed an algorithmic trading system based on DC dynamic threshold and reinforcement learning model. The proposed trading strategy is trained using the Q learning algorithm to find an optimal trading rule. This strategy is evaluated on real stock market data. The results have shown that it has improved trading efficiency and Sharpe ratio index. Research[23] uses deep learning methods to forecast the closing prices of MAPNA and Toucaril shares on the Tehran Stock Exchange. It develops One-dimensional Convolutional Neural Networks (1DCNN), Long Short-Term Memory (LSTM) networks and a combined CNN-LSTM model for stock price estimation. The efficiency of these techniques is determined using various metrics. Study[24] proposes a hybrid deep learning model, the CNN-LSTM which combines the 2D-CNN for image processing with the LSTM network for managing image sequences and classification. They transformed the top technical indicators from financial time series in to images for 21 different days. From the approach of stock market forecasting and intelligent decision-making process, reseach [25] shows the possibility of Deep Reinforcement Learning in financial markets and advantages of strategic decision-making.

PROPOSED METHOD

A conceptual framework of the research method is illustrated in FIGURE ١. As shown, the CNN deep learning classification provides a set of output after receiving inputs. The inputs of this model are the indicators of specific stock, along with a prediction of these indicators using the long-short-term memory (LSTM) algorithm. The output of the model includes the buy or sell signal.

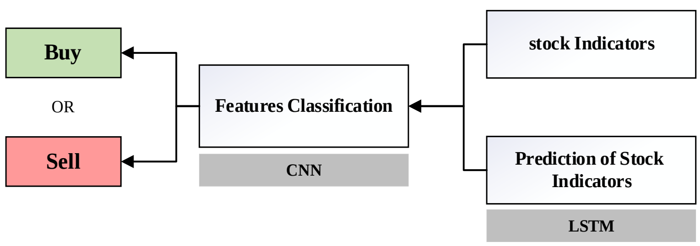


FIGURE ١
A CONCEPTUAL FRAMEWORK OE THE RESEARCH METHOD

I. Long Short-Term Memory (LSTM) Network

LSTM algorithm has a Recurrent Artificial Neural Network (RNN) architecture [26]. Unlike the common feed forward neural networks, LSTM has feedback connections; this allows the network to process not only single data points (such as images) but also entire data sequences (such as speech or video). For example, LSTM can be used for handwriting classification, speech recognition and anomaly detection in network traffic [27]. A common LSTM unit consists of a cell, including of an input, an output and a forget gate. FIGURE 2 illustrates the architecture of a unit cell in LSTM network. There are the following relationships established in LSTM network cell [28]:

$$\begin{aligned} f_t &= \sigma_g\left(W_f x_t + U_f h_{t-1} + b_f\right) \\ i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\ o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\ \tilde{c}_t &= \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \\ h_t &= o_t \circ \sigma_h(c_t) \end{aligned} \qquad (١)$$

The initial values of $c_0 = 0$ and $h_0 = 0$ and the operator $\circ$ represents the Hadamard product (element by element). The subscript t represents time steps. In Equation no.١ $x_t \in \mathbb{R}^d$ is the input vector of the LSTM unit, $f_t \in \mathbb{R}^h$ is the activation vector of forget gate, $i_t \in \mathbb{R}^h$ is the activation vector of input/update gate, $o_t \in \mathbb{R}^h$ is hidden state vector, $h_t \in \mathbb{R}^h$ is also known as the output vector of the LSTM unit, $\tilde{c}_t \in \mathbb{R}^h$ is the cell input activation vector, $c_t \in \mathbb{R}^h$ is the cell state vector, and $W \in \mathbb{R}^{h\times d}$, $U \in \mathbb{R}^{h\times h}$

and $b \in \mathbb{R}^h$ are weight and skew (bias) matrices, respectively. These values are determined in the learning process. In these equations, the superscripts d and h represent the number of input features and the number of hidden units, σ_g is the sigmoid activation function, and σ_c and σ_h are hyperbolic tangent activation functions.

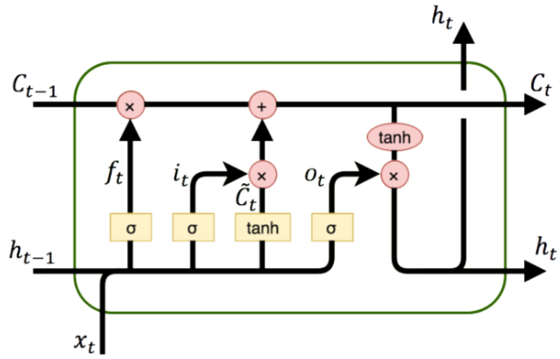


FIGURE ۲
LSTM NETWORK CELL ARCHITECTURE

II. Convolutional Neural Network(CNN)

In CNN, the hidden layers include the layers that perform the convolution. Typically, this consists a layer that performs an inner multiplication of the convolution kernel with the layer input matrix. The activation function is usually ReLU. The convolution operation produces a feature map, which contributes to the next layer. It should be noted that there are also aggregation layers, fully connected layers and normalization layers[29]. Figure 3 illustrates an example of a convolutional neural network.

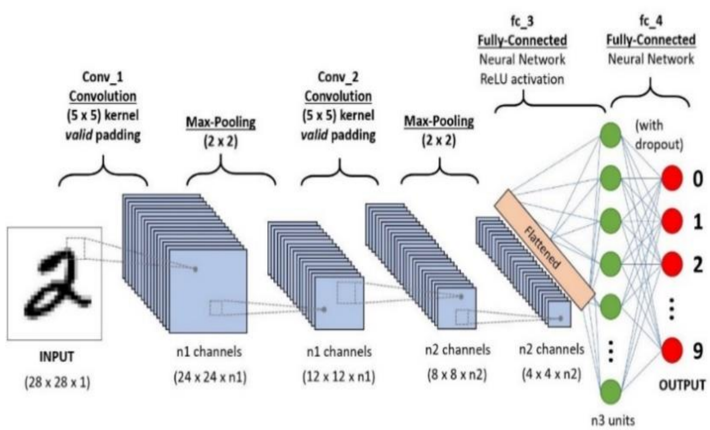


FIGURE **3**
AN EXAMPLE OF A CONVOLUTIONAL NEURAL NETWORK

- ***Convolution layer***

A convolutional layer in CNN generally has the following characteristics:

- Convolution filters/kernels that have been defined by width and height.
- Input channels and output channels.
- Additional hyper parameters of the convolution operator, such as padding, stepping and dilation.

Convolution layers convolute input and pass the result to the next layer. The fully connected feedforward neural networks can be used to learn features and classify data. In addition, convolutional neural networks are ideal for data with a lattice-like topology (e.g., images) because the spatial relationships between the individual features are considered during convolution and/or aggregation.

- ***Aggregation layer***

Convolutional networks may include local/global aggregation layers along with traditional convolutional layers. Aggregation layers reduce data dimensionality by combining the outputs into a single neuron. Local aggregation is a combination of small clusters and is usually applied with clusters of size 2 × 2. The global aggregation operates on all feature-mapped neurons [30]

- ***Fully connected layer***

Fully connected layers connect each neuron to every other neuron in the next layer. This layer is the traditional multilayer perceptron (MLP) neural network. The planar matrix passes through a fully connected layer to classify the images.

- ***Evaluation process***

In the following, each step of the model execution process is presented separately. First, the OHLC data and candidate stocks are recorded and missing data are removed. Then the data is divided into two states of learning and testing. In this research, the learning state includes 750 observations leading to the first collected test data and the testing state consists of 250 observations leading to the last collected data. The software used for data modeling and analysis is MATLAB software. In general, the steps of the data analysis process are as follows.

Step 1: In this step, contribution indicators for the learning state are extracted. In the literature, four types of indicators have been considered: trend indicators, momentum or oscillator indicators, volume indicators and volatility indicators. Using all the indicators increases calculations and sometimes leads to the overestimation or underestimation. Therefore, this research used the feature selection method to select the proper indicators to implement the classification process.

Step 2: In order to select the appropriate indicators, the ECFS feature selection method has been used. The output of a feature selection model includes the priority of each indicator. First, the ECFS algorithm was run for each candidate stock and the eigenvector value was determined for each indicator. Then, the average of the values for each indicator was used to determine the priority of each indicator. Pebble diagram analysis was used to determine the number of screened features to continue the investigation.

Step 3: In this step, an LSTM network is trained for each indicator that is selected. FIGURE 4 illustrates the architecture of this network. The input layer includes time series sequence input. The next layer is the LSTM layer. For this layer, one hundred hidden units are considered. The following layer is a fully connected layer that connects the LSTM layer to the output or the regression layer.

FIGURE 4
LONG-SHORT-TERM MEMORY NETWORK ARCHITECTURE

Step 4: In this step, each indicator is predicted for the learning time horizon of the CNN neural network. The input structure, which includes the indicators and prediction of stock values using the LSTM model, is converted into a CNN input matrix as follows. In this matrix, $\boldsymbol{i_{MN}}$ is the $\boldsymbol{M}^{\mathbf{th}}$ indicator by considering the length of $\boldsymbol{N}$ **and** $\boldsymbol{r_N}$ is the forecast of the next $\boldsymbol{N^{th}}$ day using the LSTM model.

$$\begin{bmatrix} i_{11} & i_{12} & \dots & i_{1N} \\ i_{21} & i_{22} & \dots & i_{2N} \\ \dots & \dots & \dots & \dots \\ i_{M1} & i_{M2} & \dots & i_{MN} \\ r_1 & r_2 & \dots & r_N \end{bmatrix} \qquad (۲)$$

Step 5: In the proposed CNN model of this research, the output of the model (or dependent variable) is the buy or sell signal for a share. A zigzag indicator is used to determine buy or sell signal. This indicator shows the price trend in zigzag lines.

Step 6: In this step, the convolutional neural network is trained. Common filter sizes used for convolutional layers are *3 x 3*, *5 x 5* and *7 x 7* for medium or small-size images. In addition, *2 x 2* or *3 x 3* filter sizes are used for Max-Pooling parameters.

Step 7: The trained algorithm should be checked by the test data. The indicators are the same indicators that were extracted during the learning state.

Step 8: Similar to Step 3 of the learning process, calculated indicators in previous step are predicted to design the inputs of the CNN network.

Step 9: In this step, buy/sell signal is issued. In other words, the convolutional neural network inputs that include indicators and their forecasts are entered into the network, and the buy/sell signal is issued as the output (forecast) of the network.

Step 10: After issuing a buy/sell signal for the test period, the ability of the proposed algorithmic trading model can be evaluated. For this purpose, the accuracy is checked from the following equation:
Algorithm accuracy = number of correct signals/total issued signals

Step 11: In this step, as shown in FIGURE 5, the proposed LSTM-CNN algorithmic trading model is compared with three competing models. RNN-CNN model, LSTM-MLP model and RNN-MLP model.

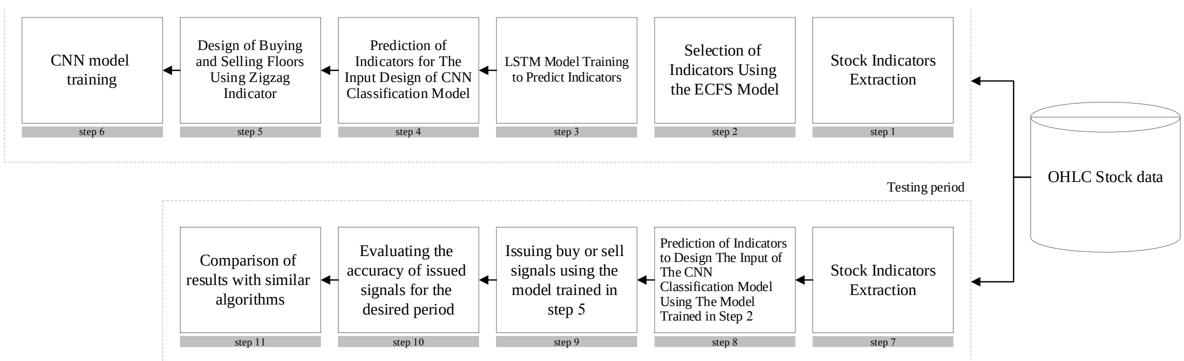


FIGURE 5
THE DEVELOPING PROCESS OF THE RESEARCH MODEL

- **Case study and data collection**

In order to evaluate the proposed model and show the managerial implications of this research, the data of five selected stocks which are presented in the list of top 50 companies have been used. The daily adjusted data for the first price, the highest price, the lowest price, the final price and trading volume are available at least for recent five years. The selected stocks are: Femeli (National Copper Industries Of Iran), Khodro (Iran Khodro), Shapena (Isfahan Oil Refinery), Vaghadir (Ghadir Investment), and Vebasadr (Iran Saderat Bank). The data is extracted from the Tehran Stock Exchange website using TSE client software.

III. EVALUATION RESULTS

TABLE 1, shows the results of the ECFS algorithm implementation and eigenvalues for each of the stocks and indicators. These results determine the rank of each indicator.

TABLE1
RESULTS OF THE FEATURE SELECTION MODEL

Indicator	Vebsader	Vaghadir	Khodro	Femeli	Shapena	Average
MACD Line	٠,١١٥١	٠,١٢٣٨	٠,١٢١٤	٠,١١١٦	٠,١٢٠٥	٠,١١٨٥
MACD Signal Line	٠,١١٦٩	٠,١١٩٠	٠,١٢١١	٠,١١٧٠	٠,١١٨٥	٠,١١٨٥
Acceleration	٠,١١٤١	٠,١١٧١	٠,١١٦٩	٠,١١٤٥	٠,١١٨٠	٠,١١٦١
Momentom of High Price	٠,٢٤٥٩	٠,٢٣١٩	٠,٢٣٥٨	٠,٢٤٨٤	٠,٢١٣٢	٠,٢٣٥٠
Momentom of Low Price	٠,١٦٥٠	٠,١٦٠٣	٠,١٦٠٣	٠,١٧٢٩	٠,١٥٣٧	٠,١٦٢٤

RSI	۰,۱٦٥۰	۰,۱٦۰۳	۰,۱٦۰۳	۰,۱۷۲۹	۰,۱٥۳۷	۰,۱٦۲٤
Highest High	۰,۱۳۳۸	۰,۱۳۲۰	۰,۱۳٤٦	۰,۱٤۱٦	۰,۱۳٤۹	۰,۱۳٥٤
Lowest Low	۰,۱۳۰٤	۰,۱۳٥۳	۰,۱۳٦۳	۰,۱۲۲٦	۰,۱۳۳۱	۰,۱۳۱٥
Chaikin Oscillator	۰,۱٤۱۸	۰,۱٤۳۹	۰,۱۳۸۹	۰,۱٤٦۸	۰,۱۳۷۹	۰,۱٤۱۹
Accumulation/Distribution Line	۰,۱٦۹۳	۰,۱٥۱٦	۰,۱٥۰۰	۰,۱٦۲۸	۰,۱۳٤۲	۰,۱٥۳٦
Accumulation/Distribution Oscilator	۰,۱٥٦۰	۰,۱٤٤۲	۰,۱٤۰۹	۰,۱٤٤۳	۰,۱۳۰۸	۰,۱٤۳۲
Chaikin Volatility	۰,۱۲۹۷	۰,۱۲۱۳	۰,۱۱٥٥	۰,۱۱۲۸	۰,۱۳۰٥	۰,۱۲۲۰
Negative Volume Index	۰,۱۱۲٥	۰,۱۲٤٥	۰,۱۱٦۸	۰,۱۱۲۳	۰,۱۲۰۱	۰,۱۱۷۲
Positive Volume Index	۰,۱۱۸٦	۰,۱۱۹٥	۰,۱۱۷۹	۰,۱۱۲۸	۰,۱۲۰٥	۰,۱۱۷۹
On Balance Volume	۰,۱٥٥٦	۰,۱٤۹۹	۰,۱۸۳۰	۰,۱۸۹۲	۰,۱٤۱۱	۰,۱٦۳۸
MiddleBollinger	۰,۱۲۳٦	۰,۱۱٦۷	۰,۱۱۸۷	۰,۱۱۲٦	۰,۱۲۰۲	۰,۱۱۸٤
Lower Bollinger	۰,۱۱٦۸	۰,۱۱۷٥	۰,۱۱۸۰	۰,۱۱۳٦	۰,۱۱۹۲	۰,۱۱۷۰
Upper Bollinger	۰,۱۲٤۹	۰,۱۱۸۱	۰,۱۲۰٦	۰,۱۱٤۳	۰,۱۱۸۸	۰,۱۱۹۳
Price Rate-of-Change	۰,۱۲۰٤	۰,۱۱۷۲	۰,۱۱۷۹	۰,۱۱۳٥	۰,۱۱۹٤	۰,۱۱۷۷
Price and Volume Trend	۰,۱۲٥۸	۰,۱۱۷٦	۰,۱۱۹۱	۰,۱۱٤۲	۰,۱۲۰۲	۰,۱۱۹٤
Volume Rate-of-Change	۰,۱۱۳٦	۰,۱۱۸۰	۰,۱۱۷٥	۰,۱۱۳۳	۰,۱۲۰٥	۰,۱۱٦٦
Stochastic %K	۰,٦۹۲۳	۰,۷۰٦۸	۰,۷۰۲۸	۰,٦۹۲۸	۰,٦۲۳٤	۰,٦۸۳٦
Stochastic % D	۰,۱۳۸۲	۰,۱۲۳٤	۰,۱۳۱۸	۰,۱۲٥٥	۰,۳۸٥۸	۰,۱۸۰۹
Weighted Closing Prices	۰,۱۱٥۱	۰,۱۱۷٤	۰,۱۱۸۲	۰,۱۱۳٦	۰,۱۲۰۷	۰,۱۱۷۰
Williams Accumulation/Distribution line High	۰,۱۳۲۰	۰,۱۳۷۸	۰,۱۲٥٥	۰,۱۳۹۲	۰,۱۳٤۸	۰,۱۳۳۹
Williams Accumulation/Distribution line Low	۰,۱۱۷٥	۰,۱۱۷۳	۰,۱۱۸۲	۰,۱۱۳٦	۰,۱۲۰۷	۰,۱۱۷٤
Williams Accumulation/Distribution line Close	۰,۱۱۱۹	۰,۱۲۱۳	۰,۱۱۷٦	۰,۱۱٤۹	۰,۱۲۲۷	۰,۱۱۷۷
Mdian Price	۰,۱٦۲۱	۰,۱٥۹۸	۰,۱٥٤۸	۰,۱۷۳۲	۰,۱٥۱۹	۰,۱٦۰٤

The Pebble diagram and the percentage description have been used to determine the number of selected features. FIGURE 6 shows this diagram. Six features with high-priority describe at least 90% of the independent variable. Therefore, these six features enter the next stage as the output of the feature selection model. These are Stochastic %K, Stochastic % D, Momentom of Low Price, Momentom of High Price, OBV and RSI.

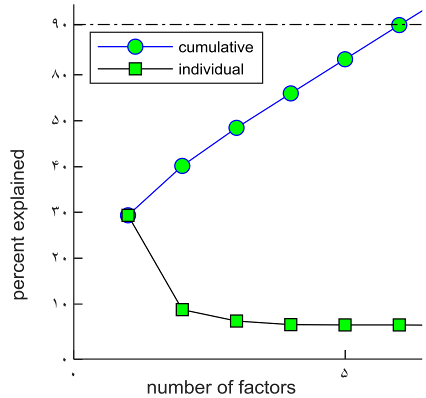
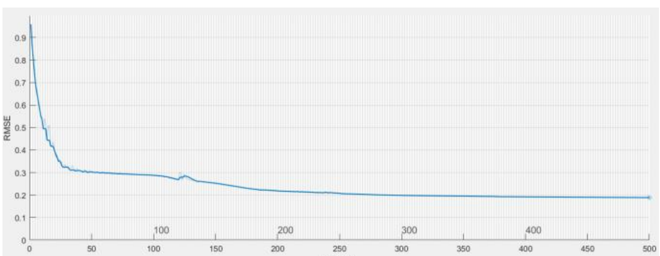
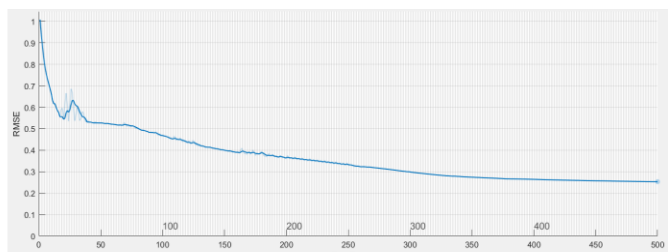


FIGURE 6
PEBBLE DIAGRAM FOR SPECIAL VECTOR VALUES OF INDICATORS

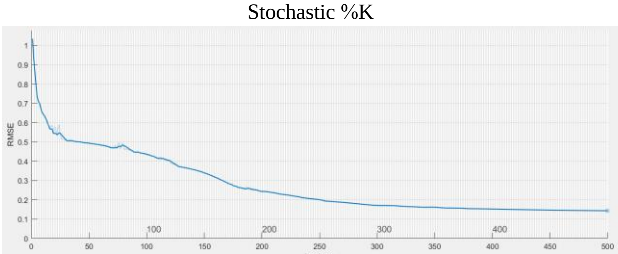
FIGURE 7 shows the learning process of the LSTM model for each of the *Vebsader* stock indicators. For the other stocks, the same process is followed. As can be seen, the value of RMSE in the iterations of the LSTM model has decreased and stabilized over time. FIGURE 8 shows the diagram of the learning process (right) and confusion matrix (left) of the LSTM-CNN model for *Vebsader* stock. The accuracy of the CNN model is stable after 600 iterations and the final learning accuracy is about 82%. FIGURE 9 shows the diagram of the trading process (right) and accuracy of issued signals (left) for *Vebsader* stock with the LSTM-CNN model. Buy signal and correct signal are shown in green and sell signal and false signal are shown in red.



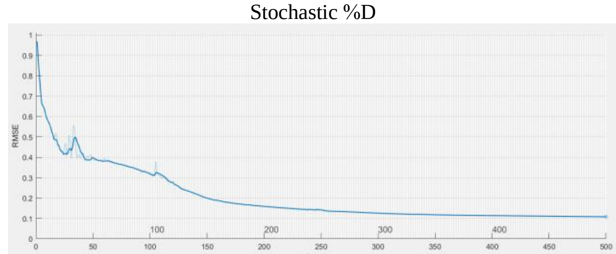
Stochastic %K



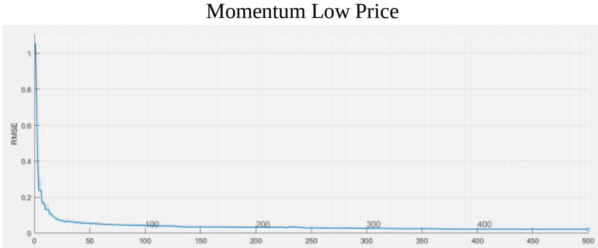
Stochastic %D



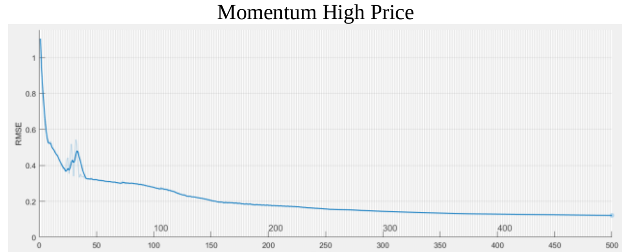
Momentum Low Price



Momentum High Price



OBV



RSI

FIGURE 7
LSTM MODEL LEARNING PROCESS DIAGRAM FOR EACH OF THE *VEBSADER* STOCK INDICATORS

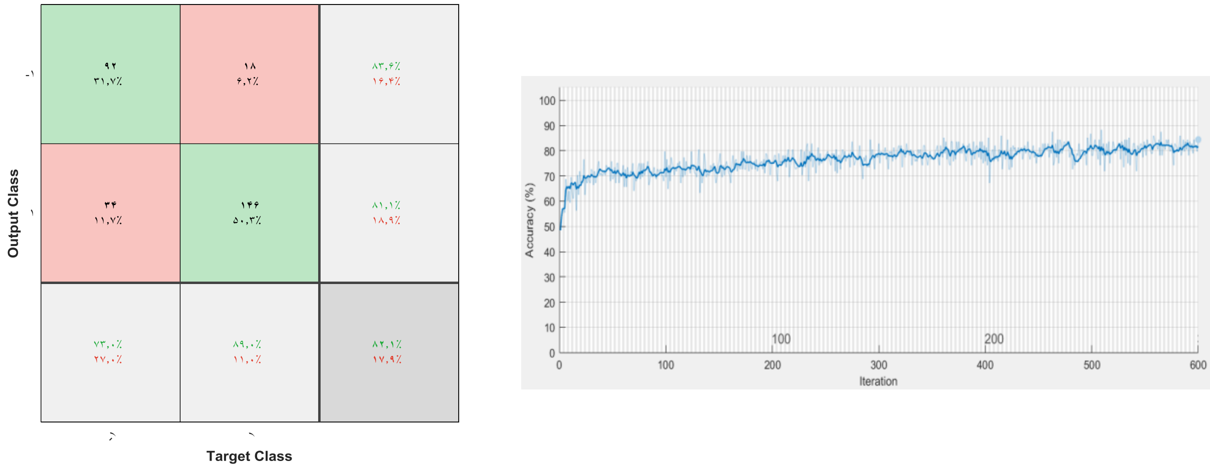


FIGURE 8
DIAGRAM OF THE LEARNING PROCESS (RIGHT) AND CONFUSION MATRIX (LEFT) OF THE LSTM-CNN Model For *Vebsader* Stock

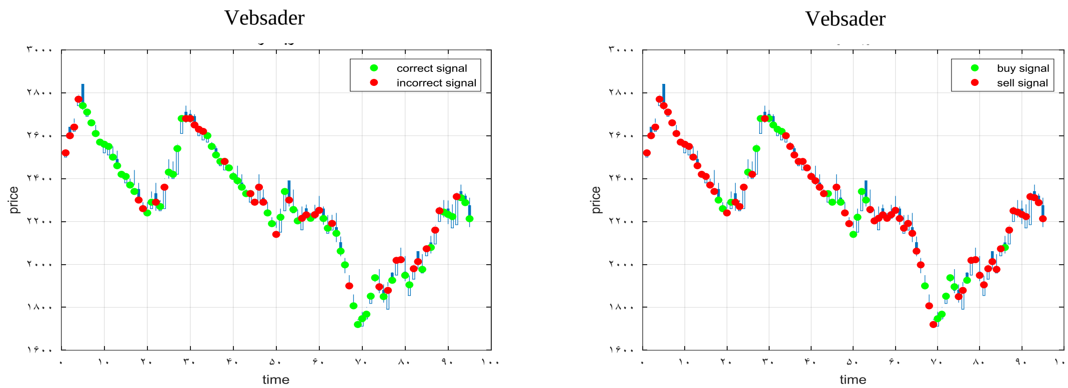


FIGURE 9
DIAGRAM OF THE TRADING PROCESS (RIGHT) AND THE ACCURACY OF THE ISSUED SIGNALS (LEFT) FOR THE *VEBSADER* STOCK WITH THE LSTM-CNN MODEL

FIGURE 10 shows the learning process diagram (right) and confusion matrix (left) of the LSTM-MLP model for the *Vebsader* stock. As can be seen, the model has reached the minimum value of mean square error (MSE) in the third epochs. In addition, the left diagram shows the confusion matrix and according to it, the model accuracy is 72%.

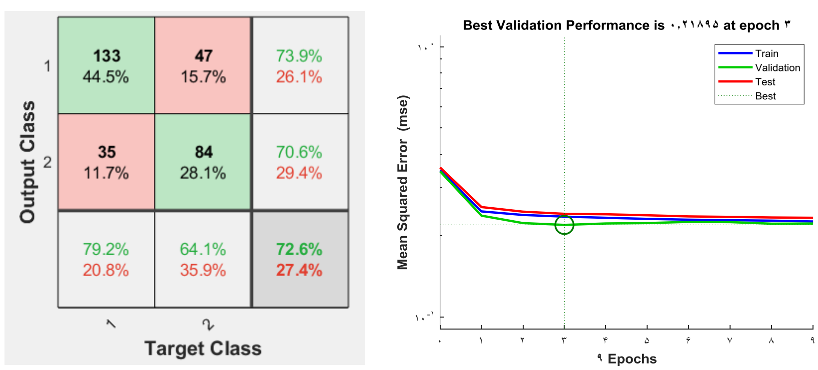


FIGURE 10
DIAGRAM OF THE LEARNING PROCESS (RIGHT) AND CONFUSION MATRIX (LEFT) OF THE LSTM-MLP MODEL FOR *VEBSADE*R STOCK

FIGURE 11 shows the trading process diagram (right) and the precision of the issued signals (left) for the *Vebsader* stock with the LSTM-MLP model.

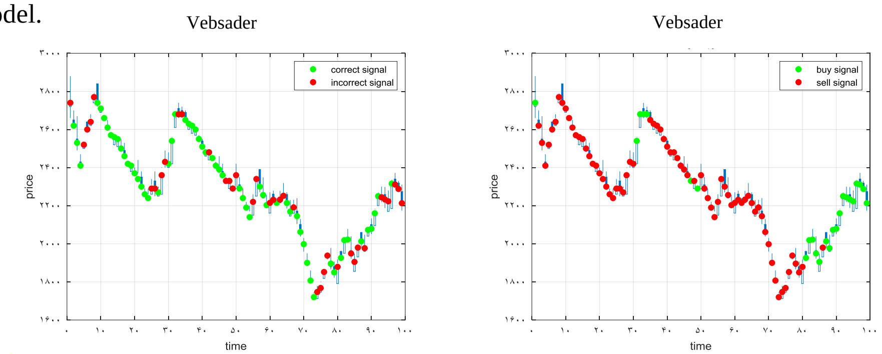
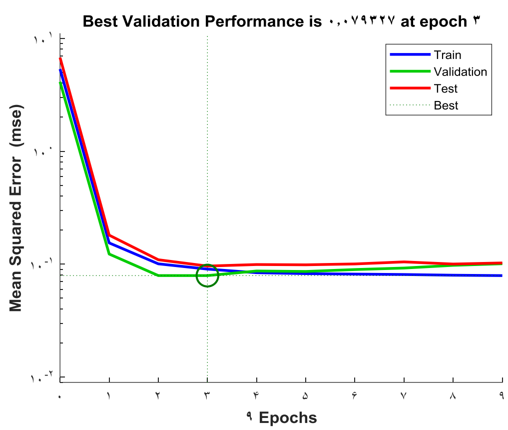


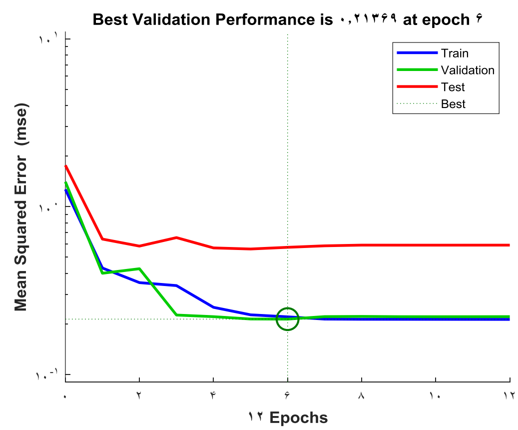
FIGURE 11

DIAGRAM OF THE TRADING PROCESS (RIGHT) AND THE ACCURACY OF THE ISSUED SIGNAL (LEFT) FOR THE VEBSADER STOCK WITH THE LSTM-MLP MODEL

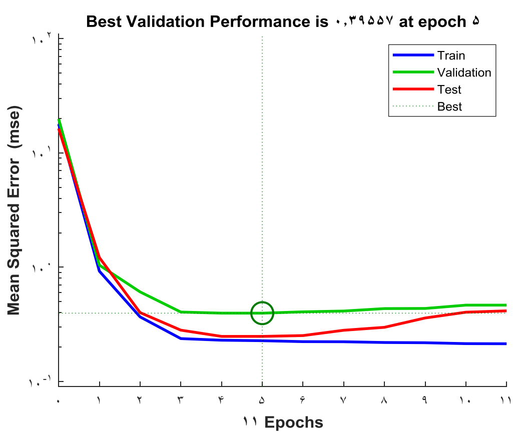
FIGURE 12 illustrates the learning process diagram of the RNN model for each Vebsader stock indicators. As can be seen, the indicators reached the minimum MSE value at different stages of the learning process(epochs).



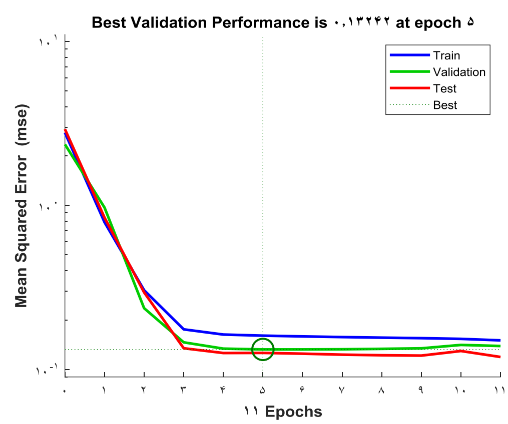
Stochastic %K



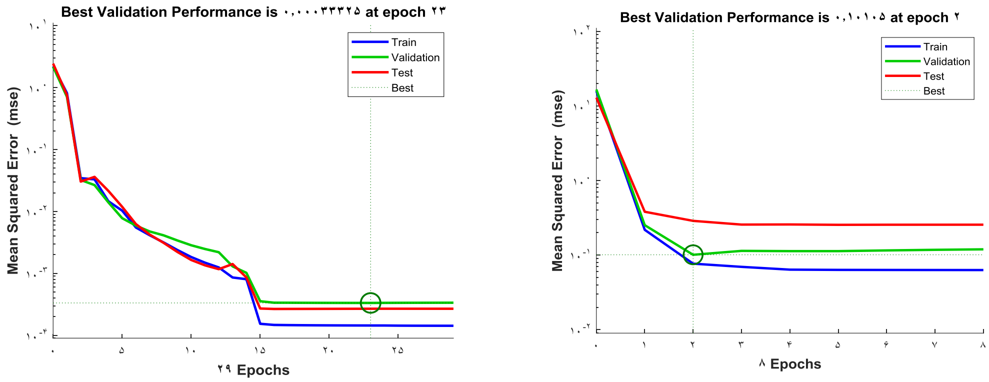
Stochastic %D



Momentum Low Price



Momentum High Price



OBV RSI

FIGURE 12
DIAGRAM OF THE LEARNING PROCESS FOR THE RNN MODEL IN EACH OF VEBSADER STOCK INDICATORS

FIGURE 13 shows the learning process of the RNN-CNN model, and the model accuracy is stable at 75%.

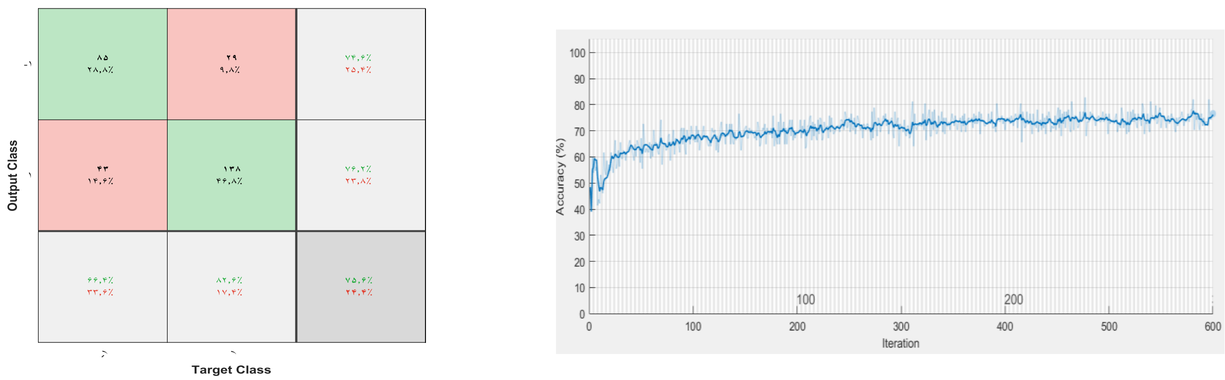


FIGURE 13
DIAGRAM OF THE LEARNING PROCESS PROCESS (RIGHT) AND THE CONFUSION MATRIX (LEFT)FOR THE RNN-CNN MODEL FOR VEBSADER STOCK

FIGURE 14 illustrates the diagram of the trading process (right) and the accuracy of the issued signals (left) for the *Vebsader* in the RNN-CNN model.

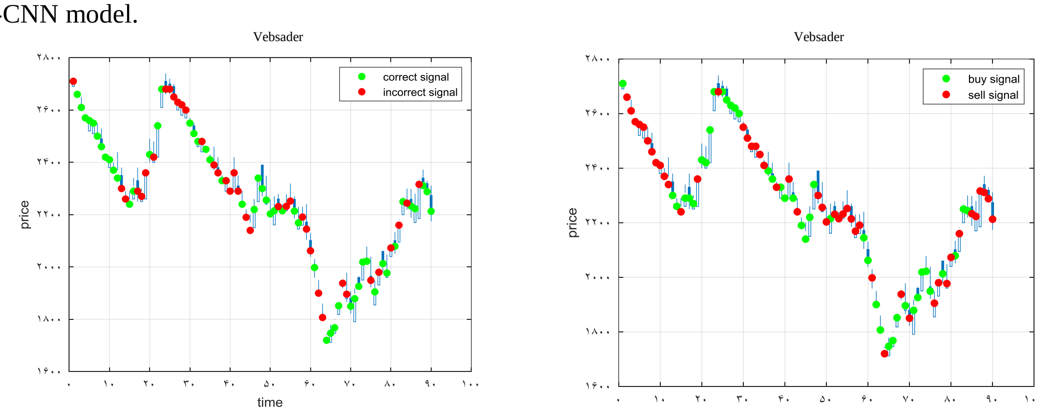


FIGURE 14
DIAGRAM OF THE TRADING PROCESS (RIGHT) AND THE ACCURACY OF THE ISSUED SIGNALS (LEFT) FOR VEBSADER STOCK IN RNN-CNN MODEL

FIGURE 15 shows the diagram of the learning process (right) and confusion matrix (left) of the RNN-MLP model for *Vebsader* stock. As can be seen, the MLP model has reached the minimum value of mean square error (MSE) in the 13th epochs and accuracy is 75.2%.

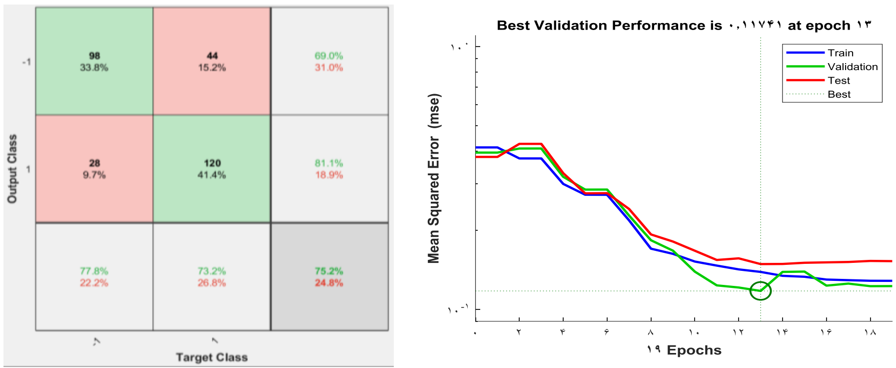


FIGURE 15
LEARNING PROCESS DIAGRAM (RIGHT) AND CONFUSION MATRIX (LEFT) OF THE RNN-MLP MODEL FOR VEBSADER STOCK

FIGURE 16 shows the diagram of the trading process (right) and the accuracy of issued signals (left) for *Vebsader* stock with the RNN-MLP model.

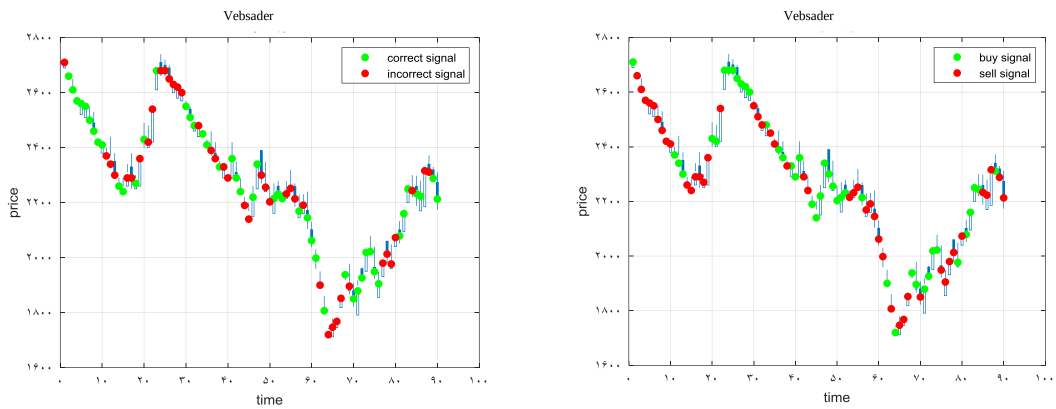


FIGURE 16
DIAGRAM OF THE TRADING PROCESS (RIGHT) AND THE ACCURACY OF THE ISSUED SIGNALS (LEFT) FOR THE *VEBSADER* STOCK MODEL IN RNN-MLP MODEL

Table 2 shows the performance results of RNN and LSTM neural networks in the learning state. These results are shown based on the RMSE index for each indicator. As can be seen, the performance of the LSTM method is better than the RNN method for all the indicators.

TABLE 2
THE PERFORMANCE RESULTS OF RNN AND LSTM NEURAL NETWORKS IN THE LEARNING STATE(RMSE)

stock	**RNN**					
	Stochastic %D	Stochastic %K	Momentum High Price	Momentum Low Price	RSI	OBV
Vebsader	۰,۵۲۶۹	۰,۲۹۳۸	۰,۴۲۲۲	۰,۵۰۹۲	۰,۳۱۵۵	۰,۰۴۱۰
Vaghadir	۰,۵۶۵۸	۰,۳۱۴۴	۰,۴۰۵۳	۰,۴۳۵۴	۰,۳۳۷۷	۰,۱۴۰۱
Khodro	۰,۵۴۹۸	۰,۳۰۹۶	۰,۴۳۸۰	۰,۴۵۷۸	۰,۳۴۵۶	۰,۳۵۱۶
Femeli	۰,۷۰۲۵	۰,۵۷۴۷	۰,۶۶۹۱	۰,۷۷۲۵	۰,۶۵۳۹	۰,۲۷۷۰
Shapena	۰,۵۰۹۷	۰,۲۸۷۹	۰,۳۴۵۷	۰,۴۸۸۲	۰,۳۲۹۷	۰,۱۷۴۹
stock	LSTM					
	Stochastic %D	Stochastic %K	Momentum High Price	Momentum Low Price	RSI	OBV
Vebsader	۰,۱۸۹۸	۰,۱۶۷۴	۰,۰۷۴۱	۰,۲۱۵۱	۰,۱۲۵۳	۰,۰۱۹۸

Vaghadir	٠,٢٢٦٤	٠,٢٠٣٤	٠,٢٥٥٥	٠,١٧٢٢	٠,١٤٦٣	٠,٠٣٦٥
Khodro	٠,٢٧٥٦	٠,١٧٢٤	٠,٢٤٦٠	٠,٢٦٥٠	٠,١٥٦٨	٠,٠٠٠٥
Femeli	٠,٢٥٦٥	٠,٢٥٥٤	٠,١٧٠٥	٠,٣٢٤٩	٠,١٤٣٧	٠,٠٤٥٩
Shapena	٠,٢٢٦٤	٠,١٥٧١	٠,١٦٦٨	٠,٢٥٦٤	٠,١١٦٦	٠,٠٠٠٥

Table 3 shows the performance results of combined algorithms in the learning state, including LSTM-CNN, LSTM-MLP, RNN-CNN and RNN-MLP algorithms. As can be seen, the accuracy of the combined LSTM-CNN method is higher than the other combined methods in 4 out of 5 investigated stocks. Only in Vaghadir stocks the combined RNN-MLP method obtained higher accuracy with a small difference from the LSTM-CNN method. Therefore, the ability of the combined LSTM-CNN method to learn the buying and selling signal is better than the other methods, approximately.

TABLE 3
PERFORMANCE RESULTS OF COMBINED ALGORITHMS IN THE LEARNING STATE(ACCURACY)

stock	LSTM-CNN	LSTM-MLP	RNN-CNN	RNN-MLP
Vebsader	٨٢,١	٧٢,٦	٧٥,٦	٧٥,٢
Vaghadir	٧٦,٦	٦٦,٩	٧١,٣	٧٧,٦
Khodro	٨٥,٩	٧٥,٦	٨٠,٧	٧٧,١
Femeli	٨٤,٥	٦٥,٦	٧٢,٧	٧٢,٤
Shapena	٨٣,٤	٦٣,٥	٧٦,١	٧٦,٨

Table 4 shows the performance results of the combined algorithms in the testing state. As the same, the performance of the combined LSTM-CNN method has achieved the highest accuracy among the other methods. From a general view, the use of deep learning model (LSTM) for stock indicator, forecasting or buying and selling classification can have acceptable performance.

TABLE 4
PERFORMANCE RESULTS OF THE COMBINED ALGORITHMS IN THE TESTING STATE(ACCURACY)

stock	LSTM-CNN	LSTM-MLP	RNN-CNN	RNN-MLP
Vebsader	٧٤,٥٦٦٧	٦٧,٠٤٦٥	٦٢,٤٠٠٠	٥٦,٨٤٤٤
Vaghadir	٥٨,٨٥٨٣	٥٦,٩٤٥٥	٤٦,٨٤٤٤	٤٩,٠٦٦٧
Khodro	٦٦,٧٨٨٩	٤٧,٨٥٤٥	٦٠,١٧٧٨	٥٣,٥١١١
Femeli	٦٥,٦٣٢٠	٥٦,٩٤٥٥	٥٦,٨٤٤٤	٦٠,١٦٢٩
Shapena	٧٠,٦٩٣٨	٦٠,٩٨٥٩	٦٢,٤٠٠٠	٦٥,٣٤٨٥

Proposed algorithmic trading system is presented in FIGURE 17. As can be shown, the price data of OHLC is stored in the database first. Then, the desired stock indicators that have been screened using the feature selection algorithm are calculated. The LSTM algorithm forecasts these indicators. The values of the indicators and their forecasts are entered into the model as inputs to the CNN algorithm. Finally, the CNN algorithm determines the buying or selling action.

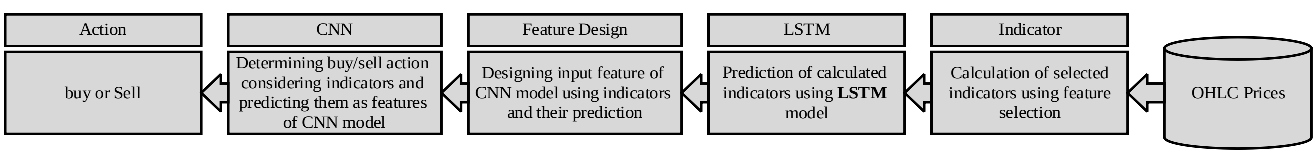


FIGURE 17
PROPOSED ALGORITHMIC TRADING SYSTEM

CONCLUSION

In recent years, modeling financial markets based on artificial intelligence methods has gained the attention of many reserchers. Artificial intelligence allows for breaking big data hidden and invisible patterns and extracting valuable information. These markets can benefit from data mining, artificial intelligence and machine learning methods. This study focuses on using and evaluating a hybrid method based on artificial intelligence for modeling price and stock indicators for algorithmic trading systems. According to this study, deep learning models(LSTM) are more efficient than the surface neural network models(RNN). It shows a higher performance in predicting time series indicators and determining buy/sell signals. The results of this research, which was evaluated at the Tehran Stock Exchange as a case study. The following practical findings and recommendations can be made to market participants to improve the investment process in this market:

- Using models based on artificial intelligence in the Tehran Stock Exchange market can increase trading efficiency and reduce trading errors to an acceptable level. Therefore, it is recommended for capital market actors, including portfolio managers, investment companies and retail traders, focus more on using these approaches.
- Since models based on artificial intelligence are always dependent on the data, in this field are advised to design the trading system based on coherent and up-to-date databases. Therefore, the various possibilities created in the market in the latest time have to be provided for the trading system.

The research can be developed in various aspects to consider some limitations of this study, as follows:

- **Explore Alternative AI Models:** This study utilized LSTM and CNN algorithms to model the financial market. Future research could explore other deep learning models, such as Bi-Directional LSTM or Deep MLP models for comparative analysis.
- **Optimize Model Parameters:** The performance of deep learning models can be improved by optimizing their parameters using meta-heuristic algorithms, such as genetic algorithms. Future research could benefit from this approach and compare the results obtained with this study.
- **Broaden Market Scope:** This research investigated only a subset of market shares. Future studies could process larger or more diverse datasets or other financial markets to evaluate the entire market and enable a comprehensive comparison of the performance of different algorithms.

REFRENCES

[١] A. Bigiotti and A. Navarra, "Optimizing Automated Trading Systems," The 2018 International Conference on Digital Science, pp. 254-261, 2018.

[٢] O. B. Sezer and A. M. Ozbayoglu, "Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach," Applied Soft Computing, vol. 70, pp. 525-538, 2018. https://doi.org/10.1016/j.asoc.2018.04.024

[٣] K. A. Althelaya, E.-S. M. El-Alfy, and S. Mohammed, "Evaluation of bidirectional lstm for short-and long-term stock market prediction," in 2018 ٩th international conference on information and communication systems (ICICS), pp. 151-156: IEEE, 2018. https://doi.org/10.1109/IACS.2018.8355458

[۴] C. Martín, D. Quintana, and P. Isasi, "Grammatical Evolution-based ensembles for algorithmic trading," Applied Soft Computing, vol. 84, 2019. https://doi.org/10.1016/j.asoc.2019.105713

[۵] M. R. Alimoradi and A. H. Kashan, "A league championship algorithm equipped with network structure and backward Q-learning for extracting stock trading rules," Applied Soft Computing, vol. 68, pp. 478-493, 2018. https://doi.org/10.1016/j.asoc.2018.03.051

[۶] J. B. Chakole, M. S. Kolhe, G .D. Mahapurush, A. Yadav, and M. P. Kurhekar, "A Q-learning agent for automated trading in equity stock markets," Expert Systems with Applications, vol. 163, 2021. https://doi.org/10.1016/j.eswa.2020.113761

[٧] Y. Li, W. Zheng, and Z. Zheng, "Deep robust reinforcement learning for practical algorithmic trading," IEEE Access, vol. 7, pp. 108014-108022, 2019. https://doi.org/10.1109/ACCESS.2019.2932789

[٨] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, "Deep direct reinforcement learning for financial signal representation and trading", IEEE transactions on neural networks and learning systems, vol. 28, no. 3, pp. 653-664, 2016. https://doi.org/10.1109/TNNLS.2016.2522401

[٩] T. Shintate and L. Pichl, "Trend prediction classification for high frequency bitcoin time series with deep learning," Journal of Risk and Financial Management, vol. 12, no. 1, p. 17, 2019. https://doi.org/10.3390/jrfm12010017

[١٠] P. Arvidsson and T. Ånhed, "Sequence-to-sequence learning of financial time series in algorithmic trading", 2017.

[١١] S. Sun, S. Wang, and Y. Wei, "A new ensemble deep learning approach for exchange rates forecasting and trading," Advanced Engineering Informatics ,vol. 46, 2020. https://doi.org/10.1016/j.aei.2020.101160

[١٢] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, and H. Fujita, "Adaptive Stock Trading Strategies with Deep Reinforcement Learning Methods," Information Sciences, 2020. https://doi.org/10.1016/j.ins.2020.05.066

[١٣] T. Théate and D. Ernst, "An application of deep reinforcement learning to algorithmic trading", In Expert Systems with Applications,2021. https://doi.org/10.1016/j.eswa.2021.114632

[١۴] L. Weng, X. Sun, M. Xia, J. Liu, and Y. Xu, "Portfolio Trading System of Digital Currencies: A Deep Reinforcement Learning with Multidimensional Attention Gating Mechanism," Neurocomputing, 2020. https://doi.org/10.1016/j.neucom.2020.04.004

[١۵] A. Nan, A. Perumal, and O. R. Zaiane, "Sentiment and Knowledge Based Algorithmic Trading with Deep Reinforcement Learning", Database and Expert Systems Applications, 33rd International Conference, DEXA 2022, Vienna, Austria, August 22–24, 2022. https://doi.org/10.48550/arXiv.2001.09403

[١۶] E. Ponomarev, I. V. Oseledets, and A. Cichocki, "Using Reinforcement Learning in the Algorithmic Trading Problem," Journal of Communications Technology and Electronics, vol. 64, no. 12, pp. 1450-1457, 2019. https://doi.org/10.1134/S1064226919120131

[١٧] C. Betancourt and W.-H. Chen, "Deep reinforcement learning for portfolio management of markets with a dynamic number of assets," Expert Systems with Applications , 2020. https://doi.org/10.1016/j.eswa.2020.114002

[١٨] K. Lei, B. Zhang, Y. Li, M. Yang, and Y. Shen, "Time-driven feature-aware jointly deep reinforcement learning for financial signal representation and algorithmic trading," Expert Systems with Applications, vol. 140, 2020. https://doi.org/10.1016/j.eswa.2019.112872

[١٩] R. Ruiz-Cruz, "Portfolio modeling for an algorithmic trading based on control theory," IFAC-PapersOnLine, vol. 51, no. 13, pp. 390-395, 2018. https://doi.org/10.1016/j.ifacol.2018.07.310

[٢٠] J. Eapen, D. Bein, and A. Verma" ,Novel deep learning model with CNN and bi-directional LSTM for improved stock market index prediction," in 2019 IEEE 9th annual computing and communication workshop and conference (CCWC), pp. 0264-0270: IEEE, 2019. https://doi.org/10.1109/CCWC.2019.8666592

[۲۱] Y. Li, S. Wang, Y. Wei, and Q. Zhu" ,A New Hybrid VMD-ICSS-BiGRU Approach for Gold Futures Price Forecasting and Algorithmic Trading," IEEE Transactions on Computational Social Systems, vol. 8, no. 6, pp. 1357-1368, 2021. https://doi.org/10.1109/TCSS.2021.3084847

[۲۲] M. E. Aloud and N. Alkhamees, "Intelligent Algorithmic Trading Strategy Using Reinforcement Learning and Directional Change," IEEE Access, vol. 9, pp. 114659-114671, 2021. https://doi.org/10.1109/ACCESS.2021.3105259

[۲۳] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997. https://doi.org/10.1162/neco.1997.9.8.1735

[۲۴] F. Bodaghi, A.Owhadi, Ar.Khalili Nasr and M.Khadem Sameni," Stock Price Forecasting in Iran Stock Market: A Comparative Analysis of Deeplearning Approaches", International Journal of Web Research, Vol. 6, No. 2, 2023. https://doi.org/10.22133/ijwr.2024.407077.1170

[۲۵] Y. Li, P. Ni and V.Chang, "Application of deep reinforcement learning in stock trading strategies and stock forecasting", Computing 102, pp.1305–1322 , 2020. https://doi.org/10.1007/s00607-019-00773-w

[۲۶] S.M. Rida, E.Hamza and Z.Taher,"From Technical Indicators to Trading Decisions: A Deep Learning Model Combining CNN and LSTM", International Journal of Advanced Computer Science and Applications,Vol. 15, No. 6, 2024. https://doi.org/ 10.14569/IJACSA.2024.0150685

[۲۷] Y. Tan, C. Hu, K. Zhang, K. Zheng, E. A. Davis and J. S. Park, "LSTM-Based Anomaly Detection for Non-Linear Dynamical System," in *IEEE Access*, vol. 8, pp. 103301-103308, 2020, doi: 10.1109/ACCESS.2020.2999065

[۲۸] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, Deep learning. MIT press Cambridge, 2016.

[۲۹] D. Ruan, F. Zhang, J. Yan, "Transfer Learning Between Different Working Conditions on Bearing Fault Diagnosis Based on Data Augmentation", vol 54, Issue 1, pp. 1193-1199, 2021. https://doi.org/10.1016/j.ifacol.2021.08.141

[۳۰] H. H. Aghdam and E. J. Heravi, "Guide to convolutional neural networks ", Springer, New York, NY, vol. 10, no 51, pp. 978-973, 2017.