

An Optimized Machine Learning Model for Stock Trend Anticipation

Nalabala Deepika*, Mundukur Nirupamabhat

Vignan's Foundation for Science, Technology & Research, Guntur, AP 522213, India

Corresponding Author Email: deepika.kitss@gmail.com



<https://doi.org/10.18280/isi.250608>

ABSTRACT

Received: 17 September 2020

Accepted: 22 November 2020

Keywords:

stock market, security market, economical-volatile, external factors, secure prediction, feature selection, trend prediction, artificial bee colony

Security market is an economical-volatile in nature as it is driven by not only based on historical prices various unpredictable external factors like financial news, changes in socio-political issues and natural calamities happened in real world; hence its forecasting is a challenging task for traders. To gain profits and to overcome any crisis in financial market, it is essential to have a very accurate calculation of future trends by for the investors. The trend prediction results can be used as recommendations for investors as to whether they should buy or sell. Feature selection, dimensionality reduction and optimization techniques can be integrated with emerging advanced machine learning models, to get improvised prediction in terms of quality, performance, security and for effective assessment external factors role in stock market nonlinear signals. In this empirical research work, a set of hybrid models were built and their predictive abilities were compared to find consistent model. This work implies the base model, boosted model and deep learning model along with optimization techniques. From the experimental result, the optimized deep learning model, ABC-LSTM was turned out superior to all other considered financial models LSSVM, Gradient Boost, LSTM, ABC-LSSVM, ABC-Gradient Boost by showing best Mean Absolute Percentage Error (MAPE) value, which was low.

1. INTRODUCTION

Stock market is a synchronized financial market where the various securities are traded and governed by stock exchange board. In the modern days, everyone's general objective is lead peaceful life and prepare them self for upcoming difficulties without any stress [1]. Hence there exists a struggle towards developing a strong financial position. In addition, stock market is the only place where we apply intelligence for building wealth in a short time. Thus, a lot of consideration had been given to the analysis and forecast of future prices and patterns of the stocks, and it has become an interesting topic of research. Conventionally, diverse statistical forecast techniques [2, 3], for example, correlation analysis, time series analysis, linear regression were utilized to anticipate the stock pattern. However, these techniques were halfway effective due to their non-reliability. As an option in contrast to these methodologies, machine learning techniques, for example, SVM, Random forest, artificial neural system are generally used to catch the non-linearity of the financial exchange. In any case, anticipating a definite or close to-correct pattern of the stock market prices is an open issue. All these mechanisms imposed on the historical data of particular stock, but this internal data was not enough for accurate prediction as the stock market is having volatile as a characteristic. In real time there are number of external factors, which allows a greater impact on trading occurring in stock market.

All things considered, there may be the need to investigate applicable factors which may affect stock price/trend for example climate [4], US economy, worldwide economy, exchange rate [5], geopolitical occasions [6], supply and

demand insights and so on and there could be some more. Based on data available, two main types of analysis related stock trading can be done; namely fundamental and technical analysis. Fundamental analysis is a strategy for assessing the basic estimation of a security and studying the elements that could impact its upcoming price. This type of assessment depends on exterior events and impacts, in addition to financial reports and industry patterns. Most of financial traders who need to assess long term venture options start with a basic examination of an organization, an individual stock, or the market as a whole. Fundamental analysis is the way toward estimating a security's natural worth by assessing all divisions of a business or market. Technical analysis is a specialized examination method for analyzing and foreseeing value movements in the economic markets, by utilizing historical price outlines and market insights. It depends on the possibility that if a trader can recognize past market trends, they can frame a genuinely precise forecast of future price or trend directions. In general, some widespread technical indicators are taken into consideration for more returns and less risk to predict the stock market. In the light of previous studies, it is hypothesized that various technical indicators may be used as input variables in the construction of prediction models to forecast the direction of movement of the stock price index [7]. We selected technical indicators as feature subsets by the review of domain experts and prior researches [8-14]. The data uploaded into social media by the public about the particular stock plays a vital role and used as essential information, which may influence the opinions of the traders on respective or relevant stocks. Figure 1 depicts the general plot of how social media affect stock market Trends.

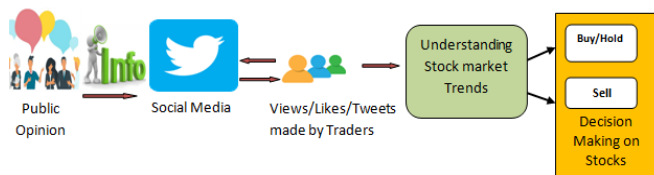


Figure 1. Social media affect on stock market trends

The public opinions regarding stocks are shared at the social media; which is common for all, namely Twitter. The machine learning algorithms can be adopted for natural language processing of public tweets to obtain the consisted sentiment value. The basic idea - “averaging scores may result in applicable information”, considered here for the calculation of stock’s overall sentiment score. The trader will make the decision regarding stocks, whether to buy/hold or sell the stock based on knowledge gained from analysis.

Optimization technique is a mathematical model development by defining cost/objective function and applying optimization procedure iteratively until optimize i.e., minimize/maximize the cost function or any satisfactory solution is found. In machine learning, numbers of optimization techniques are available; these can be implied along with base, ensemble machine learning models and with deep learning models to improve the accuracy of constructed predictive model. Machine learning algorithms are used to build, base models like linear regression, support vector machine-SVM, K-Nearest Neighbor-KNN, decision trees; ensemble models like random forest, boosted trees; deep learning models like feed forward neural network-FNN, recurrent neural network-RNN, and Long-Short Term Memory-LSTM. In deep learning to train the network, to tune the learning rate or to optimize error function at network layers. Dimensionality reduction can be adopted to make predictive modeling ease and performance upgrading.

The whole paper organized as preceding sections, the survey of the literature with respect to the work is presented in Section 2. The proposed architecture is demonstrated in Section 3 and the methodologies applied explained in section 4. Finally, the interpretations of results are shown in Section 5, followed by future work in Section 6.

2. LITERATURE SURVEY

Qiu et al. [15], developed a prediction technique to guide the stock trading. The authors implemented the LSTM along with wavelet transform to denoising the data. They made the comparative study of the three models namely LSTM model, the LSTM model with wavelet denoising and the gated recurrent unit (GRU) neural network model. The datasets utilized from the standard bench mark indices S&P 500, DJIA, HSI. The experimental results show that the prediction along with attention mechanism defeats the basic LSTM prediction model.

Sezer et al. [16], studied the time series forecasting mechanisms with deep learning mechanism for accurate stock market prediction. The authors presented a systematic review of studied research papers to guide the rest of the researchers in the same research area.

Chopra et al. [17], investigated the forecasting capability of Artificial neural networks (ANN) at the Indian currency demonetization period. The authors considered the nine stock

values and the index CNX NIFTY50 for the period of 8 years as input data set. The authors adopted the Levenberg-Marquardt algorithm for training back propagation multilayered NN. By the experimentation, the authors stated that the hidden layer neurons discrepancy does not affect MSE. And also the proposed model efficiently predicts the close price and best suitable for market with high volatile situations.

Sedighi et al. [18] proposed a novel mixture model consisting of Artificial Bee Colony (ABC), Adaptive Neuro-Fuzzy Inference System (ANFIS), and Support Vector Machine (SVM) for accurate stock prediction. The authors compared the base model with the other mixture models and found that, their model out performed rest of the models. The considered US (united states) stock exchange fifty largest stocks for the period of ten years from (2008 to 2018).

Ghanbari and Arian [19], developed a novel hybrid prediction model BOA-SVR, the Butterfly Optimization Algorithm is used to select parameters of SVR. The performance is compared with other meta heuristic optimized SVR models. The experimental results judged by both prediction performance accuracy and time consumption, revealed that the model constructed is one of the best models.

Arora [20], made a financial analysis for the efficient stock prediction using deep learning models. They made comparative analysis among SVM, Back propagation and LSTM models with mean accuracy and standard variance values. They found that, forecasting with LSTM model outperformed the rest of the two models.

Gandhmal et al. [21], made a systematic analysis and review of research papers on the concept of an efficient stock market prediction. The authors studied the various methodologies, like Artificial Neural Networks (ANN), Bayesian model, Fuzzy classifier, Neural Network (NN), Support Vector Machine (SVM) classifier Machine Learning Methods. The authors presented a clear understanding of prediction technologies, performance metrics, and gaps and challenges in the research area of stock market prediction system. They concluded their study as, for the accurate and efficient stock market prediction different factors need to be considered hence, it is a difficult task.

Das et al. [22], explored the novel work for stock market future trends forecasting of the by firefly algorithm with evolutionary frame work. They adopted the prediction models Extreme Learning Machine (ELM), Online Sequential Extreme Learning Machine (OSELM) and Recurrent Back Propagation Neural Network (RBPNN). The performance of the constructed models was evaluated for the datasets from BSE, NSE, S&P 500 and FTSE. Before feeding to the model, the input feature set is reduced with the appropriate mechanism from the list of statistical techniques namely Factor Analysis (FA), Principal Component Analysis (PCA) and optimized techniques Genetic Algorithm (GA), Firefly Optimization (FO) and along with evolutionary framework. From the experimental comparative analysis, the authors found that the prediction model developed with algorithm called Firefly with evolutionary framework along with OSELM shown the outstanding performance comparatively than the rest of the models.

Shah et al. [1], presented the novel work with the aid of artificial intelligence and neural network for predicting the stock market trends. Quick Gbest Guided artificial bee colony (ABC) learning algorithm is used to train the feed forward neural network, so they named their model as QGGABC-FFNN. The data set considered from Saudi Stock Market

(SSM). The experimental results revealed that the presented model outperformed the other conventional models.

Shynkevich et al. [23], investigated the impact of horizon and input window size on prediction accuracy by implementing KNN, ANN, SVM algorithms for 2-class and 3-class data classification. The horizon trading days and input window size considered at different values from 1 to 30. The performance metrics namely prediction accuracy, return per trade, winning rate and Sharpe ratio are utilized for algorithms' evaluation. The authors noticed that the performance metrics are high at the same value for both horizon trading days and input window size. The averaged performance metric value calculated for each individual horizon.

Jamous et al. [24], proposed an efficient technique for the stock market prediction by using an adaptive linear combiner. A particle swarm optimization with center of mass (PSOCoM) used to adjust the particles during training period by the mean square error (MSE) as an objective function. The authors used the data collected from USA stock market indices namely Standard's and Poor's 500 (S&P 500), Dow Jones Industrial Average (DJIA) and National Association of Securities Dealers Automated Quotations 100 (NASDAQ-100) evaluated their model by comparing with evolutionary computing tools such as particle swarm optimization (PSO), genetic algorithm (GA), bacterial foraging optimization (BFO), and Adaptive bacterial foraging optimization (ABFO) as an alternative of PSOCoM. They found that the adaptive linear combiner with PSOCoM defeated all with improved prediction accuracy.

Hegazy et al. [25], proposed a novel mixture model by adopting particle swarm optimization (PSO) and least square support vector machine (LS-SVM). The authors used historical stock and technical indicators data as input parameters. For better prediction accuracy, the PSO algorithm elects the appropriate parameters as input for LS-SVM to overcome over-fitting and local minima issues. The authors evaluated the constructed model with thirteen numbers of financial benchmark datasets and made comparison with artificial neural network with Levenberg -Marquardt (LM) algorithm. They found that, their model exhibited the better performance by giving more prediction accuracy and showcased the optimized LS-SVM model with the aid of PSO algorithm.

Badar et al. [26] discussed that the adoption of Artificial Intelligence techniques for active power loss minimization of widely distributed Electrical Power systems leads to an optimized result. They gave an overview of performance characteristics and enhanced versions of various AI techniques namely Ant Colony Optimization, Differential Evolution, Genetic algorithm, Particle Swarm Optimization, Simulated Annealing and Tabu Search. The authors suggested that the merging of AI techniques may result in a more optimized solution.

3. ARCHITECTURE

For getting final predicted outcome as shown in Figure 2, the three phases namely Data set formation, Data Pre-processing and Final forecasting model are to be executed as a present research work.

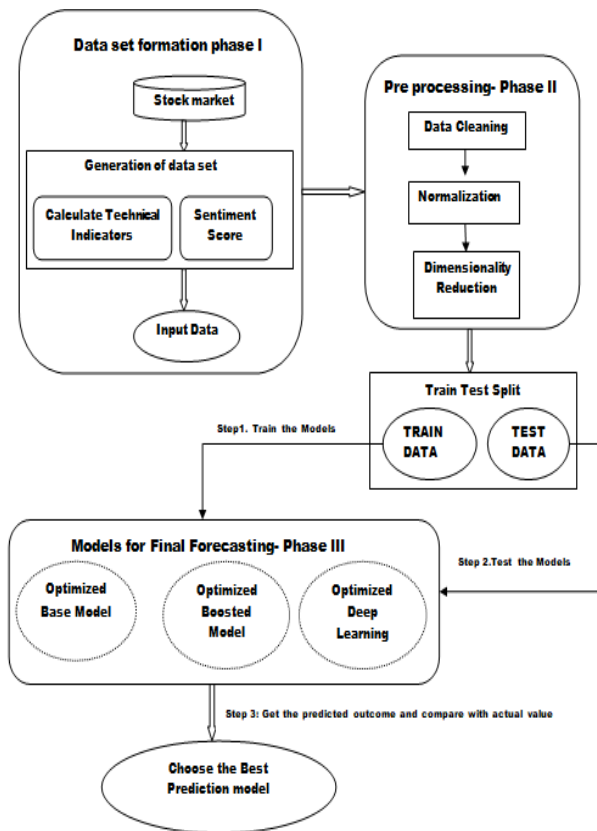


Figure 2. Layout of proposed forecasting model

3.1 Data set formation

In phase I, the input dataset is formed by extracting from Yahoo finance database of the data Indian Stock market named NSE for the six famous companies namely Apple, Amazon, Infosys, Microsoft, Oracle, TCS; under the technology and software services sector. The listing of company names and stock quotes is shown in the Table 1. On a single date, for each transaction, there are the Date, Open, High, Low, Close, Adjacent Close, Volume parameters. Using these values Technical indicators namely Moving average, Increase_in_vol, Increase_in_adj_close, ADX, CCI, MACD and RSI are calculated. Sentiment score evaluated by using text summarization and mining from the social platform collected data.

Table 1. List of tickers

Tickers	Company names	Tickers	Company names
AAPL	Apple Inc.	TCS	Tata Consultancy Services
AMZN	Amazon.com Inc.	ORCL	Oracle Corporation
INFY	Infosys	MSFT	Microsoft Corporation

3.2 Data Pre-processing

As the collected data may contain abnormal/noisy data, it is intended to be processed i.e., need to be cleaned and normalized and if necessary, dimensionality reduction need to be performed so as to reduce the features in order to model train.

3.2.1 Data cleaning

As the data to be analyzed is collected from various sources, there are many chances of duplicated/mislabeled data due to manual data entry procedures, equipment errors and incorrect measurements. A missing value can signify a number of different things in your data. As the processing step we need to identify missing values and either drop or impute them.

3.2.2 Normalization

For effective data processing we may in need of transforming from one form to another format. Decimal scaling, min-max scaling, Z-score scaling are the types of normalization.

3.2.3 Dimensionality reduction

The number of input variables or features for a dataset is called as Dimensionality. Having more input features is known as the curse of dimensionality, which causes for the task of predictive modeling more complex and machine learning technique performance degrade. In order to better fit a predictive model, the classification or a regression process need to be simplified by reducing the number of input features in dataset basing on essence of the data.

The data is further split into train and test dataset, in which train data is meant for Training the model and test data is used for testing the model trained.

3.2.4 Final forecasting model

The models adopted are ABC-LSSVM as optimized base model, ABC-Gradient boosting as optimized boosted model and ABC-LSTM as optimized deep learning model. After getting the predicted values the performance metrics MSE, MAPE are calculated to find the accurate prediction model.

4. METHODOLOGY

4.1 Technical analysis

Technical Analysis Library (TA-Lib) is an open-source library available at www.ta-lib.org which is widely used by trading software developers for performing TA of market data. This involves the process stock price determining, based on the past patterns of the stock using technical indicators calculated with time-series analysis. Using open, high, low, close the indicators namely CCI, ADX, MACD, RSI were calculated.

MACD: Moving average convergence divergence displays the trend following the characteristic of market prices.

CCI: Consumer channel index identify prices reversals, price extreme and the strength of rise prices

RSI: Relative strength index is a momentum oscillator that detect the speed and changes of price movements

ADX: Average directional index is an indicator use to determine the strength trend of prices movement. The trend can either be up or down base on the two indicators, positive directional and negative directional index.

Moving average: the mean value of Adjacent Close with rolling window size 30.

Increase_in_vol: the volume difference value of two consecutive days.

Increase_in_adj_close: the adjacent close difference value of two consecutive days.

Change of Close price: Represents how big or small change

in stock /index closing value used to capture fluctuation in market. The given day Close price rate of change is calculated using division of difference between closing values of yesterday's and today's with closing value of yesterday's.

Basic assumption in stock market is that stock Deviation, increase or decrease will depend on how given stock was doing in past. Also, the stock price for given company will also depend on how market is doing and how given sector is doing. Keeping these three things in mind Index_Deviation, Index_ROC, Sector_Deviation, Stock_Deviation, Stock_Price_ROC, Output parameters calculated for input dataset as below:

Deviation: If price of stock or index in higher than yesterday then Deviation for given day is +1 as there is an increase in price else it's -1.

Rate of change (ROC): Represents how big or small change in stock /index closing values. Try to capture fluctuation in market. For given day ROC is calculated using difference between yesterday's closing value - today's closing value divide by yesterday's closing value.

Index_Deviation: Calculated based on market performance for last 5 days. It's an average of 5 days index Deviation.

Index_ROC: Calculated as average of last 5 days ROC for index.

Sector_Deviation: Calculated based on sector performance for last 5 days. It's an average of 5 days index value for all technology company shares.

Stock_Deviation: Calculated as average for last 5 days Deviation for given company Deviation

Stock_Price_ROC: Calculated as average of last 5 days for given stock.

Output: If closing stock price for a stock today day is more than yesterday's closing stock price for the same stock, then the corresponding output is denoted by 1 else it is denoted as 0.

Hence.in one record of input data set had the dimensions: Stock Price, Index_Deviation, Index_ROC, Sector_Deviation, Stock_Deviation, Stock_Price_ROC, (Relative Strength Index) RSI6, (Average Directional Index) ADX14, (Commodity Channel Index) CCI12, (Moving Average Convergence / Divergence) MACD, sentiment, other stock's close values and Output dimension need to predict.

4.2 Feature engineering

The text summarization was applied on collected day-wise tweets for a particular company and then sentiment analysis was performed to get sentiment value. The feature namely sentiment score is calculated for every entered opinion with the aid of Vader sentiment analyzer from natural language tool kit. The idea averaging is adopted per each date to obtain the final sentiment score.

Feature selection: The Dimensionality reduction is the phase to be executed after the data pre-processing steps cleaning, normalizing the data and before model training.

4.3 Dimensionality reduction techniques

- Feature selection methods like scoring or statistical methods are for selecting only essential features and to remove irrelevant features. wrapper methods, filter methods and embedded methods are the feature selection technique three main classes.

- Wrapper method is a search process follows best first search, by preparing the number of feature subsets to wrap by fitting and evaluating the predictive model the scores are assigned according to model performance. This method uses the stochastic procedure or heuristics. An example for stochastic is hill-climbing algorithm, and the passes in forward/backward directions to add/remove features is an example for heuristics. RFE- recursive feature elimination is an example of a wrapper feature selection method.

- Filter Method is a ranking process uses statistical measure like correlation to assign score. Pearson’s correlation, Chi-Squared test and information gain are the examples of a filter feature selection method.

- Embedded Method is a learning process which identifies the features which contributes best accuracy. The regularization methods Ridge Regression, Elastic Net and LASSO are the examples of an embedded feature selection method.

- Matrix Factorization methods adopts the linear algebra for dimensionality reduction. By reducing a dataset matrix into a selected subset of its constituent ranked parts, which represents the whole dataset with salient structure of the matrix. Principal components analysis (PCA), implemented by matrix decomposition techniques like Eigen decomposition, singular value decomposition is the frequently used method for components ranking.

- Manifold Learning method adopts a projection, which maps the high dimensional data into low dimensions by conserving the relationships or salient structure representation of the data. Self-Organizing Map (SOM), Multidimensional Scaling (MDS), Sammons Mapping, t-distributed Stochastic Neighbour Embedding (t-SNE) are the examples.

4.4 Algorithms adopted

PCA: To identify linearity in considered data set by reducing the number of dimensions with less information loss. After dimensional reduction, the final features will be named as principal components also called as eigenvectors. The magnitude of these components is called as eigen values which shows the variance of the data along the new feature axes. To train a model, PCA outputs should be given as input. To embed the PCA with any other learning algorithm, the Pipeline can be used s best way, where initially the PCA transform the data and the next step is the learning algorithm that takes the transformed data as input.

LS-SVM: To impose linearity in to the model, the traditional SVM model was altered as LSSVM by using a function with equality constraints called regularized least squares. In this adaption, instead of a convex quadratic programming (QP) problem as in classical SVM, the linear problem considered and the solution was found by solving a set of linear equations. These models are known as SVM with versions of least squares to recognize patterns and analyze data for the supervised learning methods namely regression and classification analysis.

Implementation: The various types of kernels are listed as in Table 2.

Table 2. Types of kernels

S. No	Kernel Type	Kernel parameter
1	Polynomial Kernel	“poly”
2	Gaussian Kernel	“rbf”-Radial basis function
3	Sigmoid Kernel	“sigmoid”

The Least-Squares Support Vector Machine (LSSVM) model is the advanced version of basic Support Vector Machine (SVM) with slight alterations in the main functions called objective and restriction, which results in a best simplification of the optimization problem in SVM. Equation (1), (2) and (3) shows the objective function for basic SVM model optimization Problem. The SVM had a set of inequality restrictions; hence for optimization problem solution the kernel trick needs to be added as Eq. (4).

$$\text{minimize } f_0(\vec{w}, \vec{\xi}) = \frac{1}{2} \vec{w}^T \vec{w} + C \sum_{i=1}^n \xi_i \quad (1)$$

$$\text{S.T } d_i \left(\vec{W}^T \vec{x}_i + b \right) \geq 1 - \xi_i \quad i=1,2,\dots,n \quad (2)$$

$$\xi_i \geq 0, \quad i=1,2,\dots,n \quad (3)$$

$$f(\vec{x}) = \text{sign} \left(\sum_{i=1}^n \alpha_i^0 d_i k(\vec{x}_i, \vec{x}) + b_0 \right) \quad (4)$$

$$\text{minimise } f_0(\vec{w}, \vec{\xi}) = \frac{1}{2} \vec{W}^T \vec{w} + \gamma \frac{1}{2} \sum_{i=1}^n \xi_i^2$$

$$\text{S.T } d_i \left(\vec{W}^T \vec{x}_i + b \right) = 1 - \xi_i \quad i=1,2,\dots,n \quad (5)$$

The main alteration for LSSVM is changing the restrictions into the form of equality from inequality condition in the objective function Eq. (5).

Gradient Boosting: It is a dominant algorithm for the task classification or regression in machine learning which uses decision tree models and can be used for classification or regression predictive modeling problems. It follows the ensemble method namely boosting which adds models in sequential manner to improve the prior models performance by correcting the prediction errors. The basic gradient boosting is a generalized version of initial algorithm called AdaBoost with advancements in model fitting using gradient descent optimization algorithm and arbitrary differentiable loss function. To improvise the basic model performance, the enhancement can be allowed for Gradient Boosting Hyper parameters namely Tree Constraints like Number of Trees, Tree Depth; Weighted Updates like Learning Rate which is used to limit how much each tree contributes to the ensemble; Random sampling adopted for fitting trees on subsets of Samples and Features. The use of random sampling without replacement often leads to a stochastic gradient boosting algorithm.

Implementation: In python, Gradient Boosting implementation can be done with the aid of the classes namely `cross_val_score`, `Repeated Stratified K Fold`, `Gradient Boosting Classifier`, `Gradient Boosting Regressor`, `pyplot` which are imported from the packages namely `sklearn`, `numpy`, and `matplotlib`.

LSTM: The Long Short-Term Memory (LSTM) neural network is considered due to its long term predictive property demonstration in complex problem domains, specifically for sequence prediction problems. These are the recurrent neural networks and very different from other deep learning techniques, such as Multilayer Perceptrons (MLPs) and

Convolution Neural Networks (CNNs).

4.5 Optimization techniques

Optimization is the strategy adopted for neural networks-“The weights or learning rates are altered by reducing the losses”, to produce the accurate results. These are classified into first order and second order algorithms. Gradient Descent, SGD-Stochastic Gradient Descent, Mini-Batch Gradient Descent, Momentum and Nesterov Accelerated Gradient are the first order optimization algorithms with constant learning rate (η). AdaGrad, AdaDelta are the second order optimization algorithms with dynamic learning rate (η). Adam is another algorithm which adopts momentums of first and second orders. To efficiently train the neural network in less time, Adam is the best algorithm among all listed algorithms.

Artificial Bee Colony Algorithm (ABC):

ABC is an efficient population-based, stochastic evolutionary optimization algorithm, inspired by the intelligence of natural behaviour of honeybees also recognized as Swarm Intelligence (SI) concept. An artificial colony is formed by arranging total bees as three groups namely employee, onlooker and scout bees. The artificial colony aims to maximizing the nectar’s amount (solutions’ qualities/fitness value), by positioning both the employed bees and onlooker bees equally in count. And in an equal number, the food sources (solutions) are handled as one source responsible for one employed bee. The nectars’ amount is calculated for the food sources searched by the employed bees. The knowledge gained is communicated to the bees called onlooker available on hive (dance area) for make use of a nectar source. The onlooker bees take care of deallocating source assigned with the employed bees and appoint as scout bees to search the new valuable food sources.

PSEUDO CODE FOR LSSVM-ABC ALGORITHM

ABC optimizer:

Control Parameters of ABC Algorithm are set as:
 Colony size, CS = 30, dimension of the problem, D = 25, Limit for scout, L = (CS*D)/2.
 Initialize_employees
 Initialize_onlookers
 trail=0, itr =1,max_trails=m,max_itr=n_itr
 Initialize the food source positions (solutions) xi=1, ..., SN
 Evaluate the nectar amount (fitness function fiti) of food sources
 fitness=LSSVM(dim=D, gamma=1, kernel='rbf', data=DS)
 REPEAT

Employee_Bee (EB) Phase:

```
// Initialization Phase: First of all, initialization of the positions
of 15 food sources (CS/2) of employed bees is randomly done
Using uniform distribution in the range (-1, 1).
Employed bees are half of colony size. EB=CS/2;
For each EB,
    if(trail<=max_trails):
        food_sources [Fit_ ness i]
        // EmployedBees Initial fitness values
        update_ optimal_ solution:
        //produce new food source positions (solutions) as new_pos
        component= random_selection(sample)
        new_pos = pos + (pos- component) * Φ;
        //Φ is randomly produced number in the range [-1, 1].
```

```
Calculate the value Fit_ ness i
Fit_ ness i = 1 / (1 + fitness); fitness ≥ 0
            = 1 + abs (fitness); fitness < 0
If (Fit_ ness i<fitness)// solution couldn't be improved
trail++;
goto next employee_bee
else
    trial = 0
Update_ food_sources[ ]
//with updated EmployedBees replace the solution
greedy selection ( )
// select_best_food_sources and Memorize it as the best solution
End For
Calculate the probability values pi for the best solution with pi=
fitness/sum of fitness
```

Onlooker_Bee (OB) phase:

```
For each OB,
Produce new food source (solutions)
// Choose the food sources depending on pi
candidate =random_ selection(best_ food _sources)
if(trial <= max_ trials):
    component = random_ selection (candidate)
    new_ pos = candidate + (candidate - component)*Φ;
    //Φ is randomly produced number in the range [-1, 1].
    Calculate the value of Fit_ ness i
    Fit_ ness_i = 1 / (1 + fitness); fitness ≥ 0
              = 1 + abs (fitness); fitness < 0
    If(Fit_ ness i>= fitness):
        trial = 0
    else:
        trial ++
goto next onlooker_bee
        greedy selection ( ) // select_best_food_sources and
Memorize it as the best solution
End For
```

Scout Bee (SB) Phase:

```
if(assign(task(EB) → task(OB))
Free(EB_food source)
// Determine the source to be abandoned
Allocate(SB) ← EB
// allocate employed bee as scout forsearching new food sources.
greedy selection ( )
// select_best_food_sources and Memorize it as the best solution
itr=itr+1
Until (itr= =n_itr)
// n_itr=Maximum number of iteration
```

Implementation:

1. Initialize dataset as DS.
2. Assign hyper parameters of LSSVM.
3. Consider following Mean Absolute Percentage Error as objective function of optimization.

$$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

4. Pass calculated objective function to fitness parameter in ABC Optimizer.

The performance of LSSVM is based on the selection of hyper parameter’s values for the model namely cost penalty (C), insensitive-loss function (ϵ) and kernel parameter (γ). In this work, ABC optimization technique can be used to get the best suitable values for LSSVM parameters.

5. RESULTS

The response variable Y is discrete then performance can be measured with Confusion Matrix as in Figure 3.

Confusion Matrix		Actual	
		Positive	Negative
Forecasted	Positive	A True Positives	B False Negatives
	Negative	C False Positives	D True Negatives

Figure 3. Confusion Matrix

Confusion Matrix is an N X N matrix, which forms the basis for the other types of performance metrics. Here N is the number of forecasted classes.

- True Positives: The number of data point's label forecasted as Yes, where the actual labels are Yes.
- True Negatives: The number of data point's label forecasted as No, where the actual labels are also No.
- False Positives: The number of data point's label forecasted as Yes, where the actual labels are No.
- False Negatives: The number of data point's label forecasted as No, where the actual labels are Yes.

$$\text{Sensitivity} = A / (A + C)$$

$$\text{Specificity} = B / (B + D)$$

$$\text{Accuracy} = (A + D) / (A + B + C + D)$$

$$\text{Precision} = A / (A + B)$$

- Accuracy: The proportion of number of data points that are correctly classified, with respect to all data points.
- Positive Rate (Sensitivity): The amount of positive data points that are correctly classified as positive, with respect to all positive data points. This value is in the range [0,1]

- False Positive Rate (Specificity): The amount of negative data points classified as positive mistakenly. This value is in the range [0,1].

The response variable Y is continuous then performance can be measured with following measures as in Table 3.

- Mean Absolute Error: It is the measure of the prediction model performance, calculated in terms of the average of how much deviation presented in forecasted values from actual values.
- Mean Squared error: Similar to MAE, but average of the square of how much deviation presented in forecasted values from actual values.

Mean absolute percentage error: The MAE equivalent percentage, i.e. MAE with modifications, which converts the whole thing into percentages, is known as mean absolute percentage error (MAPE). It can be also stated as ratio of the residual $(Y - \hat{Y})$ over the actual (Y), multiplying by 100 % helps to convert in to percentage.

With the two forms of data set, the experimentation is carried out for predicting change of stock for each company's close price value, which is continuous value. The feature set 1 considers the dimensions namely company stock values, technical indicators namely CCI, RSI, MACD, ADX, Moving average, Increase_in_vol, Increase_in_adj_close, Momentum, change and sentiment. The feature set 2 considers the reduced dimension set after applying PCA.

The performance of LSSVM with ABC optimization for the Amazon stock data in the format of feature set 1 is as shown in Figure 4, And in the format of feature set 2 is as shown in Figure 5. This depicts that the cost is decreasing and optimization fitness increased as the number of iterations increased. With the feature set 1, the fitness value linearly increased with respect to number of iterations, finally figured to the value 3 as the fitness. And the cost also decreased as the number of iterations increased.

Table 3. Performance Metrics for prediction model

Metric	Formula	Disadvantage
Mean Error (ME)	$ME = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)$	Does not capture the underlying data if + ve and -ve data cancelled out.
Mean Absolute Error (MAE)	$MAE = \frac{1}{N} \sum_{i=1}^N y_i - \hat{y}_i $	Can't perform derivatives/calculus as absolute values are there
Mean Squared Error (MSE)	$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$	Units are getting squared.
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}$	Gets influenced by units
Mean Percentage Error (MPE)	$MPE = \frac{100\%}{N} \sum_{i=1}^N \left(\frac{y_i - \hat{y}_i}{y_i} \right)$	+ ve and -ve values gets cancelled out.
Mean Absolute Percentage Error (MAPE)	$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left \frac{y_i - \hat{y}_i}{y_i} \right $	Everything needs to be scaled by the actual value; MAPE is undefined for data points where the value is 0.

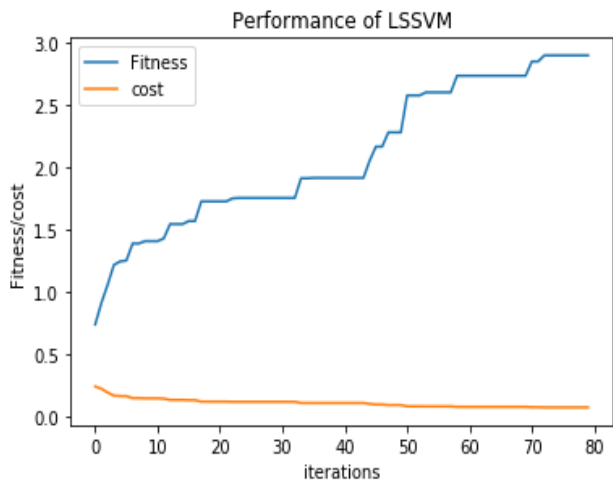


Figure 4. Performance of LSSVM along with ABC

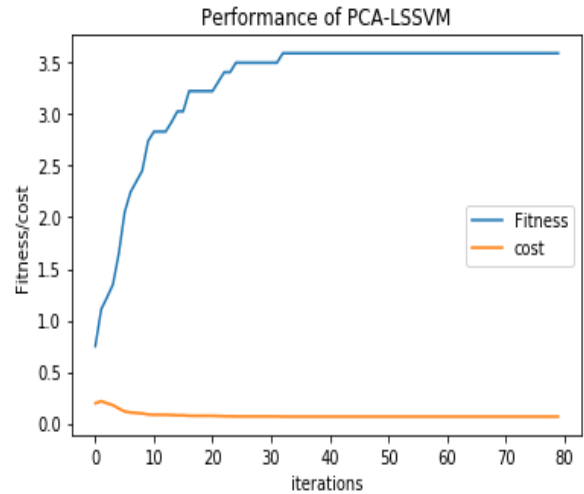


Figure 5. Performance of PCA-LSSVM along with ABC

Table 4. Comparison of accuracy and execution time consumption testing for feature set 1
Acc-Accuracy ET-Execution Time (in milli seconds)

Algorithm	ABC-LSSVM		ABC-Gradient boost		ABC-LSTM	
Metrics/Ticker	Acc	ET	Acc	ET	Acc	ET
AMZN	83.03	128262.359	83.06	122231.382	83.09	103423.638
AAPL	80.05	171432.456	80.84	170224.145	81.08	170129.586
INFY	83.06	112856.648	84.04	111672.248	84.08	110107.381
TCS	84.08	92621.024	85.02	91952.287	85.08	90807.088
MSFT	80.06	156423.032	81.02	155526.345	81.07	153256.538
ORCL	80.26	153960.298	81.65	151582.269	81.72	150422.136

Algorithm	LSSVM		Gradient boost		LSTM	
Metrics /Ticker	Acc	ET	Acc	ET	Acc	ET
AMZN	81.08	131328.025	82.09	122323.392	83.01	111313.626
AAPL	78.13	173327.384	79.27	171032.564	80.23	171696.124
INFY	82.09	113938.318	83.04	112657.784	83.06	111265.318
TCS	84.08	92702.302	85.04	92027.567	85.08	90934.008
MSFT	80.03	160234.518	80.08	164325.765	81.04	160538.274
ORCL	79.92	156173.483	80.64	153859.378	80.88	154452.459

Table 5. Comparison of accuracy and execution time consumption testing for feature set 2
Acc-Accuracy ET-Execution Time (in milli seconds)

Algorithm	ABC-LSSVM		ABC-Gradient boost		ABC-LSTM	
Metrics/Ticker	Acc	ET	Acc	ET	Acc	ET
AMZN	82.04	120328.025	83.01	122231.382	84.02	111313.626
AAPL	80.35	169867.136	81.24	170224.145	82.02	171696.124
INFY	84.02	108568.438	84.08	102762.472	85.02	101265.138
TCS	85.02	91962.254	85.06	91523.217	86.08	90549.893
MSFT	80.08	152309.328	81.04	152321.562	81.09	150216.421
ORCL	80.76	152960.298	81.99	151582.269	82.72	150452.459

Algorithm	LSSVM		Gradient boost		LSTM	
Metrics/Ticker	Acc	ET	Acc	ET	Acc	ET
AMZN	80.02	132453.651	80.04	122435.432	80.08	112334.268
AAPL	78.03	175371.384	79.07	173320.564	80.03	172896.996
INFY	82.04	114713.538	82.06	113763.386	83.02	112137.931
TCS	84.06	92802.024	84.8	92302.6324	85.02	91102.904
MSFT	79.08	168713.538	80.02	166542.765	80.08	165467.538
ORCL	79.84	157371.384	80.23	156985.562	80.36	155560.328

Here the Accuracy values, Execution times for feature set 1 & 2 are tabulated as shown in Table 4 and Table 5. The feature set 1, i.e., with all the considered dimensions, values are tabulated in Table 4. The feature set 2, i.e., the dataset after the dimensional reduction the values are tabulated in Table 5. In

the previous research works, various machine learning algorithms are used to analyze the stock market data for better prediction. In this research work, to improve accuracy an ensemble model is constructed by mixing computation intelligence through naturally inspired to the machine learning

models and also by considering reduced feature set with human opinion, basic and some new technical indicators, other stock values under the same sector. The strengths of the research work are as the number of iterations increased fitness increases and cost decreases. This research work finds that the optimized deep learning model CM_ABC-LSTM with dimensional reduction defeats the considered rest of the models, by considering Accuracy and Execution Time. The limitations of the work are, the stock market prediction is not so easy task as it is sensitive and affected with the so many internal and external factors. In this research work only few of the internal factors and external factors are considered, need to be capable of considering pandemic situations effects also while predicting. By believing the statement “gathering and analyzing more data, generates more useful information”, for better prediction data in all the dimensions need to be gathered. Hence as the future work, a model needs to be trained and used for prediction with accidental data generated in real world.

6. CONCLUSION & FUTURE SCOPE

Even though several strategies were adopted for predicting the stock market, it is observed that stock analysis is always fluctuating and doesn't possess any constant criteria or scenario for the efficient prediction, therefore there is always necessity to develop an efficient strategy for handling the prediction process thus the implementation is always directed towards the construction of the hybrid model with the optimization. From the experimental model, it is observed that the optimized models outperformed all the base, boosted and deep learning models with the best Accuracy value for the prediction of stock price with less Execution time and optimized deep learning model ABC-LSTM with dimensional reduction turned out as superior to all other considered financial models. Therefore, the result shows that the feature set 2 works better than the feature set1 for finding best prediction model in less time.

This work can be further extended by adopting of two or more optimization techniques by incorporating the better processes to build Hybrid models to enhance the performance of prediction model. By using GPUs, different sectors of stock data can be experimented and performance can be analysed.

REFERENCES

- [1] Shah, H., Tairan, N., Garg, H., Ghazali, R. (2018). A quick Gbest guided artificial bee colony algorithm for stock market prices prediction. *Symmetry*, 10(7): 292. <https://doi.org/10.3390/sym10070292>
- [2] Bollen, J., Mao, H.N., Zeng, X.J. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1-8. <https://doi.org/10.1016/j.jocs.2010.12.007>
- [3] Pesaran, M.H., Timmermann, A. (1995). Predictability of stock returns: Robustness and economic significance. *The Journal of Finance*, 50(4): 1201-1228. <https://doi.org/10.1111/j.1540-6261.1995.tb04055.x>
- [4] Deepika, N., NirupamaBhat, M., Paul, P.V. (2019). Influence of the climate change on the stock market. *International Journal of Innovative Technology and Exploring Engineering*.
- [5] Deepika, N., Paul, P.V. (2020). Analyzing effect of political situations on Sensex-Nifty-Rupee - A study based on election results. In *Proceedings of the Third International Conference on Computational Intelligence and Informatics*, Springer, Singapore, pp. 811-819. <https://doi.org/10.1007/978-981-15-1480-7>
- [6] Nalabala, D., Bhat, M.N. (2020). Predicting the e-commerce companies stock with the aid of web advertising via search engine and social media. *Revue d'Intelligence Artificielle*, 34(1): 89-94. <https://doi.org/10.18280/ria.340112>
- [7] Chen, A.S., Leung, M.T., Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan stock index. *Computer & Operations Research*, 30: 901-923. [https://doi.org/10.1016/S0305-0548\(02\)00037-0](https://doi.org/10.1016/S0305-0548(02)00037-0)
- [8] Armano, G., Marchesi, M., Murru, A. (2005). A hybrid genetic-neural architecture for stock indexes forecasting. *Information Sciences*, 170(1): 3-33. <https://doi.org/10.1016/j.ins.2003.03.023>
- [9] Diler, A.İ. (2003). Estimating the direction of the ISE National-100 index by using artificial neural networks error breedback method. *Banks Turkey Capital Markets and Economic Growth: Co Integration and Causality Analysis (1989-2000)*, 81.
- [10] Huang, C.L., Tsai, C.Y. (2009). A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting. *Expert Systems with Applications*, 36(2): 1529-1539. <https://doi.org/10.1016/j.eswa.2007.11.062>
- [11] Kim, K.H. (2003). Dollar exchange rate and stock price: Evidence from multivariate cointegration and error correction model. *Review of Financial Economics*, 12(3): 301-313. [https://doi.org/10.1016/S1058-3300\(03\)00026-0](https://doi.org/10.1016/S1058-3300(03)00026-0)
- [12] Kim, K.J., Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications*, 19(2): 125-132. [https://doi.org/10.1016/S0957-4174\(00\)00027-0](https://doi.org/10.1016/S0957-4174(00)00027-0)
- [13] Kumar, M., Thenmozhi, M. (2007). Support vector machines approach to predict the S&P CNX NIFTY index returns. In *10th Capital Markets Conference*, Indian Institute of Capital Markets Paper. <http://dx.doi.org/10.2139/ssrn.962833>
- [14] Yao, J., Tan, C.L., Poh, H.L. (1999). Neural networks for technical analysis: A study on KLCI. *International Journal of Theoretical and Applied Finance*, 2(2): 221-241. <https://doi.org/10.1142/S0219024999000145>
- [15] Qiu, J.Y., Wang, B., Zhou, C.J. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PIOS ONE*, 15(1): e0234206. <https://doi.org/10.1371/journal.pone.0227222>
- [16] 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.
- [17] Chopra, S., Yadav, D., Chopra, A.N. (2019). Artificial neural networks based Indian stock market price prediction: Before and after demonetization. *International Journal of Swarm Intelligence and Evolutionary Computation*. 8(1): 1000174. <https://doi.org/10.4172/2090-4908.1000174>
- [18] Sedighi, M., Jahangirnia, H., Gharakhani, M., Fard, S.F. (2019). A novel hybrid model for stock price forecasting based on metaheuristics and support vector machine.

- Data, 4(2): 75. <https://doi.org/10.3390/data4020075>
- [19] Ghanbari, M., Arian, H. (2019). Forecasting stock market with support vector regression and butterfly optimization algorithm. arXiv:1905.11462v1 [cs.LG] 27 May 2019.
- [20] Arora, N. (2019). Financial analysis: Stock market prediction using deep learning algorithms. In Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur-India. <http://dx.doi.org/10.2139/ssrn.3358252>
- [21] Gandhmal, D.P., Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. Computer Science Review, 34: 100190. <https://doi.org/10.1016/j.cosrev.2019.08.001>
- [22] Das, S.R., Mishra, D., Rout, M. (2019). Stock market prediction using Firefly algorithm with evolutionary framework optimized feature reduction for OSELM method. Expert Systems with Applications, (4): 100016. <https://doi.org/10.1016/j.eswax.2019.100016>
- [23] Shynkevich, Y., McGinnity, T.M., Coleman, S.A., Belatreche, A., Li, Y.H. (2017). Forecasting price movements using technical indicators: Investigating the impact of varying input window length. Neurocomputing, 264: 71-88. <https://doi.org/10.1016/j.neucom.2016.11.095>
- [24] Jamous, R.A., El Seidy, E., Bayoum, B.I. (2016). A novel efficient forecasting of stock market using particle swarm optimization with center of mass based technique. Int. J. Adv. Comput. Sci. Appl., 7(4): 342-347. <https://doi.org/10.14569/IJACSA.2016.070445>
- [25] Hegazy, O., Soliman, O.S., Salam, M.A. (2014). A machine learning model for stock market prediction. International Journal of Computer Science and Telecommunications, 4(12): 17-23.
- [26] Badar, A., Umre, B., Junghare, A. (2014). Study of artificial intelligence optimization techniques applied to active power loss minimization. IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), pp. 39-45.