International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Modelling of Hyperparameter Tuned Bidirectional Long Short-Term Memory with TLBO for Stock Price Prediction Model

**T. Swathi[1], N. Kasiviswanath[2], A. Ananda Rao[3]**

**Submitted**: 21/03/2023 **Revised**: 24/05/2023 **Accepted**: 12/06/2023

**Abstract:** Because of the stock market's importance as a forum for investors, precise forecasting of stock market developments remains a popular study topic among monetary industry executives and academics. Forecasting stock prices is an exciting endeavour that is complicated by the stock Earlier research that used accurate replicas and machine learning approaches to forecast short-term changes in stock prices was primarily focused on forecasting short-term changes in stock prices. Considering this, research develops an Hyperparameter Tuned Bidirectional Long Short-Term Memory (HPT-BiLSTM), a unique BiLSTM for stock price estimationThe recommended HPT-HCLSTM method combines forecasting, parameter optimization, and preprocessing. The HPT-BiLSTM approach is used in the HPT-BiLSTM strategy to predict stock prices. The BiLSTM technique's hyper parameters are also improved using the teaching and knowledge-based optimization (TLBO) strategy, leading to noticeably decreased error levels. A variety of imitations were carried out in order to confirm the HPT-BiLSTM method's enhanced forecast presentation, and the significances established that the technique's higher presentation could be noticed in a number of different facets.

***Keywords**: Stock price prediction, stock market, time series, prediction, deep learning, hyperparameter tuning, TLBO algorithm*

## 1. Introduction

As a result of the network's phenomenal growth, multimedia data from mobile phones, social networking sites, news websites, and financial websites is rapidly growing and having an impact on our real-world daily lives. It is essential to learn how to fully utilise big data in the age of the internet so that it can give us accurate and useful information (Yi, Y. 2019). To lower decision-making risk, investors, for instance, might estimate future price trends of financial assets using financial data (Cavalcante, R. C., 2016 & Song, H., 2020). On the other hand, it can be difficult for investors to gain crucial information on budget allocation right away. In order for investors to make wise investment decisions, certain technical or quantitative tools must be employed to foresee financial market swings.

There have been several forecasting methods for financial stock data published, and they all perform amazingly well (Thangavel, R. 2013). Traditional time series prediction techniques including the Auto Regressive (AR), Moving Average (MA), Auto Regressive and Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA) models aim to obtain the best guess (Nti, I. K., 2020).

There are several restrictions on the statistical inference that can be used by the approaches mentioned above to characterise and evaluate the relationship between variables. On the one hand, these strategies are unable to account for nonlinear variations in stock price since they presume a linear link between model structure (Tang, H., 2019). Financial time series are extremely noisy, time-varying, dynamic, etc., whereas these methods presume that data variance is constant (Chen, W., 2021).

To fix the faults, a number of machine learning techniques have been used to reproduce the nonlinear relationship in financial time series. Artificial Neural Networks (ANNs) are frequently employed to manage monetary time series because of their superior nonlinear mapping and generalisation capabilities (Sezer, O. B., 2020 & Yu, P., 2020).Unlike econometric statistical models, ANN models do not demand a rigid model structure or additional sets of assumptions. For instance, a hybrid model combining ANN with the Exponential Generalized Autoregressive Conditional Heteroscedasticity (EGACH) model may be used to predict S&P 500 volatility. The experimental results show that the hybrid model has a lesser test error than any individual econometric model. By combining

*1Research Scholar, Department of CSE, Jawaharlal Nehru Technological University, Anantapur (JNTUA), Ananthapuramu 515002, India.*
*Assistant Professor, Department of CSE,*
*G. Pulla Reddy Engineering College, Kurnool 518007, Affiliated to Jawaharlal Nehru Technological University Anantapur, India.*
*Email Id: swathitudi@gmail.com*
*2Professor & Head Department of CSE, G Pulla Reddy Engineering College, Kurnool.,518007, India.*
*Email Id: hodcse@gprec.ac.in*
*3Professor, Vice Chancellor Rayalaseema University, Kurnool,518007, India.*
*Email Id: akepogu@gmail.com*

ARMA and Exponential Smoothing (SE) linear models with recurrent neural networks, Rather et al. (2015) forecasted market returns (RNN). The results of the experiment demonstrate that RNN can enhance prediction performance much more.

LSTM networks have been offered as a possible solution to the problem of vanishing gradients in RNNs in recent years. To find long-term correlations in time series data, these systems use storage cells and entrances (Pokle, S. B. 2019). Because of the benefits it offers, LSTM has gained widespread acceptability for the willpower of dated series forecasting (Sayeed, R. F., 2015). Nelson and his colleagues used LSTM to develop a method for forecasting equity prices (Hua, Y., 2019). Several studies have found that the LSTM model surpasses other mechanism learning models such as SVM, GA, and BP Neural Network in terms of accuracy. For example, the forward LSTM forecasts the future using historical data, but the reverse LSTM forecasts the future using current data. Both of these models employ recurrent neural networks. As a result, an increasing number of academics are turning to BiLSTM for help in time series prediction. The final prediction is more accurate since the BiLSTM network can incorporate data from the past as well as the future. With the help of multitask RNNs, it is possible to use EEG to figure out what a gesture means and to predict how bad a disease will be based on how it changes over time (Ding, G., 2020).

The aim of this study was to develop a long short-term memory hybrid convolutional neural network for hyperparameter-tuned stock price forecasting (HPT-BiLSTM). The HPT-BiLSTM approach, which is suggested, can successfully develop a DL model to anticipate stock prices. BiLSTM is used in this approach. In addition, the TLBO method is used to optimize the hyperparameters of the BiLSTM technique, resulting in error values that are as small as feasible. Because of their greater computational capabilities, enhanced learning, and storage capacity, higher-order networks are more widespread than standard neural networks. This is due to the fact that higher-order networks are more similar to current computers. To the best of our knowledge, no one has tried to optimise the classification performance of a higher-order neural network (particularly the Pi-Sigma neural network).A novel population-based instructional learning-based optimization technique was used to train the neural network. The design of the TLBO model, which is used to establish the hyperparameters of the BiLSTM technique, highlights the study's originality. Extensive simulation study is carried out, and the results are reviewed from a variety of perspectives to determine whether the HPT-Bi-LSTM technique improves overall performance.

The remaining sections of the paper are arranged as follows: Section 2 presents a review of the relevant literature. In Section 3, we go over the proposed technique in great detail. In Section 4, we'll discuss how to put the proposed approach into action and assess its efficacy. Section 5 of the paper contains the conclusion.

## 2. Literature Survey

Mehtab *et al*. (2020) ML, DL, and statistical approaches were combined to produce a reliable and accurate stock price forecasting architecture. They utilise routine stock price information at 5-minute intervals from a well-known company listed on the National Stock Exchange of India (NSE). A new method of portfolio construction developed by Subulakshmi (2018) is based on a hybrid approach that blends ML stock prediction algorithms with MV portfolio selection methods. First, a hybrid strategy that combines an IFA and XGBoost is illustrated for forecasting equities for the coming few days. The hyperparameters of the XGBoost are to be optimised by the IFA. Then, portfolios were created by selecting stock prices with high potential returns using the MV technique.

Ramalingam (2021) A hybrid model integrating deep learning and machine learning was developed to forecast stock values. They will pick NIFTY 50 index values from the National Stock Exchange of India beginning in January 2015 and continuing through December 2019. Using NIFTY's historical data, they built a number of machine learning (ML) models and used them to predict the NIFTY 50's closing prices for the first few weeks of 2019. Even though they utilise a wide variety of classification models to forecast the NIFTY index's behaviour, researchers have developed a number of regression strategies for predicting the index's actual Close values. The next step is to use CNN to build a DL-based regression methodology and then to use walk-forward validation to improve the predicted accuracy of the system.

Zhang *et al*. (2021) We advocated for a novel SVR-ENANFIS method for predicting stock prices by combining the best features of the two existing models. To begin, the SVR model projects where the technical indicators' values will be in the future. The closing price is then predicted using ENANFIS based on the predictions from the first phase. Two novel hybrid techniques, EMD-CNNLSTM and CEEMD-CNN-LSTM, were developed by Rezaei et al. in 2021. These techniques are capable of extracting temporal sequences and deep features for one-step-ahead predictions. According to the presented model, a few collaborations may be found as this algorithm is integrated, perhaps enhancing this method's analytical capability.

Abe *et al*. (2020) For efficient investment management, develop a cross-sectional daily stock price prediction model using DL. Utilize market closing data to build a portfolio, then make investments, for instance, the following day when the market reopens. To confirm the model's efficacy, they also conduct experimental evaluations on the Japanese stock market. Ampomah et al. (2020) evaluate the performance of ensemble tree ML systems in predicting stock prices (RF, XG, BC, Ada, ET, and VC). To conduct the research, eight different stock data sets were employed, each selected at random from three major exchanges (NSE, NYSE, and NASDAQ).Databases for testing and training are divided into many categories.

Yu *et al*. (2020) Financial product price data is processed into a one-dimensional (1D) series utilising features acquired from chaotic system prediction, which have varying time dimensions. This price sequence is then replicated using the price sequence PSR method. (It has been discovered that the definition of the word provided in chaos-related texts is inadequate.) One concise yet nebulous definition is as follows: "Chaotic systems are characterised by their seemingly random growth over phase space and their sensitive dependency on starting circumstances."A DNN-based prediction technique is created and utilised to forecast stocks based on the DL algorithm's LSTM and PSR approaches.

N. Naik *et al*. (2019) employ both full ensemble empirical mode decomposition with adaptive noise and EMD, which stands for empirical mode decomposition, in their hybrid financial time series forecasting. (CEEMDAN) After that, an LSTM model is qualified on the remaining constituent's intrinsic mode function (IMF), and the final prediction is obtained by putting the forecasts for each constituent together. Despite the numerous studies on monetary forecasting that have been published, no model has been developed specifically for the Moroccan market. On the other hand, many works have been released expressly for markets outside of Morocco.

Lu *et al*. (2021) suggested using CNN-BiLSTM-AM to predict stock closes. CNN and (BiLSTM, and AM). CNN could extract stock features. LSTM prevents RNN's gradient explosion and gradient disappearance. Researcher’s proposed using computational intelligence to improve approximation accuracy. It combines the neural-like structure of SGT with polynomial inputs. This expansion uses second-degree Wiener polynomial (Hardas, B. M., 2017). This combination improves classification, regression, image identification, and scaling accuracy. SGTM neural-like structure maintains system speed in training and use modes.

## 3. Proposed System

This study creates a novel HPT-BiLSTM technique for accurately evaluating stock values. Pre-processing, BiLSTM-based forecasting, and TLBO-based hyperparameter tuning are all part of the HPT-BiLSTM strategy. Figu 1 illustrates the future HPT-BiLSTM approach.

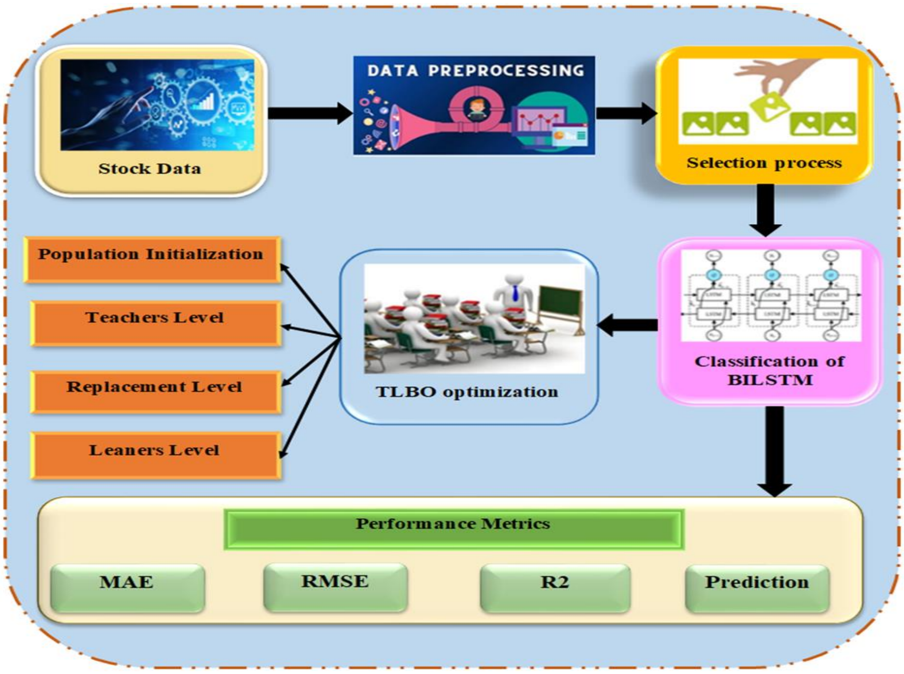


**Fig 1**. The HPT Bi-LSTM model's overall process

### 3.1 Data pre-processing

A stock's starting and closing values, as well as its highs and lows, can be studied over the course of its trading history. Investors are mainly interested about the final price. As a result, our calculations will be focused on the single most important input variable: the closing price. Our research found that the pricing of the input variables had no effect on the outcomes.

Figure 2 shows a typical exercise and an examination set, both of which contain all of the material. The use of a moving window as a form of data segmentation is commonplace. BiLSTM is constantly put through its paces using N + 1. (i.e., the closing price of the previous N days is used to forecast the closing price of the following day). Following the completion of the training set, the data from the previous N days of the training set is used to forecast the first day of the test set. Finally, one day is added to the input data rolling window, resulting in no data on Day X. It's feasible to use test data from one day to predict test results from the next day, and so on. When the last day of the test set has been forecasted, it's over.

Because each standard may have its own unique area and gauge, smoothing and normalizing stock price data is essential to decrease stock data noise and enhance stock price pattern detection. The act of transforming values measured on several balances into a single, unchanging gauge is known as data normalization. The outcomes of data smoothing are shown in the equation 1 as:

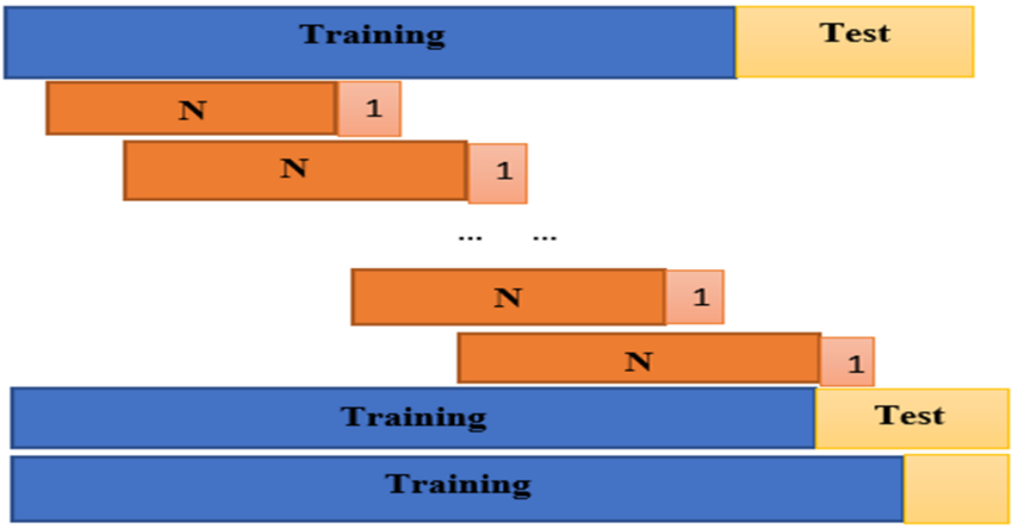


**Fig 2:** The rolling window schematic diagram

$$w_{(p,q)} = \begin{cases} 0, & q = 1 \\ \dfrac{w_q - w_{q-1}}{w_{q-1}}, & q > 0 \end{cases} \qquad (1)$$

where $w_{(p,q)}$ signifies the consequence of information flattening at the $t^{th}$ day. Here, when t = 1, we set $w_{(p,q)}$ = 0.

The knowledge of BiLSTM is in detail to get standard designs which can be exaggerated by "min–max" standardization of the data set. +The "min–max" standardization technique is publicized as equation 2.

$$w_{(m,q)} = \frac{w_{(p,q)} - w_{(p,\min)}}{w_{(p,\max)} - w_{(p,\min)}} \qquad (2)$$

where $w_{(m,q)}$ signifies the information after standardization, $w_{(m,q)}$ is unique information, $w_{(p,\min)}$ is the least among the information usual, and $w_{(p,\max)}$ is the extreme.

Because of this, denormalization and de-smoothness are compulsory at the conclusion of the procedure of forecast in order to acquire the initial price, which is strongminded in equation 3 and 4 as follows.

$$w\wedge_{(p,q)} = w\wedge_{(m,q)} \left[ w_{(p,\max)} - w_{(p,\min)} \right] + w_{(p,\min)}, \qquad (3)$$

$$w\wedge_{q} = w\wedge_{(p,q)}\, w_{t-1} + w_{t-1}, \qquad (4)$$

where $w^{\wedge}_{(m,q)}$ signifies the forecast information, $w^{\wedge}_{(p,q)}$ denotes the forecast information after previously normalized, and $w^{\wedge}_{q}$ denotes the forecast data after both previously normalized and de-smoothness.

### 3.2. Selection

The selection operator is the evolutionary algorithm's equivalent of the mutation operator. Existing individuals' fitness scores are used to choose the next generation in the BiLSTM proposed survivor selection technique.

The BiLSTM is optimised via a dynamic method (i.e., population). As a result, based on the training technique for that evolutionary generation, we may alter the fitness function and compute the fitness score for each BiLSTM. As this study demonstrates, there is no method to compare fitness results between generations. Because the change operators of the intended BiLSTM describe an assortment across various BiLSTM exercise targets, the desired offspring suggests effective training approaches. The selection tool for BiLSTM is called (,)-selection, and it takes both the appropriateness purpose and the change operatives into account.

When it's all said and done, offspring population $\{BiLSTM_i\}^{\lambda}_{i=1}$ is organized rendering to their appropriateness grooves $F_i$, the μ − best persons are designated as populace of the following group.

### 3.3. BiLSTM Based Prediction model

The next stage is to expand the BiLSTM-based forecast approach after the contribution information for the standard market has been preprocessed. The Bi-LSTM technique is used to tackle RNN's inclination evaporation problem. The Bi-LSTM has the capacity to either delete or increase the cell state statistics' quality. This capability has been available as an option on a gate basis. The Bi-LSTM recommends using three gates, referred to as the "contribute," "forget, and "manufactured" entries. The read, write, and reset procedures are provided by these entries, respectively.

**HPT-Bi-LSTM Algorithm**

***Input:*** $population\ size\ P = N, the\ number\ of\ mutations\ n_m the\ batch\ size\ m, batch\ data\ D, and\ initial\ weight\ w_o$

**output:** $close\ price\ of\ the\ next\ day$

1. $w = w_o$
2. $Initializes\ model\ parameter\ w\_o$
3. $for\ i = 1\ to \frac{m}{(Nn_m)}$
4. $param \longleftarrow w\ save\ model\ parameters$
5. $for\ j = 1\ to\ N$
6. $for\ k = 1\ to\ n_m$
7. $M(param) assign\ parameters\ to\ the\ model$
8. $Get\ a\ batch\ D\ as\ input\ x_i of\ EBiLSTM$
9. $Switch(k)$
10. $Case1: loss_{square}, param_{square} \longleftarrow M(x_i, square, param)$
11. $Case\ 2: loss_{abs}, param_{abs} \longleftarrow M(x_i, abs, param)$
12. $Case3: loss_{huber} \longleftarrow M(x_i, huber, param)$
13. $End\ switch$
14. $If\ K = n_m$
15. $Loss_{min} \longleftarrow \min\left(loss_{square}, loss_{abs}, loss_{huber}\right)$
16. $param_{new} \longleftarrow \left(loss - min, param_{square}, param_{abs}, param_{huber}\right)$
17. $w \longleftarrow param_{new}$
18. $end\ for$
19. $end\ for$
20. $end\ for$

Amongst them, $C_{l-1}$ implies the cell state in the previous component, $d_{l-1}$ refers the consequence of previous constituent, $y_l$ positions for the contemporary contribution, applied for producing novel recollection, and the subsequent information comprises the cell state $C_l$ transported advanced, original production dt.

The Bi-LSTM forgetting gate turns out to be nothing more than a valve. A significant amount of data would flow into the memory if the input gate was left open all the time. At the moment, erasing material from memory necessitates an extra step as part of the forgetting process. It's a gate that's easy to overlook. It's at (previous output) and d (l-1) (current input), and it provides all of the digits in the cell state C (l-1) a value between 0 and 1. (preceding state). An equation 5 implies that nothing was modified, whereas (0) indicates that nothing was altered at all. The following is the calculation formula:

$$F_l = Sigmoid(X_f [d_{l-1}, y_l] + b_f) \tag{5}$$

A stock's beginning and ending prices, as well as highs and lows, can be tracked over time. The final price is most important to investors. As a result, the ultimate pricing will serve as the foundation for all of our calculations. Our research found that the variable costs of input had no effect on the outcomes.

A contribution entry in an LSTM must enhance the current influence's state-of-the-art memory in order to avoid the NN from "forgetting" a portion of the previous state. It is possible to divide the input gate into two pieces. The most important component is the input threshold layer, a sigmoid layer that determines whether values should be changed. The second component, a tan*h* layer, is in charge of the grouping of a fresh application course that has been improved to get to this point. The equation 6 to 8 has been followed:

$$h_l = \sigma(Xn.[d_{l-1}, y_l] + b_n) \quad (6)$$

$$C^{\sim}{}_l = \tanh(X_m.[d_{l-1}, y_l] + b_m) \quad (7)$$

$$C_l = F_l * C_{l-1} + h_l * C^{\sim}{}_l \quad (8)$$

The letters Xn, bm, WM, and Ct stand for the weight matrix, the bias item used to update the status of the unit, the weight matrix itself, and the status of the updated memory units. The input gates h l and 〖C^~〗 in If the state of the time-step memory units requires updating, then perform the dot product as shown in Equation (8). The Forgetting Gate Ft calculates the dot product with C to remember the initial condition of the time-step memory units (l-1).

Because the subsequent gate from LSTM is an important part of the current prompt, the elimination of a newly produced state that was used for regulating the state of recollection units on this layer is required. This gate is the outcome of the current situation. The gate's current competence is seen as a function of its most recent state, the output from the gate before it, and the contribution it is now receiving. An equation 9 and 10 is the reason behind the control calculation as follows:

$$d_l = O_l * \tanh(C_l) \quad (9)$$

$$O_l = \sigma(X_o.[d_{l-1}, y_l] + b_o) \quad (10)$$

After generating an Ol value between 0 and 1 using the sigmoidal starting purpose, you should use the tanh initiation function to increase the recollection cell state Ct and multiply it with Ol, which is the layer's production. The historical highs and lows of a stock can be compared to its opening and closing prices. Investors are mainly interested about the final price. As a result, we will base all of our estimates on the final pricing. As demonstrated by our investigation, the results are independent of the input variable costs.

In the Bi-LSTM network, you may increase the amount of temporal characteristics and analyse data. According to current research, employing the CNN and Bi-LSTM algorithms together is more stable than using them alone. In this study, the CNN network and the Bi-LSTM network are coupled to generate the BiLSTM system technique, which takes full use of both systems' temporal and spatial feature appearance skills owing to their comparable meaning.

The data was first pre-processed, and the managed data was then utilised to train CNN and Bi-LSTM networks. Following that, characteristics removed by CNN and Bi-LSTM are rendered to comparable dimensional charting coatings, and their effects are associated by analogous concatenation and finally classified by SoftMax.

### 3.4. TLBO Based Hyper parameter Optimization

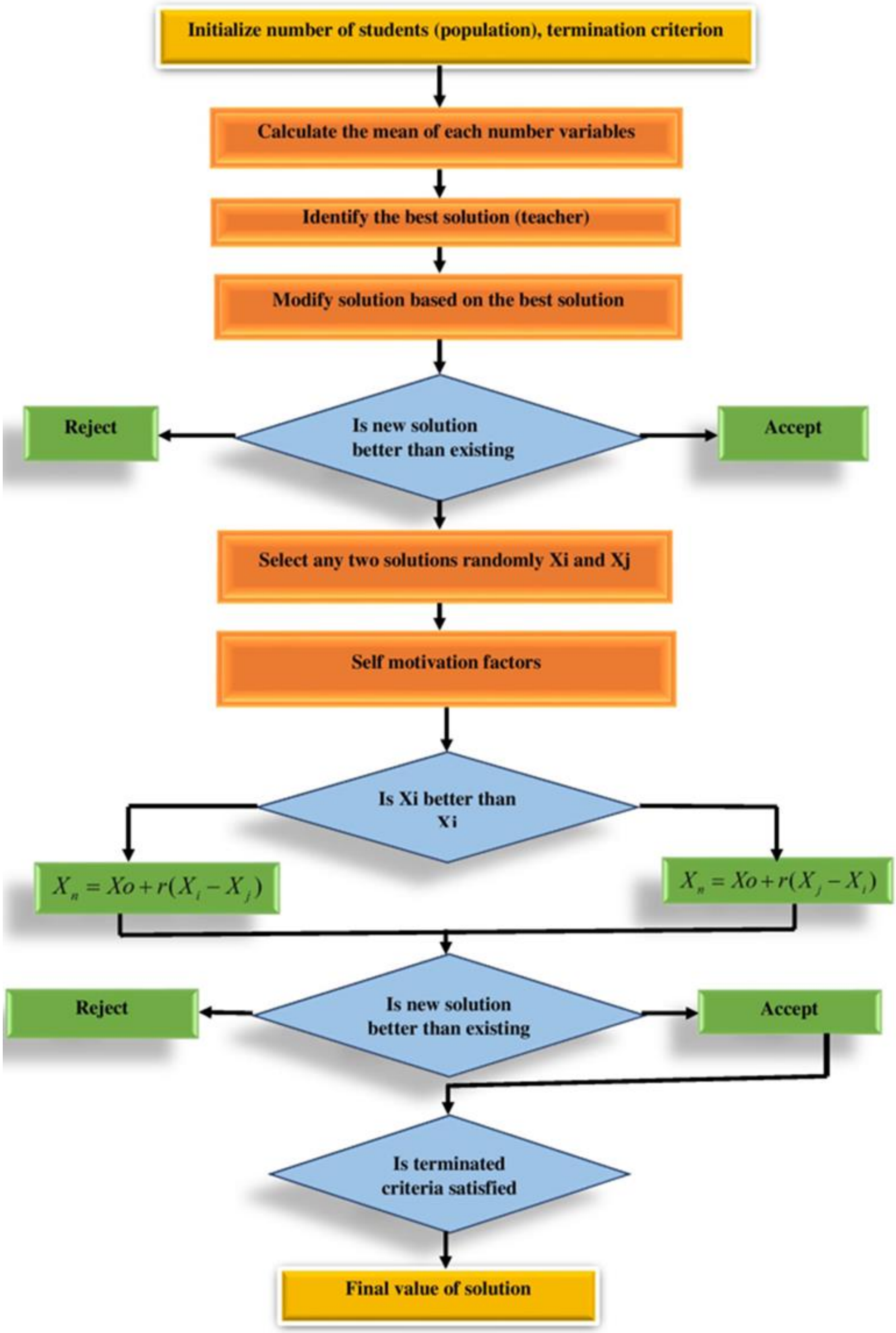


**Fig 3:** Flowchart of TLBO Algorithm

The TLBO method is used to fine-tune the hyperparameters of the HPT Bi-LSTM model. The classical education system includes both classroom-based learning and the TLBO algorithm. The core portion contrasted the best answer (teacher) and the sharing of talents among educators, as well as extra explanation from the educator chapter. Second, the learner's method for determining the optimal answer by integrating the knowledge of particular candidates with that of a randomly chosen one. These two components are critical in the TLBO approach. Both the teacher and learner phases have made sure that the level of difficulty has gone up and that there are many options.

Students are the first group to be considered when using TLBO's optimization approach. The instructor is defined by an optimum solution after the initial estimation of solutions. One answer that came from the teacher's stage was shared by the instructor and another person. The most recent solutions have been evaluated, and the finest ones have been chosen to replace the older ones. Following that, all of the solution warnings were improved by using the sharing option with additional randomly selected solutions. Following the student stage, a retention or replacement technique similar to that of the teacher was implemented to maintain or replace old learners with newly produced ones. The TLBO method is broken down into the following phases:

#### 3.4.1. Population Initialization

A key candidate explanation is identified as the lesson with nS, the student, during the TLBO procedure. This is how gratitude works. To achieve the academics who were assigned to create a random collection known as "S," the equation 11 as followed:

$$S = Lb + (Ub - Lb) \times rand(nS, nVar) \quad (11)$$

where nS is the number of students and Lb and Ub denote the project's lower and upper boundary routes, and Lb and Ub denote the project's lower and upper boundary routes, respectively.

#### 3.4.2. Teachers Level

The pupils were first intended, and the PFit (penalization objective function vector) that corresponded to them was produced. The teacher was then chosen from among the students who met the criteria for the optimum-penalization target purpose (T). An apprentice who is close to their tutor receives a step increase. The size of the step has been determined based on the tutor's dexterity as well as each student's normal capacity (AveS). The equation 12 is a phrase that best defines the teacher stage:

$$stepsize_i = T - TF_i \times Ave_s \quad (12)$$

$$newS = S + rand_{i,j} \times stepsize$$

$$i = 1, 2, ...nS \quad and \quad j = 1, 2, ....nVar$$

The tutor factor (TFi) has been preserved as a method of modifying instructor competency on the class average, which is also 1 or 2, and NewS is the vector of the new student; randi, j is an arbitrary integer between 0 and 1; and the tutor factor (NewS) has been preserved.

The value of TFi is chosen at random and is not included in the calculation. The TLBO project's schematic production of new solutions states that the most likely zone for new solutions is between two vectors: the present solution (S) and the randomised stage size. The examination of a wider area of the special solution may be aided by crossbreeding and additional augmentation techniques. It sounds just like QTLBO.

#### 3.4.3. Replacement Level

At this step, new students were evaluated and exchanged for their matching old ones using the simple greedy strategy. In this method, students are chosen to replace the current ones through an optimally penalised unbiased process. For nS students, there is now a new class.

#### 3.4.4. Learners Level

During the novice level, each student makes a purely random choice of someone who is not themselves (Srs). After that, the student expresses his ability to someone who was chosen at random. The scholar gets ahead of the other scholar when the other scholar has a higher level of skill than he does (PFit < iPFit_rs). An equation 13 is an explanation of the beginning stage:

$$stepsize_i = \begin{Bmatrix} S_j - S_{rs}; & PFit_i < PFit_{rs} \\ S_{rs} - S_j; & PFit_i \geq PFit_{rs} \end{Bmatrix} \quad (13)$$

$$newS = S + rand_{i,j} \times stepsize$$

$$i = 1, 2, ...nS \quad and \quad j = 1, 2, ....nVar$$

The fifth phase was a reiteration of the replacement process (replacement manner).

Sixth step (final condition): When the end condition is fulfilled, the operation finishes. If this is not the case, move on to step 2. Figure 2 shows an example of TLBO's flowchart.

## 4. Result and Evaluation

The data in this article was compiled using the Shanghai Stock Exchange Index's opening, closing, highest, and lowest prices from 4 January 2000 to 27 May 2021. The data were normalised for this study, and the price for the current trading day was taken from the closing price. Every 60 trading days, a candlestick chart of the Shanghai Composite Index was created, with each image being 72 pixels wide and long. The image only kept a small portion of the pixels from the candlestick chart, setting the pixel value of the blank area to 0, to reduce the impact of outside influences on the model and speed up model training. This paper converted each image into a grayscale image and performed model training to highlight the impact of stock price image morphology on stock price. 5184 samples in total were used, of which 80% were for testing and 20% were for validation. All eight qualities are present in every instance: opening price, maximum price, lowest price, closing price, volume, turnover, ups and downs, and variance. The initial price, the highest price, the lowest price, the final price, the capacity, and the change. In data analysis, the R2-squared number, the origin means square error (RMSE), and the mean absolute error (MAE) are all utilised (R2).

Figure 6 shows a comparison of the MAE of the HPT-BiLSTM approach with the MAE of other previously used methods. The MLP prototype produced unacceptable results in production, with a maximum MAE of 31.392. This supported the theory that the model had failed to function as expected. RNN, CNN, LSTM, BiLSTM, and BiLSTM-AM have all produced MAE values that are reasonably close. LSTM, RNN, CNN, and LSTM-AM have all produced MAE values. When compared to the general average, MAE values obtained through the application of the BiLSTM and BiLSTM-AM approaches are both somewhat lower than the norm, coming in at 33.67 and 30.43, respectively. The industrialized HPT-BiLSTM approach, on the other hand, has produced superior outcomes and has an MAE of 29.54 percentage points lower. According to the research that compared the RMSE of the HPT-BiLSTM method to those of other methods, the MLP approach, with an extreme RMSE value of 31.45, delivers the least exact findings. On the other hand, the MLP approach was found to be the least successful tactic in a research of the RMSE of the HPT-BiLSTM technique. Comparable RMSE values can be found between RNN, CNN, LSTM, BiLSTM, and BiLSTM-AM systems. This has happened as the organisations have become more similar. Furthermore, the RMSE values obtained by utilising the BiLSTM and BiLSTM-AM approaches were 35.76 and 32.47, respectively, indicating a slight decrease from the original values. Lastly, the HPT-BiLSTM strategy that was suggested has led to better results, and the RMSE value for this method is at least 31.26.

4.1. MAE Analysis

**Table 1:** MAE comparison of the HPT-BiLSTMmethod to existing approaches

| MAE analysis in % | | | | | | |
|---|---|---|---|---|---|---|
| No of Data/ Methods | RNN | CNN | LSTM | BiLSTM | BiLSTM-AM | HPT-BiLSTM |
| 100 | 29.47 | 35.46 | 33.52 | 31.82 | 29.11 | 26.35 |
| 200 | 32.37 | 34.82 | 31.67 | 30.56 | 28.37 | 27.51 |
| 300 | 33.45 | 35.98 | 34.94 | 29.36 | 26.45 | 25.34 |
| 400 | 31.64 | 36.48 | 33.45 | 32.48 | 31.67 | 28.76 |
| 500 | 28.45 | 35.89 | 34.23 | 33.67 | 30.43 | 29.54 |

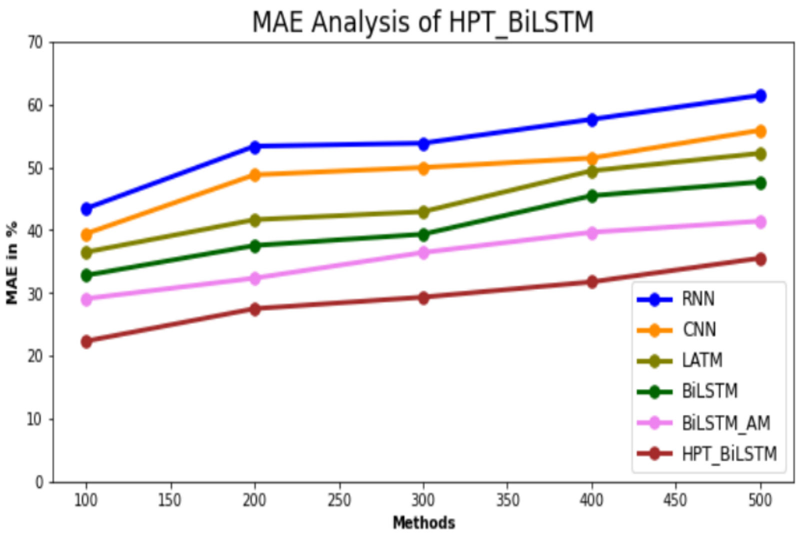


**Fig 4** MAE comparison of the HPT-BiLSTM approach with existing techniques

In Fig. 4, For a specific circumstance, the HPT-BiLSTM technique is compared to numerous existing techniques, and the MAE error rate should be reduced for improved performance. While the HPT-BILSTM technique has the lowest MAE error rate, the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM algorithms offer more balanced results. For example, the HPT-BILSTM technique with 100 data has an MAE error rate of 26.35%, but the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques have error rates of 29.47%, 33.52%, 31.82%, and 29.11%, respectively. Similarly, for 500 data points, the HPT-BILSTM technique surpasses the other techniques, with an error rate of 29.54%, compared to 28.45%, 35.89%, 34.23%, 33.67%, and 30.43% for the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques, respectively.

4.2. RMSE analysis

**Table. 2:** RMSE comparison of the HPT-BiLSTM method to existing techniques

| **RMSE analysis in %** | | | | | | |
|---|---|---|---|---|---|---|
| No of Data/ Methods | RNN | CNN | LSTM | BiLSTM | BiLSTM-AM | HPT-BiLSTM |
| 100 | 31.45 | 30.56 | 29.47 | 28.67 | 27.74 | 27.99 |
| 200 | 33.37 | 31.67 | 29.99 | 27.54 | 25.84 | 28.38 |
| 300 | 36.45 | 34.32 | 32.56 | 31.47 | 29.21 | 28.98 |
| 400 | 38.64 | 37.87 | 35.67 | 33.76 | 31.37 | 30.27 |
| 500 | 40.45 | 39.28 | 38.45 | 35.76 | 32.47 | 31.26 |

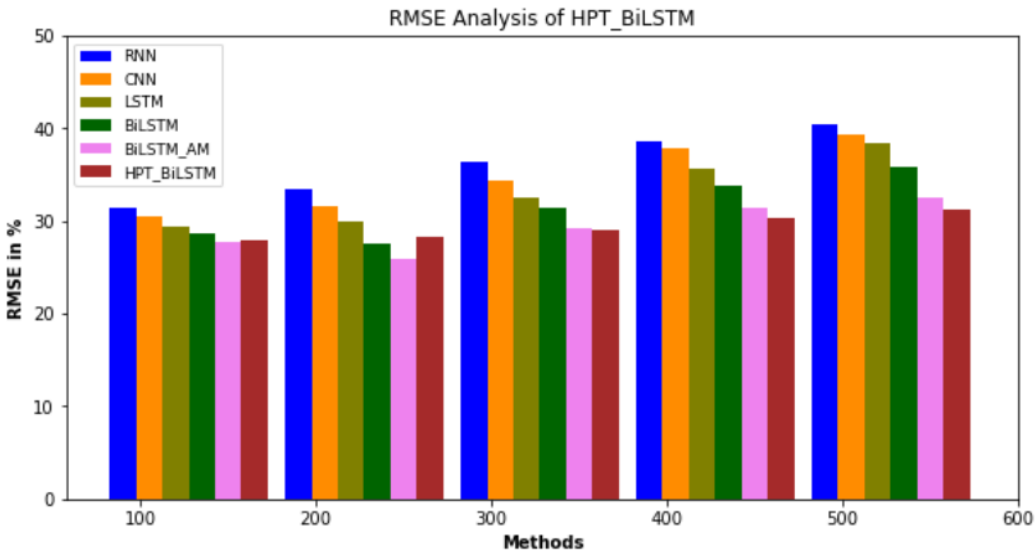


**Fig 5:** RMSE comparison of the HPT-BiLSTM method to existing techniques

In Fig. 5, For a specific circumstance, the HPT-BiLSTM technique is compared to numerous existing techniques, and the MAE error rate should be reduced for improved performance. While the HPT-BILSTM technique has the lowest RMSE error rate, the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM algorithms offer more balanced results. For example, the HPT-BILSTM technique with 100 data has an RMSE error rate of 27.99%, but the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques have error rates of 31.45%, 30.56%, 29.47%, 28.67%, and 27.74%, respectively. Similarly, for 500 data points, the HPT-BILSTM technique surpasses the other techniques, with an error rate of 31.26%, compared to 40.45%, 39.28%, 38.45%, 35.76%, and 32.47% for the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques, respectively.

4.3. R2 analysis

**Table 3:** R2 Analysis of HPT-BiLSTM method with existing techniques

| R^2 analysis in % | | | | | | |
|---|---|---|---|---|---|---|
| No. of data/Methods | RNN | CNN | LSTM | BiLSTM | BiLSTM-AM | HPT-BiLSTM |
| 100 | 33.47 | 35.84 | 36.38 | 39.38 | 41.27 | 45.43 |
| 200 | 32.04 | 34.23 | 36.87 | 39.65 | 41.59 | 45.86 |
| 300 | 32.76 | 35.27 | 36.98 | 38.89 | 40.28 | 44.34 |
| 400 | 31.26 | 36.25 | 37.27 | 38.54 | 40.86 | 43.76 |
| 500 | 31.86 | 36.95 | 38.07 | 38.95 | 39.56 | 44.37 |

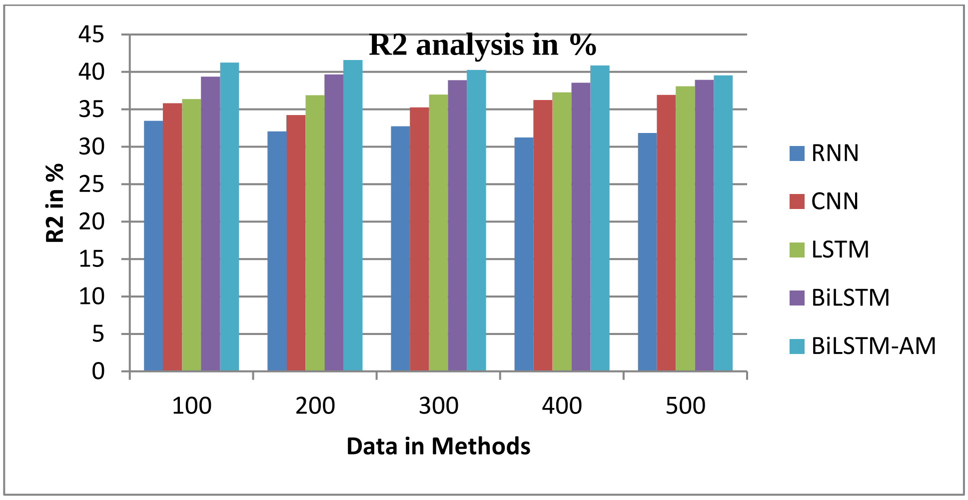


**Fig 6:** R2 comparison of the HPT-BiLSTM method to existing techniques

The HPT-BiLSTM technique is compared to numerous existing techniques in Fig. 6 for a specific case, and the R2 rate should be greater for improved performance. While the HPT-BILSTM technique has the highest R2 rate, the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM algorithms offer more balanced results. For example, the HPT-BILSTM technique with 100 data has an R2 rate of 45.43%, but the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques have error rates of 33.47%, 35.84%, 36.38%, 39.38%, and 41.27%, respectively. Similarly, for 500 data points, the HPT-BILSTM technique surpasses the other techniques, with an R2 value of 44.37%, compared to 31.86%, 36.95%, 38.07%, 38.95% and 39.56% for the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM techniques, respectively.

### 4.4 Prediction analysis

**Table 4:** HPT-BiLSTM Method Prediction Analysis Using Existing Techniques

| **Methods** | **Prediction** | **Actual** |
|---|---|---|
| RNN | 71.23 | 72.56 |
| CNN | 80.35 | 78.57 |
| LSTM | 83.78 | 81.36 |
| BiLSTM | 82.27 | 82.79 |
| BiLSTM-AM | 85.83 | 86.27 |
| HPT-BiLSTM | 88.34 | 89.03 |

Figure 7 compares the HPT-BiLSTM technique to several existing strategies for improved performance by comparing actual and expected value analysis. While the HPT-BILSTM technique has the highest value, the RNN, CNN, LSTM, Bi-LSTM, and Bi-LSTM-AM algorithms offer more balanced results. In contrast to the RNN, CNN, LSTM, and Bi-LSTM-AM techniques, which have predicted values of 71.23%, 80.35%, 83.78%, 82.27%, and 85.83%, respectively, and actual values of 72.56%, 78.57%, 81.36%, 82.79%, and 86.27%, respectively, the HPT-BILSTM technique with 100 data points has 89.03% of actual and 88.34% of predicted value.

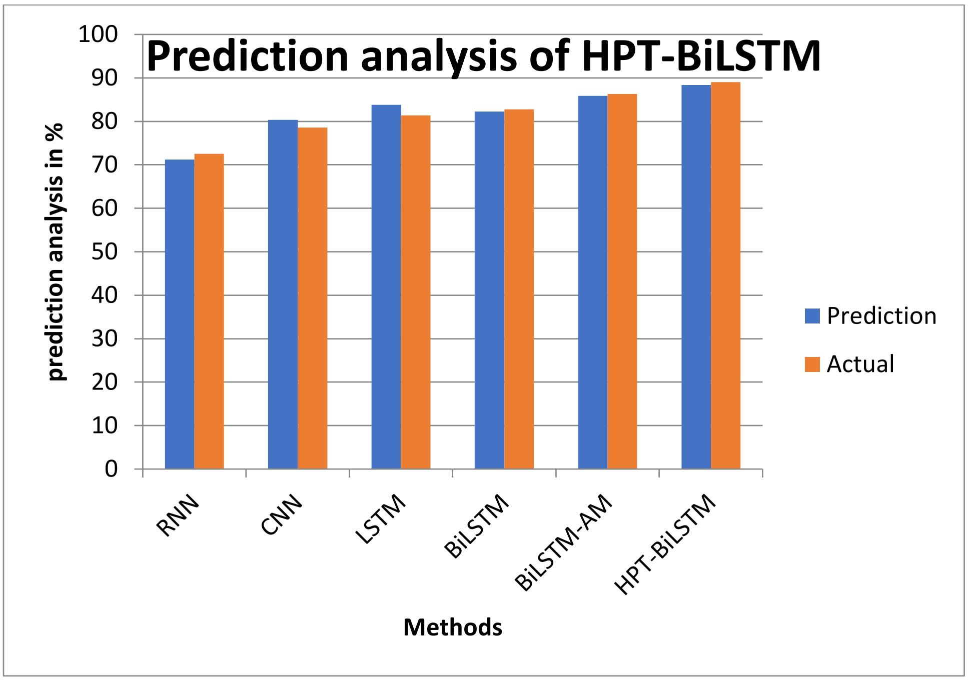


**Fig 7:** Analysis of HPT-BiLSTM Method Prediction Using Existing Techniques

## 5. Conclusions

Stock price forecasting is addressed, and an unique HPT-BiLSTM method is created. Preprocessing, HPT-BiLSTM-based prediction, and TLBO-based hyperparameter optimization make up the three main components of the HPT-BiLSTM technique's overall design. The TLBO method also successfully adjusts the HPT-BiLSTM model's hyperparameters to boost the accuracy of predictions.A thorough simulation study is conducted to evaluate the HPT-BiLSTM technique's enhanced performance, and the results are looked at from a variety of angles. As a consequence, the HPT-BiLSTM approach outperformed the other strategies over a wide range of performance metrics. During this research, an unique HPT-BiLSTM technique for reliably predicting stock values was developed. Pre-processing, a BiLSTM-based forecast, and hyperparameter tweaking comprise the HPT-BiLSTM concept. Because of TLBO model hyperparameter modifications, the BiLSTM model can deliver more accurate predictions. An extensive simulation study and an extensive analysis of the various data points are used in this investigation of the enhanced performance of the HPT-BiLSTM method. Given its superior performance across a variety of metrics, the HPT-BiLSTM strategy emerges as the clear winner in this comparison. In the future, feature selection techniques will be able to zero in on the crucial features for improved prediction accuracy. The RMSE was reduced by 31.851 percentage points using the CNN-BiLSTM-AM method and by 31.59 percentage points using the BiLSTM-AM strategy. The proposed HPT-BiLSTM method outperformed the competition with a lower root mean square error (RMSE) of 30,336 than the rest.Using feature selection algorithms, it will soon be feasible to make projections that are more accurate. There are still a few intriguing study concepts in the that might be investigated in the future.The HPT-BiLSTM, for example, has a higher chance of being able to do more tasks and get better results because it has a strong instrument. Finding a collection allocation in forecast-based trading that improves HPT-BILSTM presentation is an interesting problem.

## References


[1] Abe, M., & Nakagawa, K. (2020, May). Cross-sectional stock price prediction using deep learning for actual investment management. In Proceedings of the 2020 Asia Service Sciences and Software Engineering Conference (pp. 9-15).

[2] Ampomah, E. K., Qin, Z., & Nyame, G. (2020). Evaluation of tree-based ensemble machine learning models in predicting stock price direction of movement. Information, 11(6), 332.

[3] Cavalcante, R. C., Brasileiro, R. C., Souza, V. L., Nobrega, J. P., & Oliveira, A. L. (2016). Computational intelligence and financial markets: A survey and future directions. Expert Systems with Applications, 55, 194-211.

[4] Chen, W., Jiang, M., Zhang, W. G., & Chen, Z. (2021). A novel graph convolutional feature based convolutional neural network for stock trend prediction. Information Sciences, 556, 67-94.

[5] Ding, G., & Qin, L. (2020). Study on the prediction of stock price based on the associated network model of LSTM. International Journal of Machine Learning and Cybernetics, 11(6), 1307-1317.

[6] Hajizadeh, E., Seifi, A., Zarandi, M. F., & Turksen, I. B. (2012). A hybrid modeling approach for forecasting the volatility of S&P 500 index return. Expert Systems with Applications, 39(1), 431-436.

[7] Hardas, B. M., & Pokle, S. B. (2017). Optimization of peak to average power reduction in OFDM. Journal of Communications Technology and Electronics, 62(12), 1388-1395.

[8] Bamber, S. S. . (2023). Evaluating Performance of Beacon Enabled 802.15.4 Network with Different Bit Error Rate and Power Models. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 167–178. https://doi.org/10.17762/ijritcc.v11i2s.6040

[9] Hua, Y., Zhao, Z., Li, R., Chen, X., Liu, Z., & Zhang, H. (2019). Deep learning with long short-term memory for time series prediction. IEEE Communications Magazine, 57(6), 114-119.

[10] Lu, W., Li, J., Wang, J., & Qin, L. (2021). A CNN-BiLSTM-AM method for stock price prediction. Neural Computing and Applications, 33(10), 4741-4753.

[11] Mehtab, S., & Sen, J. (2020). A time series analysis-based stock price prediction using machine learning and deep learning models. arXiv preprint arXiv:2004.11697.

[12] Naik, N., & Mohan, B. R. (2019, May). Study of stock return predictions using recurrent neural networks with LSTM. In International conference on engineering applications of neural networks (pp. 453-459). Springer, Cham.

[13] Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. Artificial Intelligence Review, 53(4), 3007-3057.

[14] Pokle, S. B. (2019). Analysis of OFDM system using DCT-PTS-SLM based approach for multimedia applications. Cluster Computing, 22(2), 4561-4569.

[15] Ramalingam, C., & Mohan, P. (2021). An efficient applications cloud interoperability framework using I-Anfis. Symmetry, 13(2), 268.

[16] Rather, A. M., Agarwal, A., & Sastry, V. N. (2015). Recurrent neural network and a hybrid model for prediction of stock returns. Expert Systems with Applications, 42(6), 3234-3241.

[17] Krishna Shrestha (2018). Machine Learning for Depression Diagnosis using Twitter data. International Journal of Computer Engineering in Research Trends, 5(2)56-61.

[18] Sayeed, R. F., Princey, S., & Priyanka, S. (2015). Deployment of multicloud environment with avoidance of DDOS attack and secured data privacy. International Journal of Applied Engineering Research, 10(9), 8121-8124.

[19] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied soft computing, 90, 106181.

[20] Mondal , D. . (2021). Remote Sensing Based Classification with Feature Fusion Using Machine Learning Techniques. Research Journal of Computer Systems and Engineering, 2(1), 28:32. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/16

[21] Song, H., Peng, D., & Huang, X. (2020). Incorporating research reports and market sentiment for stock excess return prediction: a case of mainland china. Scientific Programming, 2020.

[22] Subbulakshmi, P., & Prakash, M. (2018). Mitigating eavesdropping by using fuzzy based MDPOP-Q learning approach and multilevel Stackelberg game theoretic approach in wireless CRN. Cognitive Systems Research, 52, 853-861.

[23] Tang, H., Dong, P., & Shi, Y. (2019). A new approach of integrating piecewise linear representation and weighted support vector machine for forecasting stock turning points. Applied Soft Computing, 78, 685-696Thangavel, R. (2013). Resource selection in grid environment based on trust evaluation using feedback and performance. American Journal of Applied Sciences, 10(8), 924.

[24] Addepalli Lavanya,Lokhande Gaurav,Sakinam Sindhuja, Hussain Seam, Mookerjee Joydeep , Vamsi Uppalapati, Waqas Ali and Vidya Sagar S.D . (2023). Assessing the Performance of Python Data Visualization Libraries: A Review. International Journal of Computer Engineering in Research Trends, 10(1), 29-39.

[25] Yu, P., & Yan, X. (2020). Stock price prediction based on deep neural networks. Neural Computing and Applications, 32(6), 1609-1628.

[26] Zhang, J., Li, L., & Chen, W. (2021). Predicting stock price using two-stage machine learning techniques. Computational Economics, 57(4), 1237-1261.