The Bioscan
AN INTERNATIONAL QUARTERLY JOURNAL OF LIFE SCIENCES
19(2): S.I (1), 154-165, 2024 www.thebioscan.com

LOCUST SWARM OPTIMIZED LOGISTIC DECISION TREE BASED TECHNICAL INDICATOR CLASSIFIER FOR STOCK MARKET PREDICTION

Mrs. N. Gowri Priya

Department of Computer Science, VET Institute of Arts and Science (Co-education), Thindal, Erode -638 012.

E- Mail ID: priyassmce@gmail.com

Dr. D. Karthika

Department of Computer Science, VET Institute of Arts and Science (Co-Education), Thindal, Erode -638 012.

E- Mail ID: karthikad@vetias.ac.in

DOI: https://doi.org/10.63001/tbs.2024.v19.i02.S.I(1).pp154-165

KEYWORDS
Stock Prediction, Machine Learning, Log First Difference transformation, Locust Swarm, BoostARoota, Logistic Decision Tree

Received on:

01-08-2024

Accepted on:

16-11-2024

ABSTRACT

As the economy has heightened swiftly in current years, more and more people have started investing their money into the stock market. Owing to this stock market prediction is considered as a pivotal venture and one that has proven to be more advantageous than others. Investors countenance notable issues making stock market-associated predictions as a consequence of dearth of movement and presence of noise. This study presents a detailed investigation of the selection of a minimal number of relevant technical indicators with the objective of increasing sensitivity, specificity, reducing training time and improving accuracy using Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier (LSBO-LDTC). The LSBO-LDTC method is split into three stages, namely, preprocessing, feature selection and classification for stock market prediction. In the first stage, data obtained from Stock Market Data - Nifty 100 Stocks (1 min) data are preprocessed to obtain cleaned ones by applying Log First Difference transformation-based Preprocessing. Then, in the second stage the preprocessed stock data is passed though the Locust Swarm BoostARoota Optimized feature selection for selecting relevant and significant technical indicators to obtain computationally efficient technical indicators for further classification. Finally in the third stage decision regarding suitable day regarding for buying or selling is performed using Logistic Decision Tree-based Classification for stock prediction. Experimental analysis is carried out on the parameters such as specificity, sensitivity, accuracy and training time with respect to number of stock data. Experimental results indicate a high performance of the proposed method for searching a global optimum stock market prediction to achieve high return on investment in comparison with other well-known machine learning methods.

INTRODUCTION

Stock market inclinations forecast is one of the most prevalent topics of interest and a notable research issue owing to its arbitrary and unbalanced nature. The stock data is typically non-stationary and features are found to be non-correlative to each other. Numerous Stock Technical Indicators (STIs) may analyze and precise stock market inclinations in an accurate manner. Technical indicators here refer to the measure taken to acquire financial time series data as input and accordingly predict price movement on the basis of statistical calculations. Technical analysts on the other hand with the aid of technical indicators predict stock price inclinations by throwing light on the results of numerous technical indicators.

A belief rule base (BRB) assessment model was introduced in [1] employing several technical indicators with the intent of forecasting stock price trend. The BRB-based model employed in this work consisted of three BRBs, namely BRB_1, BRB_2 and BRB_3. Here, BRB_1 was employed to acquire relationship between price trend of moving average (MA) and buy/sell decisions and on the other hand, BRB_2 was employed to study conditions of moving average convergence and divergence (MACD) and finally, BRB_3 was utilized to represent stochastic indicator (KD) states. By using these three indicators as base stock analysis was made to arrive at accurate stock price prediction. With this the mean square error was reduced considerably. Though minimization observed in terms of mean square error however the accuracy involved in forecasting stock price trend was not analyzed.

The technical analysis, machine learning technique and portfolio optimization model was introduced in [2] for portfolio selection and optimization. The technical analysis, K-means clustering algorithm and mean-variance portfolio optimization model was introduced to combine important analysis for portfolio investments. The average annual risk and annual rate of return data was employed for the years between 2018 and 2020 to form clusters. The designed analysis assessed the stocks correspond to investor technical strategy like Moving Average Convergence/Divergence (MACD) and Hybrid MACD with Arnaud Legoux Moving Average (ALMA), therefore minimizing the average annual risk. Despite throwing light on the results of technical indicators the sensitivity analysis was not made.

A Non-Dominated Sorting Genetic Algorithm II technique was introduced in [3] for evolving indicator parameters and for combining the indicators to generate trading strategy. The experiments were conducted with actual stocks from Stock Exchange of Thailand. The Non-Dominated Sorting Genetic Algorithm II technique generated trading strategies to address the real world security trading issues. However, the sensitivity level was not improved by Non-Dominated Sorting Genetic Algorithm II technique.

The artificial intelligence and machine learning method was introduced in [4] for forecasting the stock market with different data types. The designed method used evaluation metrics and different neural network structures to provide proposition research method. The key objective of the designed method helped the researchers to easily replicate the previous studies. But, the computational cost was not minimized by artificial intelligence and machine learning method.

Evaluated Linear Regression based Machine Learning (ELR-ML) technique was introduced in [5] to improve prediction performance. The selection of modest price of global financial data was acknowledged in any stock market prediction model based on the nonlinearities and discontinuities of factors anticipated to stock markets. ELR-ML technique was employed to forecast stock financial values of Standard and Poor's 500 index with Open, close, low, high, and volume factors. Though prediction performance was improved, the specificity level was not improved by ELR-ML technique.

Stock market (SM) indices prediction is an interesting task. A thorough analysis in this stock market indices prediction field can impart valuable information to traders, policy makers and investors in attractive SMs. A correlation feature selection model was presented in [6] with the intent of identifying significant technical indicators (TIs), that were integrated with numerous deep learning (DL) algorithms for accurate SM indices forecasting. Numerous conventional Stock Technical Indicators (STIs) may predict trends in stock market inappropriately. To analyze and validate the stock market features employing STIs and making effective decisions concerning trading, an Evolutionary Deep Learning Model was presented in [7] with the objective of identifying prices in stock trends' utilizing STIs, therefore resulting in the improvement of prediction accuracy.

Investors employ data on market activity, to name a few being, historical returns, open stock price, close stock price, low stock price, high stock price and volume of trades. In [8] a novel technique on History Bits for deriving beneficial facts from numerous established dataset for swift retrieval and circumvent data manipulation was presented. The proposed method in turn predicted the trading call out of five distinct cells in an efficient manner.

In [9], multi-source data influencing stock prices were integrated with swarm intelligence and deep learning to construct the stock price prediction model with improved prediction effect. In [10] yet another method employing stock market price prediction employing augmented text with improved precision was designed. A systematic review on decision fusion mechanisms for stock price prediction was investigated in [11]. Prior attempts were also made in trend prediction using textual information. In [12], Long Short-Term Memory Network (LSTM) and Bidirectional Encoder Representations from Transformers (BERT) were integrated with the objective of improving prediction performance in an extensive manner.

To sum the discussion, although several materials and methods were associated with stock market prediction with both technical and general indicators, these methods lacks in ensuring sensitivity, specificity and accuracy combine. It is also inferred that methods that involved optimization techniques struggles in achieving improved sensitivity and low training time in prediction stock market to making an apt decision regarding buy/sell or hold. This motivated to perform research work by proposing a Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier (LSBO-LDTC) for stock market prediction based on Locust Swarm BoostARoota Optimized feature selection and Logistic Decision Tree-based Classification. The study considers the publically available Stock Market Data - Nifty 100 Stocks (1 min) dataset to access the performance of proposed method in predicting stock market value. Initially, the input data are pre-processed by using Log First Difference transformation-based Preprocessing for better detection performance. Then, feature selection is performed by employing Locust Swarm BoostARoota Optimization model for selecting the most highly significant features (i.e., technical indicators) from the preprocessed stock data. Third, with the preprocessed stock data and highly influential technical indicators selected as input was subjected to Logistic Decision Tree-based Classification for stock prediction. Finally, by involving certain performance metrics namely sensitivity, specificity, accuracy and training time, the performance of the proposed LSBO-LDTC method is analyzed. In addition, to expore the effectiveness, the present method is equated with other existing methods.

1.1 Contributions of the work

The following is the summary of major contributions:

- To determine the best parameters (i.e., suitable day for buy and sell), we propose the Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier (LSBO-LDTC) for stock market prediction, a machine learning-based approach.
- The LSBO-LDTC method efficiently captures noise-reduced preprocessed stock market data by employing Log First Difference transformation-based Preprocessing.
- To design Locust Swarm BoostARoota Optimized feature selection model with the objective of selecting highly significant features (i.e., technical indicators) for stock market prediction.
- To propose Logistic Decision Tree-based Classification model for making wise decision regarding buy/sell or hold certain stocks in a precise and accurate manner.
- We evaluate the performance of the LSBO-LDTC method using numerous performance metrics, i.e., sensitivity, specificity, accuracy and training time and compared it with existing methods in the area of stock market prediction for making wise decision regarding buy/sell or hold.
- To determine the best parameters (i.e., suitable day for buy and sell), this paper thoroughly explains the LSBO-LDTC method and shows how it can be utilized to predict stock market concerning buying/selling and holding of stock in an efficient manner.

1.2 Organization of the work

The works is organized as follows: Section 2 imparts related works in the area of stock market prediction. Section 3 clarifies the methodology using figurative representations. Section 4 discusses the experimental setup and also elaborates a comparison with existing methods in Section 5. Finally, Section 6 includes concluding remarks.

2. Related works

Forecasting stock returns is considered as a wonderful research area and is found to be of highly significant for allocating and pricing of asset value and management of risk accordingly. However it is found to be really difficult. Researchers from wide range of areas and domains are performing research areas and topics on forecasting stock market wisely. Moreover several financial experts have attempted to address the issue, but have had only sporadic success.

Several new forms of technical indicators were analyzed in [13] for making wise and optimal stock prediction. Yet another identification of new technical indicators along with the two step economic constraint model was designed in [14] for making robust and accurate prediction. In [15] a holistic survey of recent practices followed in stock market prediction was analyzed and investigated.

However, the above said methods were found to be both laborious and cumbersome while dealing with non-stationary time series data. To address on this gap deep learning technique using stacked auto encoder was designed in [16] that by combining traditional pattern and that found in Internet provided accuracy prediction results. Yet another systematic review on stock market forecasting using deep learning and technical indicators were analyzed in [17]. An empirical analysis for both emerging equity market and technical trading rules were detailed in [18]. However only with the efficient feature or technical indicator selection would result in accurate prediction. To focus on this aspect, a survey of feature selection and extraction techniques was investigated in [19].

Research has proved that the existence of noise in data typically instigates additional errors and biases that specifically results in degradation in the precision of volatility forecasts. An additional outlier corrected mixed data sampling technique employing generalized autoregressive conditional model was employed in [20] to boosting stock volatility prediction. Yet another deep learning technique was proposed in [21] focusing on the sentimental and technical indicators involved in accurate stock prediction. A case scenario of stock prediction in Vietnam employing a plethora of machine learning algorithms was investigated in [22].

Trend technical indicator was employed in [23] based on deep learning for precise stock market related prediction. However the error factors involved in analysis was not focused. Generative adversarial networks were employed in [24] to concentrate on the mean absolute error issue. Chart patterns and attention mechanism were utilized in [25] for making accurate and precise market movement prediction. A summary of existing materials and methods used, its advantages, drawbacks and the datasets utilized are provided in Table 1.

Table 1 Existing methods summary

Reference	Work	Methodology	Advantages	Disadvantages	Dataset
[1]	forecasting stock price trend	belief rule base (BRB) assessment model	Mean square error	Forecasting accuracy	Shanghai Stock Index
[2]	Stock market optimization	Technical analysis, K-means algorithm, and mean-variance model (TAKMV)	Accuracy	Sensitivity	Philippine Stock Market data
[4]	Financial stock market forecast	linear regression based machine learning	Accuracy	Mean square error	Yahoo finance dataset
[6]	Stock price indices prediction	combining deep learning algorithms	Mean square error	Accuracy	Investpy library
[7]	Stock Prediction Based on Technical Indicators	Deep Learning Model	Prediction accuracy	Error	NSE - India
[8]	Prediction of stock market movement via technical analysis	History Bits based machine learning	Prediction accuracy	Error	Stock trading data
[14]	Forecasting stock market returns	two-step economic constraint method	Accuracy	Sensitivity and specificity	NSE - India
[20]	Enhancing stock volatility prediction	AO-GARCH-MIDAS model	Predictive accuracy	Mean square error	Shanghai Stock Exchange Composite Index

3. Methodology

In this section a method called, Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier (LSBO-LDTC) is presented for generating strategies with stock data to achieve high return on investment. Initially, different numbers of stock data is considered as an input from Stock Market Data - Nifty 100 Stocks (1 min) data obtained from https://www.kaggle.com/datasets/debashis74017/stock-market-data-nifty-50-stocks-1-min-data?resource=download. Next, preprocessing of the raw data are performed employing Log First Difference transformation-based Preprocessing model. With the obtained preprocessed stock data as output are subjected to the Locust Swarm BoostARoota Optimized feature selection for selecting relevant and significant technical indicators. Finally with the preprocessed stock data and significant technical indicators as features selected as output Logistic Decision Tree-based Classification model is applied for stock market prediction. The LSBO-LDTC method in this manner creates a trading strategy to achieve high return on investment.

3.1 Dataset description

The dataset employed in our work is Stock Market Data - Nifty 100 Stocks (1 min) data obtained from https://www.kaggle.com/datasets/debashis74017/stock-market-data-nifty-50-stocks-1-min-data?resource=download. Data uploaded in this dataset are collected from the internet. In this dataset, along with Nifty 50 stocks data, two indices data in addition to 55 technical indicators employed by Market experts are measured and made accessible. It also contains historical daily prices for Nifty 100 stocks and indices currently trading on Indian Stock Market. In addition data samples are of 5-minute intervals and the accessibility of data is from Jan 2015 to Feb 2022. Together with OHLCV (Open, High, Low, Close, and Volume) data, 55 technical indicators were created. The details pertaining to these technical indicators are given in the data description file. The whole dataset is around 33 GB and included with 100 stocks (Nifty 100 stocks) and 2 indices (Nifty 50 and Nifty Bank indices). Table 2 given below lists the feature (i.e., general features and technical indicators) present in the dataset.

Table 2 Detail about columns and technical indicators used

S. No	**Feature name**	**Description**	**S. No**	**Feature name**	**Description**
1	Date	Date of observation	31	CCI10	Commodity channel index - last 10 time frame data
2	Open	Open price of index - specific day	32	CCI15	Commodity channel index - last 15 time frame data
3	High	High price of index - specific day	33	Macd510	Moving Average Convergence/Divergence, fast period = 5 and slow period = 10

4	Low	Low price of index - specific day	34	Macd520	Moving Average Convergence/Divergence, fast period = 5 and slow period = 10
5	Close	Close price of index - specific day	35	Macd1020	Moving Average Convergence/Divergence, fast period = 10 and slow period = 20
6	Sma5	Simple moving average - 5 close price	36	Macd1520	Moving Average Convergence/Divergence, fast period = 15 and slow period = 20
7	Sma10	Simple moving average - 10 close price	37	Macd1226	Moving Average Convergence/Divergence, fast period = 12 and slow period = 26
8	Sma15	Simple moving average - 15 close price	38	Mom10	Momentum indicator of 10 close price
9	Sma20	Simple moving average for 20 close price	39	Mom15	Momentum indicator of 15 close price
10	Ema5	Exponential moving average for 5 close price	40	Mom20	Momentum indicator of 20 close price
11	Ema10	Exponential moving average for 10 close price	41	ROC5	Rate of change -: ((price/prevPrice)-1)*100 using 5 close price
12	Ema15	Exponential moving average for 15 close price	42	ROC10	Rate of change : ((price/prevPrice)-1)*100 using 10 close price
13	Ema20	Exponential moving average for 20 close price	43	ROC20	Rate of change : ((price/prevPrice)-1)*100 using 20 close price
14	Upperband	Upper band of Bollinger band	44	PPO	Percentage price oscillator
15	Middleband	Middle band of Bollinger band	45	RSI14	Relative Strength Index calculated - 14 close price
16	Lowerband	Lower band of Bollinger band	46	RSI8	Relative Strength Index calculated - 8 close price
17	HT_TRENDLINE	Hilbert Transform – Instantaneous Trendline	47	Slowk	Stochastic indicator k value
18	KAMA10	Kaufman Adaptive Moving Average - 10 close price	48	Slowd	Stochastic indicator d value
19	KAMA20	Kaufman Adaptive Moving Average - 20 close price	49	Fastk	Stochastic fast indicator k value
20	KAMA30	Kaufman Adaptive Moving Average of 30 price	50	Fastd	Stochastic fast indicator d value
21	SAR	Parabolic SAR	51	Fastksr	Stochastic Relative Strength Index k value
22	TRIMA5	Triangular moving average of 5 close price	52	Fastdsr	Stochastic Relative Strength Index d value
23	TRIMA10	Triangular moving average of 10 close price	53	ULTOSC	Ultimate oscillator
24	TRIMA20	Triangular moving average of 20 close price	54	WILLR	Williams' %R
25	ADX5	Average directional movement index - 5 close price	55	ATR	Average true range
26	ADX10	Average directional movement index - 10 close price	56	TRANGE	True range
27	ADX20	Average directional movement index - 20 close price	57	TYPPRICE	Typical price
28	APO	Absolute price oscillator	58	HT_DCPERIOD	Hilbert Transform - Dominant Cycle Period
29	CCI5	Commodity channel index - last 5 time frame data	59	BETA	Beta
30	Vol	Volume	60	MFI	Money Flow Index

3.2 Log First Difference transformation-based Preprocessing

Due to the intricacy of stock market variation or oscillation, the stock price is frequently in its entirety of random noise that would result in huge price volatility and hence causes overfitting issues. Here, the random noise has to be discarded by retaining the data trend. To be more specific, noise reduction of time series data is to discard numerous small fluctuations in the original data via function transformation that in turn assists in smoothening the curve of original data without altering the overall fluctuation trend. As a commonly used noise reduction method, logarithmic transformation better deals with time series data and preserve the features of original data as much as possible. As far as financial scenarios are concerned logarithmic transformation is extensively utilized in prediction tasks. Therefore, we choose Log First Difference transformation as the noise reduction model for stock price prediction.

The basic principle of Log First Difference transformation is to obtain logarithmic value of samples that contain important information and noise after transforming the original sample data. The resulting series of important information is larger and the coefficient of noise is smaller. The threshold is selected automatically. The resulting series greater than the threshold are considered to contain important information and hence are said to be retained whereas the resulting series less than the threshold are considered as noise and will be discarded. Figure 1 shows the structure of Log First Difference transformation-based Preprocessing model.

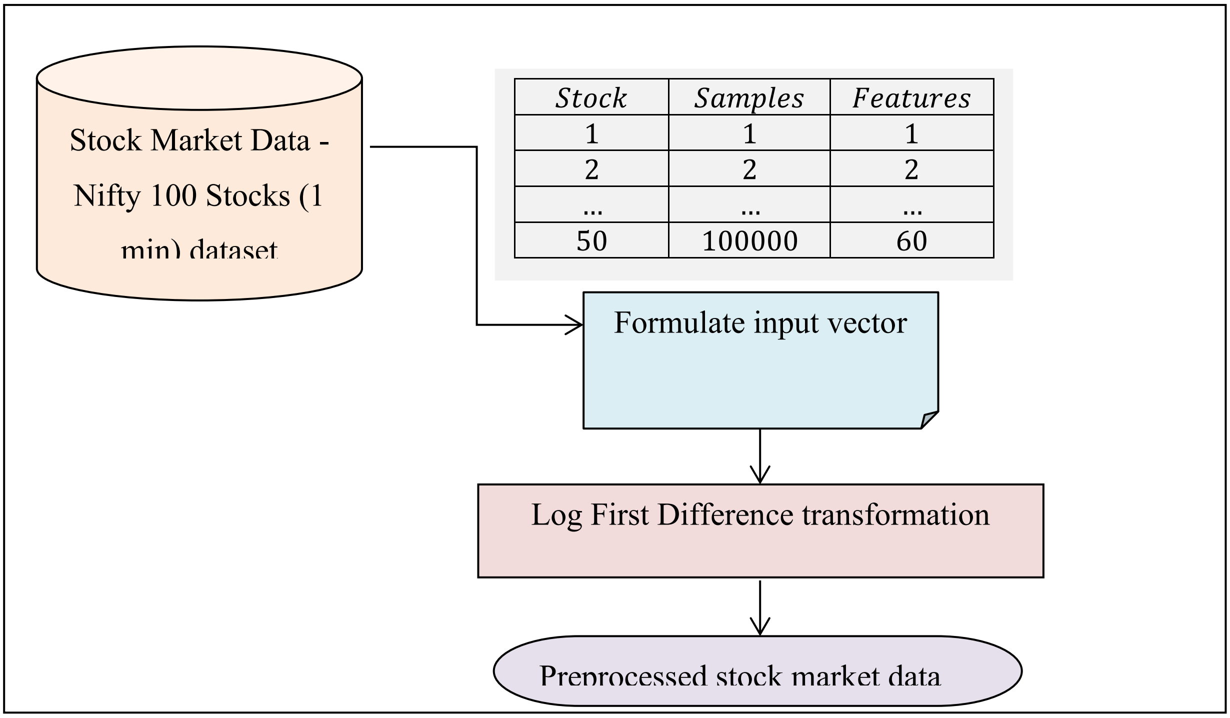

Figure 1 Structure of Log First Difference transformation-based Preprocessing

As mentioned earlier, Nifty 50 stocks data (to name a few being, ADANI PORTS & SEZ, ASIAN PAINTS, AXIS BANK, BAJAJ FINANCE, ,…ULTRATECH CEMENT, UPL,_WIPRO) and two indices data (i.e., NIFTY 50 and NIFTY BANK) , along with 55 technical indicators used by market experts are employed and provided as input vector. Then, with Stock Market Data - Nifty 100 Stocks dataset 'DS' as input, inclusive of 50 different stocks '$St_k, k = 50$', features '$F_j, j = 60$' (consisting of 3 general indicators and 55 technical indicators) and lacs of testing samples '$S_i = 100000$' is mathematically stated as given below.

$$IV = St_1 S_1 F_1, \dots St_1 S_i F_j, \dots, St_k S_1 F_1, \dots., St_k S_i F_j \quad (1)$$

From the above equation (1), the input vector 'IV' for stock prediction is performed with respect to 50 different stocks possessing 100000samples along with 60 features is first mapped and stored in the form of matrix. In our work the closing index prices are scaled with non linear scaling functions. Then, a Log First Difference transformation is utilized to preprocess the scaled down series of closing index price. Log First Difference transformation obtains the logarithmic value of series of data and then obtains the difference value to obtain the preprocessed noise discarded result. This is mathematically represented as given below.

$$PD = IV'_{u,v} = \ln(IV_{u,v}) - \ln(IV_{u,(v-1)}) \quad (2)$$

From the above equation (2), 'IV_u' and '$IV_{u,v}$' represents the data series of all closing index price and '$v-th$' day's closing index price respectively. In addition '$IV_{u,(v-1)}$' denotes the preceding day's (i.e., '$IV_{u,v}$') closing index price. The pseudo code representation of Log First Difference transformation-based Preprocessing is given below.

Input: Dataset 'DS', Samples '$S = \{S_1, S_2, \dots, S_m\}$', Stocks '$St = \{St_1, St_2, \dots, St_k\}$', Features '$F = \{F_1, F_2, \dots, F_n\}$'
Output: noise-reduced preprocessed stock data
1: **Initialize** '$m = 1{,}00{,}000$', '$k = 50$', '$n = 60$' 2: **Begin** 3: **For** each Dataset 'DS' with Samples 'S', Stocks 'St' and Features 'F' 4: Formulate input vector matrix as provided in equation (1) 5: Apply Log First Difference transformation to the input vector matrix as provided in equation (2) 6: **Return** noise-discarded preprocessed stock market data 'PD' 7: **End for** 8: **End**

Algorithm 1 Log First Difference transformation-based Preprocessing

As given in the above the execution and the trustworthiness of a neural network model considerably depend on the data quality being utilized. With neural networks playing the role of recognizing pattern, the data provided to a greater part has a positive influence on the accuracy. As a result with the data obtained from the Stock Market Data - Nifty 100 Stocks dataset is subjected to formulate input vector matrix. Followed by which Log First Difference transformations is applied to the input vector matrix to facilitate de-trending of data and spotlight indispensable association with regard to ease appropriate network learning process.

3.3 Locust Swarm BoostARoota Optimized feature selection

As far as stock market price analysis is concerned, changes in different company stock prices are controlled by several indicators, to name a few being, historical stock market data, general factors, technical indicators. The diversity of features confers a great challenge in bringing about higher stock prediction accuracy (i.e., to determine the best parameters suitable day for buy and sell). Due to this a feature selection process should be carried out to select key features from the preprocessed samples prior to the application of machine learning model to predict outcomes.

The feature selection process also assists in minimizing insignificant variables, computational cost, overfitting issue and enhances the machine learning model performance. If only small number of features is selected as input for machine learning model, predictions cannot be done and on the other hand if large number of features is selected it results in the increased running time. Hence only the most significant features influencing the results have to be selected to accomplish successful predictions.

In this work a hybrid feature selection algorithm called Locust Swarm BoostARoota Optimized feature selection is proposed that integrates Boruta utilizing XGBoost as the base model to the Locust Swarm optimization upon comparison to conventional Random Forest. The advantages of using XGBoost are that the removal of insignificant features is performed faster. Here, Boruta Feature Selection (BFS) and XGBoost to the Locust Swarm optimization are utilized in selecting the important technical indicator via two operators, solitary (i.e., exploration) and social (i.e., exploitation). The reason for selecting BFS is that it creates arbitrary shadows of the input feature.

The arbitrary mechanism enhances the model performance. Combining two feature selection models, Boruta Feature Selection (BFS) and XGBoost imparts comparatively better results upon comparison to single feature selection. Finally, by applying this hybrid model not only sufficiently explore the search space (i.e., exploration inclusive of both general and technical parameters) but also existent solutions are redefined within a defined neighborhood (i.e., exploitation). Due to this in this work Hybrid Feature Selection model is considered in the designing of technical indicators.

Initially the study considered 55 technical indicators along with 5 general indicators as listed in table 1. First, it randomizes the preprocessed sample data (i.e., locust population) by shuffling all features referred to as shadow features. Then, it trains an XGBoost on the preprocessed sample data and applies feature significance metric to measure the value of each feature. Figure 2 shows the structure of Locust Swarm BoostARoota Optimized feature selection model.

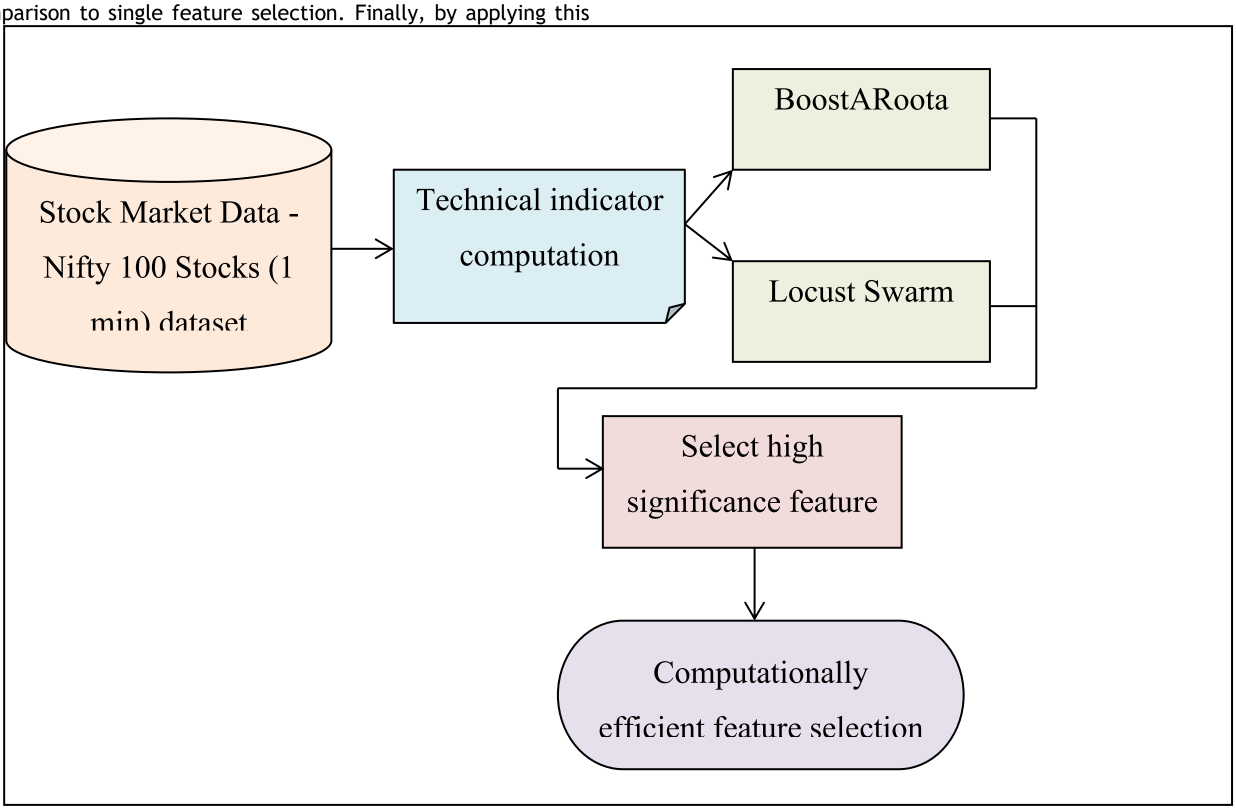

Figure 2 Locust Swarm BoostARoota Optimized feature selection model

As illustrated in the above figure, for each company stock, technical indicators are selected for every trading day '$Date$' from raw time series data that include open 'O_t', close 'C_t', high 'H_t' and low 'L_t' stock prices and trading volume 'Vol'. Hence, each data point correlates with an explicit trading day and consists of fifty input values, each equal to a certain technical indicator. The first step in the proposed model remains in creating duplicate copies of technical indicators feature.

$$DC\left(F_j\right) \rightarrow \left(Copy(F_i)\right) \quad (3)$$

From the above equation duplicate copies 'DC' of features 'F_j' are obtained by retrieving the copy '$Copy$' of the actual features 'F_i' in use. With the above obtained duplicate copies or shadow features are then subjected to random shuffle process. Here dual random shuffle process is performed to discard the correlation between independent (i.e., all features inclusive of technical and general indicators) and target features (i.e., significant technical indicators along with general indicators).

Under this Locust Swarm function, near individuals (i.e., preprocessed sample data) tend to repel with each other and on the other hand, distant individuals tend to attract with each other, maintaining the cohesion (i.e., selecting significant general and technical indicators). Different to the conventional model, in

$$DRS\left(F_j\right) \rightarrow RND.Shuffle\left(DC\left(F_j\left(RND.Shuffle\left(DC\left(F_j\right)\right)\right)\right)\right) \quad (4)$$

From the above equation (4) the results of dual random shuffle function '$DRS(F_j)$' is arrived at based on the random shuffling of corresponding generated duplicate copies of the features respectively.

Next, XGBoost algorithm is applied to identify significant technical indicator feature on the basis of higher mean values. Initially to start with the gradients and hessian values over a period of time in the past with the initialized training set 'p_i, q_i', where 'p_i' refers to the features (i.e., general and technical indicators) and 'q_i' denotes the significant selected features as output.

The Locust Swarm function applied here considers two distinct behaviors, solitary and social. Subject to the behavior, each individual (i.e., preprocessed sample data) is conducted by a set of evolutionary operators (i.e., solitary and social) that mimic different cooperative behaviors with the intent of ensuring a tradeoff between exploration and exploitation.

the proposed solitary operator (i.e., exploration), social forces are also magnified depending on the best fitness value via Gradient and Hessian processes as given below.

$$Grad_l(p_i) = \left[\frac{\partial LF(q_i, f(p_i))}{\partial f(p_i)}\right], p_i \in F \quad (5)$$

$$Hess_l(p_i) = \left[\frac{\partial^2 LF(q_i, f(p_i))}{\partial f(p_i)^2}\right]^2, p_i \in F \quad (6)$$

According to the exploration results obtained from the above obtained results based on gradient '$Grad_l(p_i)$' and hessian '$Hess_l(p_i)$' values from equations (5) and (6) a weak learner is obtained via the proposed social operator (i.e., exploitation).Exploitation here refers to the process of fine tuning existent individuals within a small neighborhood to improve their solution quality. This is performed using the training set '$\left\{p_i, \frac{Grad_l(p_i)}{Hess_l(p_i)}\right\}$' via the optimization function as given below.

$$\beta_l = argmin \sum_{i=1}^{n} Hess_l(p_i)\left[\beta(p_i) - \frac{Grad_l(p_i)}{Hess_l(p_i)}\right]^2 \quad (7)$$

$$f_l(p) = \alpha \beta_l(p) \quad (8)$$

From the above equations (7) and (8) weak learner results are fined tuned via optimization function in such a manner so as to obtain higher mean values. Followed by which Boruta Z Score is evaluated employing the mean and standard deviation as given below.

$$Boruta\ Z - score = \frac{f_l(p) - \mu}{\sigma} \quad (9)$$

From the above equation results (9) the maximum Boruta Z score on duplicate technical indicator feature are evaluated. Then the technical indicator feature are removed if '$Boruta\ Z - score$' is less than technical indicator feature and the above steps are repeated till the completion of iteration. The pseudo code representation of Locust Swarm BoostARoota Optimized feature selection is given below

.

Input: Dataset 'DS', Stocks '$St = \{St_1, St_2, \ldots, St_k\}$', Features '$F = \{F_1, F_2, \ldots, F_n\}$'
Output: Computationally-efficient feature selection
1: **Initialize** '$m = 1{,}00{,}000$', '$k = 50$', '$n = 60$', preprocessed stock market data 'PD' 2: **Initialize** general indicators '$Date, H_t, L_t, O_t, C_t, Vol$' 3: **Initialize** training set '(p_i, q_i)', learning rate '$\alpha = 0.01$', weak learners '$l = 1\ to\ L$', loss function '$LF(q, f(p))$' 4: **Begin** 5: **For** each Dataset 'DS' with Samples 'S', Stocks 'St', Features 'F' and preprocessed stock market data 'PD' **//Initialize population** **//Duplicate copy creation** 6: Formulate duplicate copies as generated in equation (3) **//Random shuffling process** 7: Formulate dual random shuffle as generated in equation (4) **//XGBoost** **//Solitary operation (i.e., exploration)** 8: Measure gradients and hessians for each stock values as given in equations (5) and (6) **//Social operation (i.e., exploitation)** 9: Obtain weak learner results using training set '$\left\{p_i, \frac{Grad_l(p_i)}{Hess_l(p_i)}\right\}$' via optimization function as given in equations (7) and (8) 10: Evaluate Boruta Z Score as given in equation (9) 11: **If** '$Boruta\ Z - score\ (PD) < PD$' 12: **Then** discard features 13: **End if** 14: **If** '$Boruta\ Z - score\ (PD) > PD$' 15: **Then** selected features 16: **Return** feature selected 'FS' 17: **End if** 18: **End for** 19: **End**

Algorithm 2 Locust Swarm BoostARoota Optimized feature selection

As given in the above algorithm with the objective of determining the best significant technical parameters (i.e., suitable day for buy stock and sell stock) or indicators along with the general parameters, a hybrid feature selection model called, Locust Swarm and BoostARoota is applied. The initial population creation for Locust Swarm is performed by utilizing the Boruta Feature Selection that first create duplicate copy, followed by which dual random shuffling is generated for further processing. The Locust Swarm BoostARoota Optimized feature selection is inspired by the behavior of the locust swarms through considering solitary behavior (i.e., exploration) and social behavior (i.e., exploitation). Here, instead of conventional Locust Swarm, both solitary behavior (i.e., exploration) and social behavior (i.e., exploitation) are performed using the XGBoost of the gradients and hessians for each stock values. The two behaviors interact with each other to allow find solution for optimization problem. Finally, stock data technical indicators with binary outcomes are performed employing high significance feature, therefore selecting features in a computationally efficient manner. The five popular technical indicators (37 features) along with general indicators (i.e., 5 features) selected in our work using the proposed model are listed below in table 3.

Table 3 Technical indicators selected

S. No	Technical indicators (i.e., features selected)	Features with corresponding technical and general indicators
1	Moving average indicator	SMA – 4 features
2	Volatility indicator	Bollinger band – 3 features
3	Volume indicator	Vol – 1 feature
4	Momentum indicator	ULTOSC – 1 feature
		William % R - 1 feature
		MFI – 1 feature
5	Trend indicator	MACD – 5 features
		EMA – 4 features
		ADX – 5 features
		RSI – 12 features
6	General indicators	Date of stock purchase – 1 feature
		High day high stock price at period 't' – 1 feature
		Low day low stock price at period 't' – 1 feature

		Open day open stock price at period 't' - 1 feature
		Close day close stock price at period 't' - 1 feature

Moving Average Indicator or simple moving average (SMA) evaluates the average price of an asset, specifically employing closing prices 'C_t', during a specified period of days (five day or 10 day or fifteen day and so on. Volatility is a statistical measure of dispersion of returns for a given market index. It is usually evaluated from either standard deviation or variance between those returns. In other words volatility simply refers to the amount of risk associated to the size of changes in stock's value. Momentum on the other hand refers to the speed of price changes in stock. Finally the trend indicator multiplies the market volume of stock by percentage change in its price.

3.4 Logistic Decision Tree-based Classification for stock prediction

Finally with the obtained preprocessed stock market data and significant technical indicators selected as feature, Logistic Decision Tree is utilized to combine the general and technical indicators for creating the best trading strategy. The Logistic Decision Tree-based Classification for stock prediction consists of a traditional decision tree format with logistic regression functions at the leaves. Upon comparison to ordinary decision tree where a test on one attributes is associated with every inner node, i.e., the node possessing two child nodes compares the attribute value to a threshold. In our work Logistic Decision Tree-based Classification for stock prediction is used. Figure 3 shows the structure of Logistic Decision Tree-based Classification model.

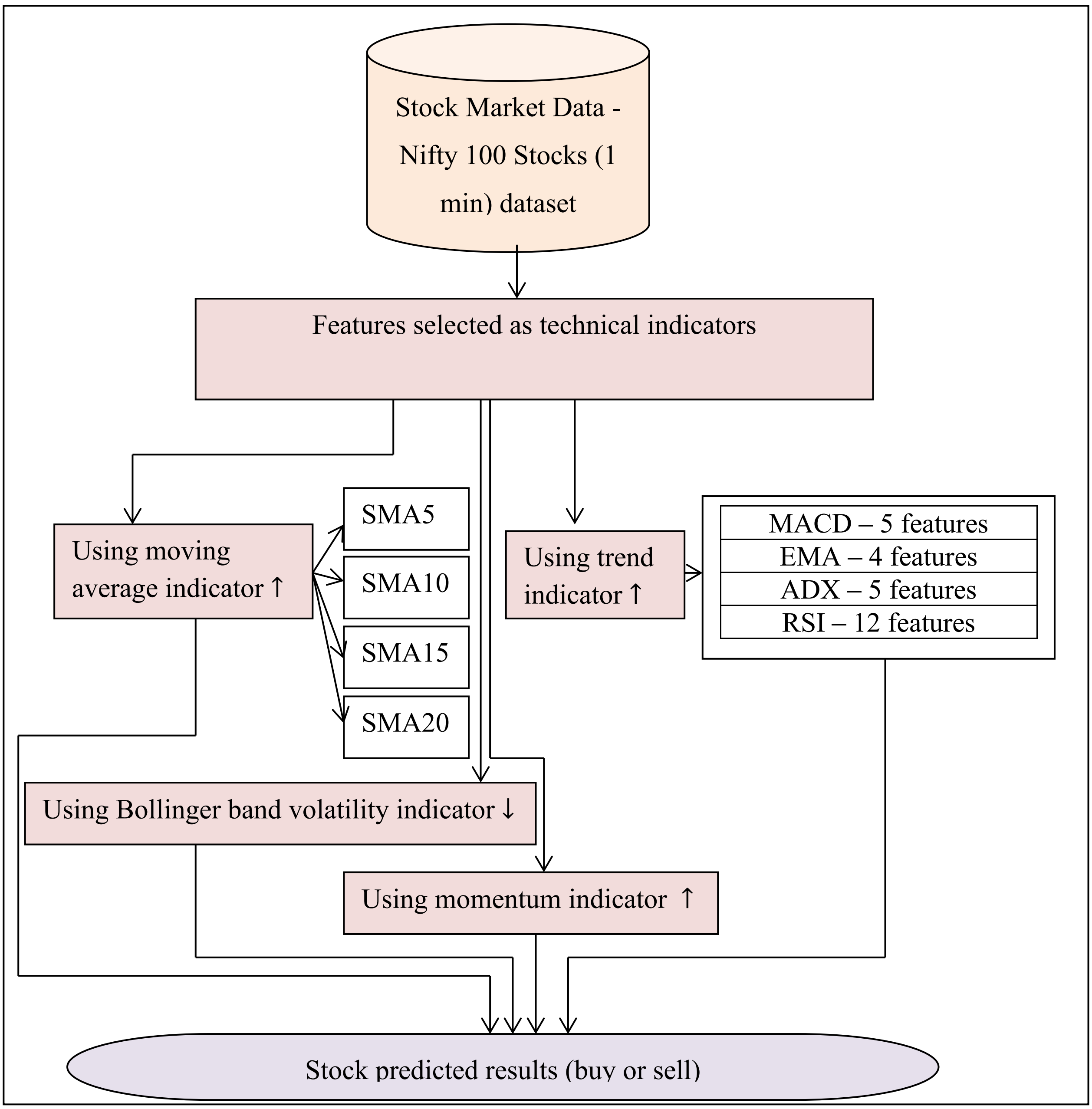

Figure 3 Structure of Logistic Decision Tree-based Classification

As illustrated in the above figure, the Logistic Decision Tree-based Classification employed in our work comprises of tree format made up of a set of non terminal nodes 'NTN' and set of terminal nodes 'TN'. Moreover, let 'IS' represent the instance space traversed by all significant features select that are present in the data. Then the tree format imparts a disjoint subdivision of 'IS' into regions 'IS_l' and every region is denoted by a leaf in the tree and is mathematically formulated as given below.

$$IS = \cup_{l \in L} IS_l \quad (10)$$

Upon comparison to the conventional decision trees, the leaves '$l \in L$' has a correlated logistic regression function 'f_l' alternatively a class label. Here the regression function 'f_l' takes into consideration a subset '$FS_l \subseteq FS$' of all selected features present and models class membership probabilities as given below.

$$Prob(G = j|S = s) = \frac{e^{F_j(s)}}{\sum_{k=1}^{j} e^{F_k(s)}} \quad (11)$$

$$F_j(s) = \gamma_0^j + \sum_{fs \in FS_l} \gamma_{fs}^j . fs \quad (12)$$

With the aid of above two equations (11) and (12) Logistic Decision Tree-based Classification model by whole logistic model tree is then mathematically formulated as given below.

$$f(s) = f_l(s).IV(s \in IS_l) \quad (13)$$

From the above equation (13) results, the best trading strategy (i.e. buy or sell) is made in an accurate and precision manner. The pseudo code representation of Logistic Decision Tree-based Classification for stock prediction is given below.

Input: Dataset 'DS', Stocks '$St = \{St_1, St_2, \dots, St_k\}$'
Output: accurate stock market prediction

Initialize preprocessed stock market data 'PD', '$k = 50$',
Initialize general indicators '$Date, H_t, L_t, O_t, C_t, Vol$'
Initialize feature selected 'FS'
 	
Begin
For each Dataset 'DS', preprocessed stock market data 'PD' and feature selected 'FS'
Impart a disjoint subdivision of 'IS' into regions 'IS_l' as given in equation (10)
Evaluate membership probability of Logistic Decision Tree as given in equations (11) and (12)
Formulate whole logistic model tree as given in equation (13)
 	
If '$s \in IS_l$'
Then '$IV(s \in IS_l) = 1$' buy/sell stock
End if
If '$s \notin IS_l$'
Then '$IV(s \in IS_l) = 0$' hold stock
End if
 	
End for
End

Algorithm 3 Logistic Decision Tree-based Classification for stock prediction

As given in the above algorithm with the objective of improving both the sensitivity and specificity rate of stock prediction, Logistic Decision Tree-based Classification model is designed. First, with the preprocessed stock data and highly significant features selected as input are subjected to the classifier for accurate and precise stock prediction. Upon comparison to the conventional decision tree model that lacks the complex calculations and their ability to overfit the preprocessed data, logistic regression and decision tree are combined to form Logistic model. Here by employing the instance space possessing all the highly significant features selected acquired as input are employed instead of attributes associated with every inner node an in depth classification with respect to five different technical indicators (average moving average in upward trend, average volatility in lower trend, average momentum in upward trend and average trend in upward trend). With the satisfaction of above five different technical indicators hypothesis stock prediction is buy/sell or stock prediction is hold. According to the logistic model tree results, with resultant value being '1' representing 'buying stock' whereas the resultant value being '0' representing 'selling stock'. By employing this strategy achieves high return on investment.

4. Experimental setup

Experimental analysis is carried out on the parameters such as specificity, sensitivity, time complexity and computational cost with respect to number of stock data. The results produced by the computation of Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier (LSBO-LDTC) for stock price is imparted in this section. In addition to the Exploratory Data Analysis (EDA), performance metrics, experimental results of proposed method is provided in this section. The execution of the process is performed in python high-level general-purpose programming language. To project the efficiency of the proposed LSBO-LDTC method for stock price prediction, comparison of existing methods, Belief Rule Base (BRB) [1], Technical analysis, K-means algorithm, and mean-variance model (TAKMV) [2] is analyzed and validated.

DISCUSSION

4.1 Performance analysis of sensitivity, specificity and accuracy

In this section performance analysis of three different metrics, sensitivity, specificity and accuracy is analyzed. Sensitivity is the ratio of true positives (correctly identified "1"s) to the total number of positives in the test dataset expressed as a percentage. In other words sensitivity refers to the prospective of a test (i.e., stock market prediction) to correctly identify stocks to either buy or sell. Sensitivity is mathematically formulated as given below.

$$Sen = \frac{TP}{TP+FN} \quad (14)$$

From the above equation (14), sensitivity 'Sen' is measured based on the true positive rate 'TP' (i.e., samples of stock to buy/sell correctly identified as buy/sell) and the false negative rate 'FN' (i.e., samples of stock to hold incorrectly identified as buy/sell). Specificity is the ratio of true negatives (correctly identified "0"s) to the total number of negatives in the test dataset expressed as a percentage. In other words specificity refers to the prospective of a test (i.e., stock market prediction) to correctly identify stocks to hold. Specificity is mathematically formulated as given below.

$$Spe = \frac{TN}{TN+FP} \quad (15)$$

From the above equation (15), specificity 'Spe' is evaluated on the basis of the true negative rate 'TN' (i.e., samples of stock to hold correctly identified as to hold) and false positive rate 'FP' (i.e., samples of stock to buy/sell incorrectly identified as hold) respectively.

Classification Accuracy (CA) or accuracy is the ratio of number of cases that are correctly classified to the total number of cases expressed as a percentage. Accuracy is mathematically represented as given below.

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (16)$$

From the above equation (16) accuracy 'Acc' is measured on the basis of the true positive rate 'TP', true negative rate 'TN', false positive rate 'FP' and false negative rate 'FN' respectively. Table 4 given below lists the comparative analysis of sensitivity, specificity and accuracy for ACC stock using LSBO-LDTC, BRB [1] and TAKMV [2].

Table 4 Comparative analysis of sensitivity, specificity and accuracy for ACC stock

Samples	Sensitivity			Specificity			Accuracy		
	LSBO-LDTC	BRB [1]	TAKMV [2]	LSBO-LDTC	BRB [1]	TAKMV [2]	LSBO-LDTC	BRB [1]	TAKMV [2]
100000	0.98	0.97	0.96	0.28	0.1	0.09	0.89	0.78	0.72
200000	0.95	0.92	0.88	0.3	0.28	0.27	0.85	0.77	0.71
300000	0.91	0.85	0.8	0.32	0.3	0.29	0.83	0.75	0.7
400000	0.88	0.8	0.77	0.34	0.31	0.3	0.85	0.79	0.72
500000	0.9	0.83	0.8	0.35	0.33	0.31	0.85	0.8	0.74

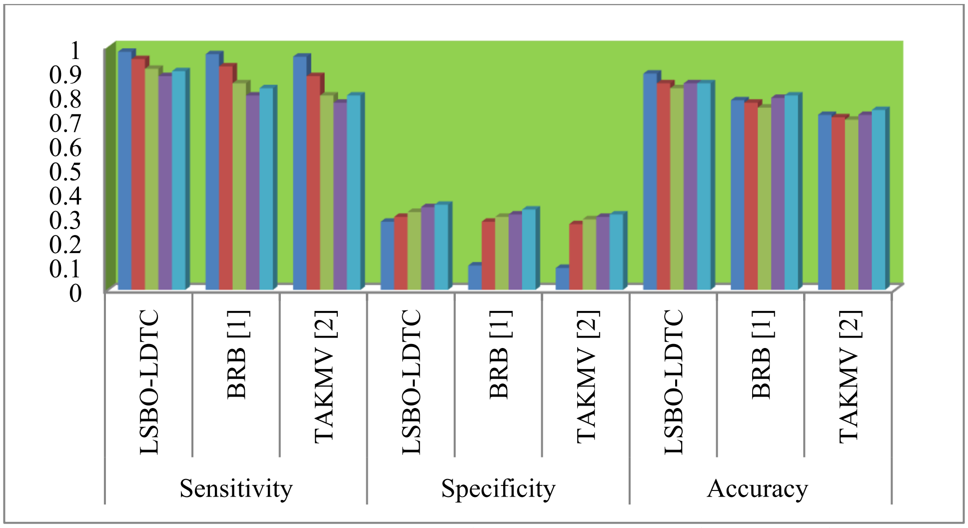

Figure 4 Sensitivity, specificity and accuracy versus samples (ACC stock)

Figure 4 given above shows the graphical representations of sensitivity, specificity and accuracy for 500000 samples acquired from the dataset. For analysis three different methods were selected, namely, LSBO-LDTC, BRB [1] and TAKMV [2]. From the above figure, with a sample of 100000 data, the sensitivity was observed to be 0.98, 0.97, 0.96, specificity was observed to be 0.28, 0.10, 0.09 and accuracy was found to be 0.89, 0.78 and 0.72 using LSBO-LDTC, [1] and [2] respectively for ACC stock. From this analysis the three performance metrics were observed to be better using LSBO-LDTC upon comparison to [1] and [2]. Table 5 given below lists the comparative analysis of sensitivity, specificity and accuracy for ASIANPAINT stock using LSBO-LDTC, BRB [1] and TAKMV [2].

Table 5 Comparative analysis of sensitivity, specificity and accuracy for ASIANPAINT stock

Samples	Sensitivity			Specificity			Accuracy		
	LSBO-LDTC	BRB [1]	TAKMV [2]	LSBO-LDTC	BRB [1]	TAKMV [2]	LSBO-LDTC	BRB [1]	TAKMV [2]
100000	0.95	0.92	0.91	0.23	0.18	0.15	0.84	0.79	0.74
200000	0.92	0.87	0.83	0.25	0.2	0.17	0.8	0.76	0.71
300000	0.88	0.8	0.75	0.27	0.22	0.19	0.78	0.74	0.69
400000	0.85	0.75	0.72	0.29	0.24	0.21	0.8	0.76	0.71
500000	0.87	0.78	0.75	0.3	0.25	0.22	0.77	0.73	0.68

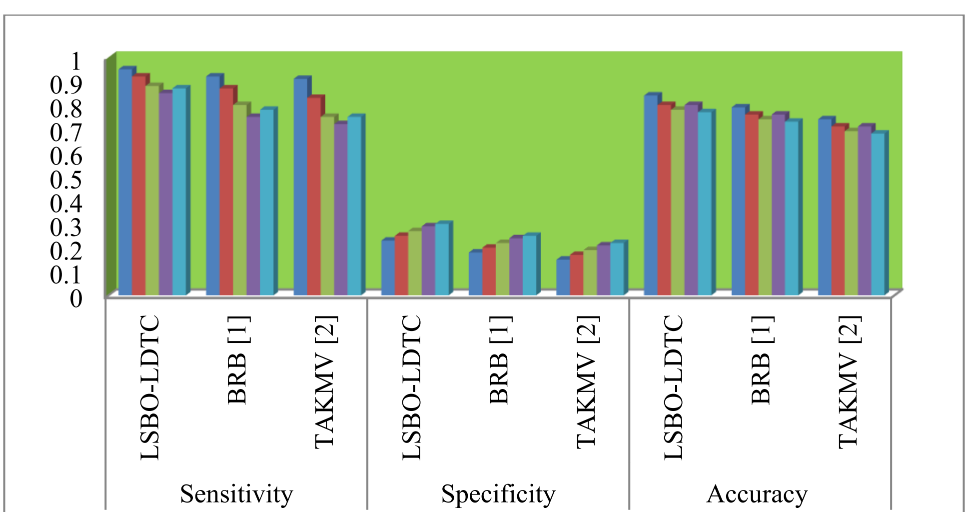

Figure 5 Sensitivity, specificity and accuracy versus samples (ASIANPAINT stock)

Figure 5 given above illustrates the graphical representations of sensitivity, specificity and accuracy for 500000 samples acquired from the dataset for ASIANPAINT stock. To make fair comparison between three distinct methods selected, LSBO-LDTC, BRB [1] and TAKMV [2] same dataset and samples were utilized. From the above figure, with a sample of 100000 data with respect to ASIANPAINT stock, sensitivity was found to be 0.95, 0.92, 0.91, specificity was found to be 0.23, 0.18, 0.15 and accuracy was observed to be 0.84, 0.79 and 0.74 using LSBO-LDTC, [1] and [2] respectively for ASIANPAINT stock. From this validation and analysis sensitivity, specificity and accuracy were found to be comparatively better using LSBO-LDTC than [1] and [2].

The reasons behind the above three performance metrics improvement, namely, sensitivity, specificity and accuracy using the proposed LSBO-LDTC method for both the ACC and ASIANPAINT stock data was due to the application of Logistic Decision Tree-based Classification algorithm for stock price prediction.

By applying this algorithm first, feature selection using Locust Swarm BoostARoota that combined Boruta utilizing XGBoost as the base model to Locust Swarm optimization was utilized. By employing this optimization model two operators, solitary (i.e., exploration) and social (i.e., exploitation) were employed to address both the exploration and exploitation involved in optimal features to be selected. Also best fitness results involved in feature selection (i.e., technical indicators) for further classification was performed via Gradient and Hessian processes. This in turn aided in reducing the false negative and false positive. With this the overall sensitivity and specificity involved in stock prediction using the proposed LSBO-LDTC method was found to be comparatively better by 6%, 42% (for ACC stock) and 9%, 19% (for ASIANPAINT stock) upon comparison to [1], [2] and [2].

Next, by exploiting the Logistic Decision Tree-based Classifier that combined both the advantages of logistic model and decision tree by taking into considerations upon comparison to conventional decision tree that takes into consideration testing on one attributes associated with every inner node upon comparison of all the inner node leaves aids in improving the accuracy. With this the overall accuracy of proposed LSBO-LDTC method was found to be better by 9%, 19% upon comparison to [1] and [2] for ACC stock and improved by 6%, 16% upon comparison to [1] and [2] for ASIANPAINT stock respectively.

4.2 Performance analysis of training time

In this section the performance analysis of training time involved in stock prediction is analyzed and validated. A considerable amount of time is said to be consumed during prediction and is referred to as training time. The training time is mathematically stated as given below.

$$TT = \sum_{i=1}^{m} S_i * Time(Pred) \qquad (17)$$

From the above equation (17) training time 'TT' is measured based on the sample data 'S_i' and the time involved in predicting stock '$Time(Pred)$' to either buy/sell or hold. It is measured in terms of seconds. Table 6 given below, list the comparative analyses of training time for ACC stock and ASIANPAINT stock using LSBO-LDTC, BRB [1] and TAKMV [2] respectively.

Table 6 Comparative analysis for training time using ACC stock and ASIANPAINT stock

Samples	Training time (sec) -ACC stock			Training time (sec) - ASIANPAINT stock		
	LSBO-LDTC	BRB [1]	TAKMV [2]	LSBO-LDTC	BRB [1]	TAKMV [2]
100000	135	185	215	125	145	165
200000	185	225	245	145	165	195
300000	215	255	285	175	190	215
400000	245	280	315	145	155	185
500000	200	250	300	180	165	215

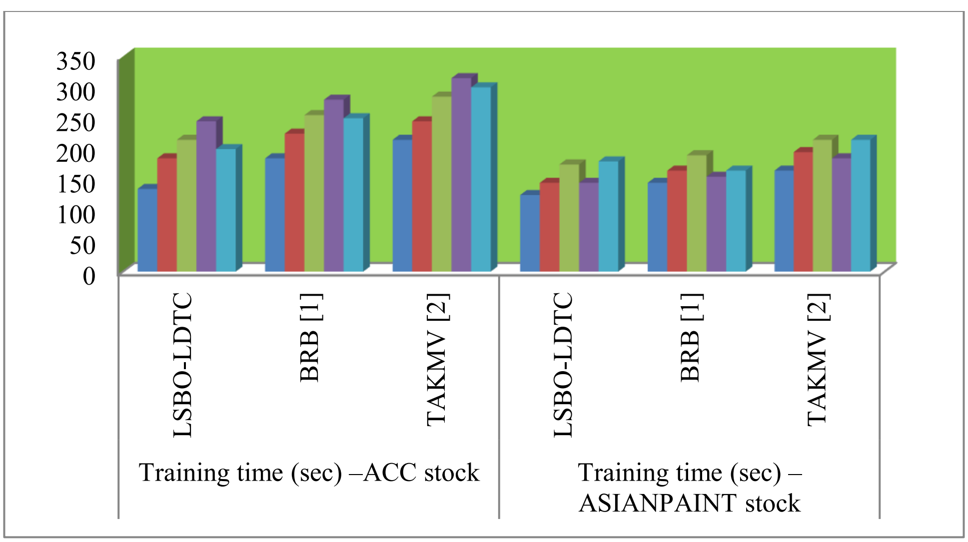

Figure 6 Training time versus samples

Figure 6 given above show the training time analyses, using 500000 training data for both ACC stock and ASIANPAINT stock respectively. To ensure fair comparison similar data were employed and validation was made for all the three methods, LSBO-LDTC, BRB [1] and TAKMV [2]. The training time for ACC stock using the LSBO-LDTC, BRB [1] and TAKMV [2] methods were observed to be 135 sec, 185 sec and 215 sec whereas using the ASIANPAINT stock it was found to be 125 sec, 145 sec and 165 sec respectively. From both the analyses it can be inferred that the training time using the proposed LSBO-LDTC were observed to be comparatively better than [1] and [2]. The reason was that by applying the Log First Difference transformation-based Preprocessing algorithm Log First Difference transformation was used as the noise reduction model for stock price prediction. Here by employing this transformation function logarithmic value of series of data were initially obtained and then difference value to obtain the preprocessed noise discarded result. This in turn minimized the training time using proposed LSBO-LDTC method by 19%, 28% upon comparison to [1] and [2] for ACC stock data and 6%, 21% upon comparison to [1] and [2] for ASIANPAINT stock respectively.

CONCLUSION

Stock market prediction refers to the process of determining the future value of a company stock traded on an exchange. Successful stock's future price prediction could give over notable gain. Hence, stock market prediction has gained a lot of interest in research community. In this study, we proposed a Locust Swarm BoostARoota Optimized Logistic Decision Tree-based Classifier

(LSBO-LDTC) for stock market prediction. The proposed model combines the Log First Difference transformation-based Preprocessing, Locust Swarm BoostARoota Optimized feature selection and Logistic Decision Tree-based Classification for stock prediction. These inputs aided the proposed LSBO-LDTC method to attain improved efficiency and accurate stock market prediction. To measure the effectiveness and performance, the proposed method was compared with existing methods. The experimental results invariably revealed that the LSBO-LDTC method outperformed the other conventional stock market prediction methods. LSBO-LDTC method achieved an accuracy of 27% with Stock Market Data - Nifty 100 Stocks (1 min) dataset showing efficiency of the proposed method in minimizing overall training time by 23%.

REFERENCES

- Yanzi Gao, Jiabing Wu, Zhichao Feng, Guanyu Hu, Yujia Chen and Wei He. "A new BRB model for technical analysis of the stock market", Intelligent Systems with Applications, Elsevier, Volume 18, May 2023, Pages 1-10 [Belief Rule Base (BRB)]
- Maricar M. Navarro, Michael Nayat Young, Yogi Tri Prasetyo and Jonathan V. Taylar, "Stock market optimization amidst the COVID-19 pandemic: Technical analysis, K-means algorithm, and mean-variance model (TAKMV) approach", Heliyon, Elsevier, Volume 9, Issue 7, July 2023, Pages 1-27 [Technical analysis, K-means algorithm, and mean-variance model (TAKMV)]
- Chawwalit Faijareon and Ohm Sornil, "Evolving and combining technical indicators to generate trading strategies", Journal of Physics: Conference Series, Volume 1195, 2018
- J. Margaret Sangeetha and K. Joy Alfia, "Financial stock market forecast using evaluated linear regression based machine learning technique", Measurement: Sensors, Volume 31, February 2024, Pages 1-7
- J. Margaret Sangeetha and K. Joy Alfia, "Financial stock market forecast using evaluated linear regression based machine learning technique", Measurement: Sensors, Volume 31, February 2024, Pages 1-7
- Abdelhadi Ifleh and Mounime El Kabbouri, "Stock price indices prediction combining deep learning algorithms and selected technical indicators based on correlation", Arab Gulf Journal of Scientific Research, Sep 2023
- Manish Agrawal, Piyush Kumar Shukla, Rajit Nair, Anand Nayyar, and MehediMasud, "Stock Prediction Based on Technical Indicators Using Deep Learning Model", Computers,Materials & Continua, Mar 2022
- Nitin Nandkumar Sakhare, Imambi S. Shaik, Suman Saha, "Prediction of stock market movement via technical analysis of stock data stored on blockchain using novel History Bits based machine learning algorithm", The Institution of Engineering and Technology, Jan 2023
- Guangyu Mu, Nan Gao, Yuhan Wang, Li Dai, "A Stock Price Prediction Model Based on Investor Sentiment and Optimized Deep Learning", IEEE Access, Jun 2023
- Salah Bouktif, Ali Fiaz, Mamoun Awad, "Augmented Textual Features-Based Stock Market Prediction", IEEE Access, Mar 2020
- Cheng Zhang, Nilam N. A. Sjarif, Roslina B. Ibrahim, "Decision Fusion for Stock Market Prediction: A Systematic Review", IEEE Aug 2022
- Kittisak Prachyachuwong and Peerapon Vateekul, "Stock Trend Prediction Using Deep Learning Approach on Technical Indicator and Industrial Specific Information", Information, Oct 2021
- Zhifeng Dai, Huan Zhu, Jie Kang, "New technical indicators and stock returns predictability", International Review of Economics and Finance, Elsevier, Mar 2021
- Zhifeng Daia, Xiaodi Donga, Jie Kanga, Lianying Hongb, "Forecasting stock market returns: New technical indicators and two-step economic constraint method", North American Journal of Economics and Finance, Elsevier, May 2020
- Prakash Balasubramanian, Chinthan P., Saleena Badarudeen1 and Harini Sriraman, "A systematic literature survey on recent trends in stock market prediction", Peer Journal of Computer Science, Jan 2024
- Xuan Ji, Jiachen Wang and Zhijun Yan, "A stock price prediction method based on deep learning technology", International Journal of Crowd, Mar 2022
- Audeliano Wolian Li, Guilherme Sousa Bastos, "Stock Market Forecasting Using Deep Learning and Technical Analysis: A Systematic Review", IEEE Access, Oct 2020
- Kevin Rink, "The predictive ability of technical trading rules: an empirical analysis of developed and emerging equity markets", Financial Markets and Portfolio Management, Springer, Aug 2023
- Htet Htet Htun, Michael Bieh, Nicolai Petkov, "Survey of feature selection and extraction techniques for stock market prediction", Financial Innovation, Springer, Mar 2023

  Ting Liu, Weichong Choo, Matemilola Bolaji Tunde, Cheongkin Wan, Yifan Liang, "Enhancing stock volatility prediction with the AO-GARCH-MIDAS model", Plos One, Jun 2024
- Haein Lee, Jang Hyun Kim, Hae Sun Jung, "Deep-learning-based stock market prediction incorporating ESG sentiment and technical indicators", Scientific Reports, Feb 2024
- Tran Phuoc, Pham Thi Kim Anh, Phan Huy Tam, Chien V. Nguyen, "Applying machine learning algorithms to predict the stock price trend in the stock market – The case of Vietnam", Humanities and Social Sciences Communications, Mar 2023
- Warda M. Shaban, Eman Ashraf, Ahmed Elsaid Slama, "SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting", Neural Computing and Applications, Springer, Nov 2023
- Jiawei Wang, Zhen Chen, "Factor-GAN: Enhancing stock price prediction and factor investment with Generative Adversarial Networks", PLOS ONE, Jun 2024
- Witawat Norasaed, Thitirat Siriborvornratanakul, "Market movement prediction using chart patterns and attention mechanism", Discover Analytics, Springer, Mar 2024