DESIGN OF HYBRID CLASSIFIERS FOR TIME SERIES PREDICTION USING STOCK MARKET DATA

Thesis submitted in fulfillment of the requirements for the Degree of

DOCTOR OF PHILOSOPHY

By

RAGHAVENDRA KUMAR

Enrolment No: 186206

Department of Computer Science Engineering and Information Technology

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

Waknaghat, Solan-173234, Himachal Pradesh, INDIA

March, 2022

SUPERVISOR'S CERTIFICATE

This is to certify that the work in the thesis entitled **"Design of Hybrid Classifiers for Time Series Prediction using Stock Market Data"** submitted by **Raghavendra Kumar** is a record of an original research work carried out by him under my supervision and guidance in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy in Computer Science and Engineering in the Department of Computer Science & Engineering and Information Technology, **Jaypee University of Information Technology, Waknaghat, INDIA.** Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

(Signature of Supervisor-I)	(Signature of Supervisor-II)
Dr. Pardeep Kumar	Dr. Yugal Kumar
Associate Professor	Assistant Professor (SG)

Department of Computer Science & Engineering and Information Technology
Jaypee University of Information Technology,
Waknaghat -173234, INDIA.

Date: 29-03-2022

DECLARATION BY THE SCHOLAR

I hereby declare that the work reported in the Ph.D. thesis entitled "**Design of Hybrid Classifiers for Time Series Prediction using Stock Market Data**" submitted at **Jaypee University of Information Technology, Waknaghat, INDIA** is an authentic record of my work carried out under the supervision and guidance of **Dr. Pardeep Kumar and Dr. Yugal Kumar**. I have not submitted this work elsewhere for any other degree or diploma. I am fully responsible for the contents of my Ph.D. thesis.

(Signature of the Scholar)

Raghavendra Kumar

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat -173234, INDIA.

Date: 29-03-2022

ACKNOWLEDGEMENTS

First of all, I would like to express my deep sense of respect and gratitude towards my guide Dr. Pardeep Kumar and Dr. Yugal Kumar in the Department of Computer Science & Engineering and Information Technology, for their support throughout this research work. I want to thank them for introducing me to the field of Time Series Analysis and giving me the opportunity to work under them. Without their invaluable advice and assistance, it would not have been possible for me to complete this thesis. I am greatly indebted to them for their constant encouragement and invaluable advice in every aspect of my academic life. I consider it my good fortune to have got an opportunity to work with such wonderful persons.

I would like to express my gratitude to our Honourable Vice Chancellor Prof. (Dr.) Rajendra Kumar Sharma and Dean Academic & Research Prof. (Dr.) Ashok Kumar Gupta to promote the research and facilitate resources in the institution. I would also like to thank my doctoral committee, Head of Department Prof. (Dr.) Vivek Kumar Sehgal, Dr. Jagpreet Sidhu and Dr. Rajiv Kumar for their valuable feedback and critical reviews during presentation and time to time help.

I am also very grateful for the support received from the JUIT, Waknaghat. In particular, I thank all the teaching, library, and non-teaching staff at the Department of computer Science & Engineering and Information Technology, JUIT, who has been extremely helpful at numerous occasions.

This note of acknowledgement will always be incomplete without the mention of my parents, my wife, my kids, my beloved & friends for encouraging me at the first place and supporting me spiritually throughout my life.

Needless to say, errors and omissions are solely mine.

Date: 29th, March, 2022 (Raghavendra Kumar)

ABSTRACT

There are a variety of theories, procedures, and approaches that make time series prediction more accurate, demanding, and exciting for researchers. Dynamic and volatile nature of stock data brings it into a sequential data series known as time series data. Stipulated time interval and stochastic behavior in data pattern makes stock market as a best fit use case for time series analysis (TSA). Deep learning (DL) approaches have made tremendous progress in anticipating stock market patterns, and they continue to pique the interest of market traders and investors. The concept of hidden layer makes DL models best fit for any time series data like the stock market. High level validity of accuracy in data brings LSTM as the most preferable approach. LSTM is a complex computational network that works on the concept of RNN. RNN and its variants are also well-known models that handle missing values and noisy data available in a dataset. LSTM is one of the best ANN topologies that deals with function mapping problems and non-linear dynamics. LSTM handles long term dependencies with its gate enabled framework. Nature inspired and evolutionary algorithms help to optimize the selection of parameters and provide stability between performance and complexity of the discussed models. Such algorithms like Artificial Bee Colony (ABC) bring significant improvements in stock forecasting accuracy.

Fusion is a state-of-the-art technique to observe the behavioral pattern from time series data. Fusion models efficiently and effectively interpret both linear and non-linear patterns that are the constraints of an individual model due to feature limitations. In this work, a three-stage fusion is proposed to handle time series data and improve stock market forecasting accuracy. In the first phase of fusion, stock market inputs that are constituted with historical data and market sentiments of the targeted stock are pooled along with established technical indicators of the stock market. Market sentiments are examined through sentiment polarity index using big data platform Hadoop. The first phase accomplishes the heterogeneous information for more accurate stock market prediction as first objective. The second phase addresses the parameter tuning issue of existing model for improving prediction accuracy. It considers ARIMA as a linear model. Time series ARIMA is optimized with the fusion of enhanced ABC and DE algorithms. This hybridization of DE and ABC preserves the steadiness between the exploitation and exploration of the hyper parameters of proposed model.

The third phase addresses the linear and nonlinear issues of time series data using hybrid classifiers. The hybrid classifier is designed with LSTM. The proposed model DE-ABC-LSTM-ARIMA induces the superior accuracy as time series classifier. In this work, experiments are performed on established and diversified reported historical datasets Apple Inc. (AAPL), Intel Corporation (INTL) and Microsoft Corporation (MSFT) are all part of the IT sector at the NASDAQ GS, as well as the India's NSE & BSE Oil & Refineries sector datasets. The proposed fusion model DE-ABC-LSTM-ARIMA outperformed the benchmark models used in this work.

Keywords: Classifier, Model, Time Series Analysis, Stock Marketplace, Prediction, Hybridization, Deep Learning, Long Short Term Memory (LSTM), Moving Average (MA), Linear Model, ARIMA, Evolutionary Algorithm, Artificial Bee Colony (ABC), Information Fusion, Differential Evolutional (DE), Algorithm Fusion, Model Fusion, Nonlinear Model, Stock Index, Stock Market Data, Dataset, Auto Regressive (AR).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATION

ACF	Auto-Correlated Function
ADF	Augmented Dickey Fuller
AR	Auto Regressive
ARIMA	Auto Regressive Integrated Moving Average
AIC	Akaike Information Center
ANEW	Affective Norms for English Words
AM	Attention Mechanism
ABC	Artificial Bee Colony
Bi-LSTM	Bidirectional Long Short-Term Memory
BA	Bat Algorithm
BSE	Bombay Stock Exchange
CS	Cuckoo Search
CEQ	Certainty-Equivalent
DE	Differential Evolution
DBN	Dynamic Bayesian Network
DJIA	Dow Jones Industrial Average
FPA	Flower Pollination Algorithm
GIWC	General Inquirer Word Count
GA	Genetic Algorithms
hABCDE	hybrid Artificial Bee Colony Differential Evolution
HDFS	Hadoop Distributed File System
HiveQL	Hive Query Language
IACF	Inverse Autocorrelation Function
LIWC	Lexicons Linguistic Inquirer Word Count
LSTM	Long Short-Term Memory
MA	Moving Average
MAPE	Mean Absolute Percentage Error

MLP	Multi-Layer Perceptron
MSE	Mean Square Error
NYSE	New York Stock Exchange
NSE	National Stock Exchange
PACF	Partial Auto-Correlated Function
PP	Philippe-Perron
PSO	Particle Swarm Optimization
RBM	Rule Based Model
RMSE	Root Mean Square Error
RKELM	Robust Kernel based Extreme Learning Machine
SARIMA	Seasonal Auto Regressive Integrated Moving Average
SWN	SentiWordNet
SCN	SenticNet
TSA	Time Series Analysis
VMD	Variational Mode Decomposition

CHAPTER 1
INTRODUCTION

1.1 Introduction

A time series is a collection of observations in chronological order that are collected in a continuous manner [1]. These data are frequently interdependent, and time series analysis is the study of this temporal reliance. Time series data, such as air quality data, weather forecasts, and stock market prediction, are huge, dynamic, and have a high dimensionality [2]. Time series analysis aids in the counting of predetermined time intervals as well as the observation of data series patterns and stochastic behavior. Stock market statistics can clearly reveal this type of activity. As a result, several studies have been presented to solve these difficulties, each with its own model, methodology, and approach.

Due to the vast number of financial data and the progress of digitalization, economic globalization is fast expanding by leaps and bounds. The research community may use this huge data to experiment with new ways and directions to help the global financial markets. As a time series data, stock market data contains one of the most intricate behavioral patterns [3]. Long-term patterns and erratic market movements make it difficult to achieve optimal performance from any model. Prediction accuracy is a logical performance measure for machine learning models in the modern era. Individual models, for example, are said to have failed due to their incapacity to read all of the patterns in time series data. Because a single model is insufficient to identify the data creation process or the features of the time series, many models are used. Combining homogeneous models leads to lower generalization variance, according to empirical data from previous studies [4]. As a result, in the contemporary period, integrating heterogeneous features and constructing fusion models are state-of-the-art procedures. Such fusion models have not only demonstrated greater performance, but they may also aid market traders and investors in developing strategies [5].

Information fusion: integrating input factors for accurate stock prediction is part of a three-stage fusion method in this work. Market-driven technical indicators derived from historical data are combined with a sentiment polarity index derived from market news and blogs in the first phase of hybridization. In this experiment, the goal of information fusion is to merge numerous components of the stock market.

The stock market is influenced by a variety of factors, including both direct and indirect influences. The stock market's primary factors are depicted in Table 1.1. Direct impact elements include global economic conditions, investor sentiments, and market history, while indirect impact aspects include disasters, pandemic circumstances, and political willingness. Because current studies of individual models do not take all of these elements into account as combined inputs to predicting models.

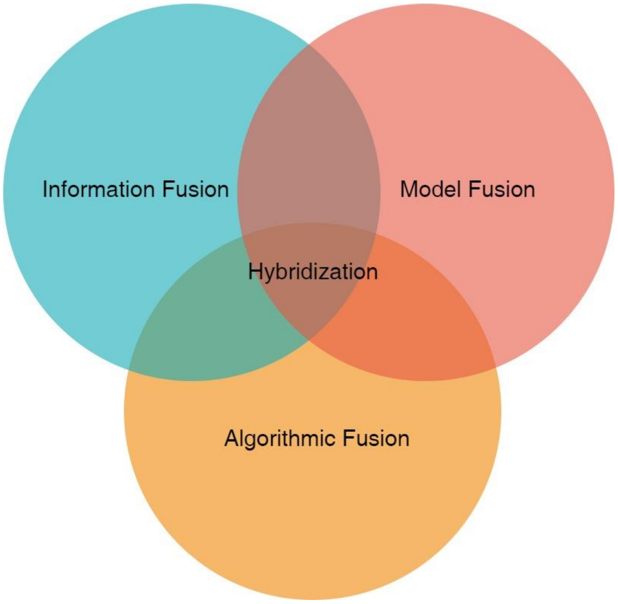

Figure 1.1: Information, Model, and Algorithmic Fusion

Significantly, the social media platform brings up many dimensions connected to news stories, blogs, and Twitter analysis to impact stock market investors' purchasing and selling decisions. Market attitudes influence the financial market, and such knowledge is particularly useful in making decisions for traders and investors. Various computational approaches are used to efficiently interpret public opinions and use them in stock forecasting [6-8]. As a result of stock market restrictions, a market continually seeks to strike a balance between efficient and inefficient markets. As a result, we fairly regard it to be a blend of both sectors, because all events and actions cannot instantaneously reflect market price and related information. Qualitative (i.e. product and strategy) and quantitative (i.e. financial data) information can be found (i.e. stock price). Because prices respond only to market-driven information, both qualitative and quantitative information reflect the stock price, regardless of whether it is true or untrue [9]. In order to assure investor perception, data from direct and indirect components (Table 1.1) is merged and mapped with time stamps depending on date in this article to maintain market efficiency.

Existing research uses two approaches to anticipate stock trends and prices. The first method uses prior stock price fluctuations as time series projections. This is often referred to as the traditional technique, and it is used in a variety of existing models [10-14]. However, due of the high dimensionality and noise availability of such models, they do not produce adequate results. On the other hand, using technical indicators to forecast the market has shown to be a successful strategy in the past. The use of technical indicators improves prediction model resilience and imprecision adjustment [15].

Table 1.1: Factors Affecting Stock Market

Direct Factors affecting Stock Market	Indirect Factors affecting Stock Market
The global economy	Decisions by the government
History of the market	Policies in the public and private sectors
News on the budget and the return on investment	Willingness to act politically
Market developments	Events involving public safety
Investors' feelings	Activism by terrorists
Other market commodities' impact	Natural disasters and epidemics

Financial professionals commonly support this approach due to its accuracy. However, it may not be recommended for newbie and ordinary investors due to complex architecture of technical indicators. Because each and every stock has its own features, choosing technical indicators is a huge challenge. A feature that is suitable for one stock's value, time zone, or market dynamics may not be suitable for another stock's value, time zone, or market dynamics. As a result, based on earlier work [16], a feature selection method is examined using ABC for the technical indicators. These indicators hold the nerves of the stock market and cover all the dimensions. The suggested model and its projected prediction outcomes are strengthened by including all aspects of the market. In the study, the ABC method is also used to choose hyperparameters of LSTM unit. Stock market forecasts are influenced by a multitude of factors. The two major groups of these elements are direct and indirect effect variables. The first direct influencing variables are the international economy, industry history, budgeting news, and market and investor sentiments. Secondary, indirect effect variables include government acts, government and non - governmental policies, terrorist operations, and environmental catastrophes.

1.2 Motivation

Time series analysis has garnered a lot of attention from the academic community in recent years, and scholars have tackled it from both a practical and theoretical basis. The stock marketplace is known as constantly changing market that can be followed throughout time. Stock market data is thought to be more complex than other types of statistical data. The exponential expansion of financial data has revealed the inefficiency of traditional solutions in terms of computational efficiency. Since predicting accuracy is a generally accepted time series analysis criterion. As a result, developing and testing such models is more difficult. Various solo, ensemble, and hybrid models have been presented in recent years to increase predicting accuracy. However, due to feature constraints, each particular model's performance is inadequate. However, by combining the error series in hybrid models, higher performance can be achieved. Because individual models fail to interpret both linear and non-linear patterns equally well, combining the results of models with different behavioral patterns should yield better results. The pros of these hybrid models are listed as.

- Address linear as well as nonlinear behavioral data patterns.
- Able to maintain the temporal relationship with sequential data series.
- Integrates heterogeneous information sources as input dataset

- In time series prediction, hybridizing algorithms handle diverse aspect.
- Solutions for multi objective optimization problems like time series forecasting.

Apart from advantages, it is also observed that several shortcomings are associated with these hybrid models.

- Exploration and exploitation processes are out of balance.
- Slow convergence rate
- Authenticity of social media data and news of stock market.
- Indirect factors affect stock market

1.3 Research Gaps

- Lack of works reported on inclusion of heterogeneous information as input parameters for stock market prediction.
- Optimal parameter tuning of existing algorithms also a considerable area of research for improving stock market accuracy.
- Linear and nonlinear behavioral patterns in time series data have significant impact on prediction accuracy.
- Improper addressing of exploration and exploitation issues of optimization algorithms.
- Consider diverse stock market issues like accuracy, trends and volatility to enhance stock market prediction accuracy.

1.4 Objectives

Based on motivation, it is highlighted that combining different features, fusion of inputs and designing hybrid models are state of the art techniques in the current era. The hybrid models are not only proven in terms of superior performance but also potentially assist market traders and investors to make their financial decisions. The objectives of this thesis are highlighted as

- To consider heterogeneous information for more accurate stock market prediction.
- To address the parameter tuning issue of existing model for improving prediction accuracy.

- To address linear and nonlinear issues of time series data using hybrid classifiers.

1.5 Thesis Organization

Organization of the thesis work is as follow.

Chapter 1: This chapter provides the introduction of stock market time series analysis along with the motivation, research gaps, objectives of the research work.

Chapter 2: This chapter provides the literature review on the stock market time series analysis motivation and objectives of the research works.

Chapter 3: It discusses the first objective of this thesis i.e., to consider heterogeneous information for more accurate stock market prediction. Additionally, this chapter examines the technical indicators of stock market through ABC algorithm.

Chapter 4: It considers the second objective of this thesis i.e., To address the parameter tuning issue of existing model for improving prediction accuracy. In this chapter, an improved ABC algorithm using DE algorithm is examined for the hyperparameter selection.

Chapter 5: This chapter examines the third objective of this thesis i.e., To address linear and nonlinear issues of time series data using hybrid classifiers.

Chapter 6: This chapter concludes the thesis work and enlighten the future perspectives.

CHAPTER 2
LITERATURE REVIEW

2.1 Review of Existing Information Fusion

Time series prediction is increasingly accurate, difficult, and intriguing for academics thanks to a variety of theories, processes, and approaches. Statistical methods (ARIMA, AR, SARIMA), ML algorithms (LR, RAF, SVR) and DL models (ANN, BPNN, RNN) are widely accepted approaches [13-18]. Support Vector Regression (SVR) and Deep Neural Networks (DNN) are examples of regularly used computational techniques [19-24]. Optimization approaches such as ABC, PSO, DE, and GA have been used to solve most stock market parameter and feature selection challenges [27, 44, 51-53]. According to a recent study, there are two types of time series forecasting models: linear and nonlinear. These DL models have made tremendous progress in anticipating stock market patterns, and they continue to pique the interest of academics. LSTM has been frequently used for prediction in the realm of TSA in recent years [25]. LSTM is a variant of RNN holds prolonged interdependence. LSTM is a cutting-edge technique in the field of TSA. This complex structure was created to solve the problem of long-term reliance previously produced data series [26, 27]. The ABC is deemed to be the most successful benchmark algorithm as per existing research. ABC has a number of advantages over its competitors, including better memory use, local search, and quality optimal solutions. To improve forecasting efficacy, ABC maintains a balance between the prediction model's performance and its complexity.

Karaboga et al. proposed it in 2005 [27] as a nature-inspired algorithm. The algorithm was inspired by the nature driven activity of bees and their hive. The algorithm is regarded as a typical intelligent nature-inspired optimization approach [28, 29].

The choice of ABC as an optimization algorithm in this work was motivated by its ease of use and superior memory consumption. The ABC is a proven local search algorithm gives quality solutions to tackle optimization issues of the TSA [30, 31].

Other direct effect elements, in addition to the stock market index, are later integrated into prediction models. From time to time, a range of extant works focused on various facets of the stock market. Integration of numerous aspects of sentiments and the creation of fusion has proven to be difficult. These concerns of sentiment integration are addressed by the suggested integration. Figure 2.1 shows the information fusion used in this study.

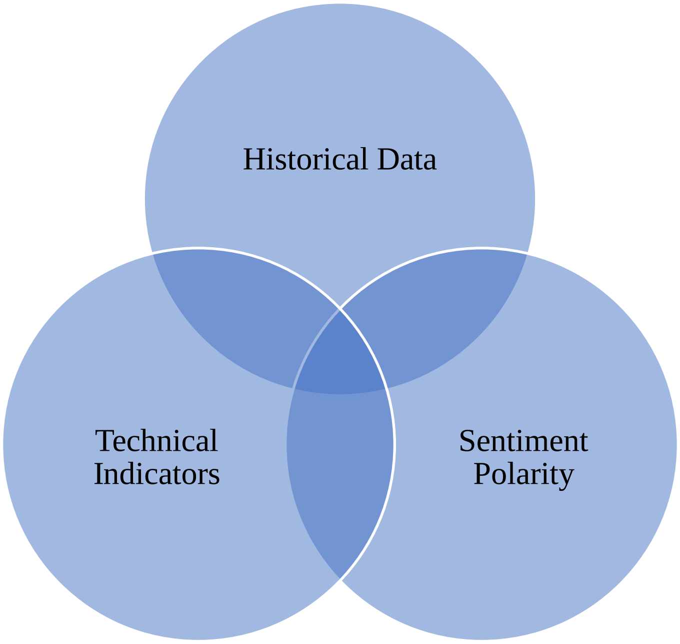

Figure 2.1: Information Fusion

An ensemble model based on the radial basis function (RBF) and metaheuristic algorithms was presented by Karathanasopoulos et al. In terms of prediction accuracy, the suggested ensemble model RBF-PSO outperform individual models [31].

Feng et al. utilized the EMD model and combine with Factorization Machine to achieve market accuracy. The EMD2FNN model examined over closing price of three diverse stock market datasets. The obtained result of EMD2FNN model brings the superior accuracy as compare to benchmark referred in the sstudy [32].

Göçken et al. looked on hybrid model based on numerous RNN and HS. The Jordan Recurrent Neural Network (JRNN) and HS models outperformed their competitors [33]. Hybrid neural network models are critical for achieving the optimum accuracy.

Yang et al. suggested a hybrid stock selection model that integrates ELM and SVR. The proposed hybrid model is optimized with PSO. The forecasting results outperformed the AR and SharpeRatio [34].

The RKELM with VMD is presented by Bisoi et al. The performance measurements MAE, RMSE, and MAPE are used to verify the effectiveness of the suggested model. The findings are compared to those of number of hybrid models [35].

A recent study found that market sentiments impact investment decisions, and that these judgments are not always rational. In stock forecasting, text sentiment analysis is crucial. In the literature, text sentiments have been divided into two categories: processes of learning techniques and ML and DL techniques. In lexical analysis methods dictionaries (SentiWordNet 3.0, Loughran and McDonald's financial lexicon, Vader and Harvard-IV, SenticNet 5 dictionary) recognize sentiments independently and then combine the positive and negative ones [36-38].

To take use of the discriminative features, Huang et al. suggested a unique picture text sentiment analysis model. The intrinsic correlation is found between semantics and graphical contents. It is identified as a driven source of sentiment analysis [36].

Dwivedi et al. proposed a RBM to measure the market sentiments. Sentiment lexicon method overwhelms the other counterparts including LIWC, SWN, SCN and ANEW. RBM does not required to handle elaborate specifications or complicated computations. In the case of a small dataset, our model outperforms others [38].

Emotional proclivity is important for accurate prediction. Jin et al. demonstrated that EMD improves prediction accuracy while lowering time delay. The attention mechanism in the revised LSTM model prioritizes crucial information [39].

Rodrigues et al. adopted Apache Hive to compute complicated multidimensional data. The proposed methodology emphasis on real-time tweets and storing them into Hive engine. Finally, when compared to the benchmark models, the findings were found to be satisfactory [40].

The vast number of reviews on the Twitter microblogging network, which are generated utilizing the map reduction method in a distributed setting, was emphasized by Skuza et al. [41]. The big data map reduction technique executed the difficult job efficiently using the Hadoop platform. With its built-in mapper and reducer methods, Map Reduction makes it simple to manipulate partitioned data [42].

Yang et al. demonstrated that big data approaches such as Hadoop and Spark are globally acknowledged techniques for measuring sentiment polarity in a cost-effective and reliable manner to deal with economic domain volatility [43].

2.2 Review of Existing Algorithmic Fusion

The ABC (Artificial Bee Colony Algorithm) is based on bee swarm foraging behavior. ABC is recognized as a potential solution to optimization issues that arise in the stock market. The ABC method [27, 28] solves multidimensional problems in the stock market. Hyperparameter optimization and feature selection are solved using multidimensional approach.

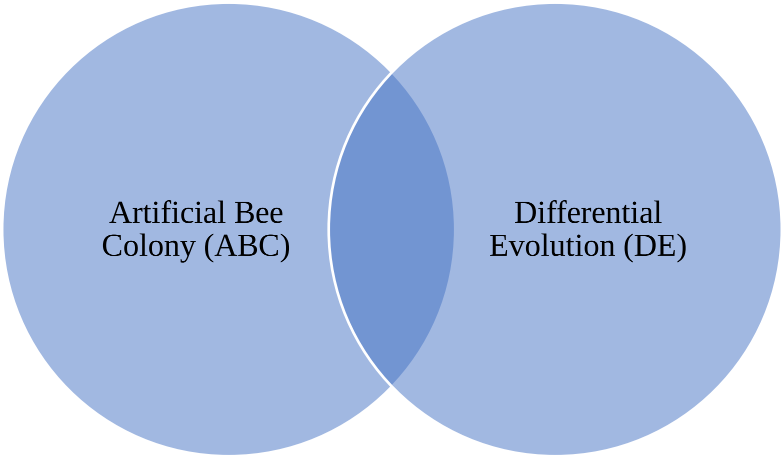

Figure 2.2: Algorithmic Fusion

Storn et al. [44] introduced the most successful algorithm, Differential Evolution (DE). DE can help with global optimization challenges in time series models like stock market forecasting. The standard DE is made up of four operations that are divided into numerous phases. Initialization of chromosome length (D), gene value range for operations (U_{min}, U_{max}), considered crossover rate (CR), discovered mutation factor (F), and population size (N).

ABC is notable for its ability to solve multi-objective problems with global exploration challenges. However, due to local exploitation, it is usually documented in local search efficiency [45-47]. Figure 2.1 shows the algorithmic fusion used in this study.

DE improves the local search process by modifying the ABC’s onlooker bee phase. Using hybrid search tactics, the solution equation improves search efficiency. It was inspired by DE's current-to-best/1 mode.

Multi-objective DE-current-to-best enhances convergence and uses numerical functions to tackle real-time optimization problems. Several prior studies are also used to support the hybrid algorithm's complexity.

Xiang et al. introduced a unique hABCDE for tackling optimization problems, which they validated using twenty benchmark functions (f1-f20). The achievement of the outcome ensured that the hABCDE outperformed other solutions [48].

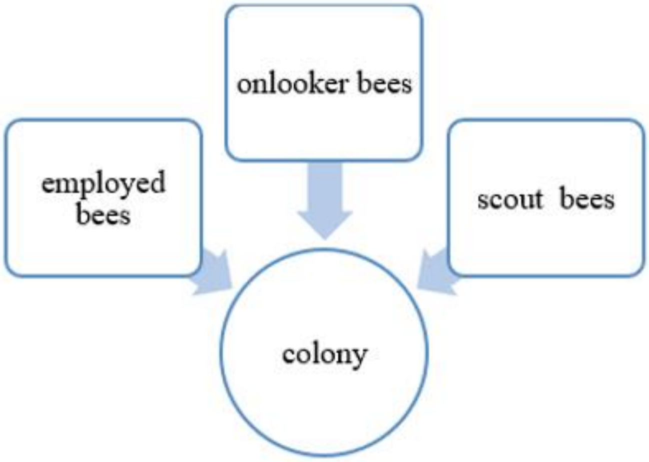

Figure 2.3: Artificial Bee Colony (ABC)

In another work, Zorarpac et al. examine the machine learning repository known as UCI to assess the performance of the suggested feature selection hybrid algorithm. The outcomes of the experiments outperformed the conventional ABC and standard DE algorithms [49].

Similarly, Jadon et al. put the HABCDE, another hybrid algorithm to the test on a set of 20 benchmark functions *(f1-f20).* The proposed work was also carried out on four real-world optimization situations *(f21-f24).* The obtained result confirms that the HABCDE outperforms its individual base algorithms in terms of accuracy, convergence speed, stability, and resilience [50].

Nature-inspired and evolutionary algorithms aid in optimizing parameter selection and maintaining a balance between performance and complexity in the models under consideration. ABC, ACO, BA, PSO, FPA, GA, CS, HS, DE, and others have shown to improve stock forecasting accuracy significantly [51-56].

DE is used to optimize the hyperparameters of the suggested hybrid prediction model, and it is compared to GA and PSO in this research. On the one hand, the GA is a natural-evolution-inspired evolutionary algorithm. It improves the performance of complicated problems with a huge search space. Selection, crossover, and mutation operators are used in the population of individuals and natural genes. A chromosome functions as a problem-solving feature encoded as a binary string [57].

PSO, on the other hand, is used to optimize nonlinear data. It's categorized as a stochastic optimization method. PSO is based on the social behavior of flocks of birds or swarms of fish. PSO optimization is accomplished by allocating memory in the search space to individuals based on past achievement.

DE is an evolutionary computer algorithm that is fast to convergence, adaptable, and robust (Price & Storm, 1995). Initialization, mutation, crossover, and selection are the four phases of DE's iterative process.

Recent research into time series data has expanded the scope of stock market forecasting. Table 1.2 summarizes recent research and investigations. To date, statistical and computational methods have been used to get the best stock prediction outcomes.

For three separate datasets, Asghar et al. suggested a model based on multiple regressions, namely the KSE 100-index, Lucky Cement Stock, and Engro Fertilizer Limited. Accuracy, precision, and recall are used as performance indicators in this study. Multiple regression has a much higher prediction accuracy than ARIMA and Random Forest [64].

Hoseinzade et al. used two deep CNN frameworks for stock market index prediction using 2D-CNNpred and 3D-CNNpred. The classifiers are validated using Sharpe ratio, and CEQ return and compared to CNN-cor [66].

Wenjie et al. propose a CNN-BiLSTM-AM hybrid model that combines CNN for input set feature extraction, BiLSTM for forecasting Shanghai Composite Index closing price over 1000 trading days, and Attention Mechanism (AM) to influence the feature states of Shanghai Composite Index stock at different time intervals. As evaluated by RMSE and MAE, model accuracy is superior to benchmark approaches MLP, CNN, LSTM, BiLSTM, and other hybrid combinations like CNN-LSTM, CNN-BiLSTM, BiLSTM-AM, and CNN-BiLSTM [67-68].

Chenglin et al. use a basic support vector machine and a cumulative auto regressive moving average to overcome the drawbacks of stock forecasting models. To predict stock price trends, the proposed technology, known as ARI-MA-LS-SVM, employs least squares support vector machines. ARI-MA-LS-SVM is a combined technique that produces better outcomes than individual models. For market investors, the proposed approach claims global market applicability and stability feasibility [69].

Ioannis et al. test CNN-LSTM fusion for gold price and movement in USD from January 2014 to April 2018, using data from Yahoo Finance. The proposed model makes use of CNN's capabilities as well as the effectiveness of LSTM layers. Different combinations of LSTM layers and different sets of hyperparameters are compared to the hybrid model CNN-LSTM [70].

To diagnose malaria, Kumar et al. suggested an ensemble learning approach. To validate the classification accuracy, the proposed model used a convolution neural network (CNN) to validate the classification matrices of malaria patients [72].

In previous work, Thomas et al. use a genetic algorithm to solve a one-dimensional cutting stock problem. By imposing a penalty function on fitness value, the proposed work improves the column generation technique. The proposed model was able to manage random behavior with a higher rate of convergence [73].

The TSA is one of the most dynamic, demonstrating both linear and nonlinear characteristics. When ABC-DE is integrated, the ARIMA model has a greater probability of acquiring the right hyperparameter tuning.

The proposed work aids in the optimization of the proposed model's hyperparameters. This hybridization maintains a balance between the utilization and investigation of the suggested model's hyper parameters.

2.3 Review of Existing Model Fusion

This work integrates the heterogeneous models and generates model fusion in the third phase. On the one hand, utilizing Auto Regressive Integrated Moving Average (ARIMA), linear aspects of stock market data are thoroughly investigated. LSTM, on the other hand, is used to detect nonlinear patterns in stock market time series.

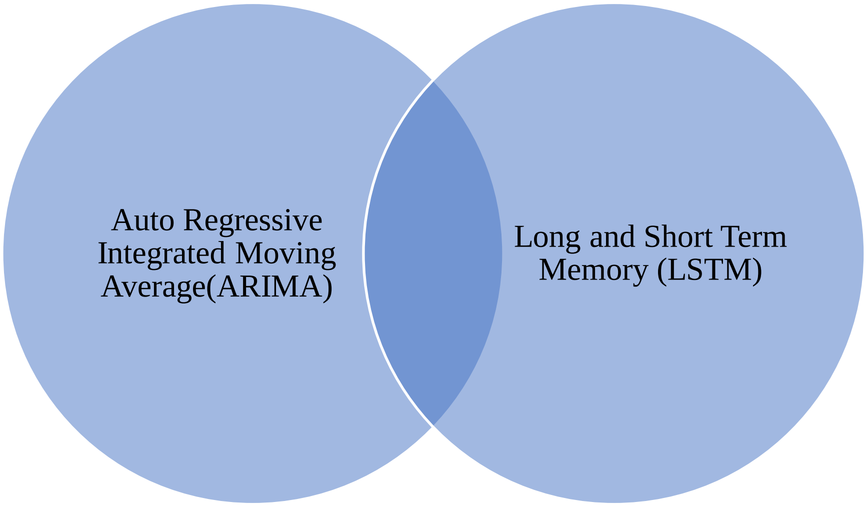
Auto Regressive Integrated Moving Average(ARIMA)
Long and Short Term Memory (LSTM)

Figure 2.4: Model Fusion

The information fetched from the previous link is correlated and made available for the next node of the network depicted in Figure1.2 by the Recurrent Neural Network (RNN) theory. RNN, on the other hand, fails to place the sequential data series in context. Hochreiter et al. developed LSTM to address the problem. The LSTM unit's goal is to preserve long-term dependencies while also correlating earlier inherent time series. Due to its memory structure, a LSTM unit is made up of states of cells [25]. Box and Jenkins established ARIMA linear model for analyzing time series data [1]. ARIMA is made up of two components: Auto-Regressive (AR), which analyses prior data to model. The other part is known as Moving Average (MA), which maintains control over deafening data from earlier instances. Pai et al. integrated ARIMA and SVM for accurate stock market forecasting. This integration brings significant accuracy as error metrics on Shanghai Share with ARIMA (0,1,0) [74].

In similar work, Babu et al. and Domingos et al. investigate ARIMA and GARCH and MLP-SVR respectively to address data patterns of TSA. The proposed hybridization is examined on diversified datasets to achieve superior outcomes [75, 76].

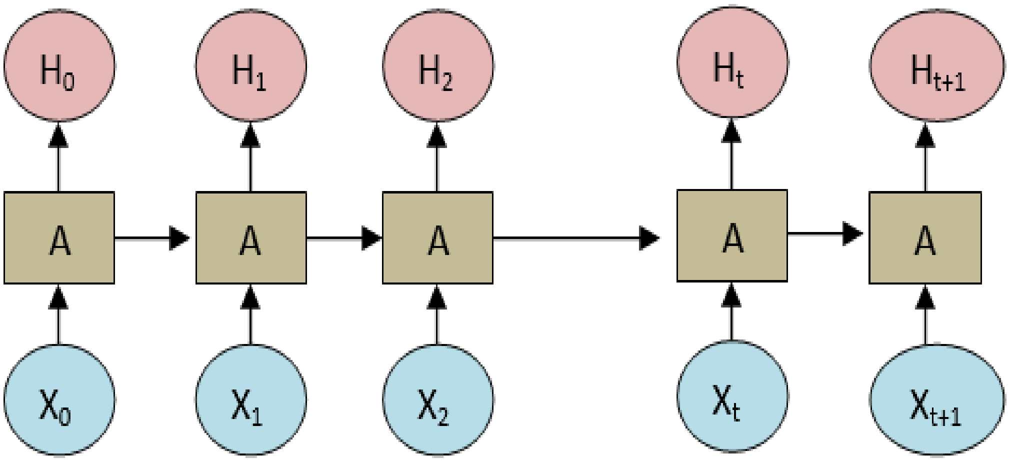

Figure 2.5: Unfolding RNN Architecture

Khashei et al. show how to increase forecasting accuracy using a hybrid model comprising linear (ARIMA) and nonlinear (ANN) models. For diverse datasets, the proposed model outperforms independent benchmark models [77, 78].

Zhou used the ARIMA with an AM-LSTM to improve online traffic TSA. AM between two LSTM layers improves the impact on hyper parameters. For additional testing, ARIMA (1,0,0) and (1,1,2) were chosen [79].

Vantuch et al. suggested a GA-ARIMA model that uses the GA and PSO to optimize ARIMA parameters. The proposed model is tested on a stock market dataset that is stationary. The GA-

ARIMA model is used to derive the parameters p, d, and q (12,2,8), as well as the lowest AIC and BIC values [80].

Musdholifah et al. take a different approach to the evolutionary hybrid model, employing the FA to extract the lowest AIC values among all the combinations tested in experiments. The proposed model brings superior accuracy while optimized with FA algorithm [81].

The ARIMA model is compared to classic ANN, such as MLP, and Neuro Fuzzy Network (NFN) by Ballini et al. This subject was investigated by the Brazilian Stock Exchange. The proposed ARIMA model is proven over its counterpart model MLP and NFN [82].

Wang develops an ARIMA model to estimate Taiwan Stock Exchange mid-term price trends. The ARIMA (1,2,1) model is used to evaluate ACF and PACF displays. The proposed model is based on an RNN with features derived from the ARIMA (1,2,1) model [83].

The existing research demonstrates a variety of approaches to time series analysis. Specifically, DE-ABC-ARIMA preserves the data trends as well as forecasting accuracy. The applicability of the ARMA-GARCH was investigated by Tang et al. in stock price prediction [84].

Duan et al. developed an integrated autoregressive DBN named as AR-DBN based on a which improved stock market volatility predictability. The proposed model added AR in the dynamic Bayesian network using adjacent observed variables. Outcome of the proposed work proved that model inferred the fluctuations of the market trends [85, 86].

In [87], the authors combined the traditional ARMA with SVM to get benefits in TSA, resulting increased the descriptive power. Traditional time-series analysis, on the other hand, relied on a relationship was better suited to sequences with stable patterns and rules, making it inappropriate for increasingly complicated nonlinear correlations.

Furthermore, there are numerous influencing aspects in the stock market, each of which has a complex impact that is neglected by simple time-series analysis methods, making prediction less successful. LSTM is a sort of time loop neural network that excels at digesting and forecasting crucial events in time series with long intervals and delays.

The traditional feedforward neural network, typified by CNN, performs brilliantly in classification tasks but is incapable of managing the complex time correlation between data in the field of deep learning.

Xingjian et al., Ma et al. and Ding et al. [88-90] applied various layered LSTM on stock market forecasting. As outcomes, efforts are made to achieve the forecasting accuracy and stability and had proven significant contribution in TSA.

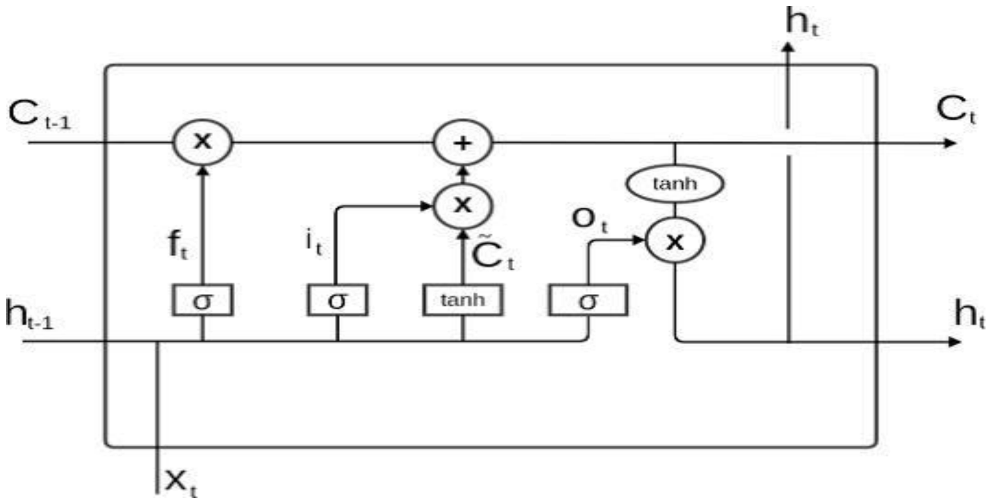

Figure 2.6: LSTM Unit by Hochreiter- Schmidhuber [25]

It can effortlessly keep track of and store past data while connecting to a new input set. ABC increases the LSTM unit's parameter selection, which has an impact on the model's correctness and performance. To sustain the flow of information, the LSTM unit's execution is separated into three portions using logical gates. The LSTM unit's input gate (I_t) is meant to store the essential information using sigmoid activation function about the cell state. This cell (C_t) combined the input of current cell ((I_t)) with the previous output value (C_{t-1}). In 2014 Yoshihara et al. employed RNN-RBM and put information as an input variable and anticipated market trends to explore the temporal consequences. In this work a recurrent deep neural network is validated on Nikkei 225. The pattern mining is applied to extract the features from massive amount of dataset [92]. Saad et al. in 2008 explored three neural network models for stock trend prediction. They applied conjugate gradient with time delay and probabilistic neural networks. RNN outperformed the rest of the models [93].

Chen et al. applied one of the search strategies, the particle swarm optimization (PSO) algorithm, and optimize various parameters of each model in 2005. This study considered the NASDAQ 100, NIFTY index, and S&P CNX values. The TSA fuzzy system is demonstrated to predict chaotic behavior of stock market [94]. Yu et al. employed SVM and PCA to extract financial time series for a stock selection system. The findings of the experiment demonstrated that PCA–SVM-selected stocks outperformed other benchmarks in terms of return in 2014 [95].

Chen et al. focused on feature weighted SVM with K-nearest neighbor to predict stock indices. The hybrid framework is designed using feature weighted SVM and K-nearest neighbor methods. This study is done in 2017 considers information gain into account for each attribute as well as its relative relevance.[96]. In other feature extraction deep learning work Chong et al. in 2017, employed three unsupervised approaches like RBM, PCA, and AE to forecast future market trends. The proposed model is applied on high-frequency intraday returns. DNN extract stock's metadata from the residuals taken from the AR model to improve forecasting accuracy [100].

In order continue to review the recent works, The Deep Learning-Based Model to Predict Corporate Performance Considering Technical Capability [101] was proposed by Lee et al. (2017). The proposed deep learning-based model utilizes the attention mechanism.

In other work, for buy–sell–hold forecasts, Sezer et al. (2017) developed a trading system based on DNN [102]. The proposed framework is found proven mechanism to invest in stock market. In this work technical parameters of the stock trading system were optimized through GA. The proposed model is developed on Apache Spark framework and considered Dow Jones30 for validation.

In 2020, Li et al. proposed a prediction model for stock market data that combined stock prices with market news feelings. The experiments are conducted on five years of dataset of Hong Kong stock exchange and used Loughran–McDonald Financial Dictionary a finance domain-specific [103]. In 2020, Yasin et al. proposed the Stock price modelling utilizing localized multiple kernel learning support vector machine prediction model. The suggested LMKL-SVM model investigates the use of a Localized Multiple Kernel Learning Support Vector Machine to forecast daily stock prices [104].

For multiple stock brands prediction, Rikukawa et al. suggested a RNN inspired stock market prediction model. Dynamic Time warping is explored to get the superior outcome using the proposed model [105]. There is various hybridization that anticipate TSA and stock market using deep learning techniques. Deep learning approaches have shown amazing success in a range of prediction problems because they can extract valuable features automatically throughout the learning process.

Table 2.1: Recent Studies of Individual and Fusion Models

Article [Ref]	Dataset	Dataset Duration	Fusion Methods	Performance Metrics
Göçken M. et al. [15]	BIST100 index	Jun 2005-May 2013	ANN +GA, ANN+HS	MAE, RMSE, MARE, MSRE, MAPE, MSPE
Hiransha M. et al. [16]	NSE, NYSE	Jan 1996-June 2015	MLP, RNN, CNN, LSTM	MAPE
Zhigang J. et al. [18]	AAPL	March 2013-Feb 2018	LSTM, EMD	RMSE, MAE, MAPE
Ritika S. et al. [19]	NASDAQ	Aug 2004-Dec 2015	(2D)2 PCA	RMSE
Kusuma R et al. [22]	Taiwan TW50, Indonesia ID10	Jan 2000 -Dec, 2016	DCNN, candlestick charts	Sensitivity Specificity Accuracy, MCC
Fischer C.T. et al. [26]	S&P 500	Dec 1992-Oct 2015	LSTM, RAF, LOG, DNN	Mean SD, ASRA
Karathana et al. [31]	SPDR S&P Power Shares QQQ Trust	Jan 2006 - Dec 2015	RBF-PSO, RBF-DE and RBF-GA	MSE, MAPE

Feng Z. et al. [32]	NASDAQ, S&P 500	Sept 2016-Dec 2016 Jan 2007-Apr 2007	EMD2FNN	RMSE, MAE, MAPE
Göçken M. et al. [33]	ECILC, EREGL, AFYON	Apr 2013-Sept 2015	HSNN, HS-JRNN, HS-GLM, HS-RT, HS-GPR	MAPE, MAE, RMSE, TheilU, and DS values
Yang F. et al. [34]	A-share market	Jan 2006-Dec 2016	ELM, BPNN, GA, SVR DE, PSO	MAPE, RMSE, SharpeRatio, HitRatio
Bisoi R. et al. [35]	BSE, S&P 500, HS Index	Jan 2010-12 Jan 2016	DE-VMD-RKELM	MAE, RMSE, MAPE
Xi Z. et al. [39]	A-share market HK Index	Jan 2015-Dec 2015	SVM, PCA+SVM. TeSIA	Correlation factor
Hyejung C. et al. [56]	KOSPI	Jan 2000-Dec 2016	GA, LSTM	MSE, MAE, MAPE
Asghar M. et al. [64]	KSE 100 index, Lucky Cement Stock, Engro Fertilizer Limited	Jan 2014–Dec 2014	MR, NN, SVM, RF, EA	Multiple R-square
Ehsan H. et al. [66]	S&P 500, NASDAQ, DJAI, NYSE	Jan 2010-Nov 2017	CNN, PCA+ANN, 2D,3D-CNNpred	MAPE, RMSE
Khashei M. et al. 2019 [77]	SZII DJIAI Nikkei 225	(Jan 1993-Dec 2010) (Jan 1991-Dec 2010) (Mar 2006-Apr2010)	ARIMA-MLP	RMSE, MAE, MAPE, SME

Khashei M., et al. 2011 [78]	Wolf's sunspot data, Canadian lynx series Exchange rate (British pound/US dollar	(1700-1987) (1821-1934), (1980-1993)	ARIMA-ANN	MAE, MSE
Li X. et al. [103]	Hong Kong Exchange (HKEx)	Jan 2003 - March 2008	LSTM	Accuracy, F1 Score
Yasin H. et al. [104]	PT. XL Axiata (EXCL), PT. Unilever Indonesia (UNVR) PT. Indofood Sukses Makmur (INDF)	Jan 2014 - May 2016.	LMKL-SVM	MAPE
Rikukawa S. et al. [105]	Tokyo Stock Exchange	Jan 2016 - Nov 2016	RNN+DTW	RMSPE, MAPE

SD: Standard Deviation, ASR: Annualize Sharpe ratio, TeSIA : Tensor-based learning approach, EMD: empirical modal decomposition, CNN: Convolution Neural Network, ELM: Extreme Learning Machine, REML: Recurrent Extreme Learning Machine, LSTM: Long-Short Term Memory, CEQ: Certainty-Equivalent, ASRA: Annualized Sharpe Ratio Accuracy, MAE: Mean Absolute Error, MLP: Multi-Layer Perceptron, MAPE: Mean Absolute Percent Error, MSPE: Mean Squared Percentage Error, ACO: Ant colony optimization, PCA: Principal Component Analysis, PSO: Particle Swarm Optimization, DE: Differential Evolution, DTW: Dynamic Time Warping, RMSPE: Root Mean Square Percentage Error, HS: Harmony Search..

CHAPTER 3

CONSIDER HETEROGENEOUS INFORMATION FOR MORE ACCURATE STOCK MARKET PREDICTION

3.1 Introduction

This chapter proposed an integrated deep learning approach using ABC optimized LSTM. ABC is the ideal suitable for hyperparameter selection in LSTM models, reducing exploitation and exploration challenges. Managing multidimensional social media reviews is a major undertaking. This chapter develops numerous features of market sentiments and computes the sentiment polarity index using the dependable platform, Hadoop a well-known ecosystem. The proposed ABC-LSTM hybrid model is contrast with DE-LSTM and GA-LSTM. To evaluate hybrid forecasting models, social media data and historical price data of IT sector funds AAPL, MSFT and INTL from NASDAQ GS are considered. The hyperparameters of LSTM like batch size, learning rate, epochs, window size and dropout probability are tuned using the proposed algorithm ABC-LSTM. The proposed algorithm ABC-LSTM is evaluated using the loss functions RMSE and MAPE.

3.2 Contribution of the chapter

The original contribution of this chapter can be streamlined as follows.

1. This chapter progresses the hybrid model ABC-LSTM to consider heterogeneous information for more accurate stock market prediction.
2. To examine the proposed model sentiment polarity index is computed from textual news articles in Apache Hive a superior distributed environment suitable for stock market data.
3. Sentiment polarity index is finally incorporated with selected technical indicators and stock price as information fusion. The proposed ABC-LSTM model is examined on 10 years (Sept 2010 to Sept 2020) of market driven sentiments and stock price of AAPL, MSFT and INTL (IT sectors) from NASDAQ.

Table 3.1: ABC Optimized Technical indicator for market trends analysis

ID	Variable	Market Dimension
1	Previous close price	Trends
2	Previous highest price	Trends
3	Previous lowest price	Trends
4	EMA (10 day) of close price	Trends
5	EMA (20 day) of close price	Trends
6	Ichimoku	Momentum
7	Stochastic K%	Momentum
8	Bollinger higher band	Volatility
9	William’s %R	Momentum
10	Chaikin volatility	Volatility
11	Keltner Channels	Volatility
12	Average True Range	Volatility
13	Momentum open price	Momentum
14	Momentum highest price	Momentum
15	MACD	Trends
16	Chaikin Money Flow	Volume
17	On Balance Volume	Volume
18	Relative strength index (RSI)	Momentum
19	Stochastic D%	Momentum
20	Keltner Channels	Volatility

3.3 Market Trend Analysis

Because stock data is very dynamic and impacted by many market forces, capturing market trends is a huge difficulty. In the global market, these dynamics are affected in upwards and downwards trends exponentially. However, historical data from NASDAQ between September 2010 and September 2020 is used in this analysis [71]. The NASDAQ Stock Market is a global select market with over 1400 stocks in its portfolio. Technical indicators are tried-and-true methods that keep the stock market's nerves calm. Market trends and volatility are all depicted as features in these indicators. Table-3.1 illustrates the ABC inspired selection of technical indicators. After all, final technical indicators with stock price attributes are among the 20 essential factors that encompass many aspects of the stock market. This work established market trends that bolstered the suggested model's outcomes. These indicators are based on the market factors like volume, momentum, trends and volatility. In this study indicators selection maintain the equilibrium of stock market diversification.

3.4 Sentiment Analysis for Time Series Prediction

In fields where time series forecasting is used, such as the stock market, sentiment research plays an important role. Historical data alone is insufficient to forecast future stock market volatility tendencies. As a result, calculating market sentiments for those stocks, as well as upgrading the dataset (past data and sentiment polarity), are critical for predicting stock trends. Such input is growing in the form of comments and blogs on social medias such as Facebook, Instagram, and other platforms.

Market sentiments are a type of information that attracts market stakeholders to make decisions while preparing an investment in the stock market. The suggested model's second phase shows how to compute the emotion polarity index utilizing Big Data, which was briefly covered in this section. According to various research, Big Data analysis is the most effective method for text analysis [39-41].

Hadoop's architecture allows massive datasets to be stored on HDFS and processed by distributed nodes. Another advantage of this strategy is that it uses Apache Flume to deal with streaming data. Bigdata platform’s Apache Flume maintains a continuous stream of data from

Twitter's source and saves it to the HDFS file system. Flume [42, 43] gathered trending tweets from several sources and loaded them into the HDFS. Because there was a lot of noisy and undesired data in the extracted data, it needed to be preprocessed. On the one hand, the downloaded tweets file has three features (End user, Tweets, Date) to conduct experiments. On the other hand, raw news files have four features including News_text, Date, Source and URL link.

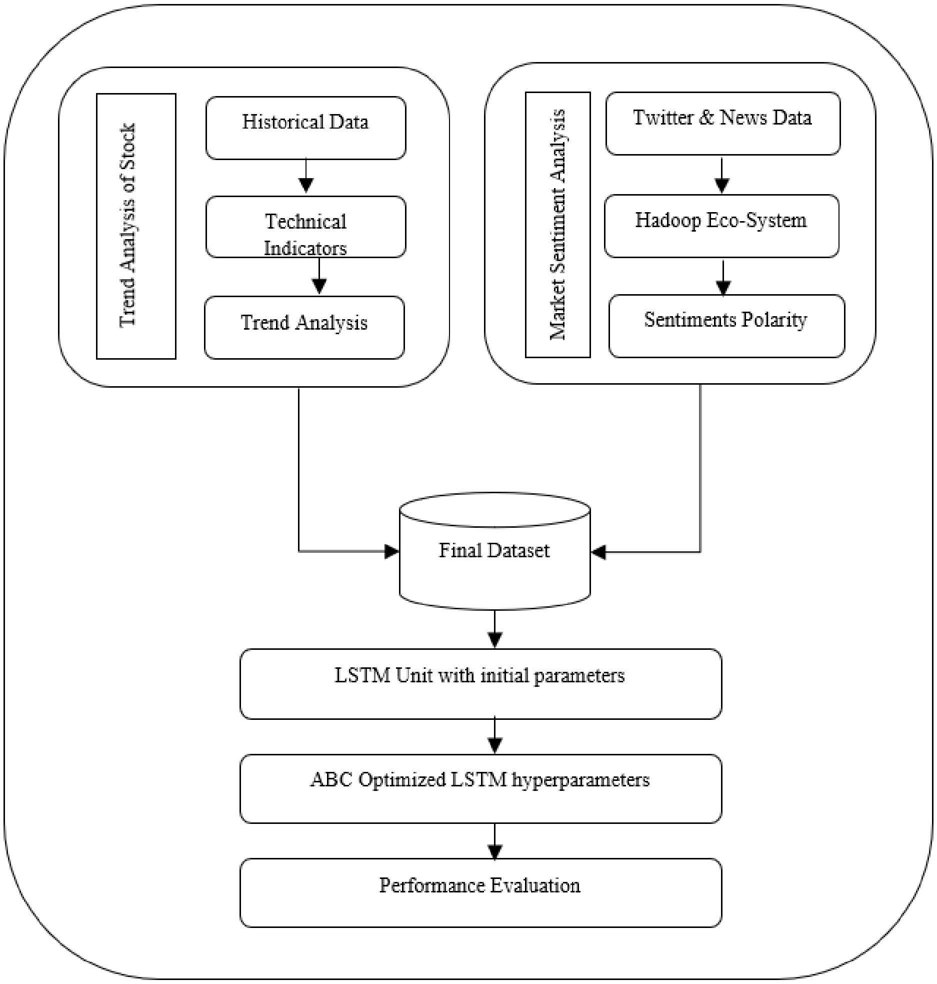

Figure 3.1: Proposed ABC-LSTM Model

3.5 Sentiment’s Polarity Computation

Tweets and news are transformed into tokens in the first round of processing. Following tokenization, a noisy data removal procedure will be used to remove undesired and duplicate statements. The sentiment polarity index is calculated using HiveQL (Hive Query Language) lexical methods [43]. Cleansed HDFS data is processed in HiveQL and then separated into words as a parsing phase. Tokenization is the term for this procedure. The lexical classification determines the sentiment polarity of tokens. Based on market sentiments, positive, negative, and neutral sensations are used to classify sentiments. The positive score of the planned procedure is updated if the feeling word belongs to a positive lexicon (a set of phrases that create positive emotions). If the sentiment term belongs to a negative lexicon, the negative score is modified (a set of words that express negative emotions). The sentiment index is derived by summing the sentiment scores of each news text in the dataset. To assess stock market trends, all three scores are calculated and compared. The prediction value of the LSTM model, on the other hand, is crucial for stock movement in upwards, downwards, or remain unchanged over time. The sentiment polarity is determined by the sum of positive, negative, and neutral scores. The polarity of the sentiment becomes 0 if the neutral score rises using equation-3.1.

$$\text{Sentiment Polarity} = (\sum \text{pos \_score} \;||\; \sum \text{neg\_score} \;||\; \sum \text{neutral\_score}) \qquad (3.1)$$

3.6 Processing of LSTM Unit

This chapter uses deep LSTM to create a novel stock prediction model [25]. It can effortlessly keep track of and store past data while connecting to a new input set. ABC increases the LSTM unit's parameter selection, which has an impact on the model's correctness and performance. To sustain the flow of information, the LSTM unit's execution is separated into three portions using logical gates. The LSTM unit's input gate (I_t) is meant to store the essential information using sigmoid activation function about the cell state. This cell (C_t) combined the input of current cell ((I_t)) with the previous output value (C_{t-1}). Equation-3.2 is used to calculate the value of the input gate (I_t) for the candidate layer, with output in the range of [0-1]. With equation-3.4, the (C_t) is added to the internal state. Equation-3.5 uses the forget gate (f_t) to compute the forget gate value using single tensor and bias factor. Because of the sigmoid function, the gate's output value is found in the range of 0 to 1. As output 1 indicates that current information will be passed to next layer unit as an output (f_t). When the output value (f_t) is 0, the previous internal state has been fully forgotten.

According to equation-3.6 regulates the flow of data within the cell. Output (O_t) and (C_t) are used to estimate hidden layer output (h_t) using equation-3.7. Output value (f_t) is 0 shows the previous internal state is not being considered and has been fully forgotten. Final Output (O_t) of LSTM unit is passed as C_t and hidden layer output as h_t. Detailed equations are given as follows.

$$I_t = (W_i[h_{t-1}]+b_i) \tag{3.2}$$

$$C_t = (W_i\,[h_{t-1}, X_t]+b_i) \tag{3.3}$$

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{3.4}$$

$$f_t = (W_f\,[h_{t-1}, X_t]+b_f) \tag{3.5}$$

$$O_t = (W_o[h_{t-1}, X_t]+b_o) \tag{3.6}$$

$$h_t = O_t * tanh\,(C_t) \tag{3.7}$$

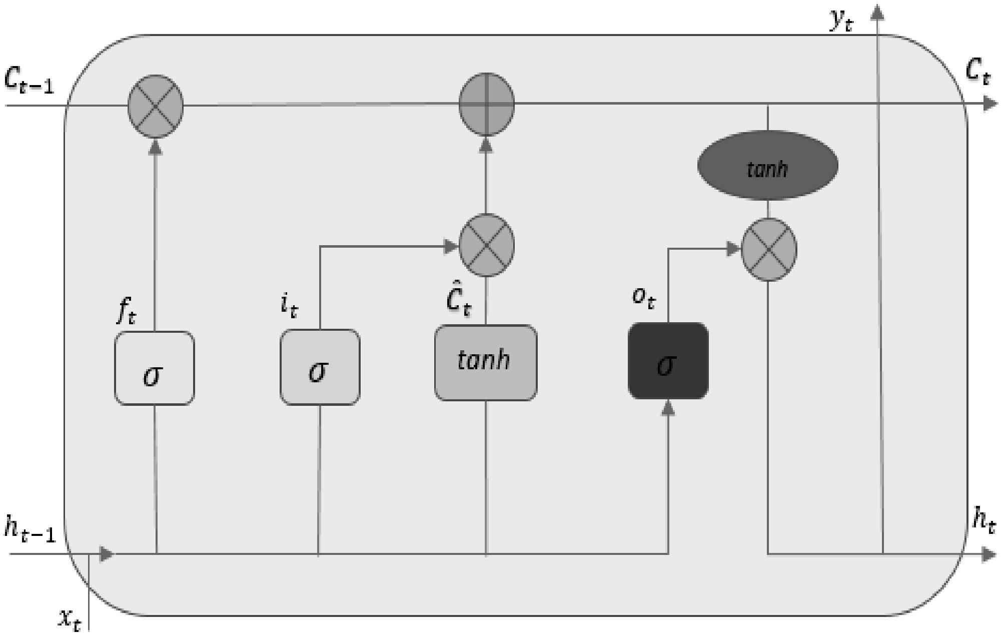

Figure 3.2: Single LSTM Unit [25]

Algorithm 3.1: Proposed Algorithm for Sentiment polarity index using Apache Hive

Input: Dataset (D) contains news events and tweets

```
Step-1 Input parameters {pos_score←0, neg_score←0, neutral_score) ← 0},
Step-2    for each news_text (N) of Dataset (D)
Step-3        Cleaned data into HiveQL and splits into words,
Step-4        Removal of unwanted and duplicate statements from dataset,
Step-5        Data normalization,
Step-6          for every word($w_i$) in news_text (N)
Step-7                If word($w_i$) in positive lexicon
Step-8                  pos_score +=1,
Step-9                else if word($w_i$) in negative lexicon
Step-10                 neg_score +=1,
Step-11               else
Step-12                 neutral_score +=1,
Step-13           end of for loop
Step-14       end of for loop
Step-15       If ($\sum$pos_score > $\sum$neg_score)
Step-16           Sentiment Polarity = +1,
                  Stock ←Upward trends,
Step-17       Else if ($\sum$pos_score < $\sum$neg_score)
Step-18           Sentiment Polarity = -1,
                  Stock ←Stagnant trends,
Step-19       Else
Step-20           Sentiment Polarity = 0,
                  Stock ←Downwards trends,
       End
```

3.7 Modified ABC

The ABC algorithm was inspired by the natural activity of bees introduced in 2005 by Karaboga D [27]. It is regarded as a typical swarm intelligent parameter optimization algorithm [28]. The bees are categorized into three types based on their behavior: employed bees, onlooker bees, and scout bees. The employed bees are expected to go out in quest of food. The colony's goal is to obtain the most nourishment, which is known as nectar. It is the obligation of onlooker-bees to use the food depending on input passed down. The onlooker bees do not leave their Hive unless they receive signs and communication from working bees. The scout bees seek for fresh prospective nectar near their hive at the end of the day. When employed-nectar bee's runs out, it transforms into scout-bee. Now scout bees are expected to look for new food item as its new job. Employed-bees and onlooker-bees are constrained by nature, whereas scout-bees are boundless and unrestricted while foraging for nectar. ABC has been found as a well-proven, high-quality optimum method for multidimensional problems. ABC uses SN solutions on search space to produce food placements (sources) at random. The complete ABC process is illustrated with detailed notations. Equation-3.8 is used to obtain solution (x_i) in the range of 1, 2…SN.

$$x_{ij} = x_{min,j} + rand[0,1](x_{max,j} - x_{min,j}) \tag{3.8}$$

An employed bee generates solution v_{ij} with equation-3.9.

$$v_{ij} = x_{ij} + rand[-1,1](X_{max,j} - X_{min,j}) \tag{3.9}$$

Equation 3.10 produces the nectar probability (p_i) using fitness value.

$$p_i = \frac{fit_i}{\sum_1^{SN} fit_n} \tag{3.10}$$

Where fit represents the fitness value for i for the objective function f_i using equation-3.11.

$$fit_i = \begin{cases} \frac{1}{1+f_i} & fi \geq 0 \\ 1 + |f_i| & fi < 0 \end{cases} \tag{3.11}$$

Algorithm3.2: Proposed Artificial Bee Colony (ABC) for LSTM hyper parameter selection

```
Input:     Initial Population $x_i$, Training & Testing dataset (X, X`)
           No of optimization parameters D, Possible solutions (SN)
Output:  Optimization hyperparameters for LSTM
BEGIN
Step-1 Load training dataset (X)
Step-2 Generate the initial population $x_i$ where i(1, 2,…, SN).
Step-3 Evaluate the fitness ($fit_i$) of population where i(1, 2,…, SN).
Step-4 Set iteration $i$ to 1
Step-5     Repeat
Step-6 for each employed-bee
        {
            Produce a new solution $v_{ij}$ using equation- 3.9
            Compute the fitness ($fit_i$) for $v_{ij}$  using equation-3.11
            Using greedy approach
        }
Step-7 Compute probability ($p_i$) for $x_i$ using equation-3.10
Step-8 for each onlooker-bee
        {
        Apply selection on solution $x_i$ based on $p_i$
        Produce a new solution $v_{ij}$ using equation-3.9
        Compute the fitness ($fit_i$) for $v_{ij}$  using equation-3.11
        Using greedy approach
        }
Step-9 Replace SN with new solution produced by scout-bee using equation-3.8
Step-10 Repeat step-9 until ($x_{ij}$ > limit) and gets optimized value using equation-3.12
Step-11 Iteration i=i+1
Step-12 Until i=max (i)
END
```

Scout-bees are now expected to hunt fresh over the whole workstation in order to obtain a globally optimized solution. The set limit is several iterations treated as a critical parameter may be calculated using equation-3.12 with optimization parameters and coefficient factor between range 0.5 to 1.

$$limit = D * n_e * c \tag{3.12}$$

3.8 Solution Design for Optimized ABC-LSTM

The ABC optimized LSTM model were illustrated in this chapter. Table-3.2 shows the list of hyperparameters. The following procedures have been proposed for LSTM neural network optimization.

Step-1 The data preprocessing stage includes the partition of datasets into 80%-20% partition as training and testing dataset. The training dataset represents 80% of total occurrences, whereas the testing dataset represents the most recent examples of the remaining 20%. To produce the best fit LSTM prediction model, validation set is separated as 25% of the training dataset instances.

Step-2 ABC algorithm, which is addressed in algorithm-3.2, develops an initial population, optimization parameters and SN solutions.

Step-3 Initially LSTM hyperparameters are used to train deep LSTM. RMSE, on the other hand, is the least fit value. Employee bees use equation-3.9 to alter the hyperparameters combination based on the lowest RMSE prediction values.

Step-4 By analyzing the objective function using equation-3.10 and reducing RMSE, onlooker bees choose the optimum LSTM hyperparameters. Employee bees, on the other hand, became scout bees when the hyper parameter values fell below the limit. Scout bees continue to work on a new LSTM hyperparameter combination to be tested using equation-3.8.

Step-5 This cycle is performed until the optimal LSTM value using the least RMSE value is found. Because the objective function's fitness value stated in equation-3.10 has the smallest RMSE.

Table 3.2 illustrates the LSTM unit's local and global hyperparameters, which yield significantly better results than the initial values used to start the experiments.

3.9 Result Analysis

The suggested algorithm-3.1 and algorithm-3.2 presented, are tested on commodity hardware with 2TB secondary storage, 12GB RAM, and a 1.90GHz processor. For the sentiment research, Apache Hadoop 2.0 service was chosen for a 10-year data range for selected NASDAQ companies. To acquire the forecasting findings, the following subsections are evaluated in turn. ABC-LSTM Hyperparameters (Table 3.2)

Table 3.2 ABC- LSTM Hyperparameters

Hyper-parameters	Initial Value	ABC Optimized Hyper-parameters Value		
		AAPL	MSFT	INTL
LSTM_units	50	100	100	100
Dropout_prob	0.5	0.6	0.6	0.5
Epochs	100	148	154	120
Batch_size	32	64	64	128
Learning Rate	0.01	0.01	0.01	0.01
Loss Function	RMSE	RMSE	RMSE	RMSE

3.9.1 Data Collection and Preprocessing

The selection of historical stock data, as well as relevant twitter and financial news data for equities in the IT sector, is described in the data collecting portion. The NASDAQ Stock Market is a global select market with over 1400 stocks in its portfolio. For sentiment analysis, Twitter was chosen as the social media data source. Between September 2010 and September 2020, social media data relating to selected stocks are deemed similar time spans for experimentation. The data preprocessing stage includes the partition of datasets into 80%-20% partition as training and testing dataset. The training dataset represents 80% of total occurrences (2517), whereas the testing dataset represents the most recent examples of the remaining 20%. To produce the best fit LSTM prediction model, validation set is separated as 25% of the training dataset instances. Finally, to improve the performance of the ABC-LSTM model by normalizing the dataset inputs and scale them between 0 to 1. Because the raw data range of the selected stock fluctuates significantly, the range from 0 to 1 obtains using equation-3.13.

$$X` = X - X_{min} / X_{max} - X_{min} \quad (3.13)$$

3.9.2 Performance Measure

For each stock, the ABC-LSTM optimizes the window size, dropout probability and batch size for the AAPL dataset were 3,64 and 0.5, respectively. Similarly, the identical set of parameters were discovered in MSFT and INTL at 3,32,1 and 3,128,0.9, respectively. The batch size for the datasets is shown in Fig. 3.3 to Fig. 3.5 and the dropout probability for the datasets are shown in Fig. 3.6 to Fig. 3.8. The ABC-LSTM model obtained a minimal loss of 0.2220, 0.3245, and 0.3062 for each selected dataset. Sensitivity analysis is used to analyze the performance improvement strategies using performance indices. ABC-LSTM outperforms the other two algorithms, according to this study. RMSE is measured as squared value, resulting known as a scale-dependent metric. On the other side, MAPE is a scale-independent measure based on percentage error. The root value of MSE restores more significant scale of the RMSE. True observation is used to standardize the MAPE for improving the accuracy of stock market price for TSA.

$$\mathrm{RMSE} = \sqrt{\textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}}(\mathrm{Xi} - \mathrm{X'i})^2/\mathrm{n}} \qquad (3.14)$$

$$\mathrm{MAPE} = 1/\mathrm{n}\textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}}|\mathrm{Xi} - \mathrm{X'i}/\mathrm{Xi}| * 100 \qquad (3.15)$$

Different MAPE and RMSE values for datasets with and without sentiment polarity index are shown in Table-3.3-3.4. The AAPL stock has a MAPE of 3.1721 and an RMSE of 8.4435, according to the ABC-LSTM experiment. MSFT stocks have a MAPE of 3.8975 and an RMSE of 3.9868. The third instance provides MAPE percent and RMSE of 4.0417 and 2.4366, respectively, using the INTL dataset. This result is produced without the use of polarity index of considered stocks. Similarly, Table-3.4 shows the proposed model (ABC-LSTM) produced error MAPE 2.0216% for the AAPL stock. However, for RMSE same stock obtained 7.3563. Both the cases are examined with market sentiments polarity. MSFT stocks have a MAPE of 1.7500 and an RMSE of 3.8678. Similarly, INTL returns MAPE 2.1698% and error RMSE as 1.6568, respectively, in the third example. All of the equivalents that were chosen to compare and established as current work in recent time were agreed upon by ABC optimized LSTM. Figure 3.9-3.11 show impact of loss function on RMSE as well as MAPE. Total occurrences of training, testing, and validation are included in the obtained results. The model's performance is reflected in the differences in values.

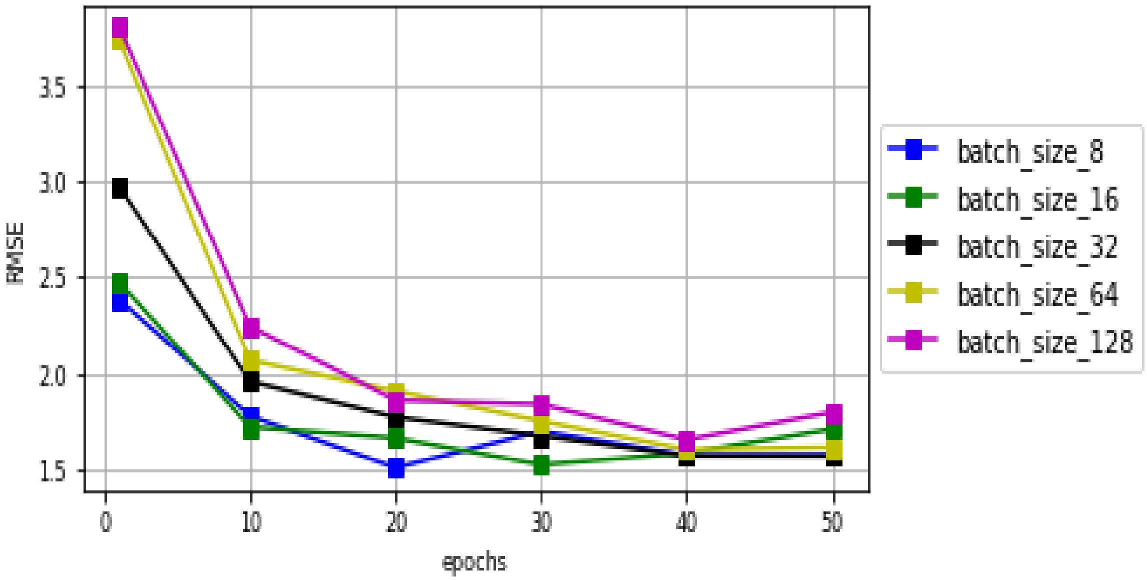

Figure 3.3: Minimization of fitness value based on batch size (AAPL)

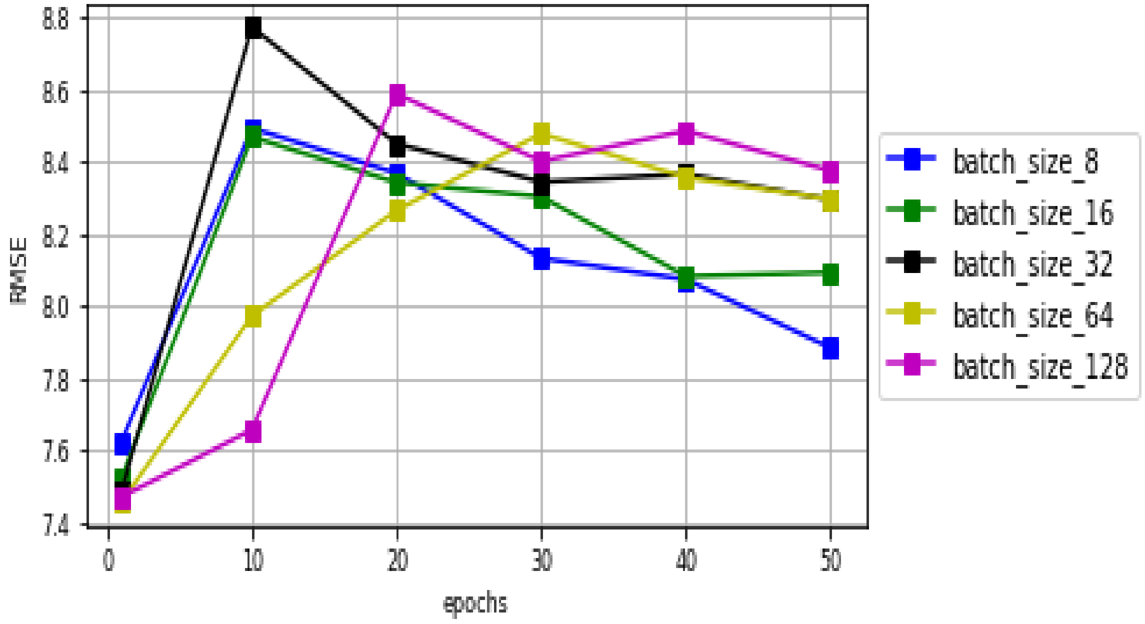

Figure 3.4: Minimization of fitness value based on batch size (MSFT)

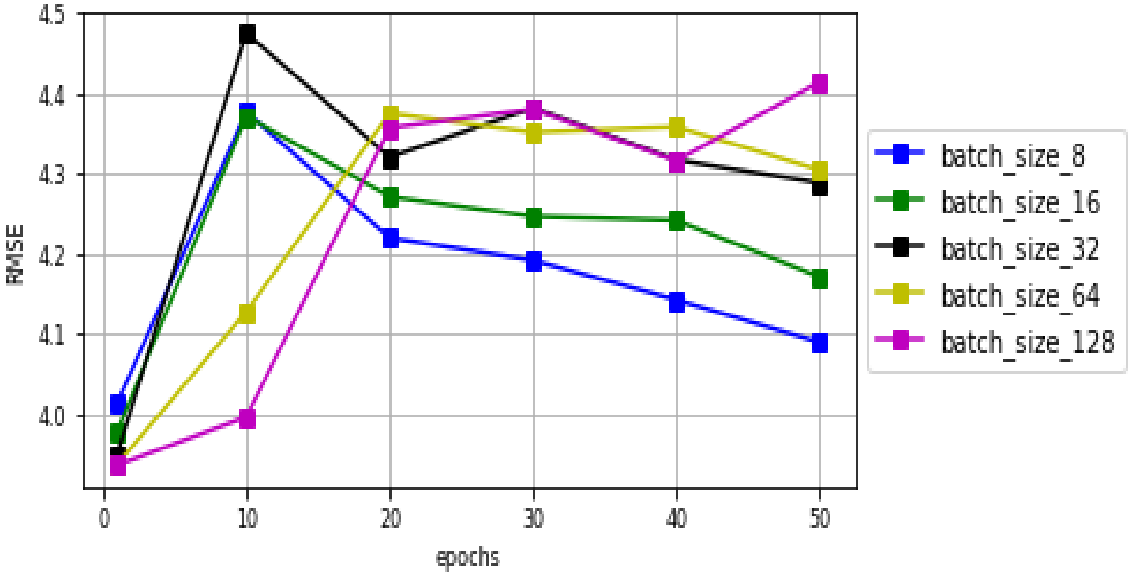

Figure 3.5: Minimization of fitness value based on batch size (INTL)

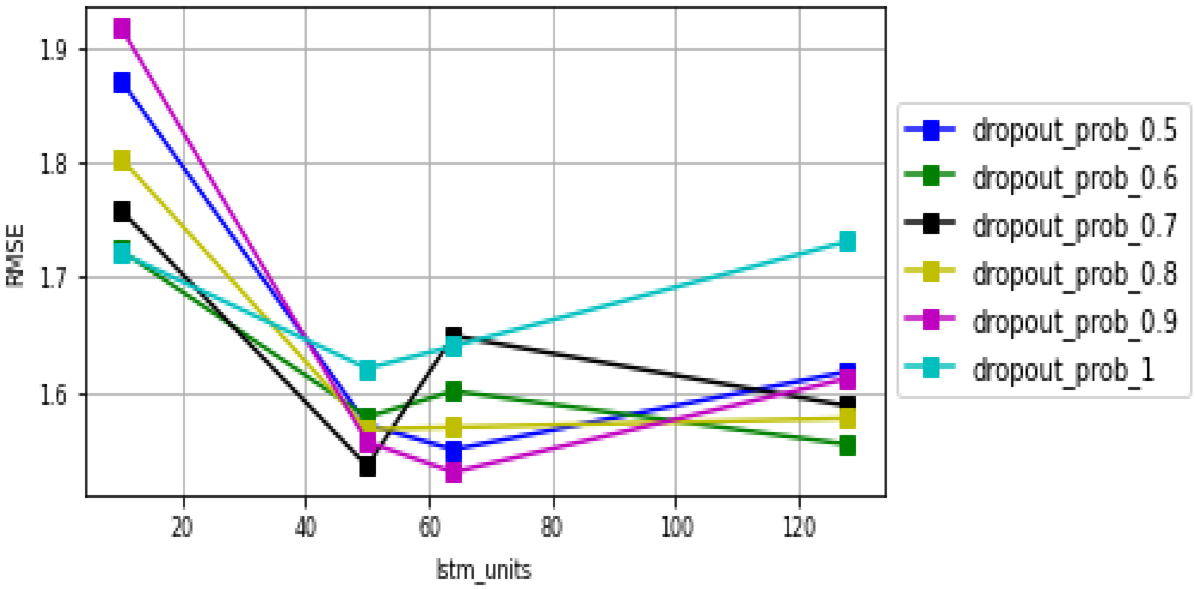

Figure 3.6: Minimization of fitness value based on dropout probability (AAPL)

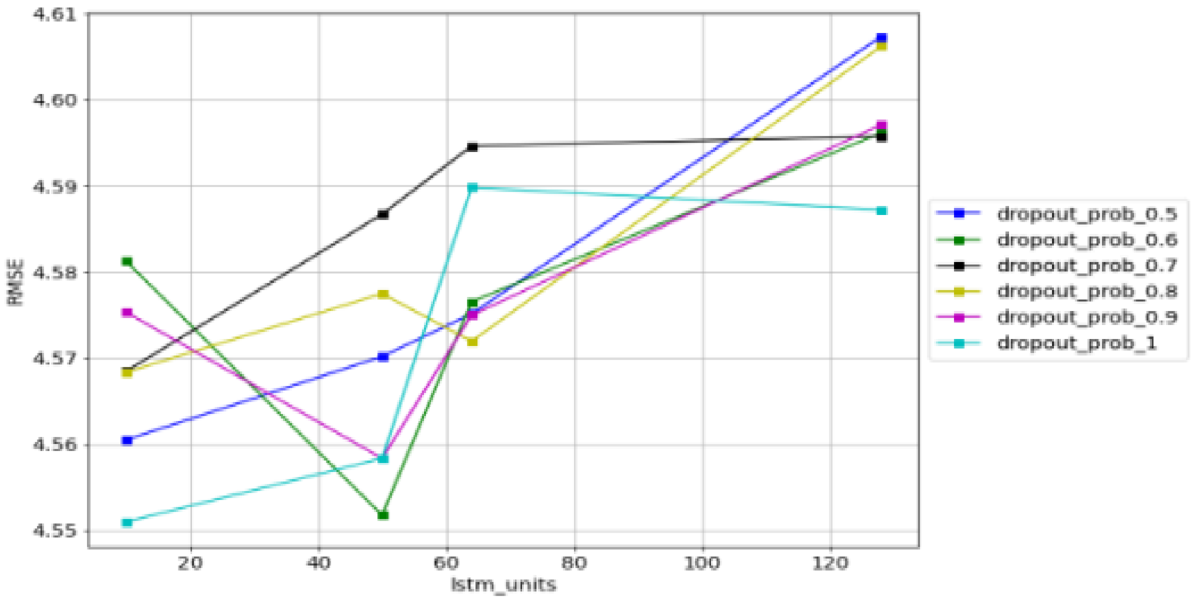

Figure 3.7: Minimization of fitness value based on dropout probability (MSFT)

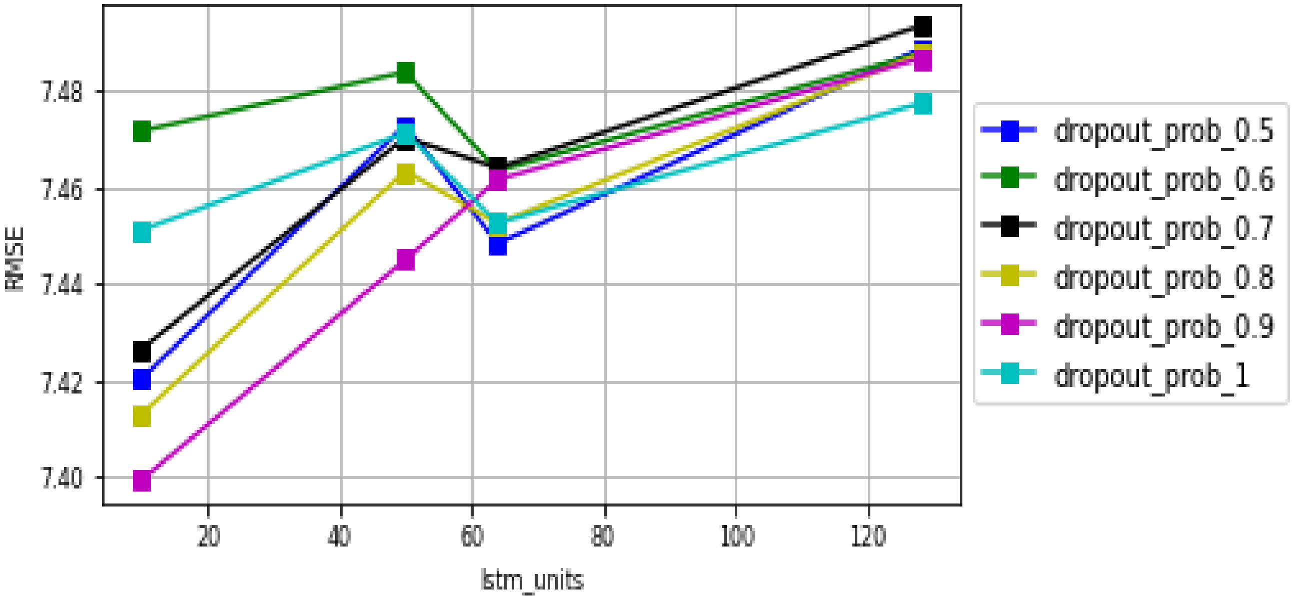

Figure 3.8: Minimization of fitness value based on dropout probability (INTL)

Table. 3.3: Experiments without Sentiments Polarity

Stock	Performance Metrics	LSTM	DE-LSTM	GA-LSTM	ABC-LSTM
AAPL	MAPE (%)	4.1400	3.0220	3.2342	3.1721
	RMSE	14.1271	13.9280	8.4558	8.4435
MSFT	MAPE (%)	3.8110	3.7490	3.4305	3.8975
	RMSE	6.9403	5.3120	3.7643	3.9868
INTL	MAPE (%)	4.9303	4.8639	4.1459	4.0417
	RMSE	4.2601	4.2574	3.7266	2.4366

Table. 3.4: Experiments with Sentiments Polarity using Hadoop

Stock	Performance Metrics	LSTM	DE-LSTM	GA-LSTM	ABC-LSTM
AAPL	MAPE (%)	3.4005	2.0220	2.4180	2.0216
	RMSE	12.5127	11.9280	8.5770	7.3563
MSFT	MAPE (%)	2.8110	1.8749	2.0480	1.7500
	RMSE	6.4027	4.3960	4.4340	3.8678
INTL	MAPE (%)	2.9303	4.5390	2.1910	2.1698
	RMSE	2.2601	3.5740	1.6550	1.6568

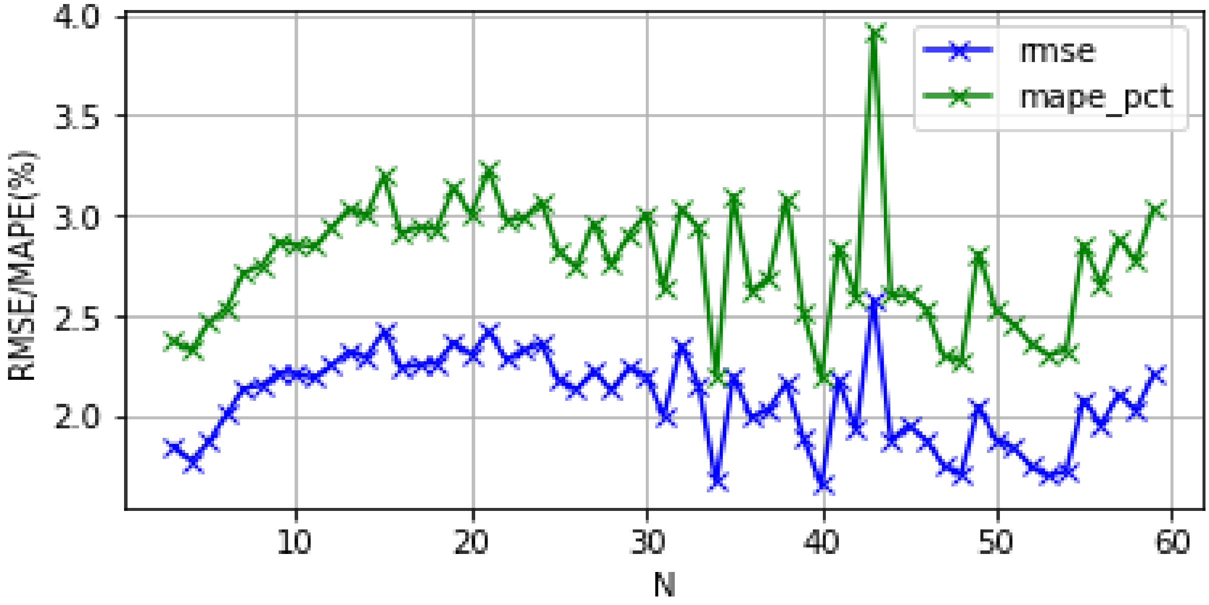

Figure 3.9: RMSE and MAPE plots for AAPL Dataset

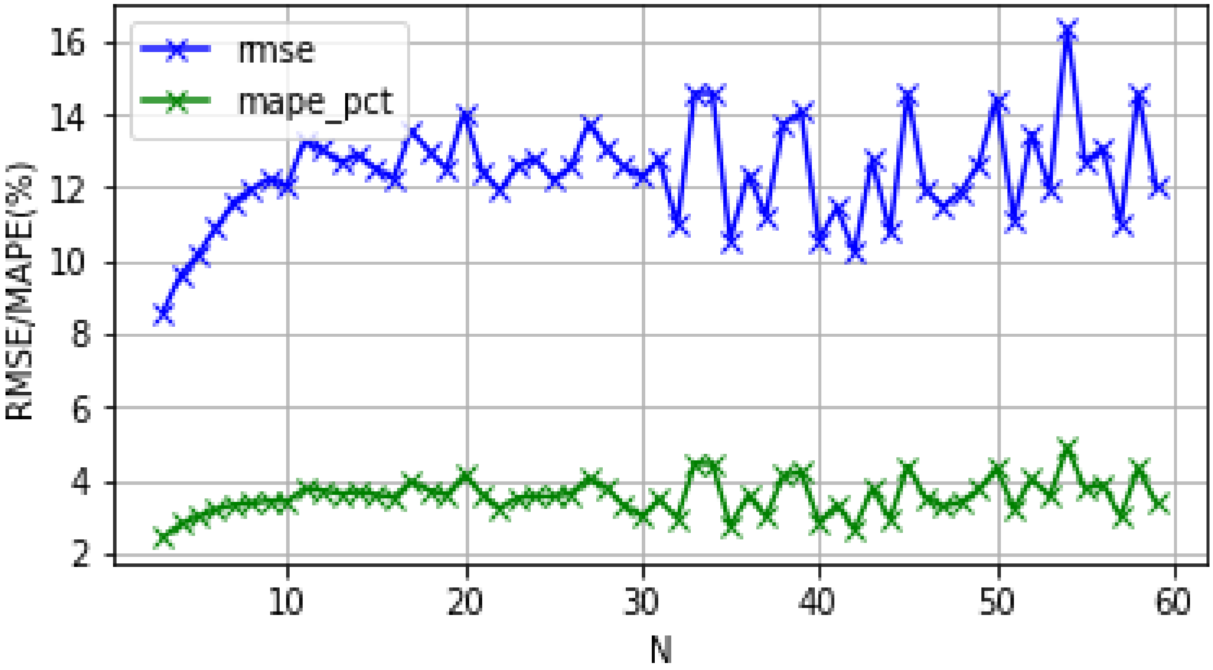

Figure 3.10: RMSE and MAPE plots for MSFT Dataset

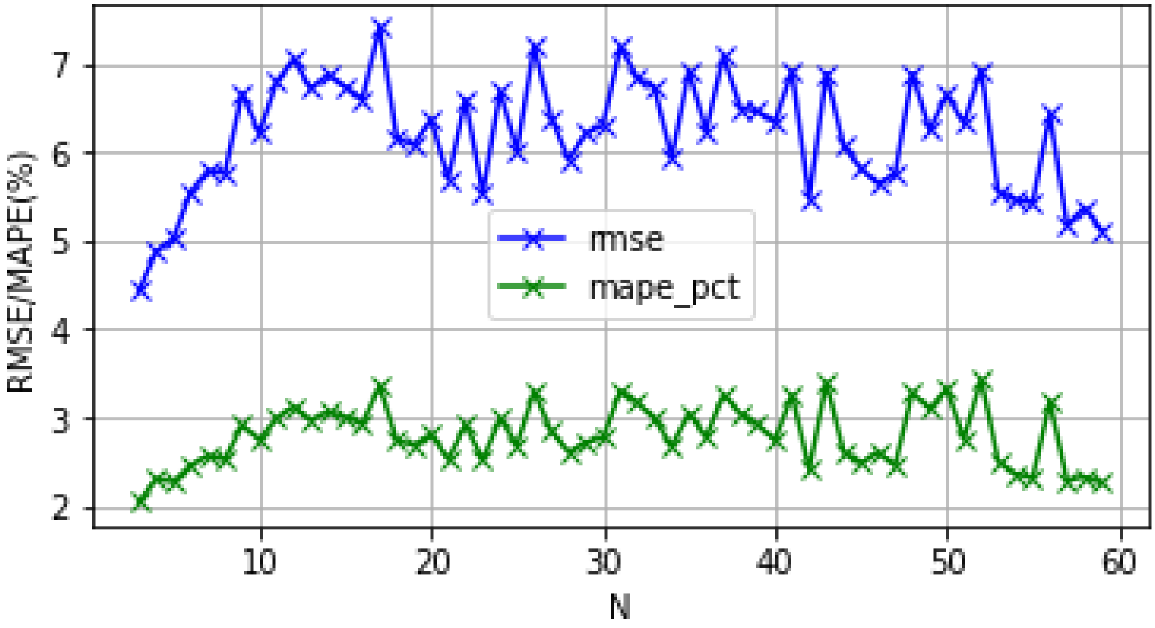

Figure 3.11: RMSE and MAPE plots for INTL Dataset

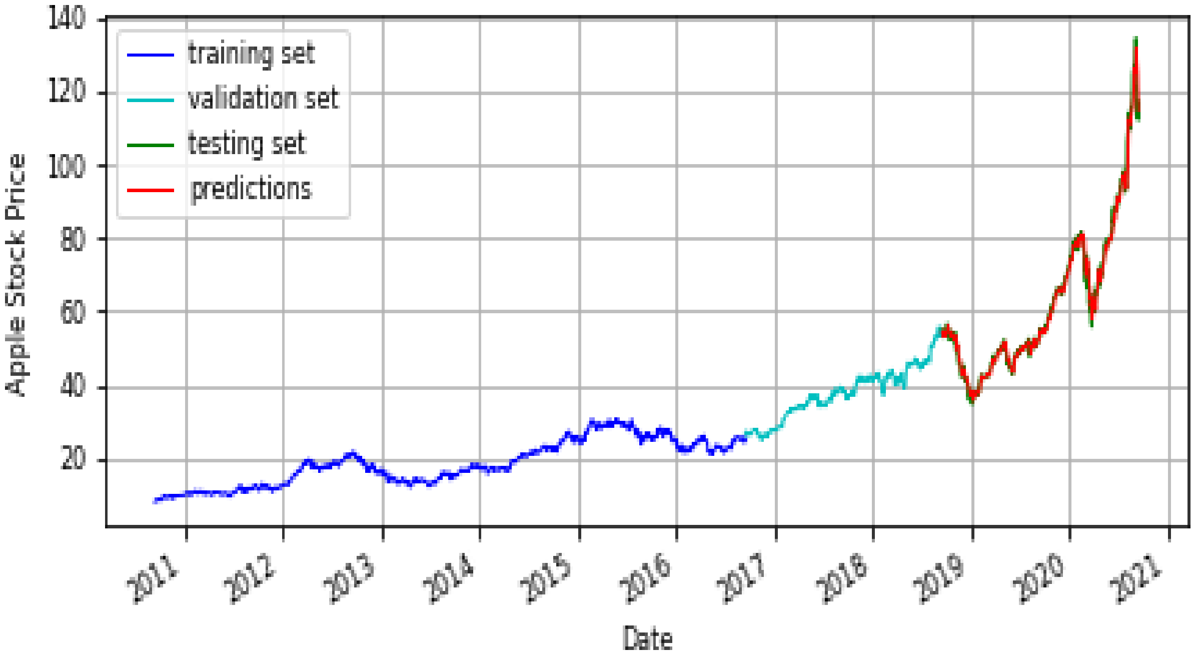

Figure 3.12: AAPL, NASDAQ Stock Prediction result

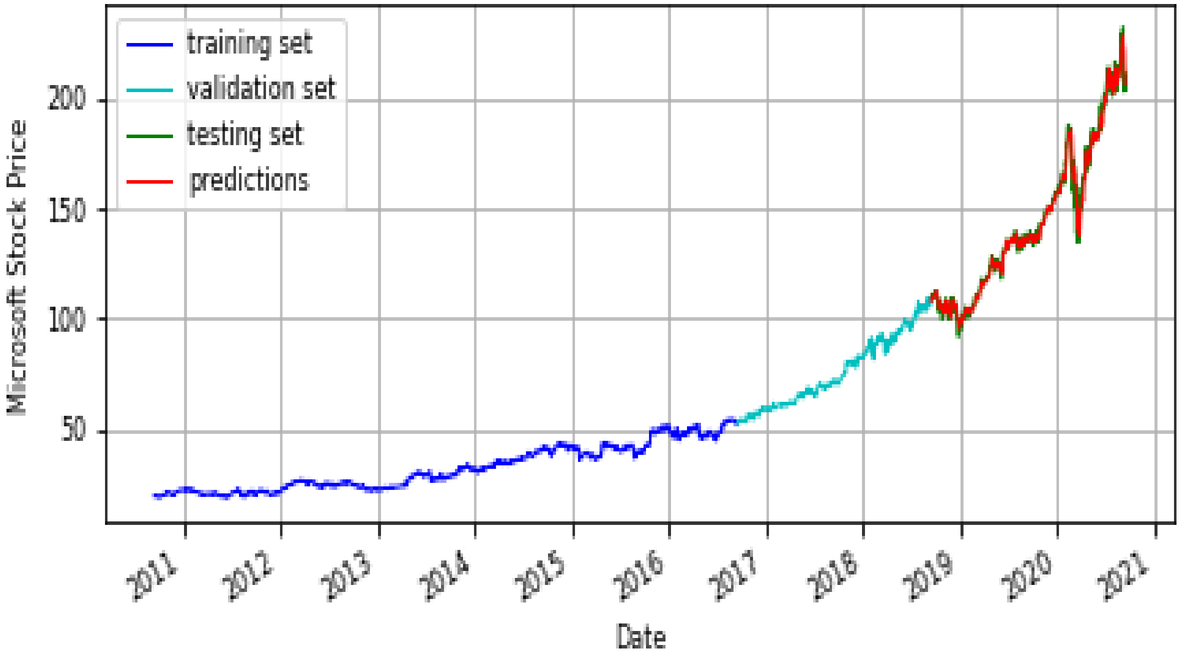

Figure 3.13: MSFT, NASDAQ Stock Prediction result

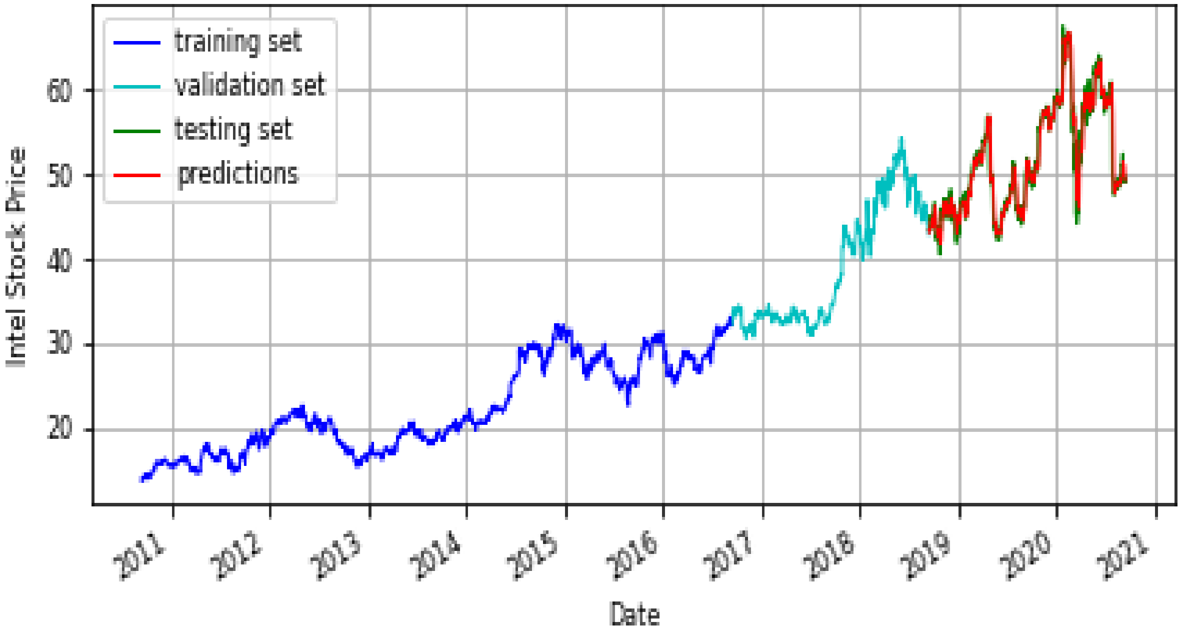

Figure 3.14: INTL, NASDAQ Stock Prediction result

3.10 Summary

In this chapter ABC-LSTM is tried-and-true hybrid paradigm for balancing concerns about exploitation and exploration. The major features of the stock market, including as volume, volatility trends, and momentum are discussed in this chapter. Technical indicators, for the most part, cover all areas, but choosing indicators is a tedious operation. The ABC method is used in this chapter to create a final feature selection pool comprising the indications. The polarity index is used to calculate sentiments. The Hadoop Ecosystem is used to analyze data in the Hadoop Distributed File Format (HDFS). On the Hive service, sentiment analysis is performed using lexicon-based dictionaries. Because MAPE is a human-friendly characteristic, the error margins for AAPL, MSFT, and INTL were 2.0216, 1.7500, and 2.1698, respectively. In terms of constraints, all stocks and related markets are judged to be extremely volatile and rely on factors other than their previous values and market history. Market and investor attitude, political willpower, the global economy, and other factors that could influence stock index growth up or down could all have an impact on stock prices in the future. Complex deep learning algorithms may be used to improve the polarity score of sentiment. These points can be thought of as future possibilities in this field.

CHAPTER 4

ADDRESS THE PARAMETER TUNING ISSUE OF EXISTING MODEL FOR IMPROVING PREDICTION ACCURACY

4.1 Introduction

Time series forecasting is commonly utilized in sequential data as single step ahead and multi-step ahead prediction. The ARIMA is a frequently used time series prediction model. In this study, we combined the DE and ABC algorithms to optimize the ARIMA model. Modified algorithms maintain exploration and exploitation methods using a combination of evolutionary algorithms in TSA. In comparison to conventional ARIMA models, the improved ABC with DE Optimization promotes better generalization and efficient performance. From September 1, 2010 to August 31, 2020, experiments are conducted on the dataset of the Refineries sector of the NSE & BSE India. The obtained results show that the proposed strategy based on a modified DE-ABC-ARIMA outperforms its competitors and improves forecasting accuracy while still retaining data patterns.

4.2 Contribution of the chapter

The original contribution of this chapter can be streamlined as follows.

1. This chapter evolves the hybrid model DE-ABC-ARIMA to address the parameter tuning issue of existing model ARIMA for improving prediction accuracy.
2. The proposed algorithmic fusion DE-ABC improves the equilibrium between the exploitation and exploration of the hyper parameters of ARIMA therefore, the prediction accuracy.
3. The proposed DE-ABC-ARIMA model examines public and private sectors companies of oil and gas, refineries sectors from NSE & BSE, India from the 10 years period between September 1, 2010 to August 31, 2020.

4.3 ARIMA

Box and Jenkins established ARIMA linear model for analyzing time series data [1]. ARIMA is made up of two components: Auto-Regressive (AR), which analyses prior data to model. The other part is known as Moving Average (MA), which maintains control over deafening data from earlier instances. The unit root tests ADF and PP are used by ARIMA to check the stationary process. Over time, ADF test is used to ensure that the mean of the instances and variance of the same instances remain consistent. The ARIMA on the other hand, displays a large error rate when data linearity is lost. ARIMA can only cope with linear data and is therefore unsuitable for complex non-linear models [1]. In AR, the lags function is derived using equation 4.1 if the autoregressive coefficient ($\emptyset$) and the number of prior instances is p.

$$\mathrm{X_t} = \emptyset_1 \mathrm{X_{t-1}} + \emptyset_2 \mathrm{X_{t-2}} - - - - \ + \emptyset_\mathrm{p} \mathrm{X_{t-p}} = \sum_{j=1}^{p} \emptyset_\mathrm{j} \mathrm{X_{t-j}} \qquad (4.1)$$

In MA, if moving average coefficient (θ_j), When q is the order of the MA term and the prior innovation process ($\varepsilon_\mathrm{t-j}$) is taken into account, the MA equation is

$$\mathrm{X_t} = \theta_1 \varepsilon_\mathrm{t-1} + \theta_2 \varepsilon_\mathrm{t-2} - - - - \ + \theta_\mathrm{q} \varepsilon_\mathrm{t-q} = \sum_{j=1}^{q} \theta_\mathrm{j} \varepsilon_\mathrm{t-j} \qquad (4.2)$$

The ARIMA parameters hold the order of AR (p), MA terms (q), and significant difference (d). These hyperparameters of ARIMA construct the time series data and ensure stationarity. In most circumstances, d is assumed to be 1. Finally, ACF and PACF are used to estimate model parameters. ACF and PACF charts are used to determine the values of p and q. The ARIMA parameter trends are determined by the behavior of both graphs. SARIMA is the abbreviation for the generalized ARIMA, where S stands for seasonal state. P, D, and Q in SARIMA stand for Auto-Regressive (p), Moving Average (q), and Differencing (d), respectively, where m is a periodicity (i.e. monthly, daily, etc).

$$\mathrm{X_t} = \mathrm{SARIMA(P, D, Q)(p, d, q)_m} \qquad (4.3)$$

AR and MA values are investigated using ACF and PACF graphs of selected equities. In the following sections, several ARIMA (p, d, q) combinations and derived AIC values will be examined in order to choose the optimum ARIMA model. Existing literature-based models have failed to attain the best forecasting accuracy while retaining data patterns [17-19]. Hybrid models have always been shown to be preferable in terms of achieving the greatest results in both categories. Following the research, it was discovered that the majority of the models were only used for one step forward prediction, which is a fundamental flaw in the existing literature. This study investigates a modified ABC utilizing DE. The ABC-ARIMA model, when compared to existing core and hybrid models, produces much better outcomes.

4.4 Proposed Model ABC-ARIMA

In 2005, Karaboga D. et al presented the ABC [27], which is a highly effective typical swarm algorithm. ABC is based on the foraging behavior of bee swarms. ABC is seen as a viable solution to numerical optimization problems such as those encountered in the stock market. The ABC provides multidimensional solutions for TSA such as feature selection and hyperparameter optimization. [28, 29].

With SN random solutions, the ABC algorithm generates food sources. For the problem under consideration, optimization parameters are denoted by the letter D. X_{min} and X_{max} as lower and upper boundary, respectively.

Equation 4.5 is used by employed-bee in the process of producing possible solutions. Onlooker-bees use equation 4.6 to forage solutions depending on nectar probability p_i. The onlooker-bee evaluates the source at this point, and equation 4.7 aids in the formulation of the objective function (f_i).

$$x_{ij} = x_{min,j} + rand[0,1](x_{max,j} - x_{min,j}) \tag{4.4}$$

$$v_{ij} = x_{ij} + rand[-1,1](X_{max,j} - X_{min,j}) \tag{4.5}$$

$$p_i = \frac{fit_i}{\sum_1^{SN} fit_n} \tag{4.6}$$

$$fit_i = \begin{cases} \frac{1}{1+f_i} & fi \geq 0 \\ 1 + |f_i| & fi < 0 \end{cases} \tag{4.7}$$

$$limit = D * n_e * c \tag{4.8}$$

4.5 Differential Evolution (DE)

DE is proposed by Storn, R. et al. [44] as a global optimization solution for time series models. The DE is made up of four operations that are divided into numerous phases. Initialization of chromosomal length (D), gene value range for operations (U_{min}, U_{max}), assumed crossover rate (CR), detected mutation factor (F), and chosen population size (N) [44-46].

$$X_{ij\,=}\,U_{min}+ rand*(U_{max}- U_{min}) \qquad (4.9)$$

A mutation is a single alteration in a gene that can be observed using the equation below. Where r1, r2, and r3 are random numbers.

$$V_i^{G+1} = X_{r1}^{G}{}_{+}\ F*(X_{r2}^{G}- X_{r3}^{G}) \qquad (4.10)$$

The term "crossover" refers to the process of exchanging genes. In a crossover procedure (j) is considered as individual gene dispersed in the range of 0-1 where rand is considered as random number.

$$u_{tj}^{G+1} = \begin{cases} v_{tj}^{G+1}, & ifrand(j) \leq cr \text{ } or \text{ } j = randn(t) \\ & x_{tj}^{G}\ {}_{,}otherwise \end{cases} \qquad (4.11)$$

Selection is based on offspring and parents in which the better performs better than the rest based on fitness.

$$x_i^{G+1} = \begin{cases} u_i^{G+1}, \ if \ f\left(u_i^{G+1}\right) < f\left(x_i^{G}\right) \\ \qquad x_i^{G} \quad , otherwise \end{cases} \qquad (4.12)$$

4.6 Modified DE optimized ABC

ABC is notable for its ability to solve multi-objective problems with global exploration challenges. However, due to local exploitation, it is known as local search efficiency. Using hybrid search tactics, the solution equation improves search efficiency. Several prior studies are also used to support the hybrid algorithm's complexity. W. Xiang et al. introduced a unique hABCDE for blocking optimization challenges, which they validated using twenty benchmark functions *(f_1-f_{20})*. The achievement of the outcome ensured that the hABCDE outperformed other algorithms [48]. In another work, Zorarpac, E. et al. examine the UCI machine learning repository to assess the performance of the suggested hybridization. In terms of F-measure values, the outcomes of the experiments outperformed the conventional ABC and standard DE algorithms [49]. Similarly, Jadon, S. S., et al. put the hybrid algorithm HABCDE to the test on a set of 20 benchmark functions *(f_1-f_{20})* [50]. The proposed work was also carried out on four real-world optimization situations *(f_{21}-f_{24})*. Time series data as stock market is exhibiting linear as well as nonlinear behavior. The ARIMA model has a better chance of getting the proper hyperparameter tuning when ABC-DE is combined.

$$v_{ij} = x_{ij} + rand[-1,1](x_{best\ i} - x_{ij} + x_{r1,j} - x_{r2,j}) \qquad (4.13)$$

Step-1 Historical datasets of Refineries are analyzed for stationary test as a preprocessing phase from September 1, 2010 to August 31, 2020.

Step-2 Plots are generated based on the allocated values and assessed for the least residual. The coefficient factors and residual of variance are held during parameter estimation for ARIMA models. ARIMA model with the acquired parameter values is used for predicting (p, d, q).

Step-3 The suggested hybrid approach improves the hyperparameters. if the parameter values do not perform well for residuals plots of ACF and PACF are regenerated, and the residual is found using the new hyper parameter values. The optimal model for time series forecasting is the one with the lowest RMSE and AIC.

4.7 Simulation Result

The experiment examined the 10 years dataset from September 1, 2010, to August 31, 2020 with 2460 instances [71] as shown in Table-4.1. The validation set is partitioned into 25%, or 492 instances, to remove overfitting and underfitting limitations from the training dataset.

$$X` = X-X_{min}/ X_{max} -X_{min} \qquad (4.14)$$

The scaled values of raw data X are used in the following equation, and the stock price range is regarded between X_{max} and X_{min}. Finally, outcome X` represents the scaled value of original X instance of the dataset.

4.7.1 Time Series Analysis

In TSA white noise test is one of the statistical methods for validating a hypothesis [106]. In several cases, the obtained autocorrelations diverge from 0 for the provided To-Lag range [6,12,18, and 24]. However, because the majority of cases are near to 0, the ARIMA model may produce the best results with this financial time series dataset. The time series plot of the specified historical dataset is displayed in each stock panel. The Autocorrelation Function (ACF) behaves differently depending on the range of lags. The Inverse Autocorrelation Function (IACF) is calculated using Yule-Walker equations in the proposed hybrid ABC-ARIMA.

Algorithm 4.1: Proposed DE Optimized Artificial Bee Colony (ABC) for ARIMA for Stock Price

```
Input: Initial Population $x_i$
        Training & Testing dataset (X, X`), No of optimization parameters D,
        Possible solutions (SN)
Output: Optimization parameters for LSTM
BEGIN
Step-1 Load training dataset (X)
Step-2 Generate the initial population $x_i$ where i(1, 2,…, SN).
Step-3 Evaluate the fitness ($fit_i$) of population where I (1, 2.., SN).
Step-4 set iteration $i$ to 1
Step-5      Repeat
Step-6 for each employed-bee
          {
                    Produce a new solution $v_{ij}$ using equation-4.5
                    Compute the fitness ($fit_i$) for $v_{ij}$  using equation-4.7
                    Using greedy approach
          }
Step-7 Compute probability ($p_i$) for $x_i$ using equation-4.6
Step-8 for each onlooker-bee
          {
                    Apply selection on solution $x_i$ based on $p_i$
                    Produce a new solution $v_{ij}$using equation-4.5
                    Compute the fitness ($fit_i$) for $v_{ij}$  using equation-4.7
                    Generate new candidate solution $v_{ij}$ as DE-mutation using equation- 4.13
                    Using greedy approach
          }

Step-9 Replace SN with new solution produced by scout-bee using equation-4.4
Step-10 Repeat step-9 until ($x_{ij}$> limit) and gets optimized value using equation-4.8
Step-11 Iteration i=i+1
Step-12 Until i=max (i)

END
```

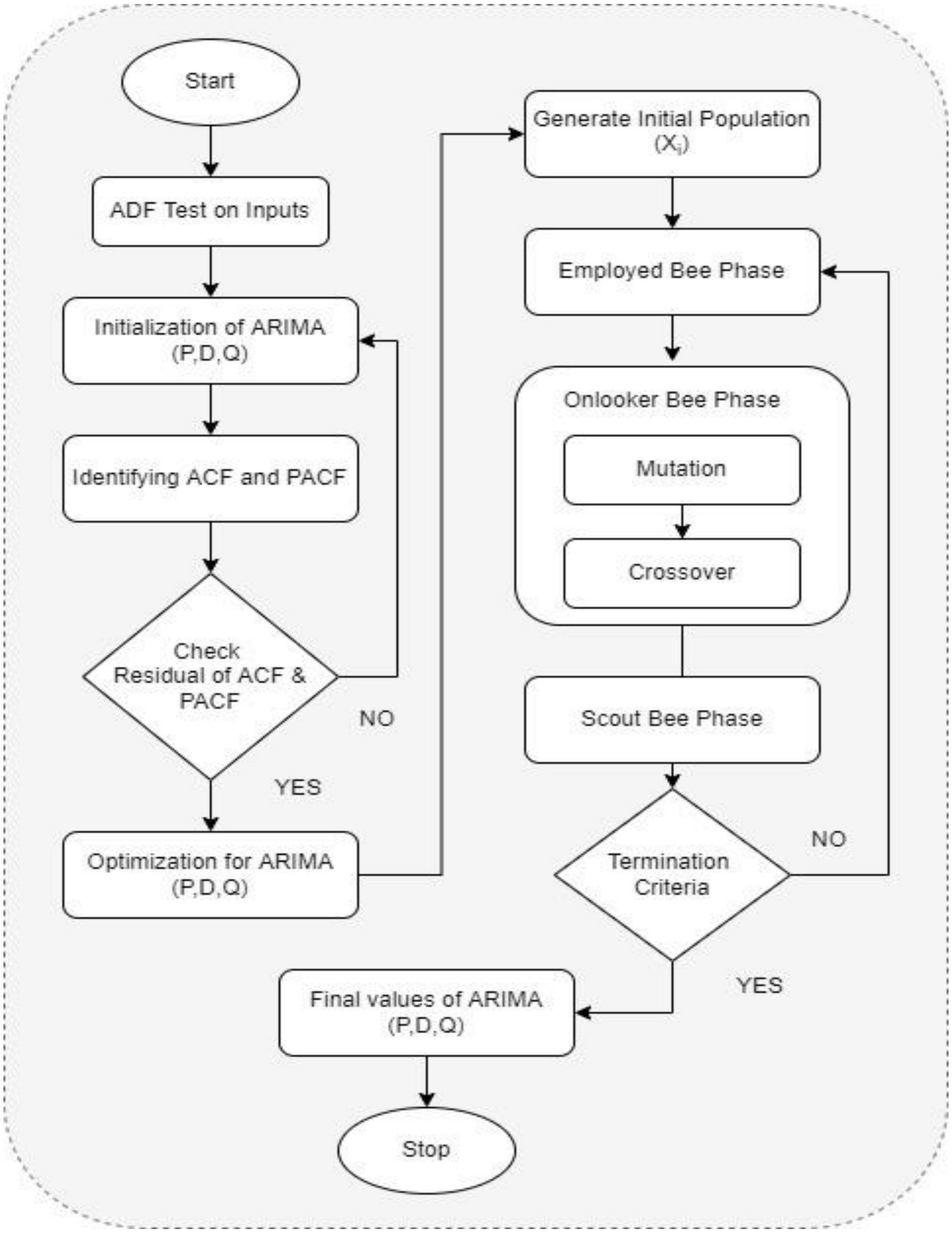
Start
ADF Test on Inputs
Initialization of ARIMA (P,D,Q)
Identifying ACF and PACF
Check Residual of ACF & PACF
NO
YES
Optimization for ARIMA (P,D,Q)
Generate Initial Population (X_i)
Employed Bee Phase
Onlooker Bee Phase
Mutation
Crossover
Scout Bee Phase
Termination Criteria
NO
YES
Final values of ARIMA (P,D,Q)
Stop

Figure 4.1: Modified DE-ABC-ARIMA Algorithm Flowchart

Table 4.1: Dataset Description based on Closing Value [71]

Ticker Value	**count**	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
BPCL	2460	212.91391	141.29129	39.906464	65.261169	202.25322	344.11088	516.3291
GAIL	2460	123.12976	28.673799	69.400002	100.50175	118.568	139.7215	196.89999

GSPL	2460	132.23215	53.106373	49.049999	84.225	124.975	180.275	262.5
HINDPETRO	2460	182.15559	114.03113	36.2444	75.241699	169.739	268.31249	488.35001
IGL	2460	164.16574	124.80265	38.669998	64.352503	93.84	270.775	522.29999
IOC	2460	110.57096	45.485315	48.224998	76.384377	94.549999	144.76875	227.35001
MRPL	2460	69.224573	25.262513	22.299999	53.537499	65.050003	77.75	142.35001
OIL	2460	220.27907	49.747255	70.349998	187.45625	228.05	253.52	330.60001
ONGC	2460	178.67292	40.75766	60	157.54575	180.60001	198.13775	310.43301
RELIANCE	2460	700.97274	390.22876	338.04999	434.15001	501.425	929.96248	2177.7

Moving Average is used to calculate the high order autoregressive model (MA). To put it another way, IACF is an ACF model that only employs moving averages. The influence of IACF decreases when the considered lag value is greater than observed p value. The stationary characteristic of time series data is demonstrated by ACF plots for selected indices, which show large declines with respect to k delays. The proposed model fitness is validated using the diagnosed data and plots of trends and correlation analyses. The model was evaluated using the DE-ABC-ARIMA. If a model fails to obtain a minimum AIC for the goal function, the process of parameter selection and model verification is frequently repeated. Surprisingly, all selected oil drilling and exploration equities, as well as refineries in the oil and gas industry, display identical patterns. The time series plot of the specified historical dataset is displayed in each stock panel. The Autocorrelation Function (ACF) behaves differently depending on the range of lags. The obtained ACF plots and PACF graphs depict the consistent performance of datasets stationarity over a 10-year period.

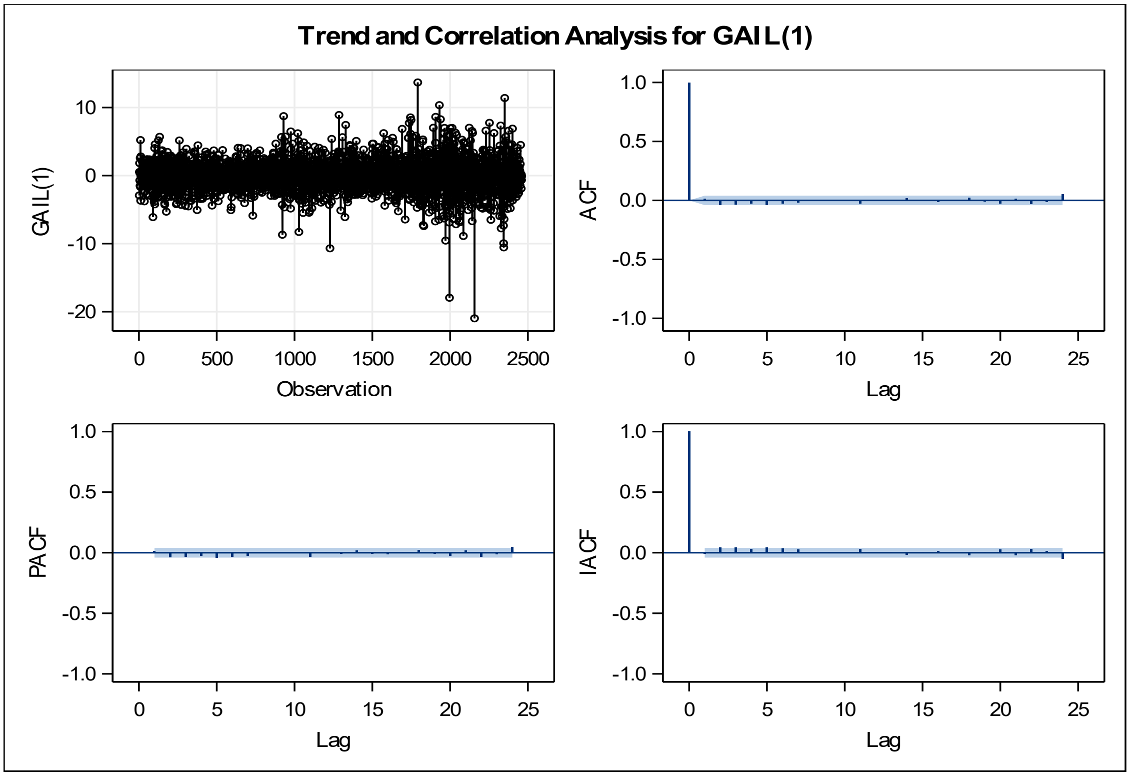

Figure 4.2: Correlation Analysis and Trend for GAIL

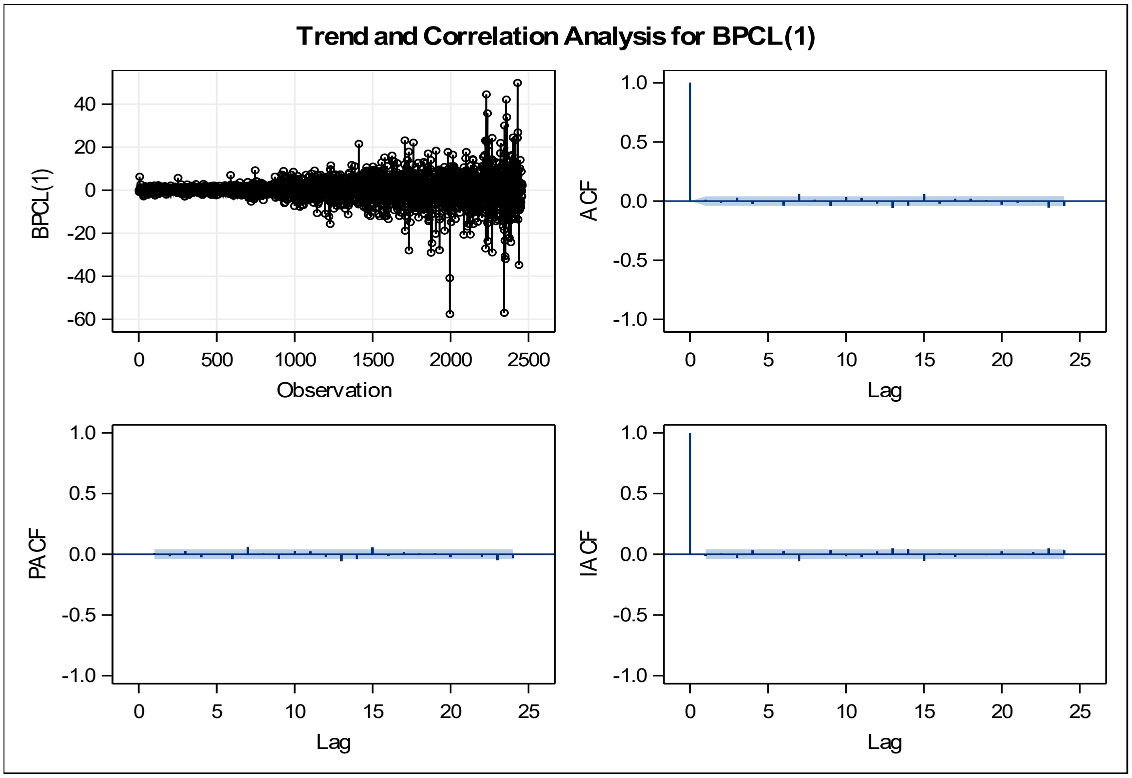

Figure 4.3: Correlation Analysis and Trend for BPCL

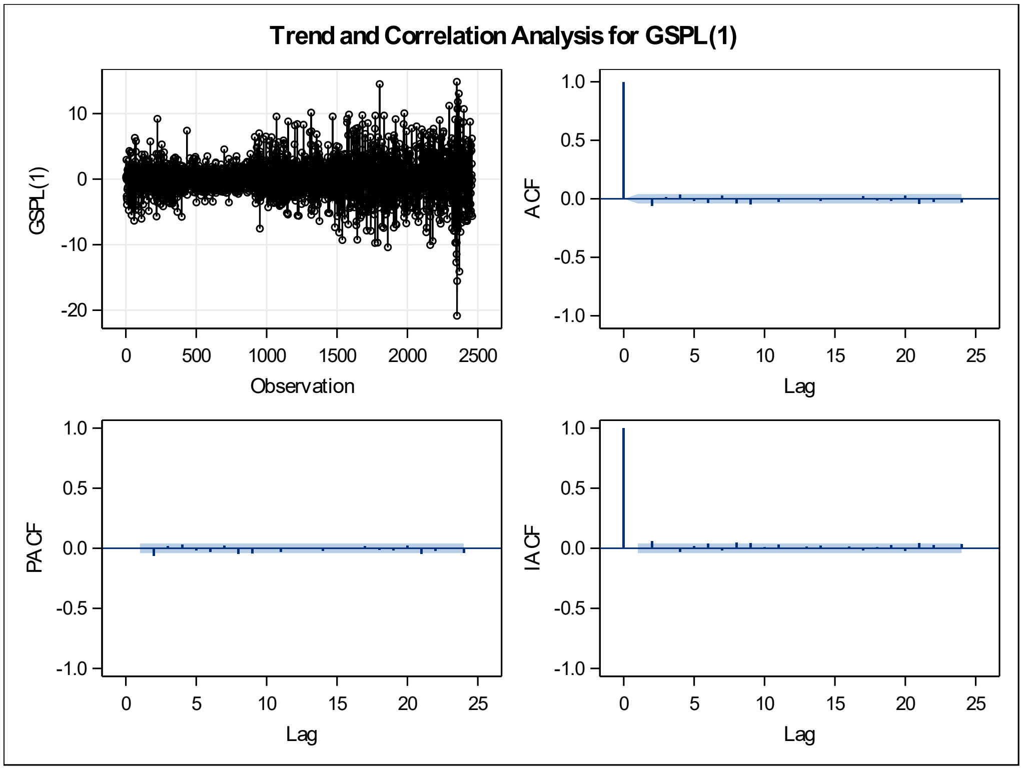

Figure 4.4: Correlation Analysis and Trend GSPL

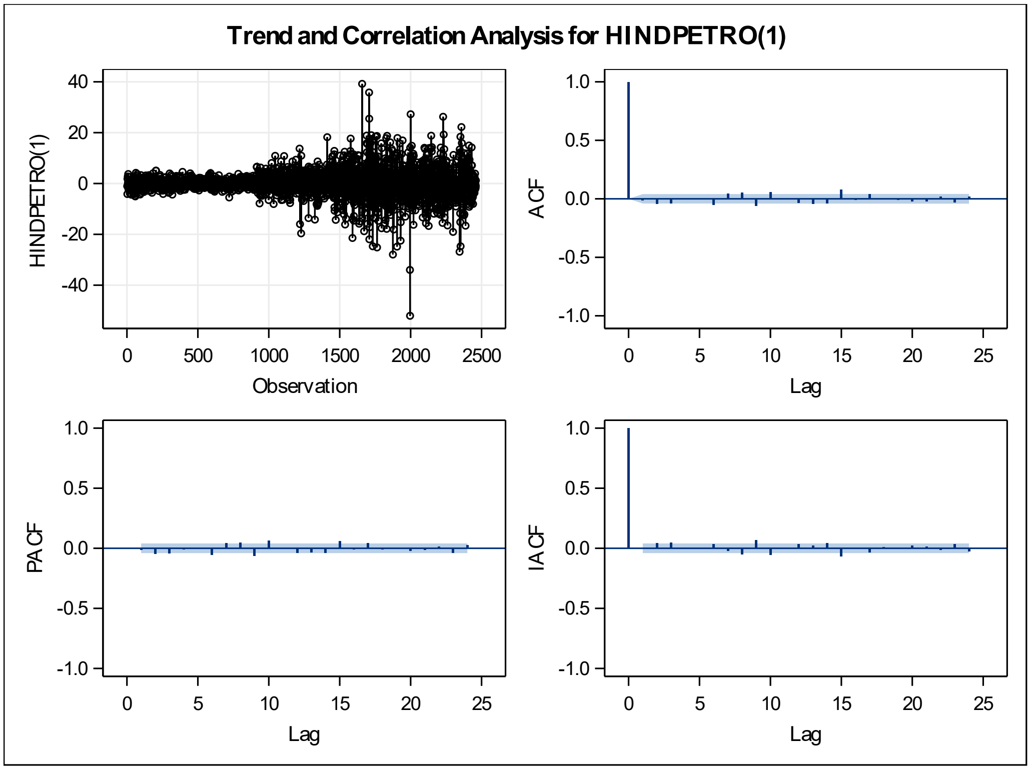

Figure 4.5: Correlation Analysis and Trend HINDPETRO

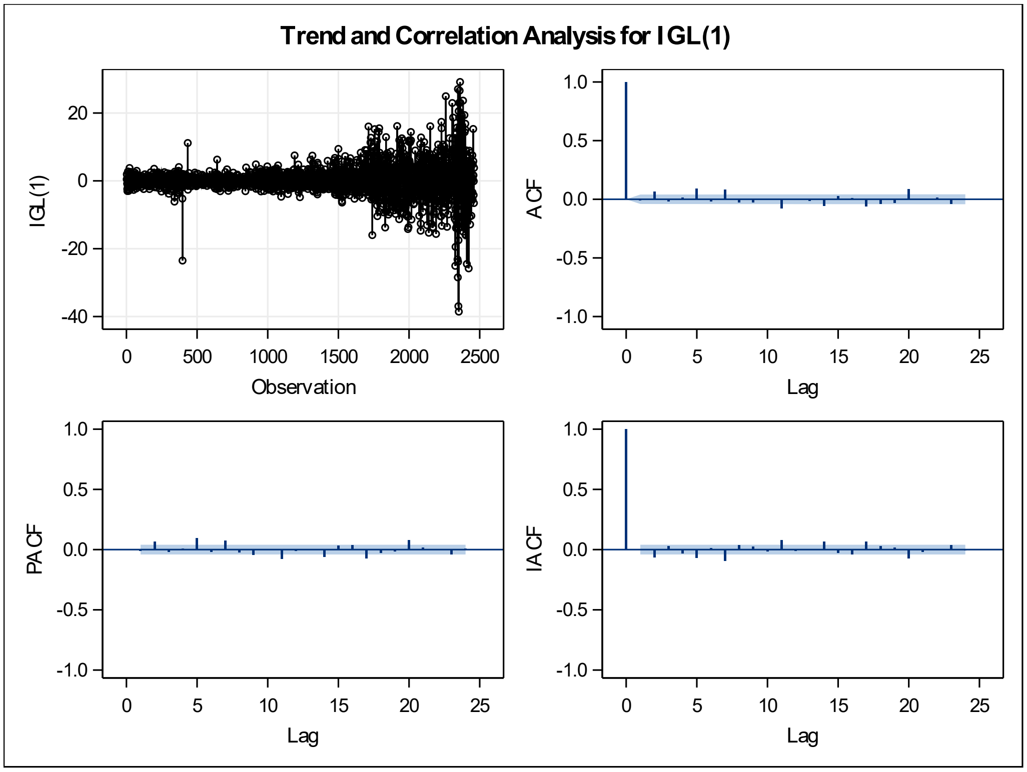

Figure 4.6: Correlation Analysis and Trend for IGL

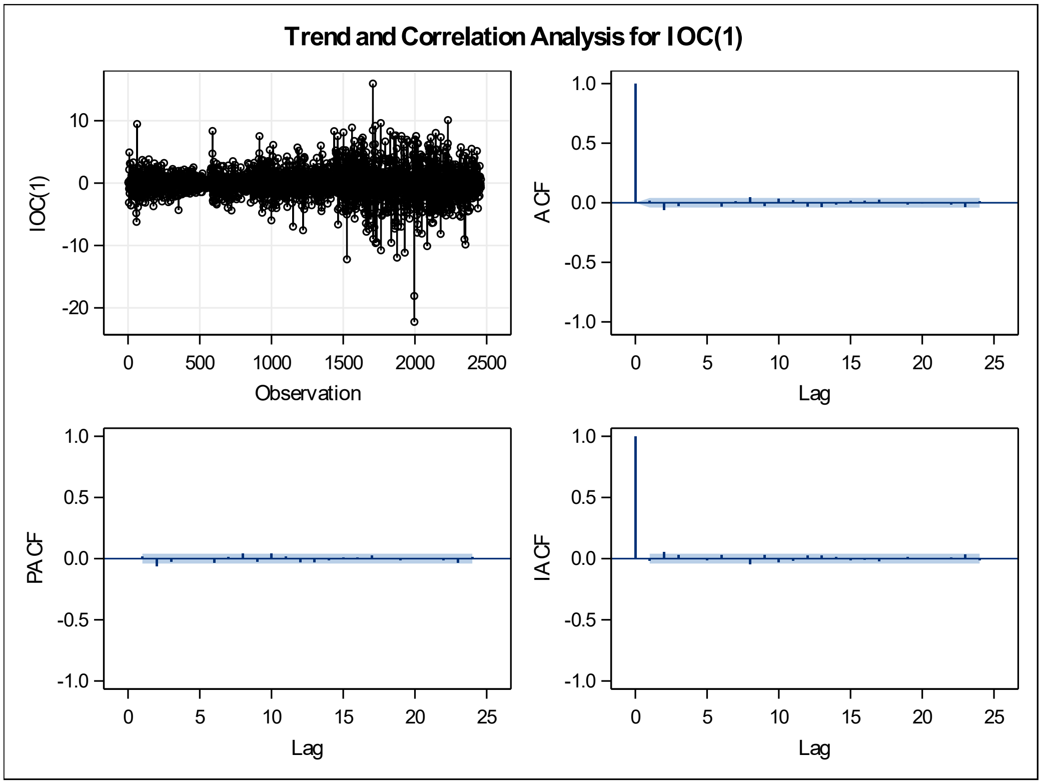

Figure 4.7: Correlation Analysis and Trend for IOC

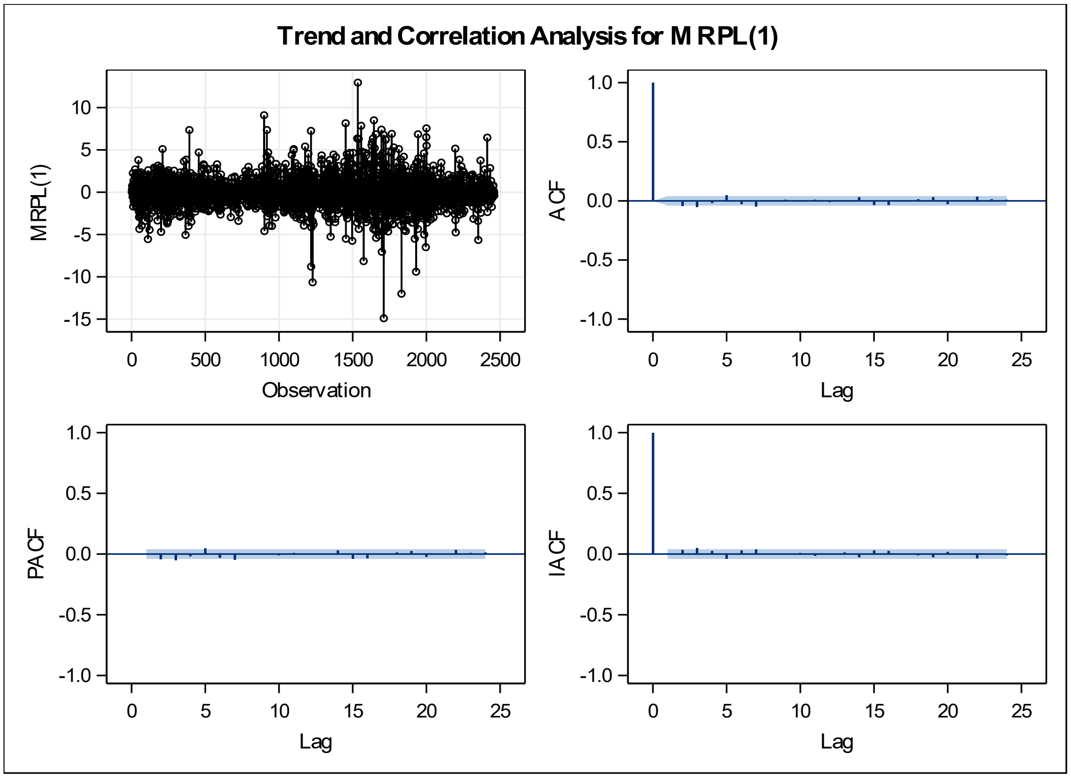

Figure 4.8: Correlation Analysis and Trend for MRPL

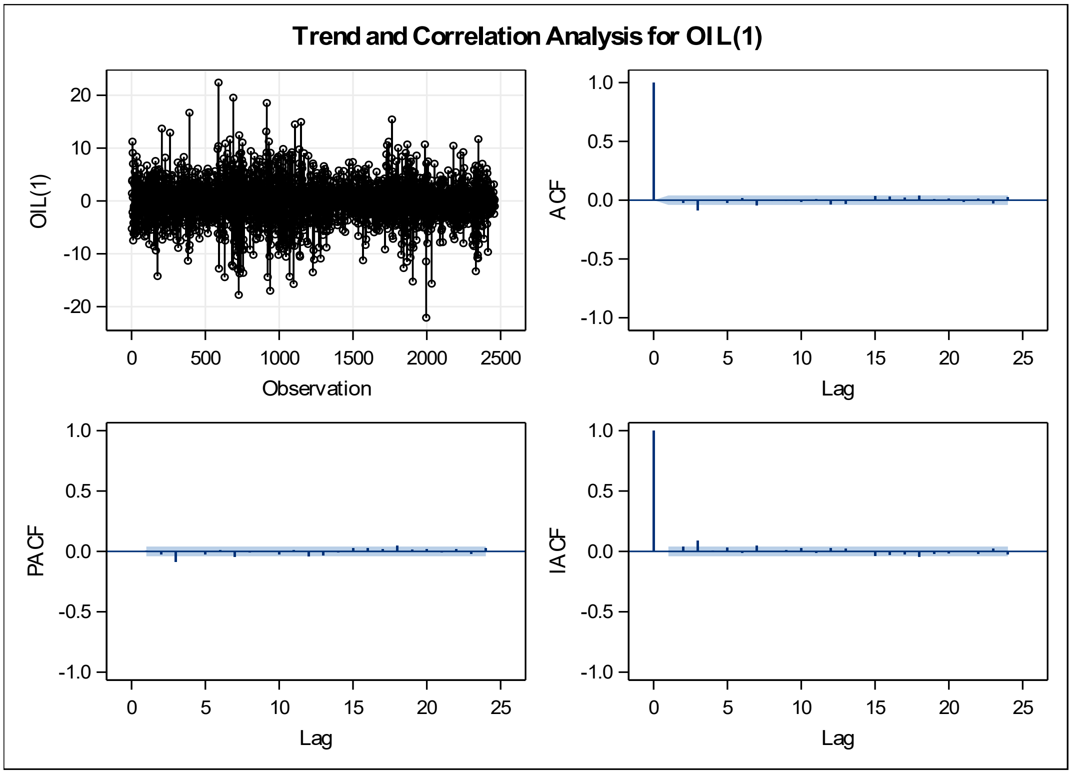

Figure 4.9: Correlation Analysis and Trend for OIL

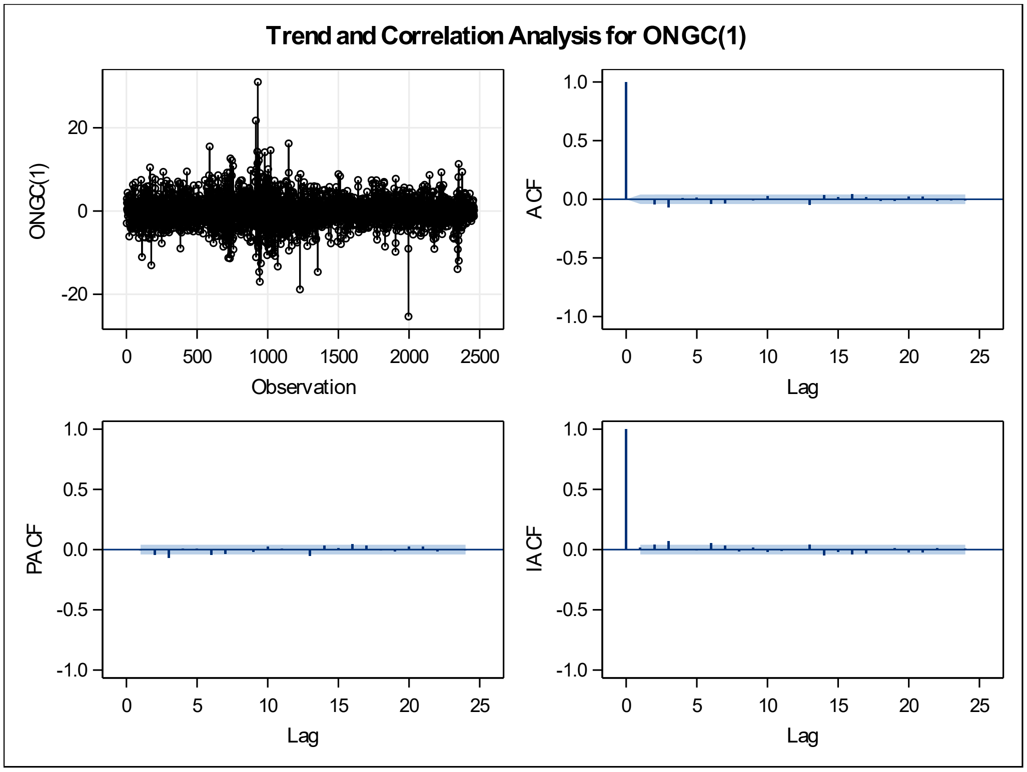

Figure 4.10: Correlation Analysis and Trend ONGC

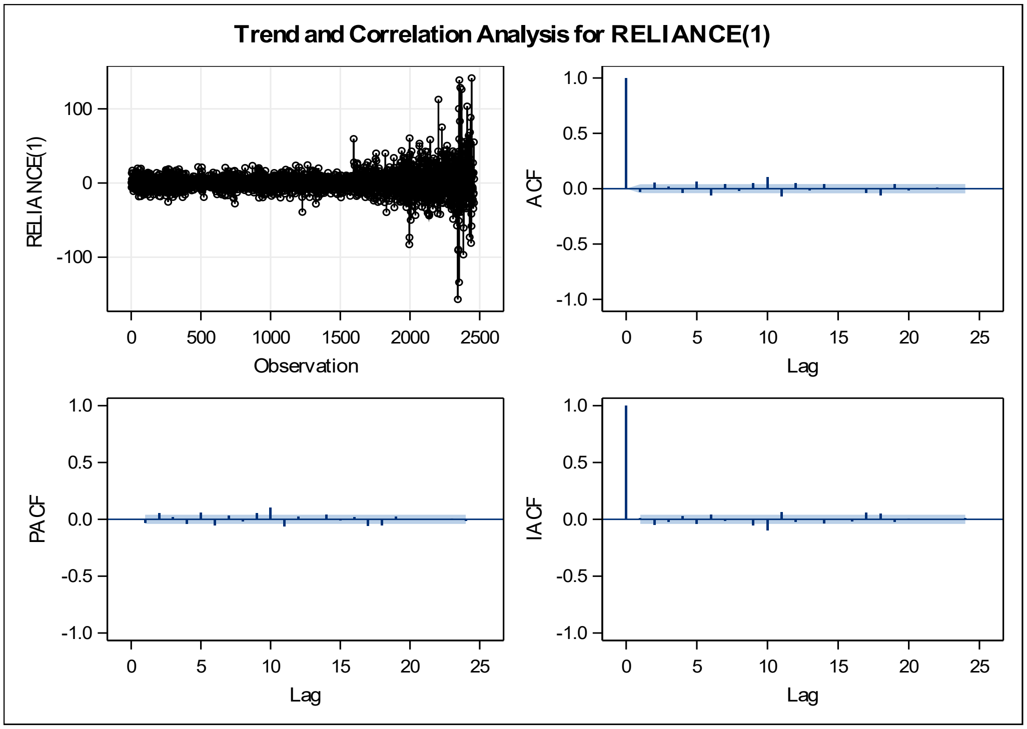

Figure 4.11: Correlation Analysis and Trend RELIANCE

4.7.2 Forecasting Analysis

Previous research [80, 81] presented a range from (p=0, d=0, and q=0) to (p=0, d=0, and q=1) to calculate the minimum AIC. The number of engaged bees of ABC are equal to the number of ARIMA model parameters. As a result, the number of ARIMA parameters is equal to the length of ABC's food supply (3). For historical dataset experiments, 100 iterations are taken into account. For the suggested ABC-ARIMA model, the selected stocks on the NSE and BSE in India gain RMSE. Table-4.2 shows the low AIC and BIC values of some of the experiments that were run to get the best prediction accuracy. RMSE is measured as squared value, resulting known as a scale-dependent metric that determines the discrepancy between both the stock's actual worth and its expected values using the model provided. Surprisingly, based on stock values and erratic behavior, the RMSE improved by 2.40 % to 24.33 %.

$$\text{Root Mean Square Error}(RMSE) = \sqrt{\sum_{i=1}^{n}(Y_t - Y'_t)^2/N} \qquad (4.15)$$

4.7.3 Multi-step Ahead Prediction

The future projection of a selected stock is explored in this work as a case study of multi-step TSA. Multi-step TSA considers market uncertainty and momentum to aid in financial decisions such as stock returns and portfolio optimization. Taieb et al. present five strategies for multi-step forward forecasting based on previous work including multiple input and multiple output (MIMO) and recursive direct [107, 108].

The direct recursive technique, which is the most intuitive way among all, is used in this study for multi-step ahead prediction. Model (h) from f_h for TSA described in equation 4.16 [109], the Direct Recursive (DirREC) technique is used.

$$Y_{t+h} = f_h(Y_{t+h-1} \ , \dots \dots \dots Y_{t-d+1}) \qquad (4.16)$$

Where Y_{t+h} is the projection of closing values of specified NSE & BSE equities from September 1, 2010 to August 31, 2020, as shown in Table-4.1. SAS studio is used to plot the calculated values of the proposed model. The suggested model supports the premise and outperforms traditional ARIMA and ABC-ARIMA models in terms of performance. The equities stated, have a 12-month future prediction with a 95 % confidence limit. Initially, the parameters of the ARIMA model were set to (0,0,0-12,12,12), after which the suggested model optimized the parameters and obtained (2,1,3) as the best fit combination. The anticipated values of chosen stocks are close enough to the actual values, as seen in Figures 4.12-4.21. This study establishes and defines a 12-month consistent trend as multi-step ahead prediction for all the selected refineries equities. The suggested model ABC-DE-ARIMA supports the premise and outperforms traditional ARIMA and ABC-ARIMA models in terms of performance. The equities stated, have a 12-month future prediction by considering 95 % confidence limit in the experiments.

Table 4.2: Performance Analysis of Proposed Model

STOCK	ARIMA (RMSE)	ABC-ARIMA (RMSE)	ABC-DE-ARIMA (RMSE)
BPCL	12.6714	10.4522	10.1395
GAIL	11.0056	9.8686	8.8556
GSPL	4.8904	3.3467	3.2345
HINDPETRO	6.5674	5.8763	4.9765
IGL	10.5784	9.3434	9.459
IOC	5.6575	4.2129	4.878
MRPL	4.7723	3.8903	3.7357
OIL	4.8975	3.9	3.878
ONGC	3.8965	3.2105	3.5711
RELIANCE	16.3453	13.9045	13.4578

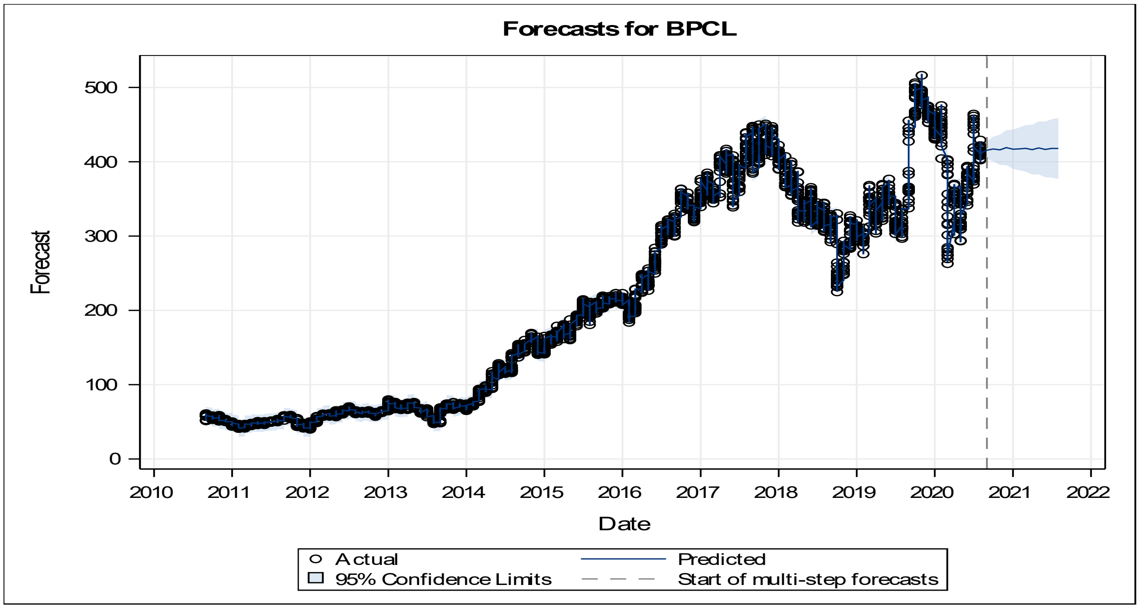

Figure 4.12: Multi-step ahead Forecasting for BPCL

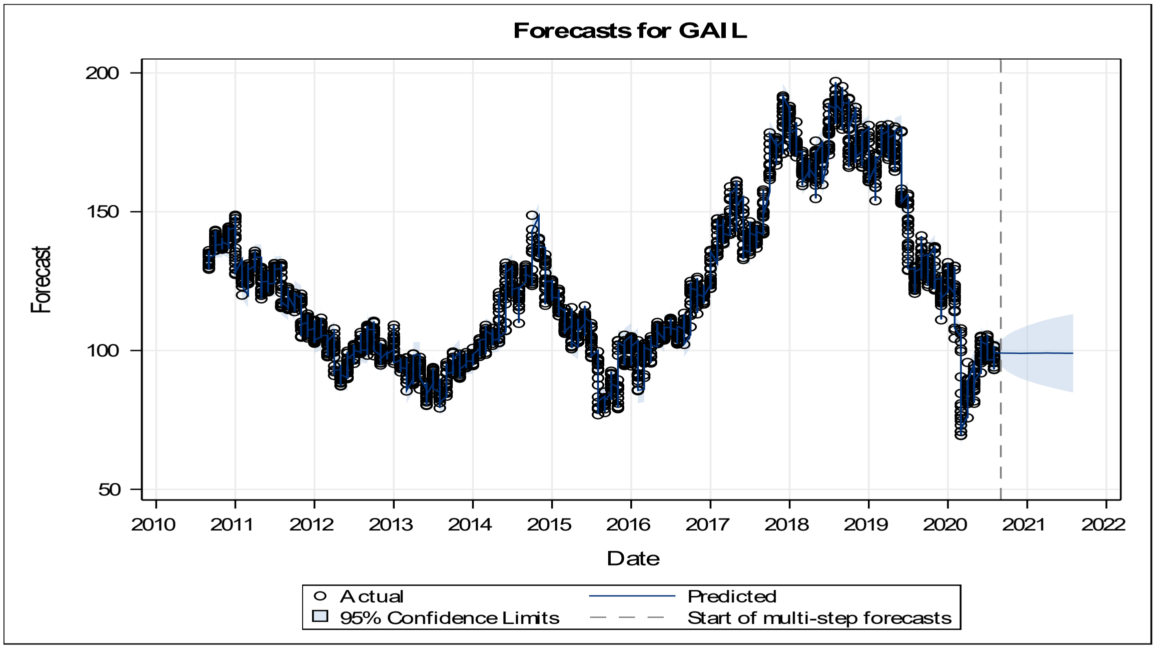

Figure 4.13: Multi-step ahead Forecasting for GAIL

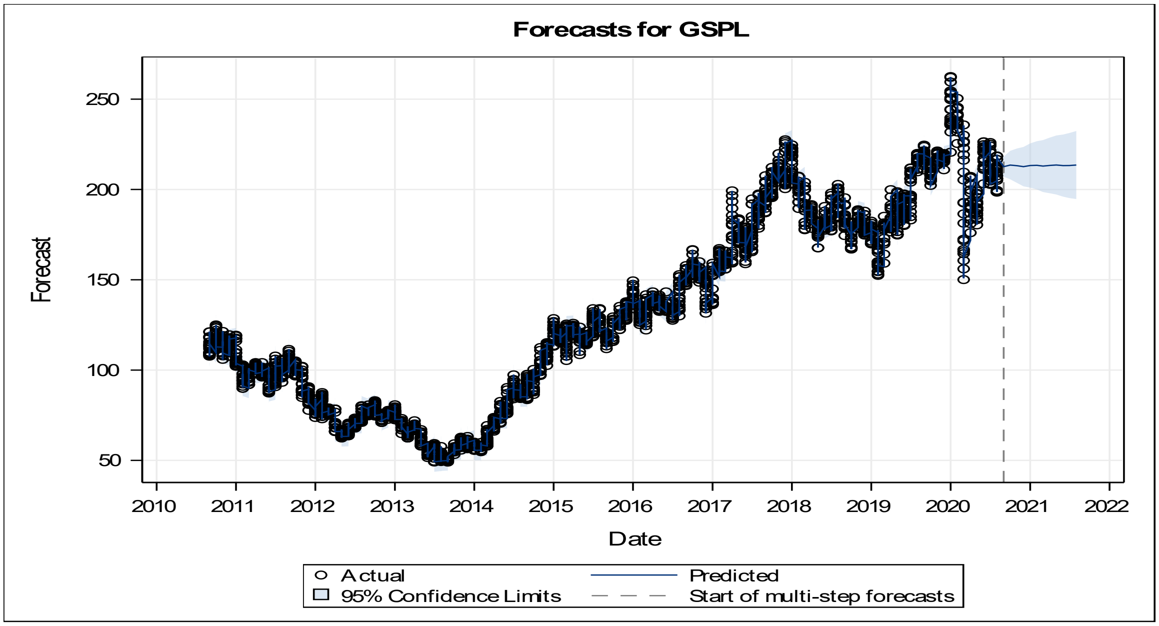

Figure 4.14: Multi-step ahead Forecasting for GSPL

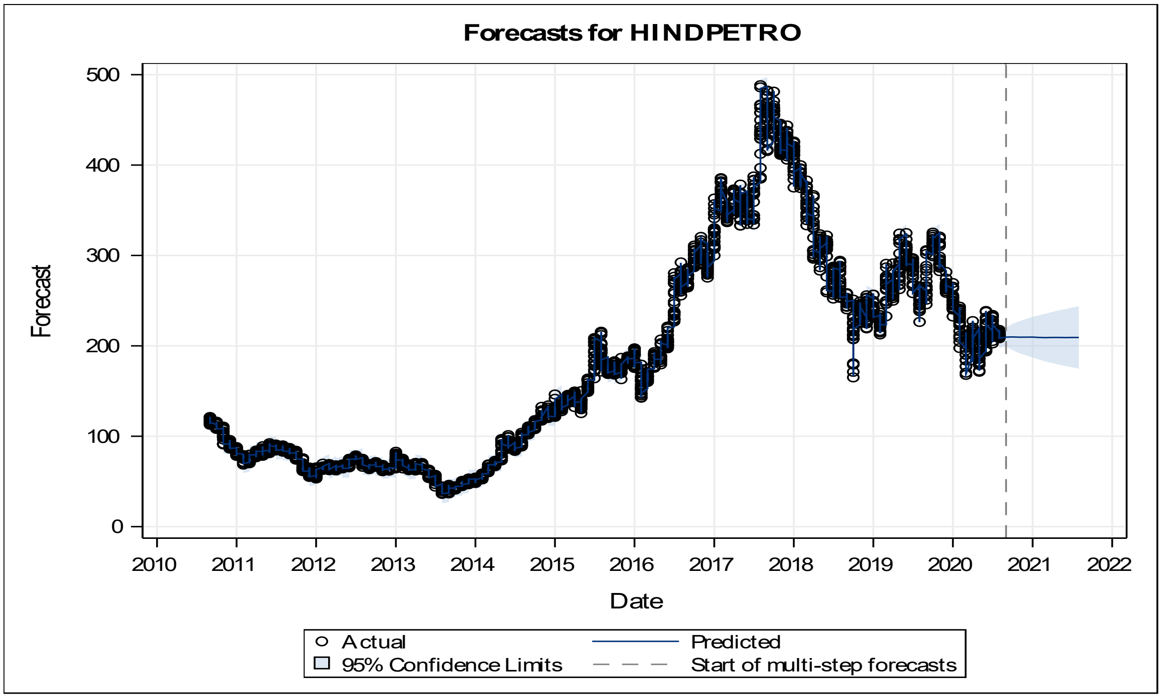

Figure 4.15: Multi-step ahead Forecasting for HINDPETRO

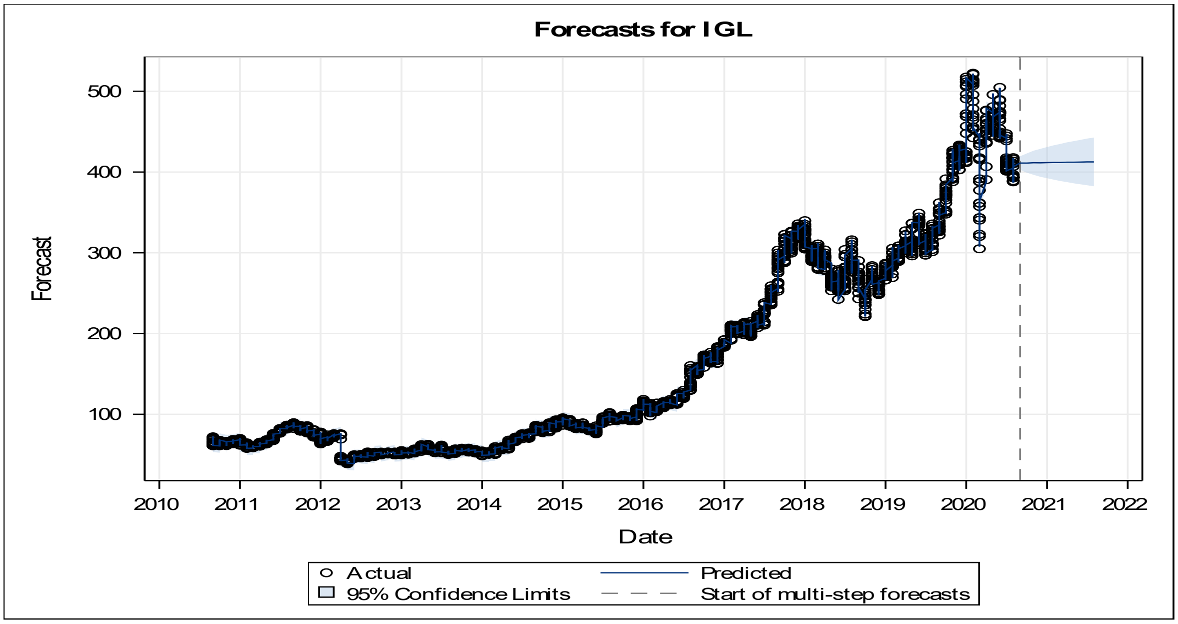

Figure 4.16: Multi-step ahead Forecasting for IGL

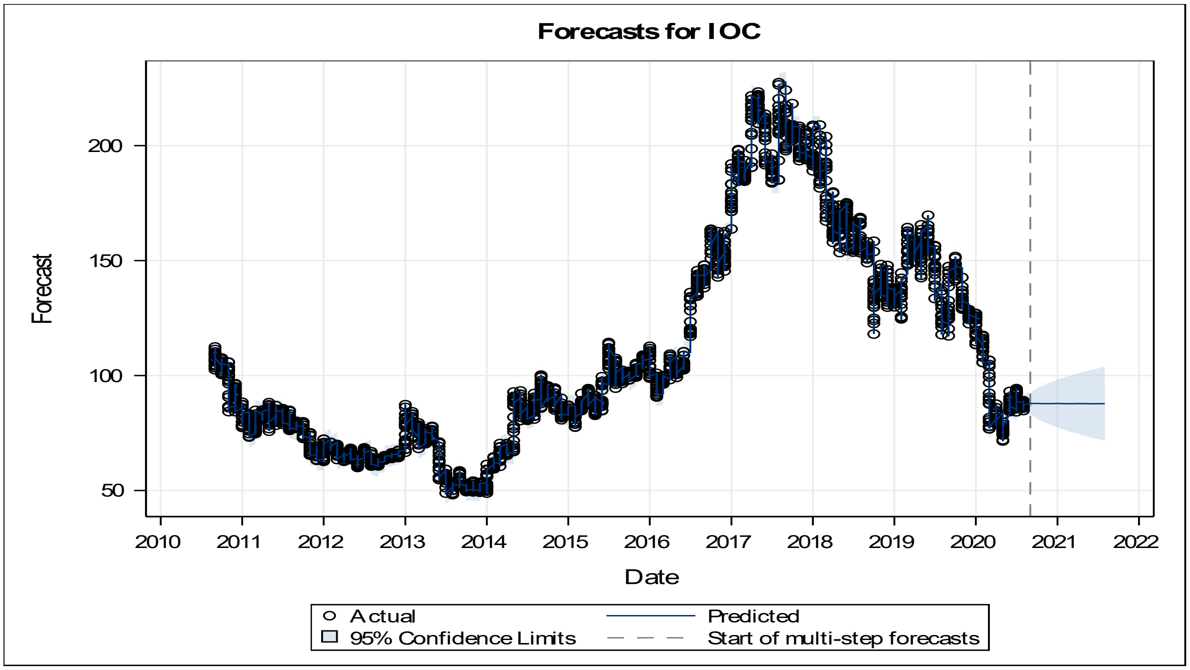

Figure 4.17: Multi-step ahead Forecasting for IOC

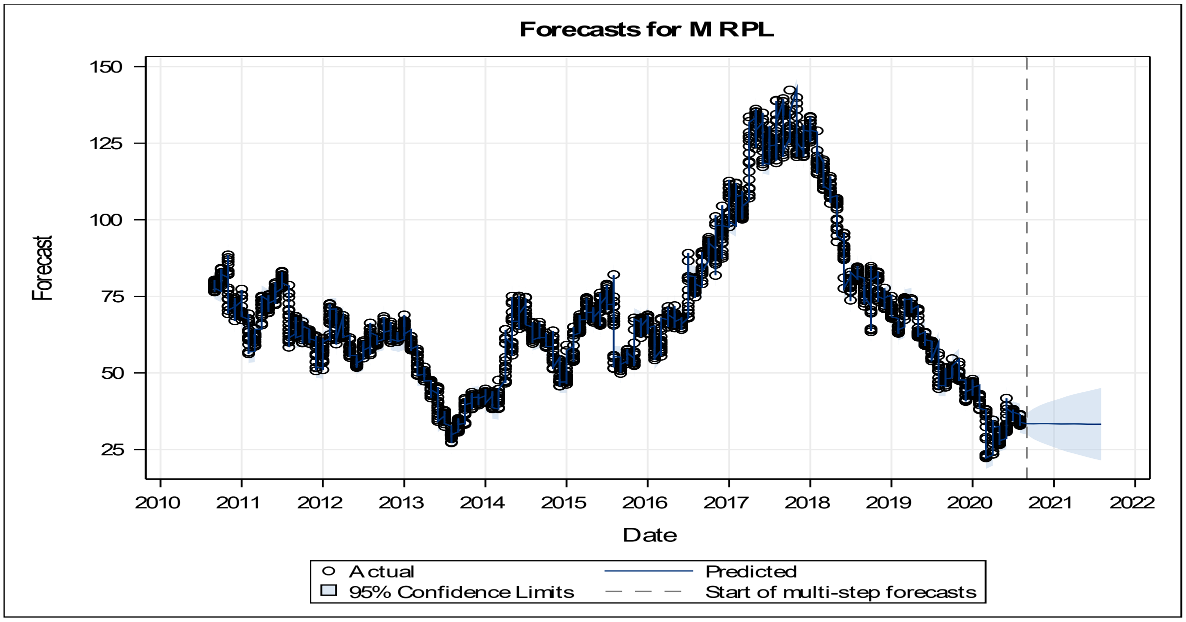

Figure 4.18: Multi-step ahead Forecasting for MRPL

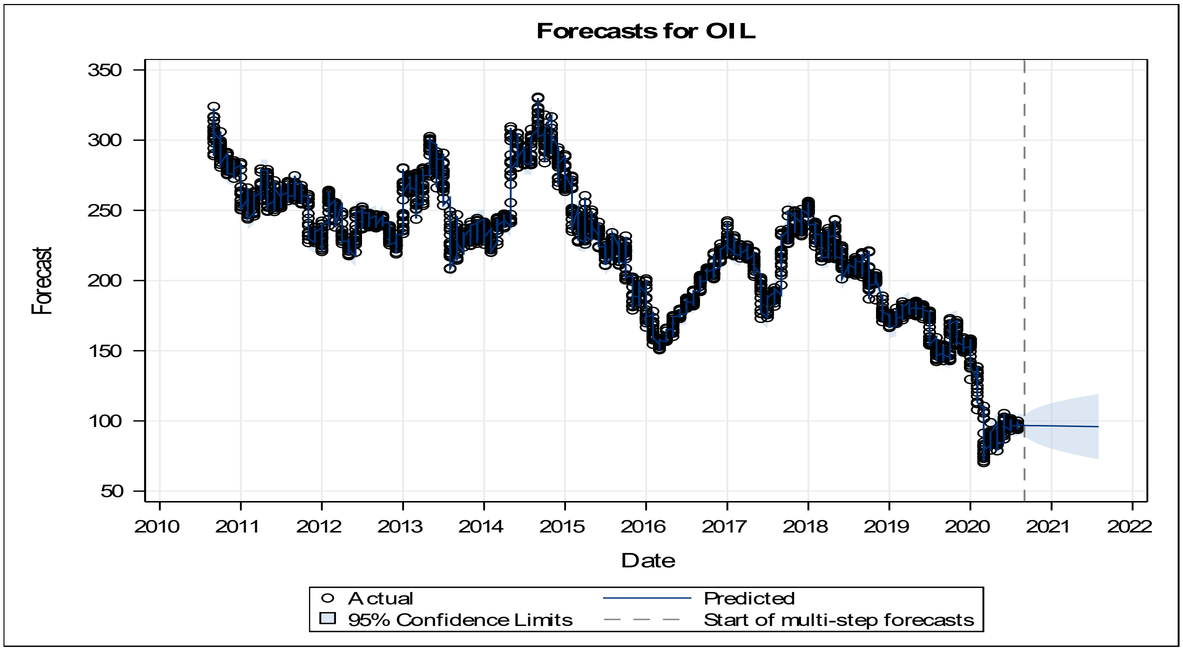

Figure 4.19: Multi-step ahead Forecasting for OIL

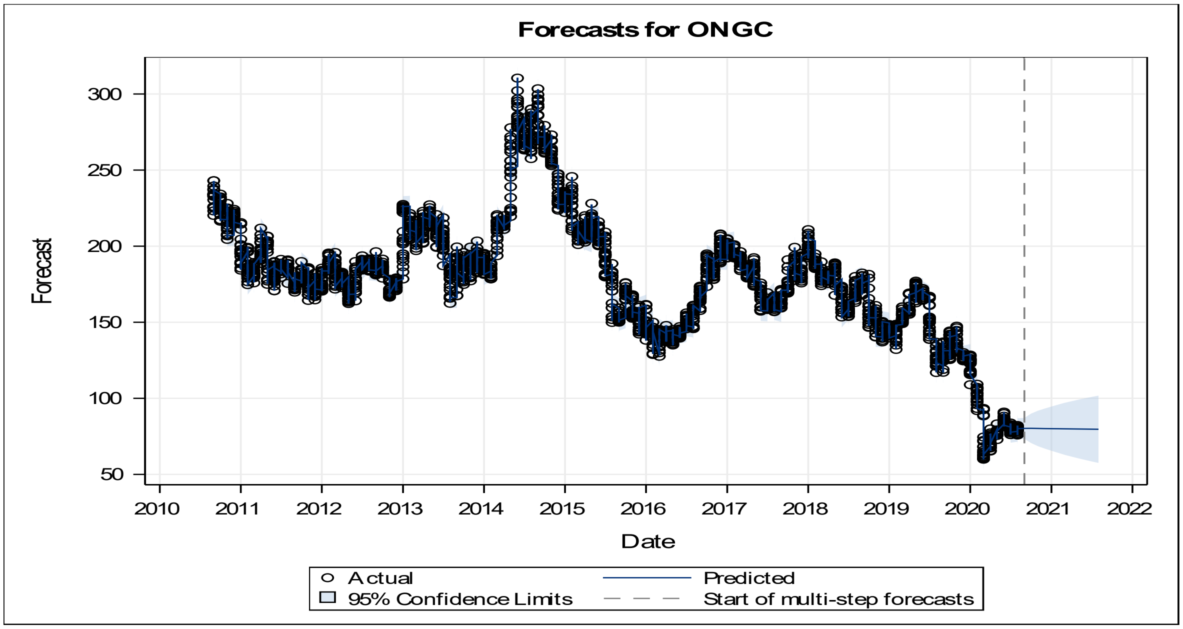

Figure 4.20: Multi-step ahead Forecasting for ONGC

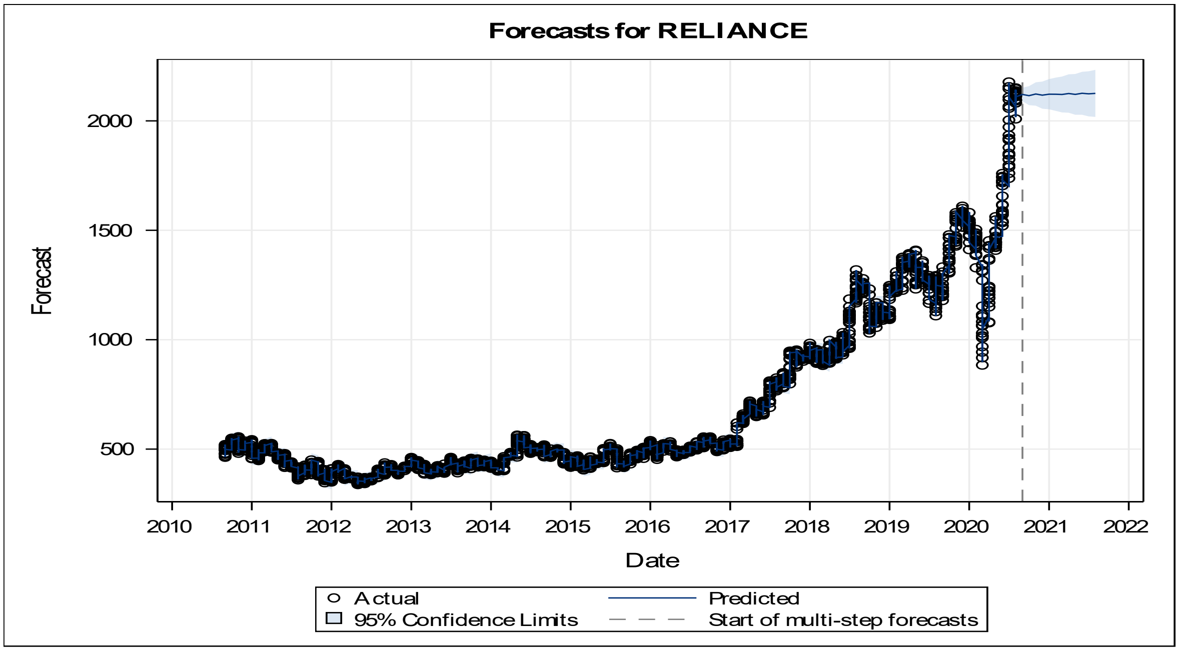

Figure 4.21: Multi-step ahead Forecasting for Reliance

4.8 Summary

One-step ahead prediction approach works effectively in the stock market to obtain higher accuracy. Multi-step ahead prediction, on the other hand, has yet to establish itself as a widespread tool and strategy among investors due to its complexity. This chapter developed a modified ABC-ARIMA hybrid model based on DE. The proposed technique DE-ABC-ARIMA reports the exploitation challenges and exploration issues of standard algorithm. The proposed fusion improves data trends conducted on 10 year’s refineries datasets. Surprisingly, due to volatile behavior given stock value range of RMSE improves the MRPL and GSPL 2.40 % to 24.33 %, respectively. The obtained results show that in multi-step TSA, the suggested modified ABC-ARIMA hybrid model outperforms its competitors. Examining nonlinear patterns is a major subject in the TSA, particularly in stock market. The nonlinear LSTM is well-established in the research community. Hybridization of such models can be useful for dealing with both linear and nonlinear dataset patterns. As a result, this topic can be considered a future prospect of this research that should be investigated further.

CHAPTER 5

ADDRESS LINEAR AND NONLINEAR ISSUES OF TIME SERIES DATA USING HYBRID CLASSIFIERS

5.1 Introduction

This chapter handles the linear and nonlinear issues of time series data using hybrid classifier DE-ABC-LSTM-ARIMA. In this chapter ARIMA and LSTM are merged in the initial phase of hybridization to synthesize linear and non-linear aspects of the dataset. The proposed LSTM-ARIMA hyperparameters are tuned with DE improved ABC. From September 1, 2010 to August 31, 2020, experiments are conducted using datasets of refineries sectors NSE & BSE. The collected findings indicate that hybrid approach outperforms the benchmark models ARIMA, LSTM, and hybrid ARIMA-LSTM.

5.2 Contribution of the chapter

Original contributions of this chapter are as follows.

1) This chapter proposed a novel framework DE-ABC-LSTM-ARIMA for model fusion and algorithmic fusion to improve the prediction accuracy for selected stock indices.
2) Heterogeneous datasets of refineries sectors from NSE & BSE considered as benchmark datasets of 10 years duration from September 1,2010 to August 31, 2020, for the experiments.
3) In addition, hyperparameters of the proposed model were optimized through improved algorithmic fusion ABC-DE. The DE-ABC-LSTM-ARIMA hybridization ensure the steadiness between the exploration and exploitation challenges of the hyper parameters of proposed model.
4) The two-phase hybridization (LSTM-ARIMA and DE-ABC) developed in this study outperformed benchmark models ARIMA-LSTM and individual models ARIMA and LSTM.

5.3 Proposed Hybrid Model

According to recent works, increasing stock market forecasting accuracy is a critical and difficult task for traders, investors, and financial decision-makers. Because of the requirement of the financial market, improvements in the efficiency of prediction models are never halted. Individual model performance is low due to feature restrictions, according to recent research. On the other hand, the outcomes of ensemble forecasting or the creation of hybrid models to combine error series lead to improved performance. The second technique, which assumes that the dataset contains both linear and nonlinear patterns in itself, has been shown to be better for time series prediction in recent years. Because individual models are unable to read both patterns equally well, integrating the results of models with diverse behavioral patterns is likely to produce better outcomes. In this chapter, the LSTM-ARIMA is suggested to assemble both linear and nonlinear features. The ARIMA model addresses the dataset's linear features and generates a linear forecast, whereas the LSTM model solves over-fitting, under-fitting, and other well-known non-linear model flaws. In proposed model where L_{ts} is a linear pattern and N_{ts} is non-linear pattern of at time series Y_{ts}.

$$Y_{ts} = L_{ts} + N_{ts} \tag{5.1}$$

Initially, linear components are modeled through ARIMA and then non-linear components are modeled through the LSTM unit. The residuals of linear model ARIMA hold the non-linear structure of selected stock market time series datasets. Hence, non-linear segments can be addressed using residuals (e_{ts}) using equation 5.1.

$$e_{ts} = Y_{ts} - L_{ts} \tag{5.2}$$

Considering, LSTM window length (n) and random error ($\mathcal{E}_{ts}$) for the residual e_{ts} is obtained through equation 5.2.

$$e_{ts} = f(e_{ts-1} , e_{ts-2} , e_{ts-3} ------, e_{ts-n})+ \mathcal{E}_{ts} \tag{5.3}$$

In the second approach where nonlinear structures are handled by LSTM units then the residuals will hold the linear structure only. Hence, linear structure will be addressed by the ARIMA model. Thus, the proposed model imposes the strength of ARIMA as well as LSTM units. The proposed LSTM-ARIMA is used firstly to handle non-linear patterns from the real time series datasets used in the experiment section. If the residual of LSTM unit and N_{ts} is considered as an output at time series (ts) which contains the linear pattern which is handled by ARIMA model in equation 5.3.

$$e`_{ts} = Y_{ts} - N_{ts} \tag{5.4}$$

Considering that, the number of lags (m) for ARIMA model and random error ($\mathcal{E}_{ts}$) for the residuals ($e`_{ts}$) obtained through equation 5.4.

$$e`_{ts} = f(e`_{ts-1}\ ,\ e`_{ts-2}\ ,\ e`_{ts-3} - - - - - -,\ e`_{ts-m}\ ) + \mathcal{E}_{ts} \tag{5.5}$$

In addition, the comparative study between obtained results demonstrated by both the hybrid models is found far better than its individual performances. However, LSTM-ARIMA yields superior performance for the stock market time series than its counterpart ARIMA-LSTM.

5.4 Improved ABC using DE (Algorithmic Fusion)

The difficult challenge is to choose Hyper parameters for the proposed hybrid LSTM-ARIMA model. On the other hand, for multi-objective optimization problems like time series forecasting, evolutionary algorithms and nature-inspired algorithms are the best answers. A previous study found that evolutionary algorithms suffer from huge population sizes, resulting in premature convergence and long computation times. As a result, choosing ABC and DE algorithms for a hybrid model allows for successful exploration and exploitation of search space. However, due to design constraints and the use of commodity hardware in the trials, the enhanced algorithm's computational complexity is on the upper side.

The ABC algorithm was inspired by the natural activity of bees and their hive. For parameter optimization, it is regarded a typical swarm intelligence method. The bees are categorized into three types based on their behavior: employed bees, observer bees, and scout bees. The bees that have been hired are intended to look for food sources. The colony's goal is to obtain the most nourishment, which is known as nectar. Onlooker bees do not leave their hive until they receive signals and information from working bees. Scout bees seek for fresh prospective nectar near their hive at the end of the day. When the employed bee's nectar runs out, it transforms into a scout bee, looking for a new job. Employed bees and observer bees are constrained by nature, whereas scout bees are unbounded and unrestricted while foraging for nectar. ABC has been found as a tried-and-true high-quality optimum algorithm for multidimensional problems [28]. ABC is notable for its ability to solve multi-objective problems with global exploration challenges. However, due to local exploitation, it is usually reported in local search efficiency [29].

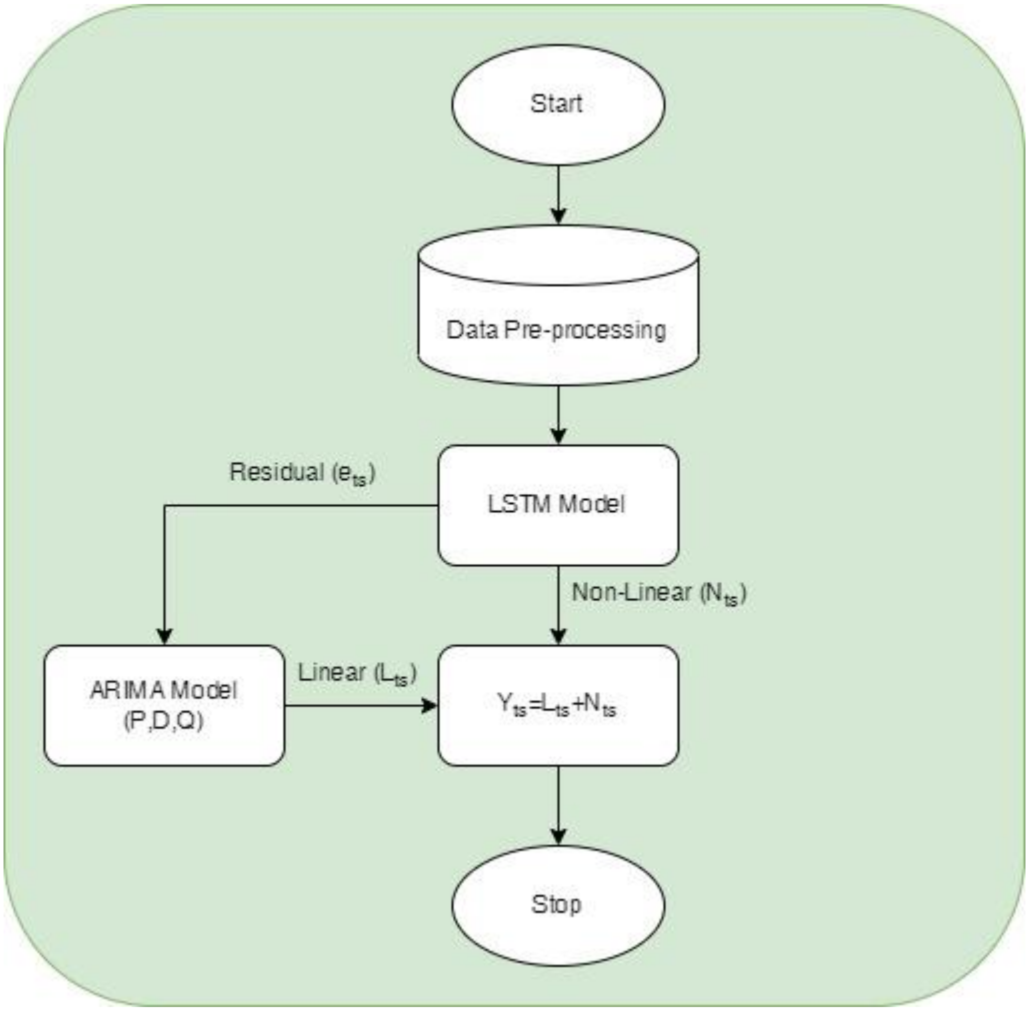

Figure 5.1: Proposed Hybrid Model of LSTM-ARIMA

Differential Evolution (DE), on the other hand, excels at finding strategies in real-world challenges. Using employed bees, the standard ABC method manages global exploration difficulties. The exploitation abilities are handled by the observer bees phase. However, the suggested model LSTM-hyper ARIMA's parameter selection encounters a local search difficulty and is subject to exploitation limits. As a result, for the hybrid model LSTM-ARIMA, this work developed a new hybrid algorithm to address multi-objective issues such as hyperparameter selection. DE improves the local search process by modifying the onlooker bee phase of the ABC algorithm. Using hybrid search strategies, it improves search efficiency. The suggested algorithm's flow chart (Fig.5.2) is based on DE's current-to-best/1 mode [44-46].

Algorithm 5.1: DE Optimized Artificial Bee Colony (ABC) for Hybrid LSTM-ARIMA

```
Step-1 Initialization of population
Step-2 Employed bees movement to food source and compute nectar amount
Step-3 Onlooker bees movement based on nectar amount
Step-4 Scout bees movement for exploiting new food source
Step-5 Retain best food source
Step-6 Compute best fitness value based on initial papulation for DE
Step-7 Do
          For i =1, where N is a number of particles
                Do
                   {
                       Crossover
                       Mutation
                   }
       End For
Step-8 If termination criteria get failed go to step-2

END
```

Several existing research [48-50] are also used to assess the method's complexity and evaluate the updated approach. W. Xiang and colleagues [48] suggested a unique hABCDE hybrid method and tested it on twenty benchmark functions *(f_1-f_{20})*. The proposed hABCDE method is intended to handle numerical optimization problems. The obtained findings demonstrated that the hABCDE algorithm outperformed other state-of-the-art methods [48]. Zorarpac, E. et al. ran experiments utilized UCI machine learning library. F-measure values as outcomes of the experiments outperformed the conventional ABC and standard DE algorithms [49]. Over 20 benchmark functions, Jadon, S. S., et al. tested HABCDE with *(f_1-f_{20})*. The proposed work was also carried out on four real-world optimization situations *(f_{21}-f_{24})*. The DE-ABC obtained result confirms the individual ABC and DE in terms of accuracy, convergency speed, stability, and robustness [50].

The precision and stability of the hybrid algorithm ABC-DE were found to be significantly better than previous work [48-50]. As a result, the suggested model LSTM-hyper ARIMA's parameter selection is based on a hybridization of ABC and DE.

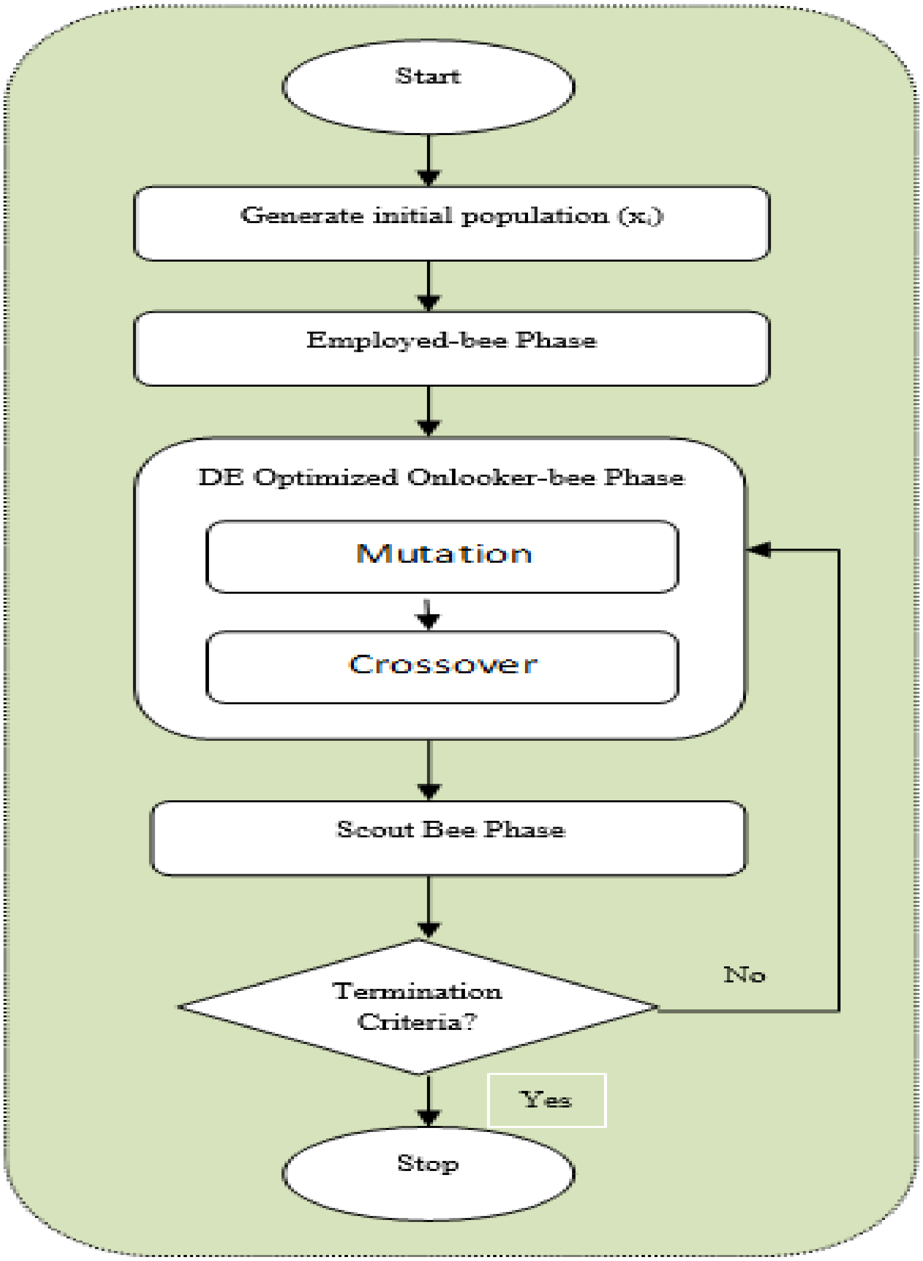

Figure 5.2: Flow chart of improved ABC using DE for LSTM-ARIMA hybrid model

ABC-DE hybridization optimizes hyperparameter selections and minimum RMSE is considered as the fitness value. As DE, you'll take care of the Onlooker Bee phase and choose the appropriate LSTM hyper parameters by evaluating the objective function and minimizing the RMSE. Employee bees evolved into scout bees to continue investigating whether the hyperparameter values had deviated from the exceeding limit. on the other hand, scout bees continue to work on a new LSTM-ARIMA hyper parameter combination to be examined.

5.5 Simulation Result

The suggested algorithm is tested on Python3.0. It is used to preprocess historical data and compute the suggested hybrid model LSTM-ARIMA for a given stock range. SAS Studio is used to perform time series analysis and modelling. To acquire the forecasting findings, the following subsections are evaluated in 5.5.1 Description of the Dataset. The simulation includes the scikit learn, pandas, seaborn, keras, statsmodel, TSA, API for deep learning and TSA implementation.

5.5.1 Dataset Description

From September 1, 2010, to August 31, 2020, a 10-year dataset NSE & BSE are used for the experiment of 2460 instances [71]. Yahoo Finance provides historical datasets with 2460 instances shown in Table.5.1. To obtain realistic performance of the hybrid model, dataset is divided into 80%-20% as training and testing datasets. However, in order to avoid overfitting, a 10k fold validation set is fixed as 25% of training set. Finally, dataset is adjusted using the z-score normalization approach using equation 5.6.

$$X` = X - X_{min} / X_{max} - X_{min} \qquad (5.6)$$

5.5.2 Time Series Analysis and White Noise Test

The white noise test is used to validate a hypothesis. If the residuals are found to be white noise with constant means and variance, the processed series of the considered stocks is called pre-whitened [106]. Table. 5.2 depicts the autocorrelation of selected oil and gas industry benchmark equities. As feature matrices, Table. 5.2 comprises the test statistics (chi-square value), Degree of Freedom (DF), p values, and autocorrelation values. For specified lags, observed p values meet the self-correlation requirements. In several cases, the obtained autocorrelations diverge from 0 for the provided To-Lag range [6,12,18, and 24]. However, the majority of cases are near to 0, indicating that the LSTM-ARIMA model may outperform this financial time series benchmark dataset.

5.5.3 Trend and correlation analysis

The autocorrelation and trend for selected equities are depicted in Figures 5.3-5.12 trend and correlation analysis. Each stock panel displays the Stock ACF, IACF, and PACF as time series plot. (i.e. BPCL, GAIL). With respect to various lag ranges, the ACF's behavior displays. The IACF is calculated using Yule-Walker equations in the proposed hybrid ABC-ARIMA. Moving Average is used to calculate the high order autoregressive model. IACF is an ACF model that uses just moving averages. When the lag value is bigger than the p value, the influence of IACF decreases. In a TSA, the PACF function is modified by an autoregressive Gaussian process at lag k. The stationary characteristic of time series data is demonstrated by ACF plots for selected indices, which show large declines with respect to k delays. If a model fails to achieve the objective function as a minimum of AIC, the process of gets repeated until optimized parameters is established. Surprisingly, all of the stocks display identical patterns, and the ACF and PACF graphs demonstrate the dataset’s continual performance of stationarity.

5.5.4 Residual Normality Diagnostics Analysis

Due to the time series nature of data, applying a machine learning model to a dataset right away is not a wise idea. As a result, residual normality diagnostics study with residual and quantile is performed in this article. This test reveals that the residuals obtained are regularly distributed for each stock in this study. Using QQ plots, Fig.5.13 to 5.22 show the residual and quantile for historical datasets. The departure from a straight line was found to be small in shape in the majority of the plots generated. For time series stock market data, observations around a straight line prove that the distribution is based on normal distribution. The highest frequency is located in the center of the histogram of a normal distribution.

5.5.5 Improved ABC-LSTM Hyper-Parameters Selection

ABC optimal hyper parameters for LSTM models are demonstrated in this research. Table-5.2 shows the list of hyper parameters at the start and after optimization.

The first stage in data preprocessing is to divide selected datasets into training (80% of the time, 1968) and testing datasets (20 % instances, 492). The training dataset represents 80% of total occurrences, whereas the testing dataset represents the most recent examples of the remaining 20%. To obtain realistic performance of the hybrid model, dataset is divided into 80%-20% as training and testing datasets. However, to avoid overfitting, a 10k fold validation set is fixed as 25% of training set. Input length range, number of hidden layer units, and max training data are hyper parameters for the LSTM unit to optimize the parameters.

Step-2 After data preparation, white noise test is used to validate the hypothesis. Step-3 With initial hyperparameters computes the RMSE, as fit value. Employee bees alter the hyper parameter combination based on the lowest RMSE prediction results.

Step-4 By evaluating the objective function and minimizing RMSE, onlooker bees choose the best LSTM hyper parameters.

Step-5 This cycle is repeated until the optimal hyper parameter value is determined by utilizing the lowest RMSE value as the fitness value of the objective function.

5.5.6 Improved ABC-ARIMA Hyper-Parameters Selection

The residual of the LSTM unit is sent to ARIMA to detect linear features of datasets based on the proposed model. As a result, hyper parameters of the ARIMA model are required in this phase, and the actions below are taken to accomplish this.

Step-1 The ADF test is applied on historical datasets are analyzed for stationary test as a preprocessing phase from September 1, 2010, to August 31, 2020.

Step-2 For the ARIMA model, the parameters are initialized with default values at step 2 of the algorithm process. The values of the parameters must be assigned as (0,1,0).

Step-3 Different plots are generated based on the allocated values and assessed for the least residual. The coefficient factors and residual of variance are held during parameter estimation for

ARIMA models. Minimum residual helps to improve the ARIMA with the derived parameter values of p, d, and q.

Step-4 The residual is calculated using the updated hyper parameter values, and the ACF and PACF graphs are rebuilt. The model with the lowest RMSE is the best for time series forecasting, according to AIC.

Figure 5.2 shows the flow chart of the DE-optimized ABC algorithm. For the suggested LSTM-ARIMA model, the selected NSE & BSE equities in India achieve RMSE. By taking the Root of MSE, this error measure compares the unit to the original unit. True observation is used to standardize the Mean Absolute %age Error (MAPE). It plays an important role in improving the accuracy of time series data, such as the stock market, as shown in equation-16[18]. Initially, the parameters of the ARIMA model were set to (0,1,0-12,1,12), after which the suggested model optimized the parameters of p, d, and q and obtained (2,1,3) as befitting values.

Table 5.1: Dataset Description based on Closing Value [71]

Ticker Value	**count**	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
BPCL	2460	212.91391	141.29129	39.906464	65.261169	202.25322	344.11088	516.3291
GAIL	2460	123.12976	28.673799	69.400002	100.50175	118.568	139.7215	196.89999
GSPL	2460	132.23215	53.106373	49.049999	84.225	124.975	180.275	262.5
HINDPETRO	2460	182.15559	114.03113	36.2444	75.241699	169.739	268.31249	488.35001
IGL	2460	164.16574	124.80265	38.669998	64.352503	93.84	270.775	522.29999
IOC	2460	110.57096	45.485315	48.224998	76.384377	94.549999	144.76875	227.35001
MRPL	2460	69.224573	25.262513	22.299999	53.537499	65.050003	77.75	142.35001
OIL	2460	220.27907	49.747255	70.349998	187.45625	228.05	253.52	330.60001
ONGC	2460	178.67292	40.75766	60	157.54575	180.60001	198.13775	310.43301
RELIANCE	2460	700.97274	390.22876	338.04999	434.15001	501.425	929.96248	2177.7

Table 5.2: Improved ABC- LSTM Hyper Parameters

Hyper-parameters	Initial Value	BPCL	GAIL	GSPL	IGL	HINDPE	IOC	MRPL	OIL	ONGC	RELIANCE
lstm_units	50	10	10	128	128	10	128	128	128	128	128
dropout_prob	0.5	0.7	0.5	0.7	0.5	0.6	0.5	0.8	0.5	0.6	0.9
Epochs	10	50	48	50	50	80	50	50	50	50	40
batch_size	32	128	128	64	64	64	64	64	64	128	128
Learning Rate	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Table 5.3: Performance Analysis of Proposed Model for AIC and BIC value

Model	Variable	BPCL	GAIL	GSPL	HINDPETRO	IGL	IOC	MRPL	OIL	ONGC	RELIANCE
ARIMA	AIC	-9535.673	-10200.465	-10001.886	-9286.231	-10430.648	-10045.663	-9428.112	-10481.538	-10034.546	-10712.761
	BIC	-10302.865	-10990.251	-10856.144	-9188.579	-10653.444	-10377.549	-9198.845	-10388.125	-10090.782	-10777.054
ABC-ARIMA	AIC	-9647.235	-10335.148	-10201.012	-9344.815	-10580.438	-10245.543	-9430.112	-10480.024	-10188.006	-10722.112
	BIC	-9812.745	-10690.251	-10156.144	-9176.124	-10660.234	-10437.449	-9288.338	-10511.005	-10114.052	-10807.503
DE-ABC-ARIMA	AIC	-9947.857	-10663.208	-10410.25	-9362.11	-10773.208	-10492.343	-9443.802	-10735.044	-10268.356	-10988.622
	BIC	-9936.243	-10761.594	-10398.635	-9350.496	-10761.594	-10480.729	-9432.188	-10723.43	-10256.742	-10977.008

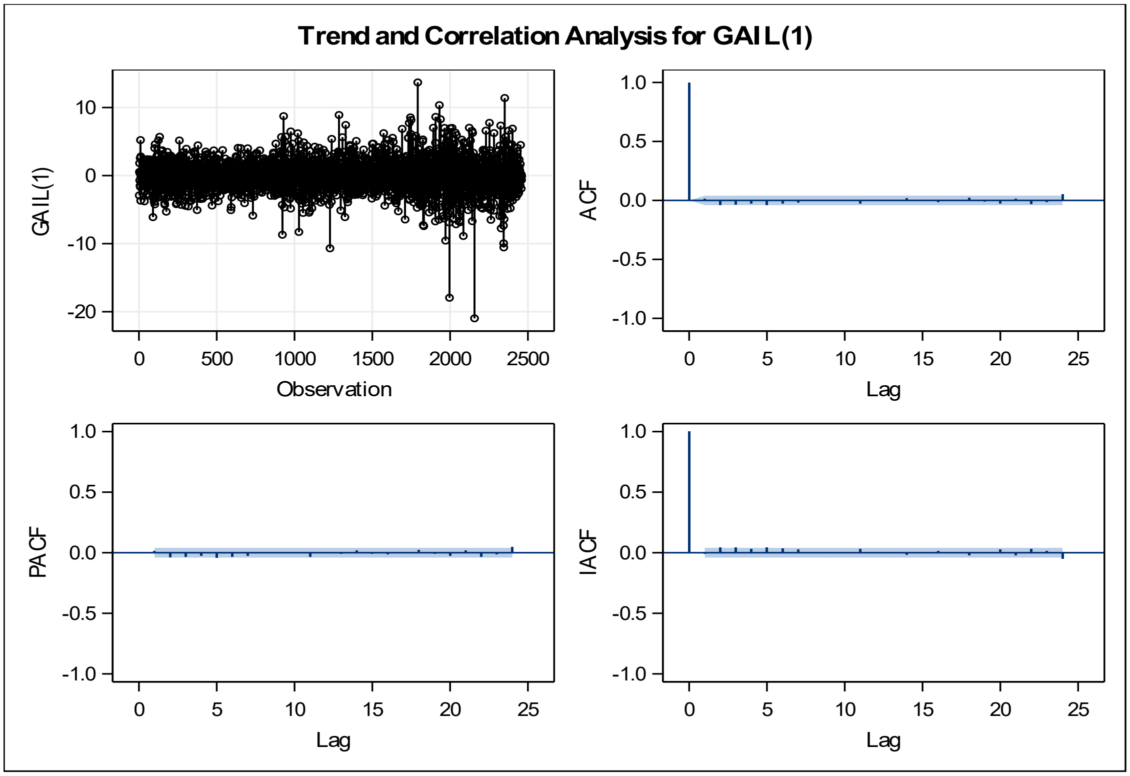

Figure 5.3: Correlation Analysis and Trend for GAIL

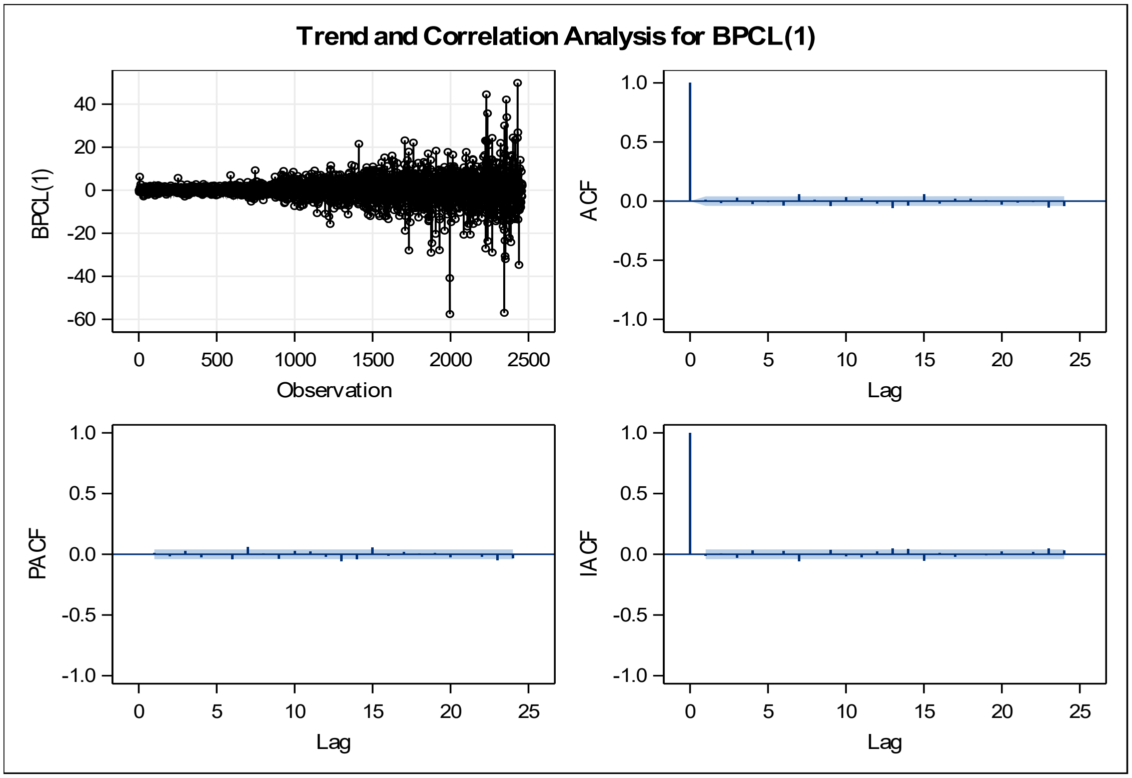

Figure 5.4: Correlation Analysis and Trend for BPCL

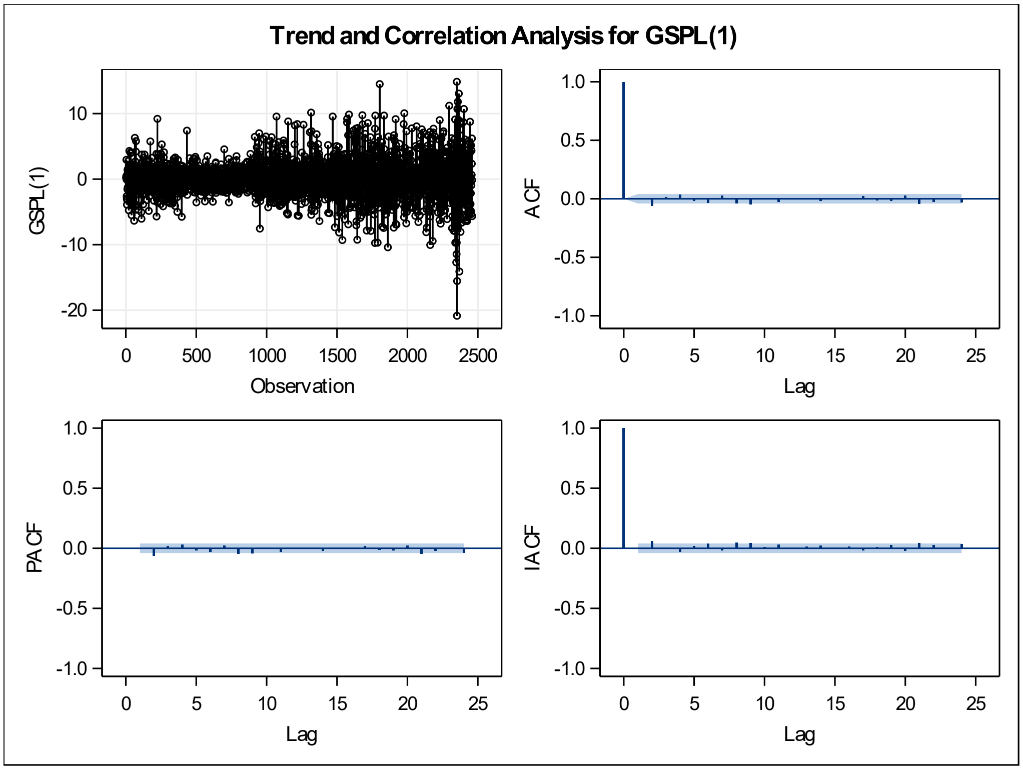

Figure 5.5: Correlation Analysis and Trend GSPL

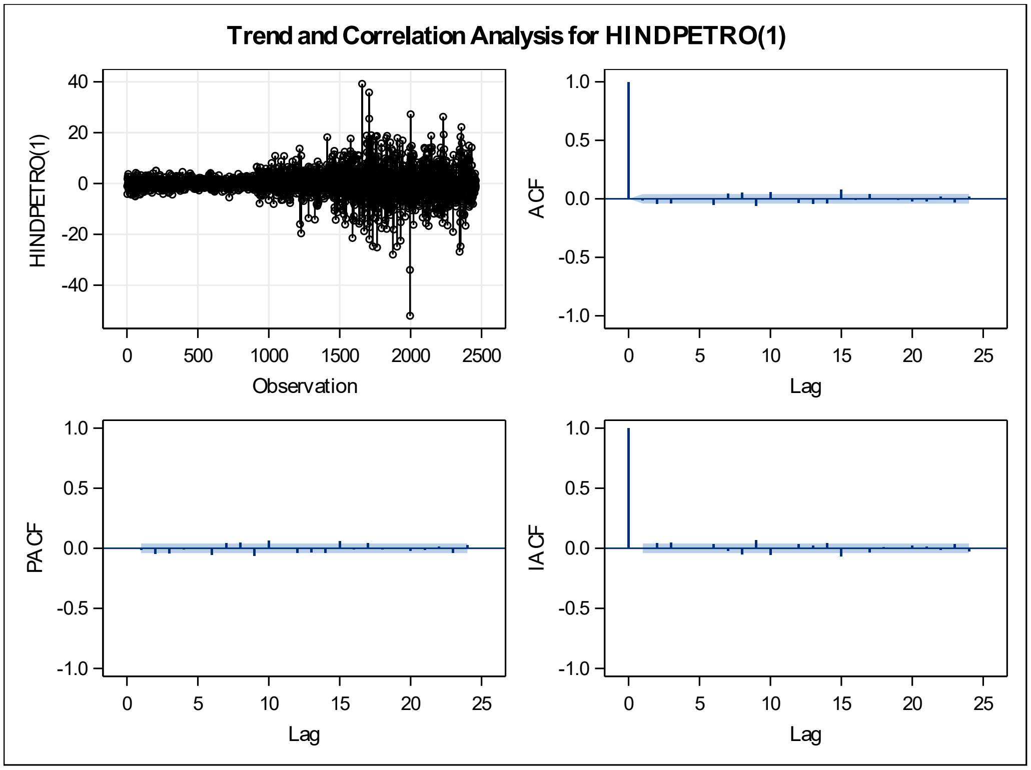

Figure 5.6: Correlation Analysis and Trend HINDPETRO

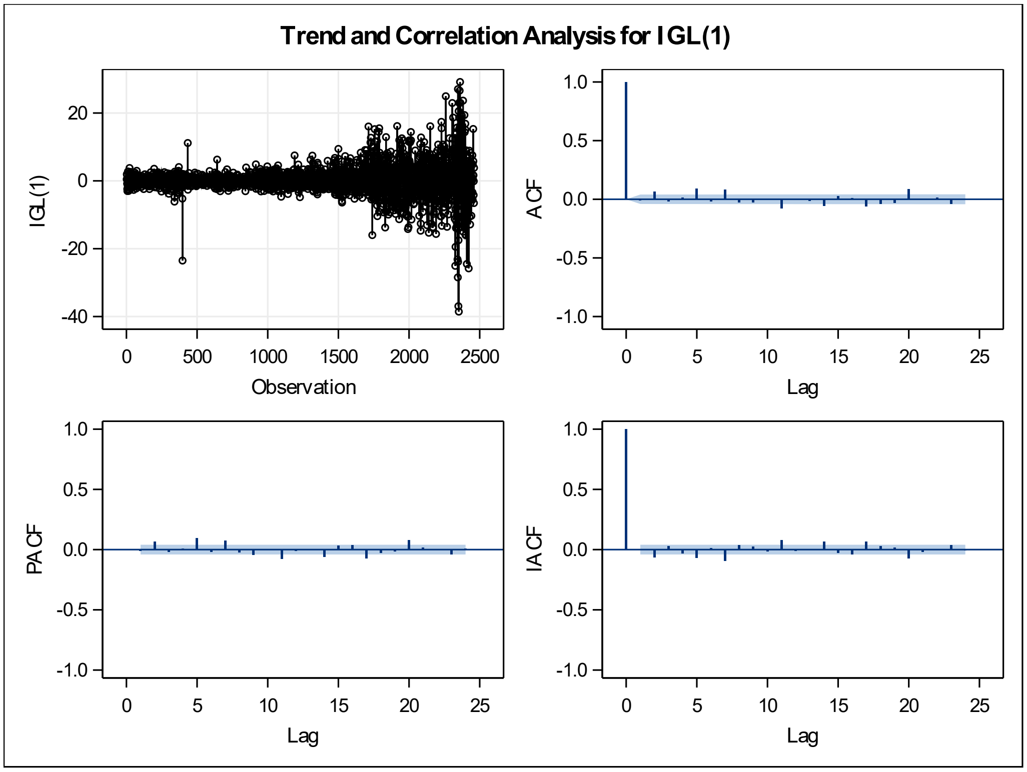

Figure 5.7: Correlation Analysis and Trend for IGL

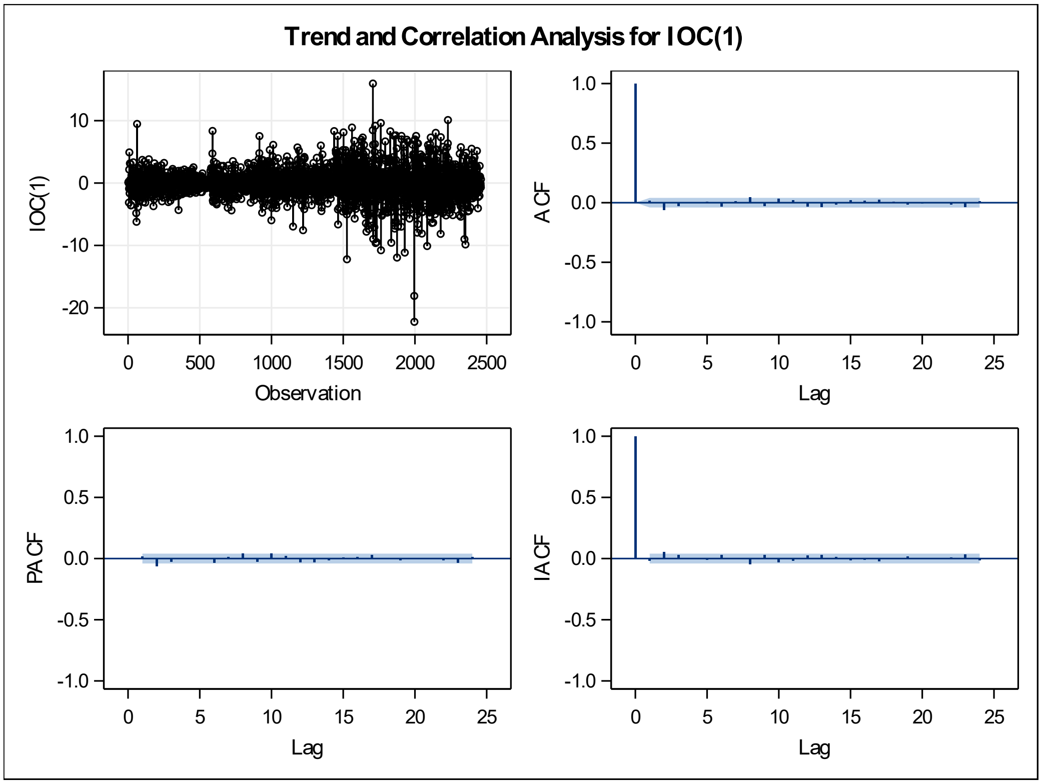

Figure 5.8: Correlation Analysis and Trend for IOC

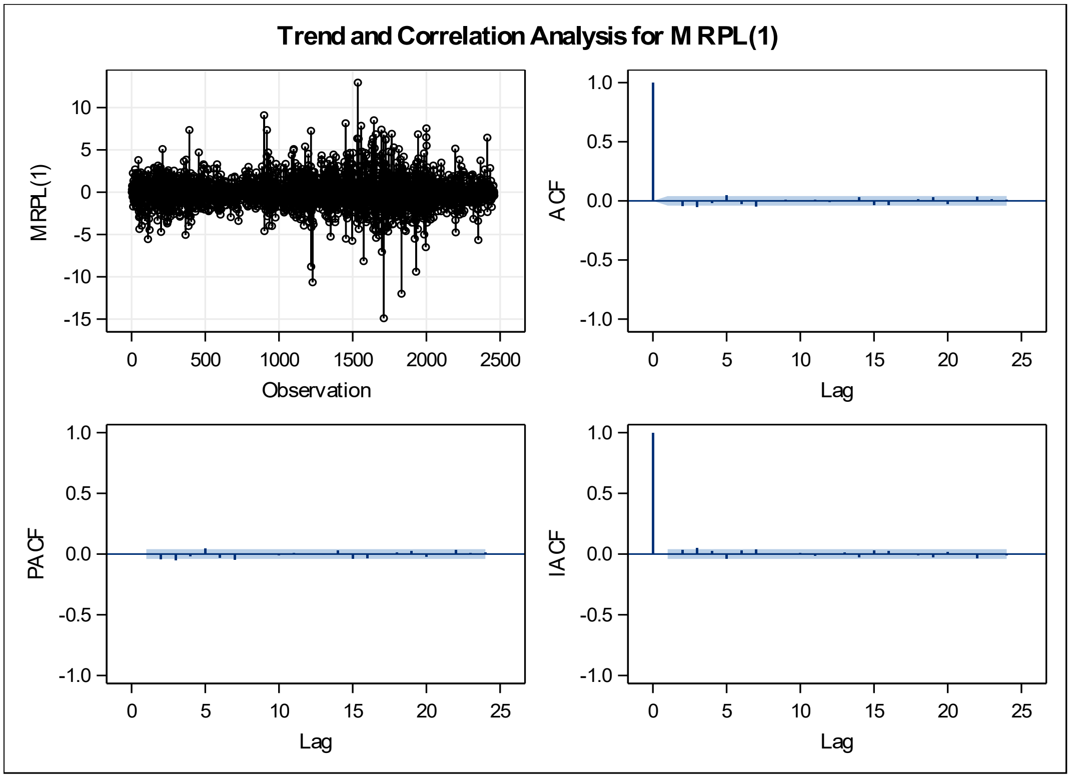

Figure 5.9: Correlation Analysis and Trend for MRPL

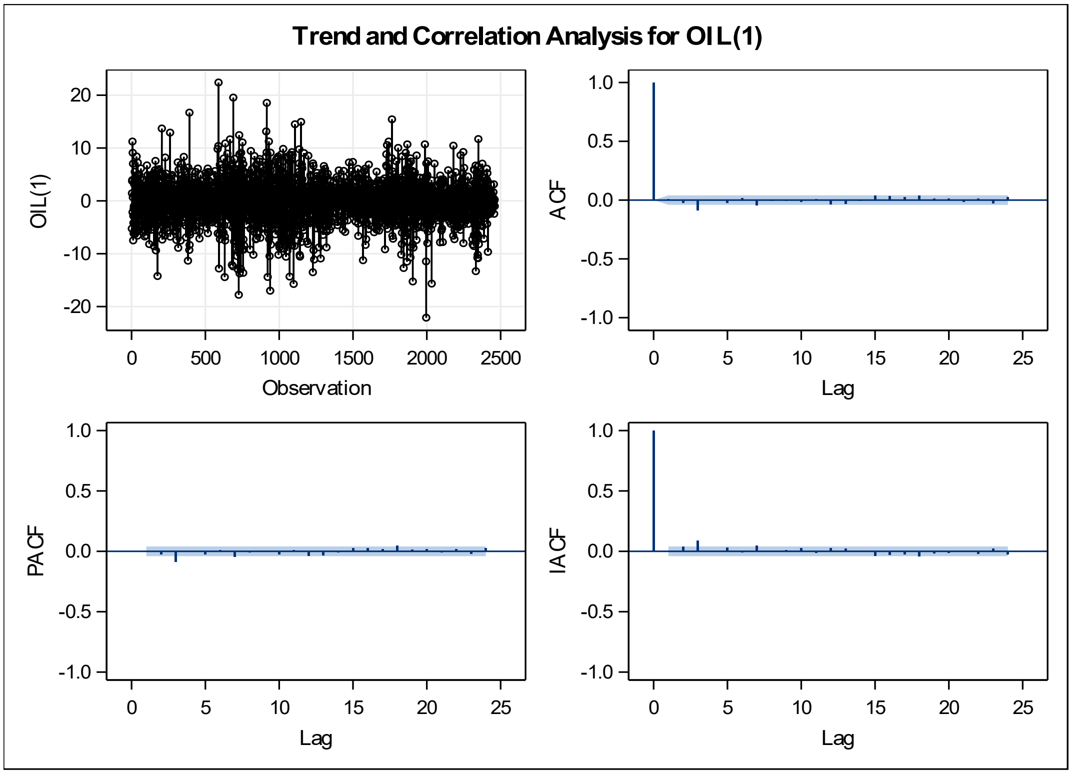

Figure 5.10: Correlation Analysis and Trend for OIL

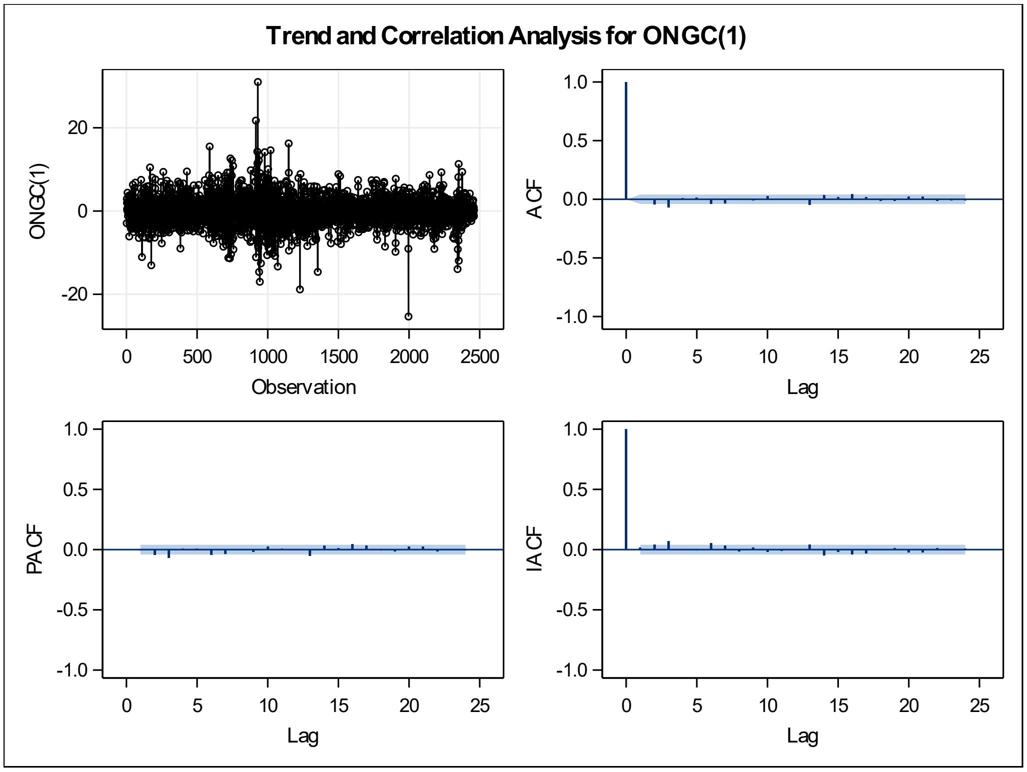

Figure 5.11: Correlation Analysis and Trend ONGC

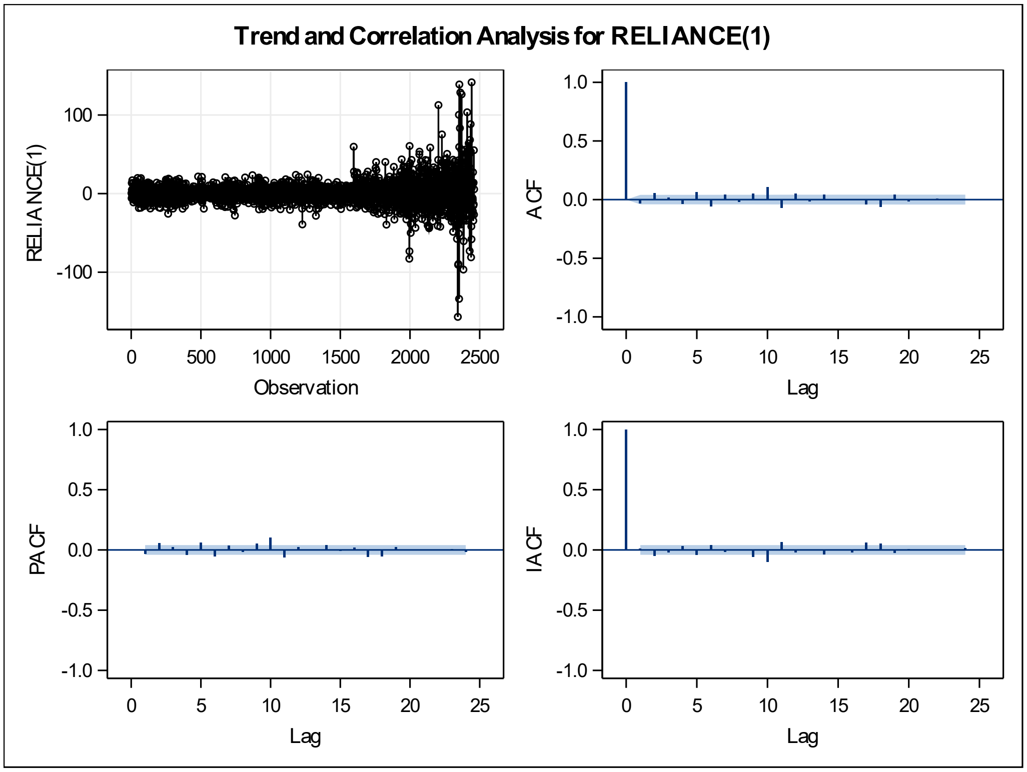

Figure 5.12: Correlation Analysis and Trend RELIANCE

Table 5.4: Autocorrelation Check for White Noise

Dataset	To Lag	Chi-Square	DF	Pr>ChiSq	Autocorrelations					
BPCL	**6**	7.36	6	0.2892	0.011	-0.014	0.026	-0.023	-0.005	-0.038
	12	24.82	12	0.0157	0.058	0.010	-0.041	0.030	0.023	-0.021
	18	46.60	18	0.0002	-0.059	-0.037	0.055	-0.018	0.018	0.018
	24	59.74	24	<.0001	0.005	-0.028	-0.012	-0.007	-0.053	-0.039
GAIL	**6**	14.61	6	0.0235	0.015	-0.039	-0.035	-0.025	-0.038	-0.029
	12	17.66	12	0.1263	-0.019	0.002	0.003	0.004	-0.029	0.007
	18	20.88	18	0.2854	-0.004	0.020	-0.006	-0.013	0.008	0.025
	24	32.89	24	0.1065	-0.009	-0.026	0.017	-0.031	-0.015	0.051
GSPL	**6**	17.28	6	0.0083	0.006	-0.062	0.017	0.035	-0.018	-0.036
	12	30.59	12	0.0023	0.027	-0.041	-0.048	0.002	-0.026	0.002
	18	33.85	18	0.0131	-0.000	-0.020	-0.007	-0.000	0.025	-0.016
	24	45.37	24	0.0053	-0.020	0.029	-0.043	-0.027	-0.002	-0.030
HINDPETRO	**6**	16.41	6	0.0117	-0.012	-0.045	-0.041	-0.006	0.001	-0.052
	12	49.30	12	<.0001	0.045	0.053	-0.062	0.058	-0.004	-0.035
	18	77.05	18	<.0001	-0.044	-0.038	0.079	-0.011	0.039	-0.001
	24	84.10	24	<.0001	-0.010	-0.023	-0.024	0.019	-0.030	0.020
IGL	**6**	33.73	6	<.0001	-0.009	0.067	-0.018	0.013	0.091	-0.018
	12	70.95	12	<.0001	0.084	-0.029	-0.028	0.004	-0.080	0.005
	18	93.50	18	<.0001	-0.013	-0.055	0.026	0.011	-0.060	-0.039
	24	120.00	24	<.0001	-0.030	0.090	0.003	0.013	-0.040	-0.005

IOC	**6**	14.71	6	0.0226	0.018	-0.061	-0.029	0.002	0.006	-0.033
	12	28.09	12	0.0054	0.011	0.045	-0.025	0.035	0.021	-0.031
	18	34.96	18	0.0096	-0.037	-0.015	0.018	0.015	0.024	0.006
	24	39.94	24	0.0217	-0.017	-0.005	-0.002	-0.013	-0.037	0.012
MRPL	**6**	19.79	6	0.0030	0.007	-0.043	-0.051	-0.019	0.050	-0.027
	12	26.23	12	0.0100	-0.048	-0.004	0.011	-0.000	0.010	-0.010
	18	35.80	18	0.0075	-0.003	0.032	-0.037	-0.034	0.004	0.016
	24	44.76	24	0.0062	0.034	-0.027	-0.007	0.037	0.014	0.011
OIL	**6**	21.97	6	0.0012	0.007	-0.024	-0.087	0.001	-0.021	0.017
	12	30.59	12	0.0023	-0.044	-0.004	-0.005	-0.016	0.009	-0.035
	18	43.83	18	0.0006	-0.030	-0.007	0.036	0.032	0.023	0.039
	24	48.93	24	0.0019	0.011	0.012	-0.015	0.014	-0.027	0.025
ONGC	**6**	21.04	6	0.0018	-0.007	-0.044	-0.068	0.011	0.016	-0.039
	12	26.49	12	0.0091	-0.037	0.004	-0.009	0.027	0.005	-0.001
	18	43.41	18	0.0007	-0.051	0.035	0.019	0.046	0.020	-0.013
	24	48.17	24	0.0024	-0.016	0.024	0.025	-0.016	-0.011	-0.010
RELIANCE	**6**	33.79	6	<.0001	-0.032	0.057	0.018	-0.037	0.065	-0.059
	12	91.95	12	<.0001	0.043	-0.021	0.052	0.107	-0.069	0.049
	18	110.89	18	<.0001	-0.015	0.044	0.001	0.006	-0.040	-0.062
	24	115.92	24	<.0001	0.041	-0.015	-0.002	0.009	0.001	0.004

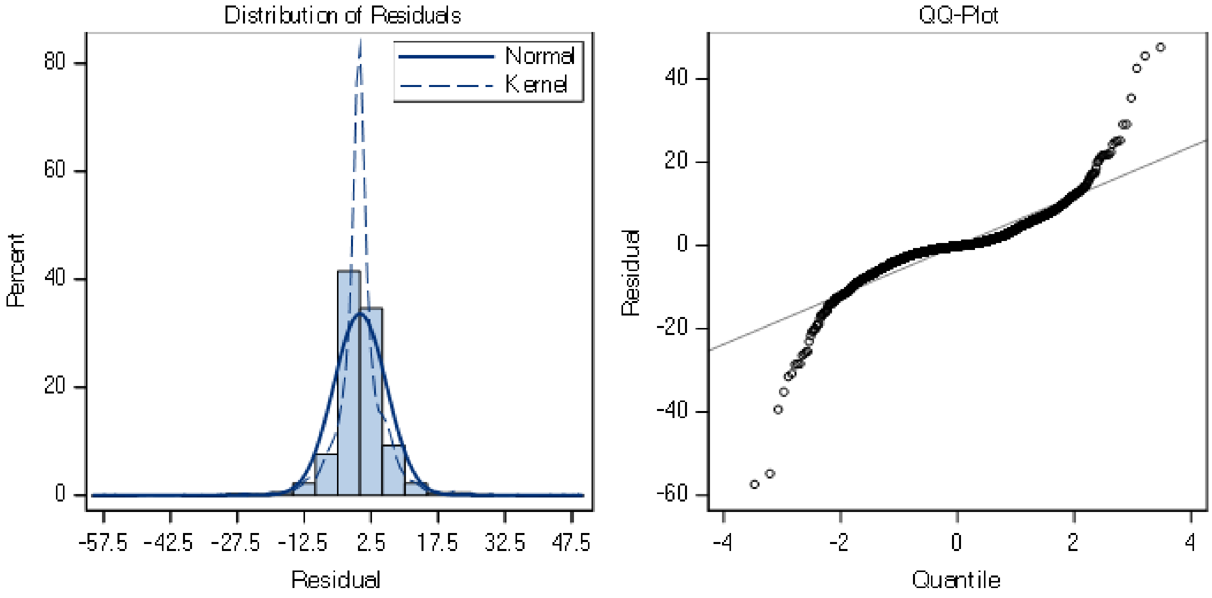

Figure 5.13: Residual Normality Diagnostics analysis of BPCL historical stock's data

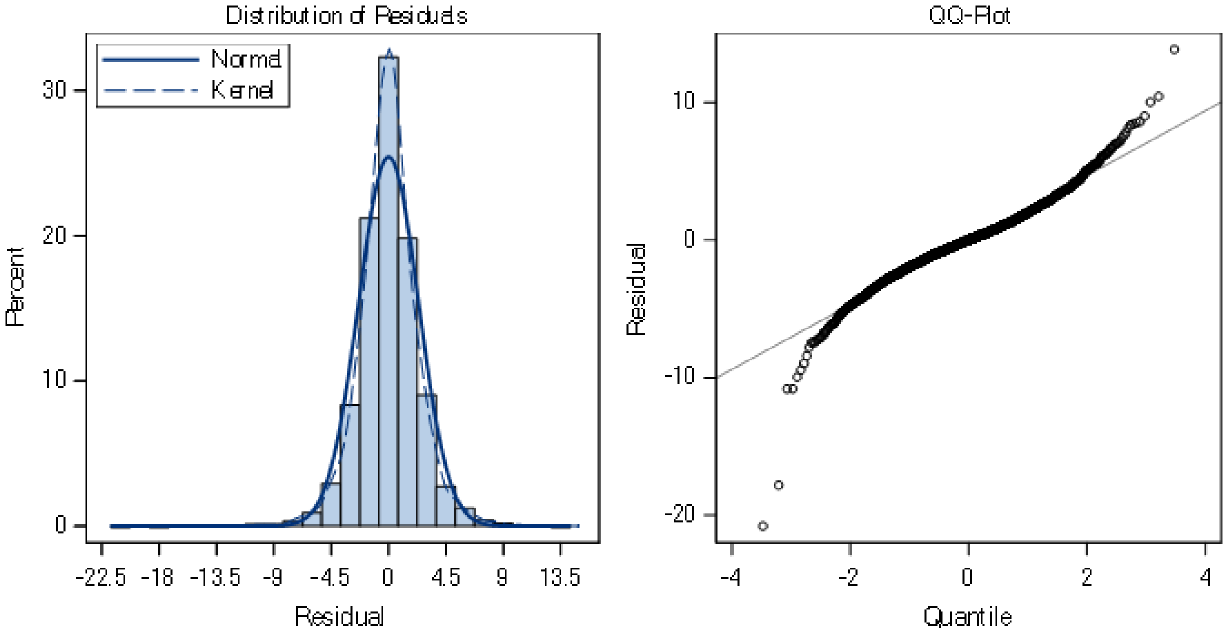

Figure 5.14: Residual Normality Diagnostics analysis of GAIL historical stock's data

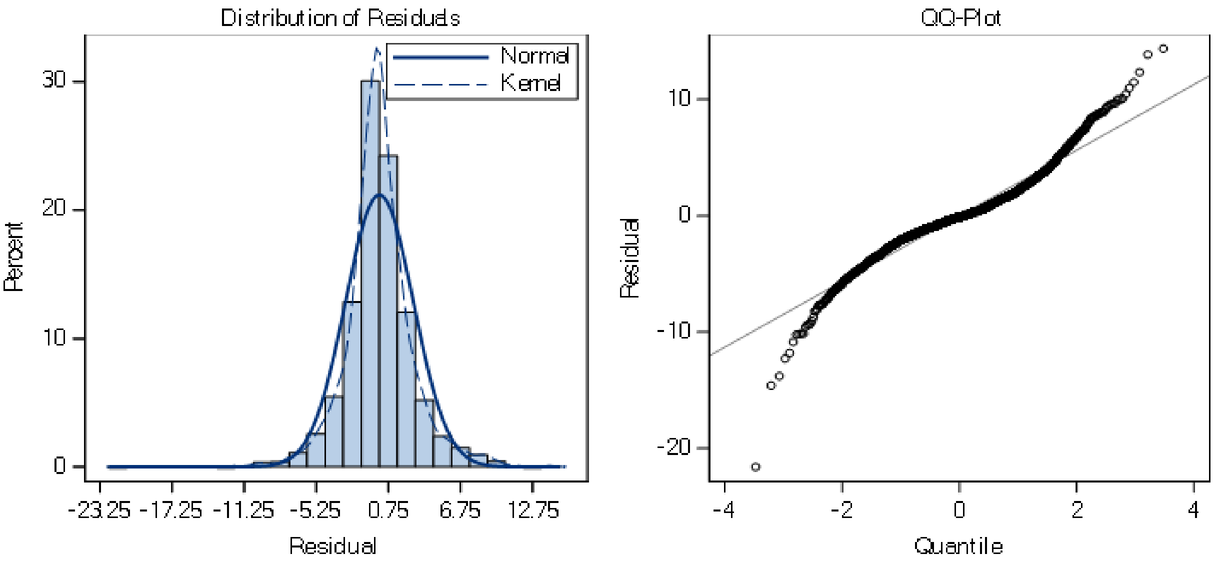

Figure 5.15: Residual Normality Diagnostics analysis of GSPL historical stock's data

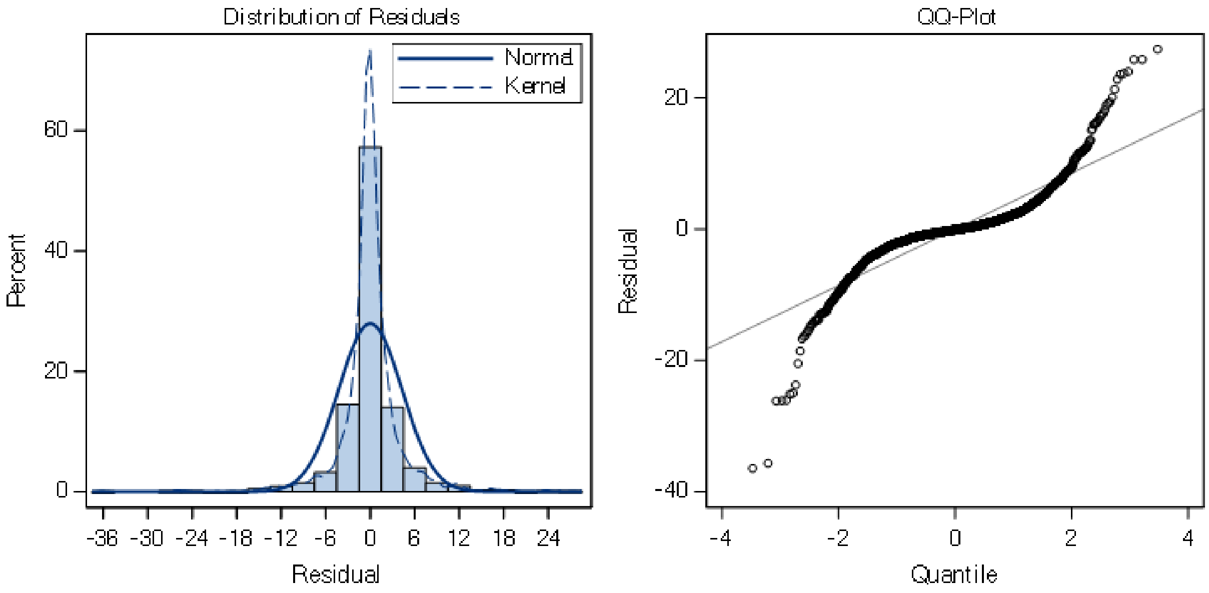

Figure 5.16: Residual Normality Diagnostics analysis of IGL historical stock's data

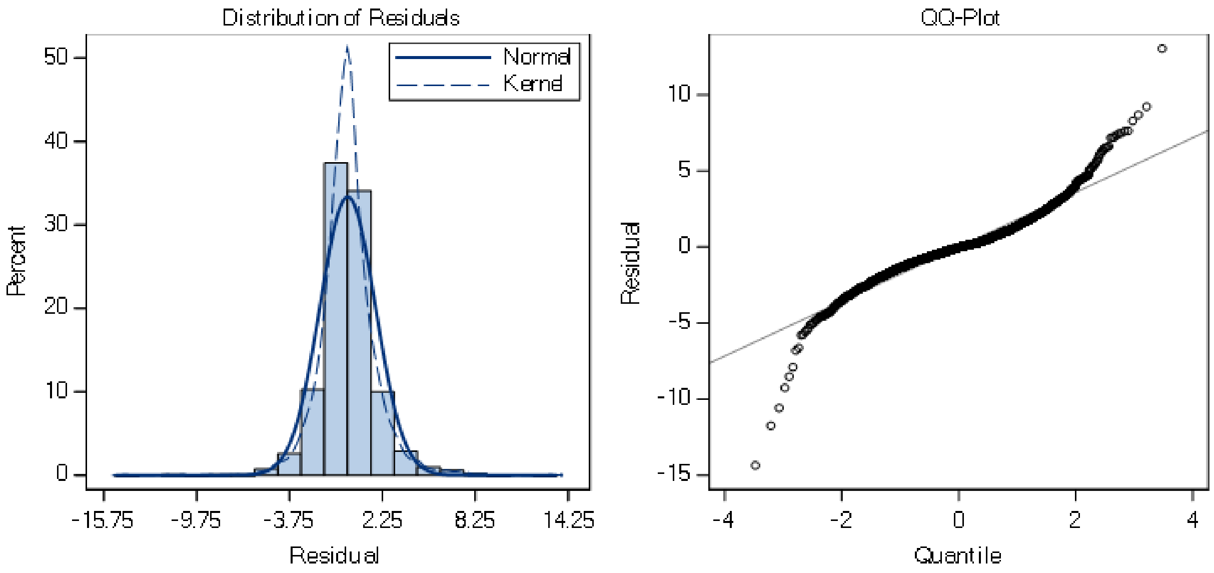

Figure 5.17: Residual Normality Diagnostics analysis of MRPL historical stock's data

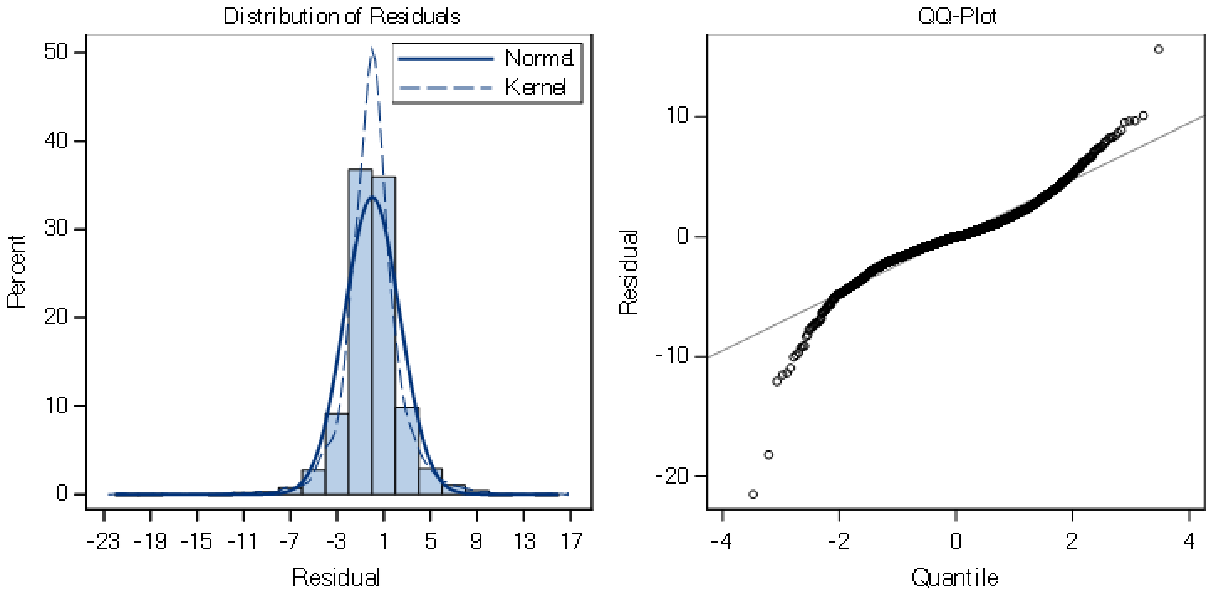

Figure 5.18: Residual Normality Diagnostics analysis of IOC historical stock's data

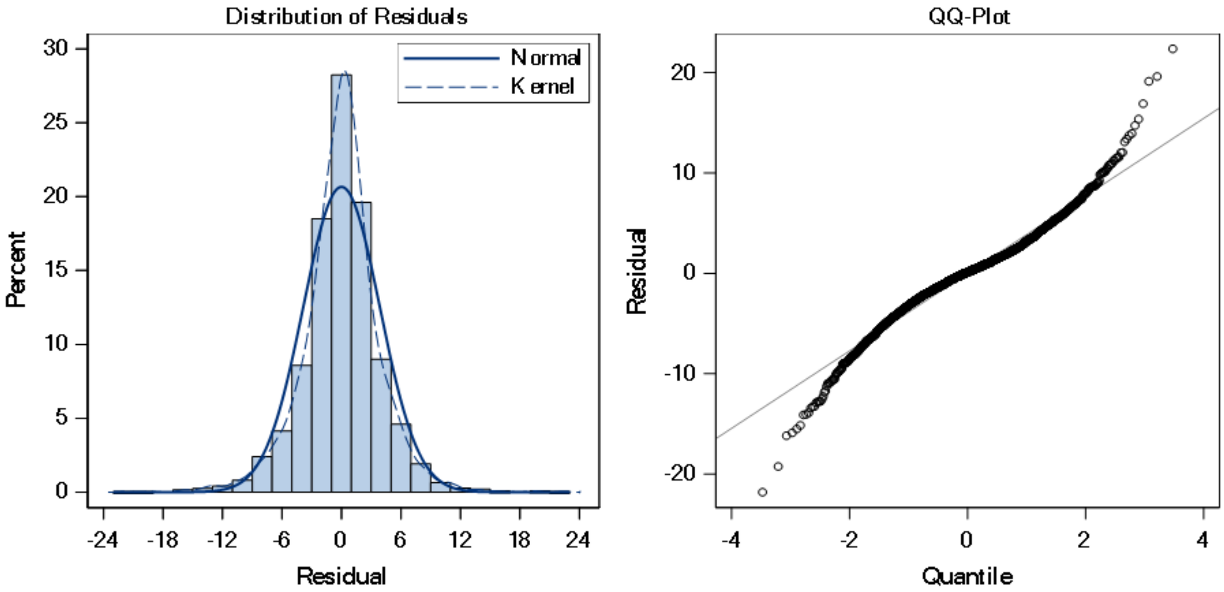

Figure 5.19: Residual Normality Diagnostics analysis of OIL historical stock's data

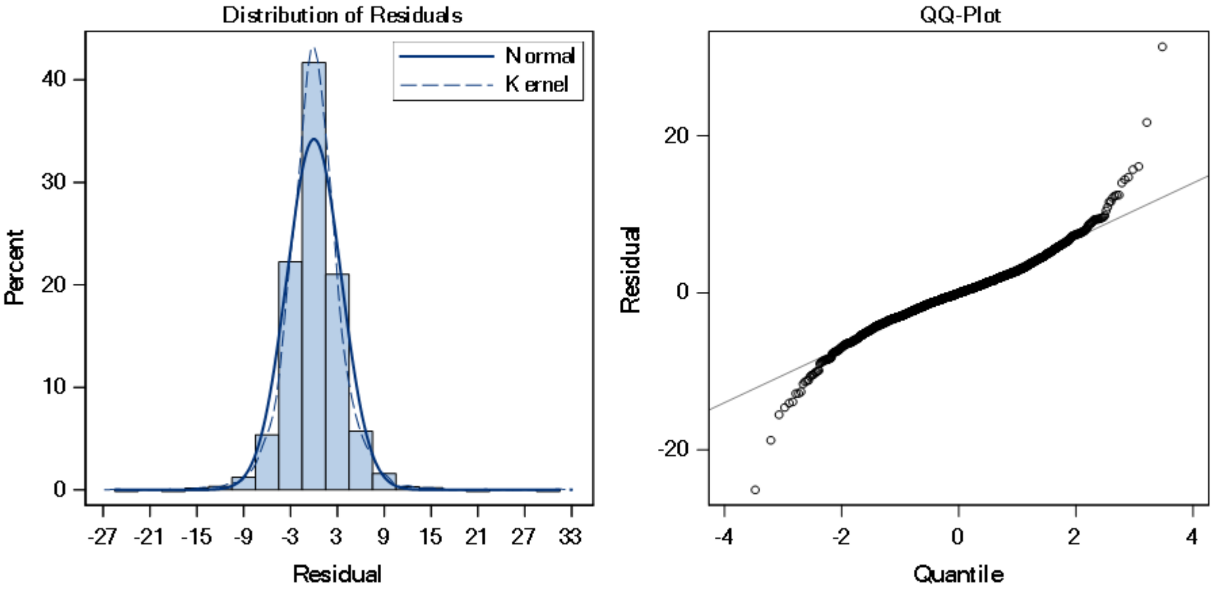

Figure 5.20: Residual Normality Diagnostics analysis of ONGC historical stock's data

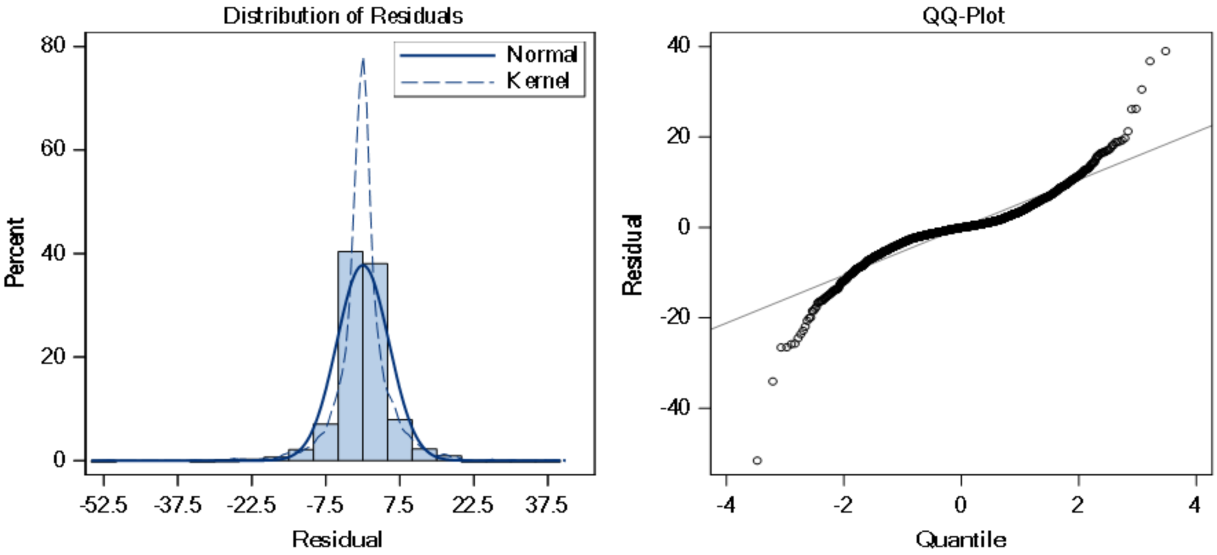

Figure 5.21: Residual Normality Diagnostics analysis of HINDPETRO historical stock's data

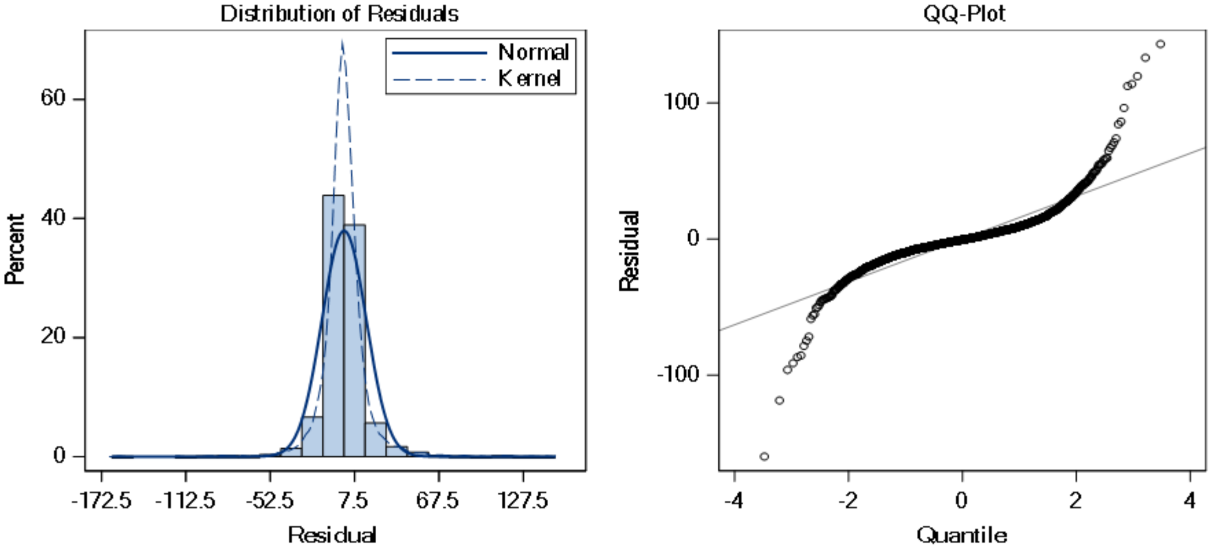

Figure 5.22: Residual Normality Diagnostics analysis of RELIANCE historical stock's data

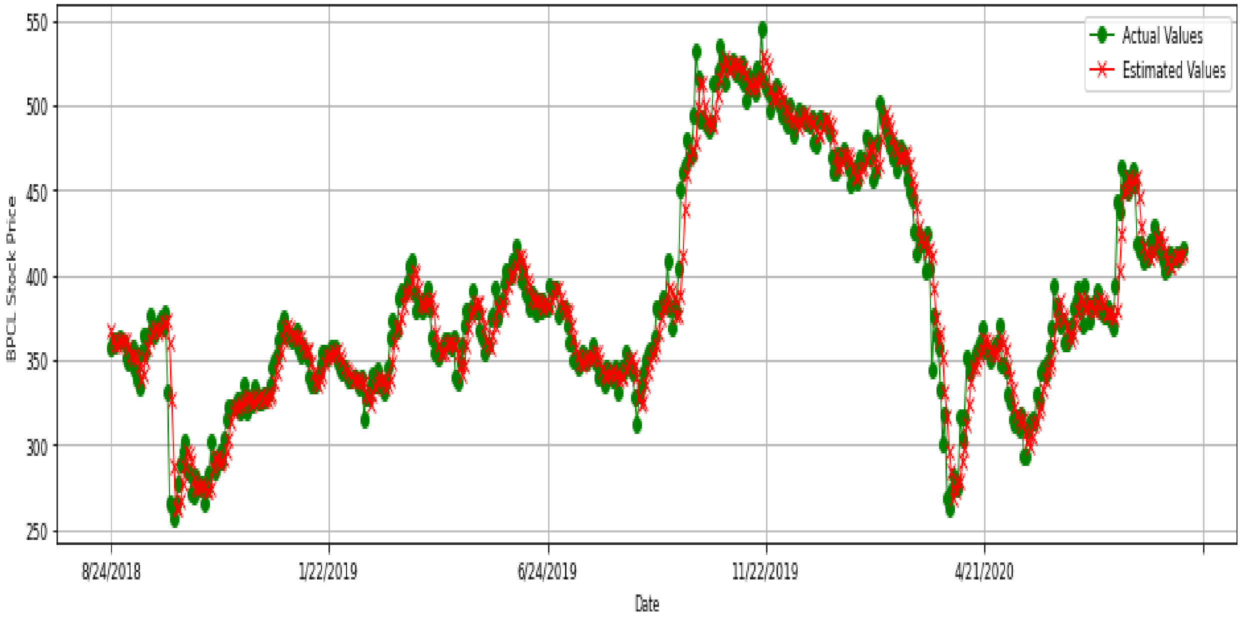

Figure 5.23: Time Series Forecasting for BPCL

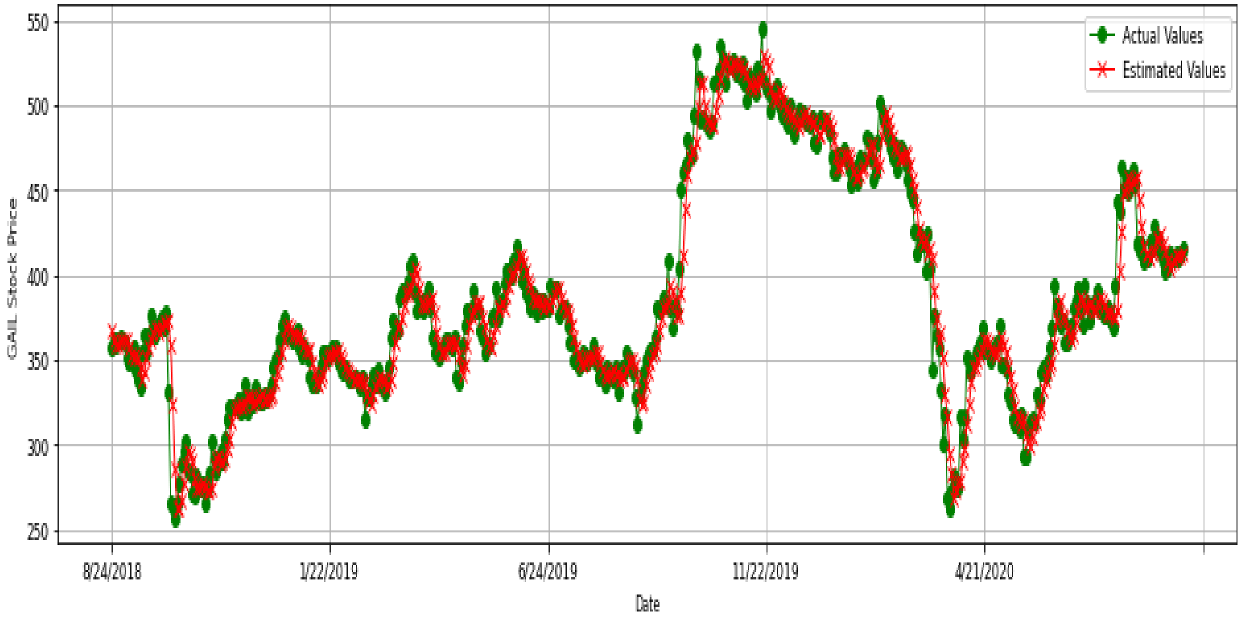

Figure 5.24: Time Series Forecasting for GAIL

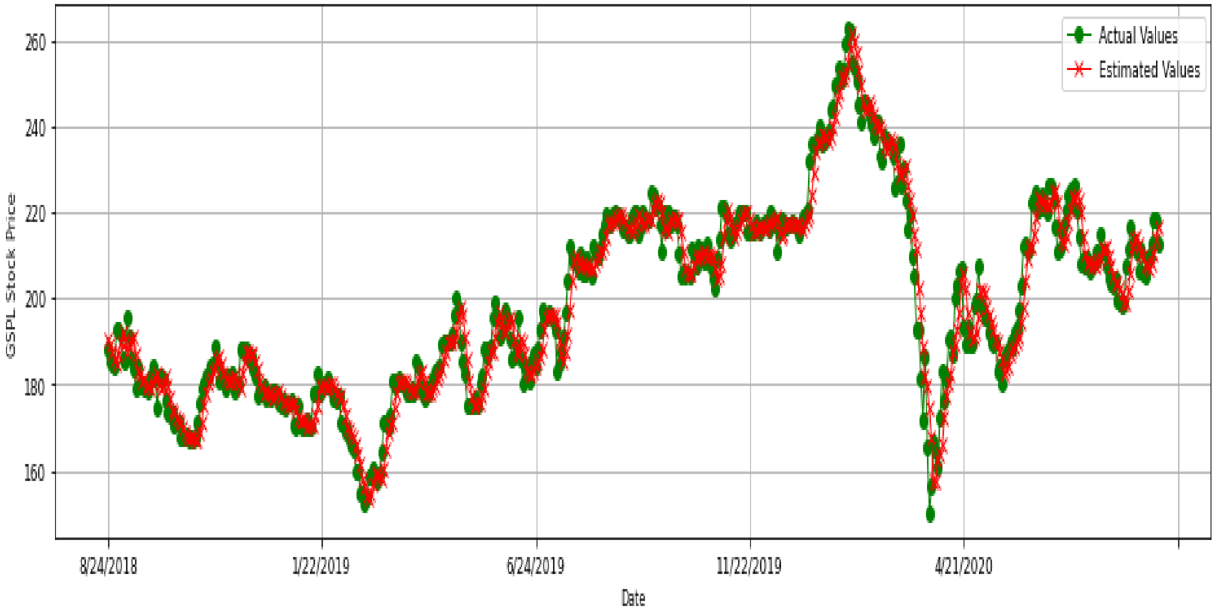

Figure 5.25: Time Series Forecasting for GSPL

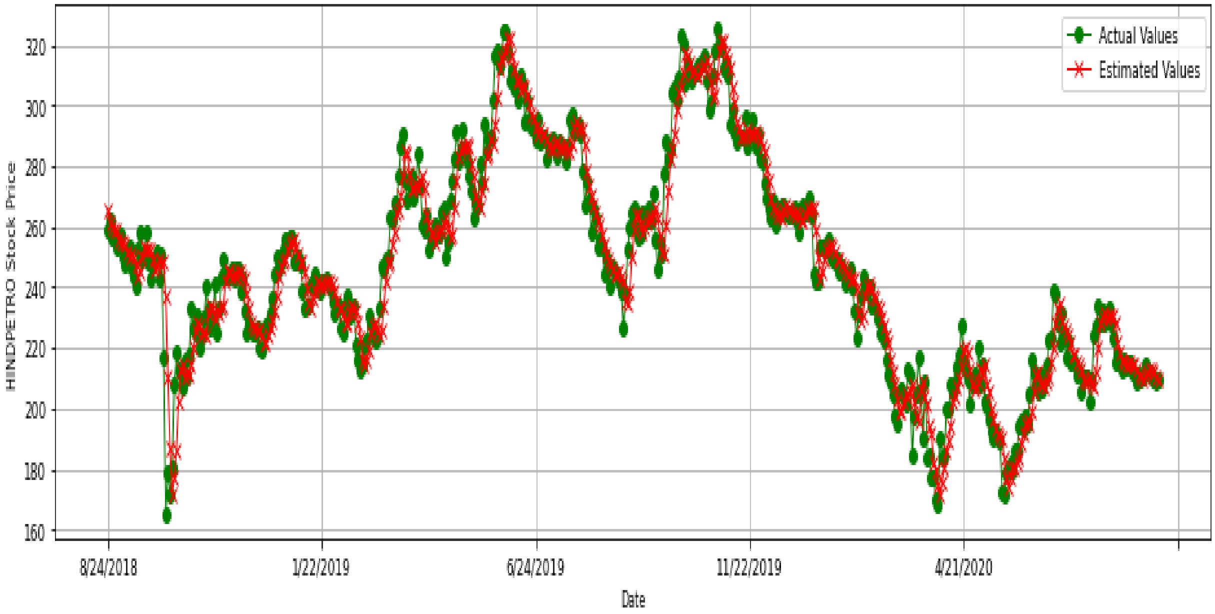

Figure 5.26: Time Series Forecasting for HINDPETRO

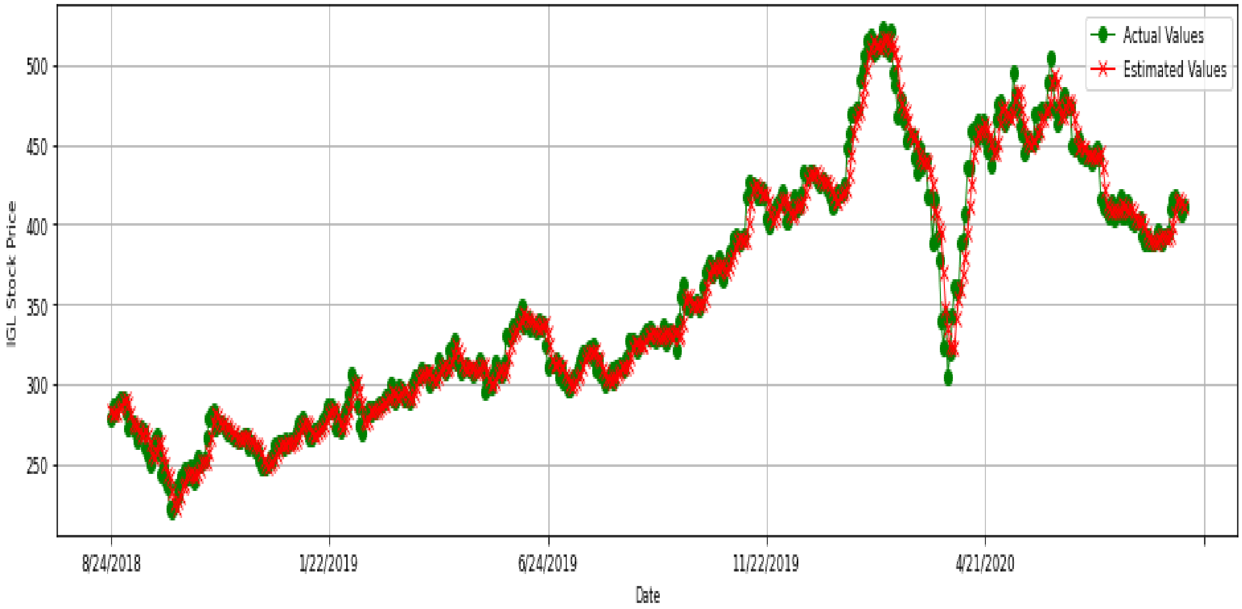

Figure 5.27: Time Series Forecasting for IGL

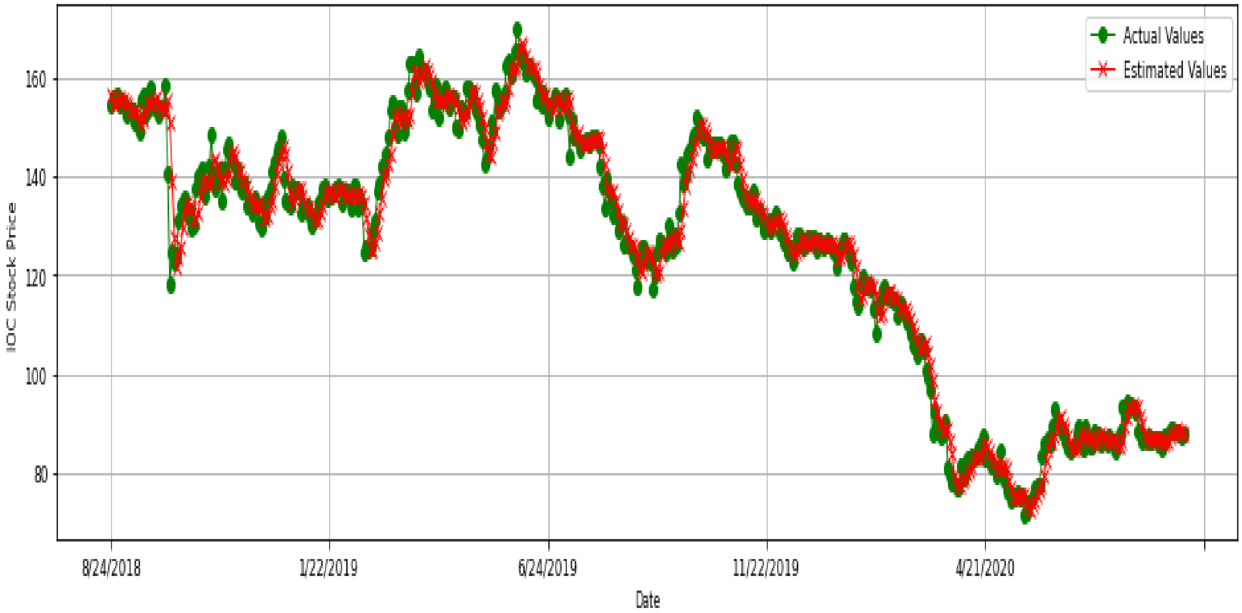

Figure 5.28: Time Series Forecasting for IOC

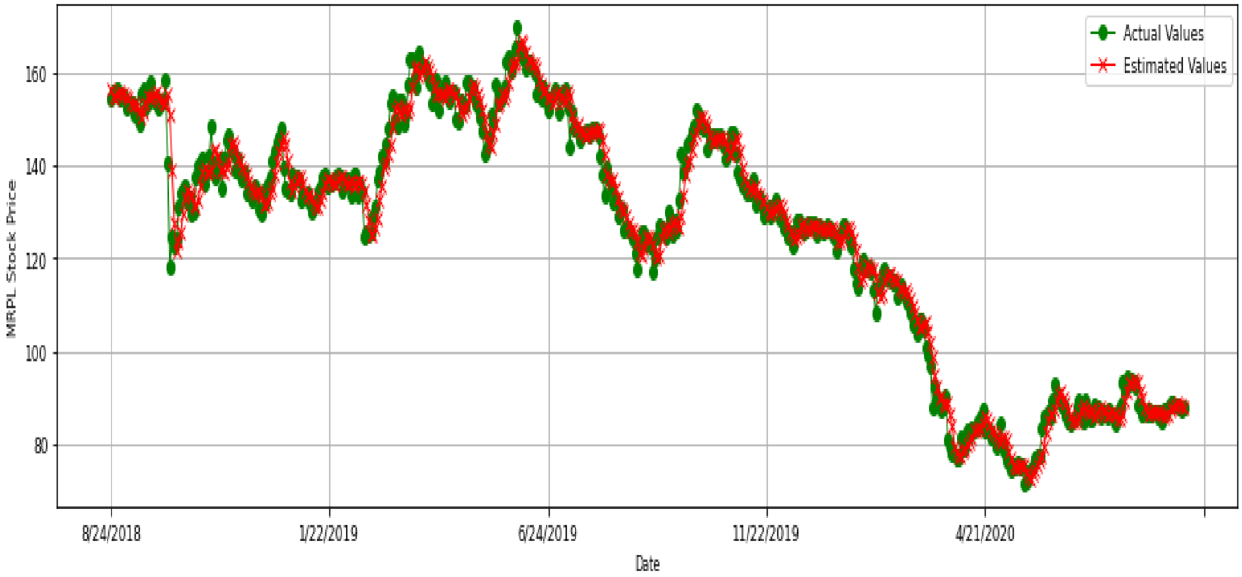

Figure 5.29: Time Series Forecasting for MRPL

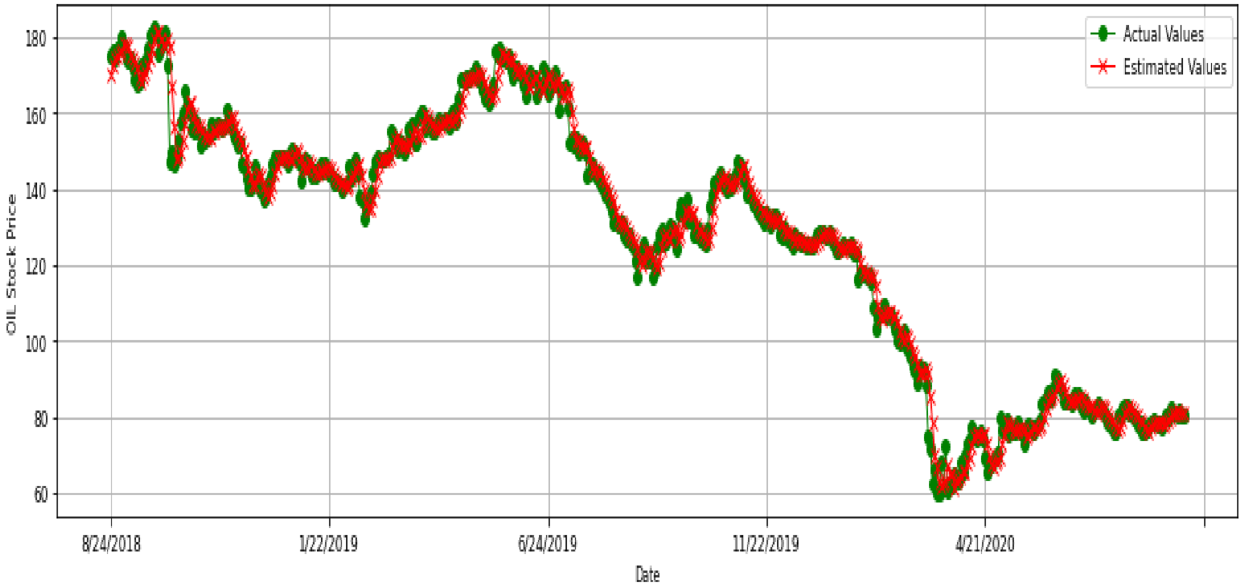

Figure 5.30: Time Series Forecasting for OIL

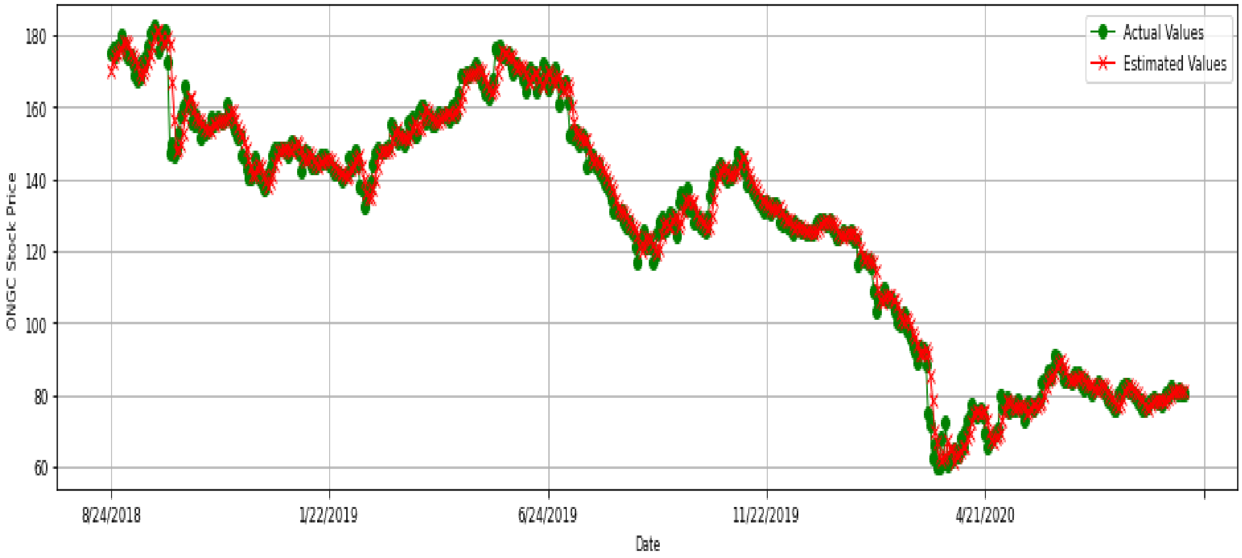

Figure 5.31: Time Series Forecasting for ONGC

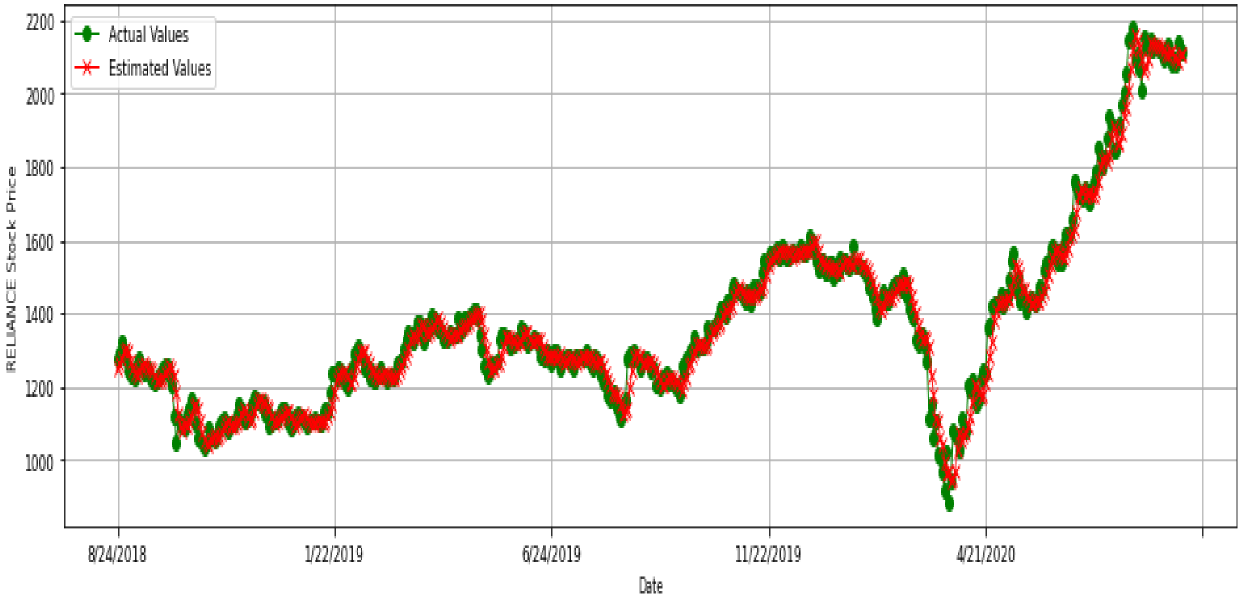

Figure 5.32: Time Series Forecasting for RELIANCE

5.5.7 Forecasting Analysis

As a result, the number of ARIMA parameters is equal to the length of ABC's food supply (3). For the historical dataset experiments, 100 iterations are taken into account.

Table 5.5 shows the estimated values of the proposed model LSTM-ARIMA with improved ABC. In comparison to individual linear model ARIMA, the proposed model LSTM-ARIMA (with improved ABC) improves the RMSE and MAPE for stock BPCL (45.98 %, 53.66 %), GAIL (37.80 %, 27.13 %), GSPL (41.38 %, 45.33 %), IGL (42.93 %, 17.27 %), HINDPETRO (35.72 %, 51.13 %), IOC (55.21 % (35.29 %, 14.20 %). Noticeably, as volatile behavior as shown in Table-5.5, a maximum 57.03 % improvement in RMSE and 56.06 % in MAPE was attained for Oil India Ltd. (OIL).

$$\mathrm{RMSE} = \sqrt{\textstyle\sum_{i=1}^{n} (\mathrm{Xi} - \mathrm{X'i})^2 / \mathrm{N}} \qquad (5.7)$$

$$\mathrm{MAPE} = 1/\mathrm{n} \textstyle\sum_{i=1}^{n} |\mathrm{Xi} - \mathrm{X'i}/\mathrm{Xi}| * 100 \qquad (5.8)$$

In comparison to individual nonlinear model LSTM, hybridization of LSTM-ARIMA (with improved ABC) brought the RMSE and MAPE values for the selected stocks BPCL (34.80 %, 51.04 %), GAIL (23.90 %, 29.01 %), GSPL (31.82 %, 39.24 %), IGL (29.76 %, 13.79 %), HINDPETRO (31.56 %, 26.97 %), IOC (39.59 (33.17 %, 12.74 %). The maximum RMSE improvement was 50.98 %, and the MAPE improvement was 47.55 % for the same OIL stock dataset. Based on observations 1 and 2, it can be concluded that the proposed hybrid model LSTM-ARIMA is the best match for the Oil India Ltd. (OIL) dataset, since the model achieved significant accuracy improvements. Improvements to the ABC employing DE play a significant role in optimizing hyper parameters for the proposed model LSTM-ARIMA, according to observation #3. In comparison to hybrid LSTM-ARIMA (without ABC optimization), optimization of LSTM-ARIMA (with improved ABC) obtained RMSE and MAPE values for the selected stocks BPCL (5.56, 15.95), GAIL (11.48, 18.23), GSPL (7.68, 9.71), IGL (3.77, 5.52), HINDPETRO (10.14, 12.66), IOC (13.28, 14.79), MRPL (1.58, 16.31), OIL (5 (7.76, 12.20). It's worth noting that the RMSE for Oil and Natural Gas Corporation (ONGC) stock improved by 14.95 %, while the MAPE for Oil India Ltd. (OIL) stock improved by 18.47 %.

Table 5.5: Forecasting Results based on performance metrics

Models	Performance Metrics	BPCL	GAIL	GSPL	IGL	HINDPE	IOC	MRPL	OIL	ONGC	RELIANCE
Proposed LSTM-ARIMA with improved ABC	RMSE	6.8441	7.9865	2.8664	3.7475	6.7995	2.5337	2.1667	2.1044	1.7979	10.5765
	MAPE (%)	0.8779	1.272	1.0739	1.3685	1.5239	1.0485	1.0235	0.7668	1.2406	1.2893
LSTM-ARIMA	RMSE	7.2478	9.0232	3.1049	3.8945	7.5673	2.922	2.2015	2.2331	2.114	11.4667
	MAPE (%)	1.0446	1.5557	1.1895	1.4485	1.7448	1.2306	1.223	0.9406	1.3934	1.4686
ARIMA-LSTM	RMSE	9.8123	10.5681	3.9834	5.1824	9.8647	3.8936	3.9528	2.9364	3.0853	13.3578
	MAPE (%)	1.2373	1.6495	1.5869	1.4903	1.9408	1.4211	1.395	1.2893	1.2205	1.2953
LSTM	RMSE	10.4983	10.4956	4.2046	5.3357	9.9358	4.1945	4.192	4.2927	3.2134	15.8264
	MAPE (%)	1.7933	1.7919	1.7675	1.5875	2.0868	1.6507	1.6497	1.4620	1.3413	1.4776
ARIMA	RMSE	12.6714	11.0056	4.8904	6.5674	10.5784	5.6575	4.7723	4.8975	3.8965	16.3453
	MAPE (%)	1.8945	1.7456	1.9645	1.6543	3.1189	1.8903	1.6665	1.7452	1.5726	1.5028

5.6 Summary

Not only for market traders and investors, but also for the research community, stock market analysis is the most difficult assignment. Individual models like ARIMA and LSTM, according to a recent study, have their own constraints due to feature limitations. At the same time, hybrid and ensemble models produce better results. As a result, this research suggested the LSTM-ARIMA hybrid model for empirical stock market time series forecasting. According to research, evolutionary algorithms are crucial for parameter selection and optimization. The ABC was shown to be the best optimization approach for selecting hyperparameters. Maximum improvements in analyzed stock market values were 14.95 % in RMSE for ONGC and 18.47 % in MAPE for OIL using the proposed approach. Despite this, the two-phase hybridization (LSTM-ARIMA and DE-ABC) developed in this study outperformed benchmark models. However, the proposed model has the potential to be upgraded to a three-phase hybrid model, in which the input (historical datasets) can be combined with market sentiments and technical indicators to create a hybrid nature. This method can be viewed as both a limitation and a potential possibility for this chapter.

CHAPTER 6

CONCLUSION AND FUTURE WORK

The three-stage hybridization described in this thesis is used to handle time series data and increase stock market forecasting accuracy. In the first step of hybridization, stock market inputs such as historical data and market sentiments for the target stock are combined with stock market technical indicators. The sentiments polarity index, which is based on the Hadoop big data platform, is used to investigate market sentiments. In the second phase, the dataset's linear and non-linear features are observed using a combination of ARIMA and LSTM. In the third phase, the hyperparameter selection of the proposed DE-ABC-LSTM-ARIMA model for stock market forecasting is investigated using an upgraded DE-ABC algorithm.

This thesis examines the stock market's major aspects, such as trends, volatility, momentum, and volume. Technical indicators, for the most part, cover all areas, but choosing indicators is a tedious operation. The polarity index is used to calculate sentiments. The Hadoop Ecosystem is used to analyses data in the HDFS.

On the one hand, the one-step ahead prediction approach works effectively in the stock market to obtain higher accuracy. Multi-step TSA, on the other hand, has yet to establish itself as a widespread tool and strategy among investors due to its complexity. This thesis suggested a modified ABC-ARIMA hybrid model based on DE optimization techniques. Surprisingly, the range of RMSE improvements for MRPL and GSPL was 2.40 % to 24.33 %, respectively. The obtained results show that in multi-step time series forecasting, the suggested modified ABC-ARIMA hybrid model outperforms its competitors.

This thesis proposed the LSTM-ARIMA hybrid model for empirical stock market time series forecasting. According to research, evolutionary algorithms are crucial for parameter selection and optimization. The ABC algorithm was shown to be the best optimization approach for selecting hyperparameters. The exploration and exploitation concerns in time series prediction are addressed by this new hybridization of DE-ABC. From September 1, 2010, to August 31, 2020, an experiment is conducted on a refinery's dataset. LSTM-ARIMA improves the individual ARIMA, LSTM, and hybrid model. The suggested LSTM-ARIMA model with improved ABC algorithm obtains much lower RMSE and MAPE values for the selected stocks. Maximum improvements in analyzed stock market values were 14.95 % in RMSE for ONGC and 18.47 % in MAPE for OIL using the proposed approach.

6.1 Future Work

Direct and indirect factors of the market growth up or down of stock indexes and potentially affect the future value of stocks. Complex deep learning algorithms may be used to improve the polarity score of sentiment. This thesis outlines the conceptual features of the LSTM-ARIMA and DE-ABC Algorithms so that practitioners can adapt the suggested model to additional time series domains with large volumes of data. To improve stock forecasting accuracy, an event-based sentiment polarity technique can be used instead of textual sentiment analysis in the future scope of the proposed work.

REFERENCES

[1] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, "*Time series analysis: forecasting and control,"* John Wiley & Sons, 2015.

[2] T. C. Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, *24*(1), 164-181, 2011.

[3] O. Bustos, and A. Pomares-Quimbaya, "Stock market movement forecast: A systematic review," *Expert Systems with Applications*, *156*, 113464, 2020.

[4] O. B. Sezer, M.U. Gudelek, and A. M. Ozbayoglu, "Financial time series forecasting with deep learning: A systematic literature review: 2005–2019," *Applied Soft Computing*, *90*, 106181, 2020.

[5] A. Thakkar, and K. Chaudhari, "Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions," *Information Fusion*, vol. *65*, pp. 95-107, 2021.

[6] R. P. Schumaker, and H. Chen, "Textual analysis of stock market prediction using breaking financial news: The AZFin text system," *ACM Transactions on Information Systems (TOIS)*, vol. *27*(2), pp. 1-19, 2019.

[7] Y. Ruan, A. Durresi, and L. Alfantoukh, "Using Twitter trust network for stock market analysis," *Knowledge-Based Systems*, vol. *145*, pp. 207-218, 2018.

[8] X. Zhang, Y. Zhang, S. Wang, Y. Yao, and S. Y. Philip, "Improving stock market prediction via heterogeneous information fusion," *Knowledge-Based Systems*, vol. *143*, pp. 236-247, 2018.

[9] Z. Wang, Y. Huang, B. Cai, R. Ma, and Z. Wang, "Stock turnover prediction using search engine data," *Journal of Circuits, Systems and Computers*, vol. *30*(07), 2150122, 2020.

[10] P.C. Chang, Y. W. Wang, and W. N. Yang, "An investigation of the hybrid forecasting models for stock price variation in Taiwan," *Journal of the Chinese Institute of Industrial Engineers*, vol. *21*(4), pp. 358-368, 2009.

[11] A. S. Chen, M.T. Leung, and H. Daouk, “Application of neural networks to an emerging financial market: forecasting and trading the Taiwan Stock Index,” *Computers & Operations Research*, vol. *30*(6), pp. 901-923, 2003.

[12] T.J. Hsieh, H. F. Hsiao, and W.C. Yeh, “Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm,” *Applied soft computing*, vol. *11*(2), pp. 2510-2525, 2011.

[13] V. Marmer, “Nonlinearity, nonstationarity, and spurious forecasts,” *Journal of Econometrics*, vol. *142*(1), pp. 1-27, 2018.

[14] R. Engle, “GARCH 101: The use of ARCH/GARCH models in applied econometrics,” *Journal of economic perspectives*, vol. *15*(4), pp. 157-168, 2001.

[15] M. Göçken, M. Özçalıcı, A. Boru, and A.T. Dosdoğru, “Integrating metaheuristics and artificial neural networks for improved stock price prediction,” *Expert Systems with Applications*, vol. *44*, pp. 320-331, 2016.

[16] M. Hiransha, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, “NSE stock market prediction using deep-learning models,” *Procedia computer science*, vol. *132*, pp. 1351-1362, 2018.

[17] S. Radha, and M. Thenmozhi, “Forecasting short term interest rates using ARMA, ARMA-GARCH and ARMA-EGARCH models,” In *Indian Institute of Capital Markets 9th Capital Markets Conference Paper, 2006*.

[18] A. A. Ariyo, O. Adewumi, and C. K. Ayo, “Stock price prediction using the ARIMA model,” In *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, pp. 106-112, 2014.

[19] R. Singh, and S. Srivastava, “Stock prediction using deep learning,” *Multimedia Tools and Applications*, *76*(18), 18569-18584, 2017.

[20] E. Guresen, G. Kayakutlu, and T. U. Daim, “Using artificial neural network models in stock market index prediction,” *Expert Systems with Applications*, vol. *38*(8), pp. 10389-10397, 2011.

[21] W. Wang, “A big data framework for stock price forecasting using fuzzy time series,” *Multimedia Tools and Applications*, vol. *77*(8), pp. 10123-10134, 2018.

[22] R. Kusuma, T. Ho, W. Kao, Y. Y. Ou, and K. L. Hua, “Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market,” *arXiv preprint arXiv:1903.12258 ,2019*.

[23] L. Wang, Y. Zeng, and T. Chen, “Back propagation neural network with adaptive differential evolution algorithm for time series forecasting,” *Expert Systems with Applications*, vol. *42*(2), pp. 855-863, 2018.

[24] L. Y. Wei, and C. H. Cheng, “A hybrid recurrent neural networks model based on synthesis features to forecast the Taiwan stock market,” *Int. J. Innov. Comput. Inf. Control*, vol. *8*(8), pp. 5559-5571, 2012.

[25] S. Hochreiter, and J. Schmidhuber, “Long short-term memory,” *Neural computation*, *9*(8), 1735-1780, 1997.

[26] T. Fischer, and C. Krauss, “Deep learning with long short-term memory networks for financial market predictions,” *European Journal of Operational Research, 270*(2), 654-669, 2018.

[27] D. Karaboga, and C. Ozturk, “Neural networks training by artificial bee colony algorithm on pattern classification,” *Neural Network World*, *19*(3), 279, 2009.

[28] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, “A comprehensive survey: artificial bee colony (ABC) algorithm and applications,” *Artificial Intelligence Review*, *42*(1), 21-57, 2014.

[29] D. Karaboga, “An idea based on honeybee swarm for numerical optimization Technical report-tr06,” *Erciyes university, engineering faculty, computer engineering department*, vol. 200, pp. 1-10, 2005.

[30] M. Mernik, S. Liu, D. Karaboga, and M. Črepinšek, “On clarifying misconceptions when comparing variants of the Artificial Bee Colony Algorithm by offering a new implementation,” *Information Sciences*, *291*, 115-127, 2015.

[31] A. Karathanasopoulos, M. Sovan, M., Chun Lo, C., A. Zaremba, and M. Osman, “Ensemble Models in Forecasting Financial Markets,” *Journal of Computational Finance*, *Forthcoming ,2019*.

[32] F. Zhou, H. Zhou, Z. Yang, and L.Yang, “EMD2FNN: A strategy combining empirical mode decomposition and factorization machine based neural network for stock market trend prediction,” *Expert Systems with Applications*, vol. *115*, pp. 136-151, 2019.

[33] M. Göçken, M. Özçalıcı, A. Boru, and A.T. Dosdoğru, "Stock price prediction using hybrid soft computing models incorporating parameter tuning and input variable selection," *Neural Computing and Applications*, *31*(2), 577-592, 2019.

[34] F. Yang, Z. Chen, J. Li, and L. Tang, "A novel hybrid stock selection method with stock prediction," *Applied Soft Computing, 2019*.

[35] R. Bisoi, P. K. Dash, and A. K. Parida, "Hybrid Variational Mode Decomposition and evolutionary robust kernel extreme learning machine for stock price and movement prediction on daily basis," *Applied Soft Computing*, *74*, 652-678, 2019.

[36] F. Huang, X. Zhang, Z. Zhao, J. Xu, and Z. Li, "Image–text sentiment analysis via deep multimodal attentive fusion," *Knowledge-Based Systems*, *167*, 26-37, 2019.

[37] J. Xu, Z. Li, and S. Y. Philip, "Social Image Sentiment Analysis by Exploiting Multimodal Content and Heterogeneous Relations," *IEEE Transactions on Industrial Informatics, 2020*.

[38] R. K. Dwivedi, et al., "Sentiment analysis and feature extraction using rule-based model (RBM)," In *International Conference on Innovative Computing and Communications* (pp. 57-63). Springer, Singapore, 2019.

[39] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM," *Neural Computing and Applications*, 1-17, 2019.

[40] A. P. Rodrigues, and N. N. Chiplunkar, "Real-time Twitter data analysis using Hadoop ecosystem," *Cogent Engineering*, *5*(1), 1534519, 2018.

[41] M. Skuza, and A. Romanowski, "Sentiment analysis of Twitter data within big data distributed environment for stock prediction," In *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)* (pp. 1349-1354), 2015.

[42] P. K. Gupta, T. Ören, and M. Singh, "*Predictive intelligence using big data and the internet of things*," (Eds), IGI Global, 2018.

[43] R. Yang, L. Yu, Y. Zhao, et al., "Big data analytics for financial Market volatility forecast based on support vector machine. *International Journal of Information Management*, vol. *50*, pp. 452-462, 2020.

[44] R. Storn, and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*,*11*(4), 341-359, 1997.

[45] S. Das, S. S. Mullick, and P. N. Suganthan, “Recent advances in differential evolution– an updated survey,” *Swarm and Evolutionary Computation*, vol. *27*, pp. 1-30, 2016.

[46] A.F. Ali, and M. A. Tawhid, “A hybrid PSO and DE algorithm for solving engineering optimization problems,” *Appl. Math. Inf. Sci*, *10*(2), 431-449, 2016.

[47] D. Pelusi, R. Mascella, L. Tallini, J. Nayak, B. Naik, and Y. Deng, “Improving exploration and exploitation via a hyperbolic gravitational search algorithm,” *Knowledge-Based Systems*, *193*, 105404, 2020.

[48] W. Xiang, S. Ma, and M. An, “Habcde: a hybrid evolutionary algorithm based on artificial bee colony algorithm and differential evolution,” *Applied Mathematics and Computation*, *238*, 370-386, 2014.

[49] E. Zorarpacı, E., and S.A. Özel, “A hybrid approach of differential evolution and artificial bee colony for feature selection,” *Expert Systems with Applications*, *62*, 91-103, 2016.

[50] S. Jadon, R. Tiwari, H. Sharma, J. Bansal, “Hybrid artificial bee colony algorithm with differential evolution,” *Applied Soft Computing*, *58*, 11-24, 2017.

[51] B. J. Glover, “*Understanding flowers and flowering*,” Oxford University Press, 2007.

[52] A. H. Gandomi, S. Yang, and A. H. Alavi, “Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems,” *Engineering with computers*, *29*(1), 17-35, 2013.

[53] K. Khan, and A. Sahai, “A comparison of BA, GA, PSO, BP and LM for training feed forward neural networks in e-learning context,” *International Journal of Intelligent Systems and Applications*, *4*(7), 23, 2012.

[54] B. Fernández, et al., “Technical market indicators optimization using evolutionary algorithms,” In *Proceedings of the 10th annual conference companion on Genetic and evolutionary computation*, pp. 1851-1858, 2008.

[55] C.A.C Coello, and G. B. Lamont, “*Evolutionary algorithms for solving multi-objective problems”* vol. 5, pp. 79-104, New York: Springer, 2007.

[56] H. Chung, and K. S. Shin, “Genetic algorithm-optimized long short-term memory network for stock market prediction,” *Sustainability*, *10*(10), 3765, 2018.

[57] L. Peng, S. Liu, and L. Wang, "Effective long short-term memory with differential evolution algorithm for electricity price prediction," *Energy*, vol. *162*, pp. 1301-1314, 2018.

[58] S. Kumar, D.K. Yadav, & D.A. Khan, "Artificial bee colony based test data generation for data-flow testing," *Indian Journal of Science & Technology*, vol-*9*(39), 1-10, 2016.

[59] P. Jain, S. Jain, O.R. Zaïane, and A. Srivastava, "Anomaly detection in resource constrained environments with streaming data," *IEEE Transactions on Emerging Topics in Computational Intelligence*,2021.

[60] M. Pant, and V. Snasel, "Design optimization of water distribution networks through a novel differential evolution", IEEE Access, 9, 16133-16151, 2021.

[61] J. K. Chhabra, " Many- objective artificial bee colony algorithm for large-scale software module clustering problem," soft computing, 22(19), 6341-6361, 2018.

[62] S. Das, P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE transactions on evolutionary computation*, vol. *15*(1), pp. 4-31, 2010.

[63] B. Leiding, P. Sharma, A. Norta, "The Machine-to-Everything (M2X) Economy: Business Enactments, Collaborations, and e-Governance," *Future Internet*, *13*(12), 319, 2021.

[64] M. Z. Asghar, et al., "Development of stock market trend prediction system using multiple regressions," *Computational and Mathematical Organization Theory*, 1-31, 2019.

[65] W. C. Hong, "Hybrid evolutionary algorithms in a SVR-based electric load forecasting model," *International Journal of Electrical Power & Energy Systems*, *31*(7-8), 409-417, 2009.

[66] E. Hoseinzade, and S. Haratizadeh, "CNNpred: CNN-based stock market prediction using a diverse set of variables," *Expert Systems with Applications, 2019*.

[67] M. M. Aliabadi, H. Emami, M. Dong, and Y. Huang, "Attention-based recurrent neural network for multistep-ahead prediction of process performance," *Computers & Chemical Engineering*, *140*, 106931, 2020.

[68] W. Lu, et al., "A CNN-BiLSTM-AM method for stock price prediction," *Neural Computing and Applications*, *33*(10), 4741-4753, 2021.

[69] C. Xiao, W. Xia, and J. Jiang, "Stock price forecast based on combined model of ARI-MA-LS-SVM," *Neural Computing and Applications*, *32*(10), 5379-5388, 2020.

[70] I. E. Livieris, E. Pintelas, and P. Pintelas, "A CNN–LSTM model for gold price time-series forecasting," *Neural computing and applications*, *32*(23), 17351-17360, 2020.

[71] National Stock Exchange (NSE)., Bombay Stock Exchange (BSE)., (2020, September 30). *Historical Datasets*. Retrieved from https://finance.yahoo.com/quote/ Last Accessed: Sept 2020.

[72] R. Kumar, A. Gupta, and A. Mishra, "Design of Ensemble Learning Model to diagnose malaria disease using convolutional neural network," In *International Conference on Innovative Computing and Communications*, pp. 1165-1176, Springer, Singapore, 2021.

[73] J. Thomas, and N. S. Chaudhari, "An integrated genetic algorithm approach to 1D-cutting stock problem," *International Journal of Operational Research*, *27*(1-2), 23-46, 2016.

[74] P. F. Pai, and C. S. Lin, "A hybrid ARIMA and support vector machines model in stock price forecasting," *Omega*, *33*(6), 497-505, 2005.

[75] C. N. Babu, and B. E. Reddy, "Prediction of selected Indian stock using a partitioning–interpolation based ARIMA–GARCH model," *Applied Computing and Informatics*, *11*(2), 130-143, 2015.

[76] S. D. O. Domingos, J. F. de Oliveira, and P. S. de Mattos Neto, "An intelligent hybridization of ARIMA with machine learning models for time series forecasting," *Knowledge-Based Systems*, *175*, 72-86, 2019.

[77] M. Khashei, and M. Bijari, "An artificial neural network (p, d, q) model for time series forecasting. *Expert Systems with applications*, *37*(1), 479-489, 2010.

[78] M. Khashei, and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Applied Soft Computing*, *11*(2), 2664-2675, 2011.

[79] K. Zhou, et al. "Comparison of Time Series Forecasting Based on Statistical ARIMA Model and LSTM with Attention Mechanism," In *Journal of Physics: Conference Series* (Vol. 1631, No. 1, p. 012141). IOP Publishing, 2020.

[80] T. Vantuch, and I. Zelinka, “Evolutionary based ARIMA models for stock price forecasting,” In *ISCS 2014: Interdisciplinary Symposium on Complex Systems* pp. 239-247, Springer, Cham, 2015.

[81] A. Musdholifah, and A. K. Sari, “Optimization of ARIMA Forecasting Model using Firefly Algorithm,” *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *13*(2), 127-136.

[82] R. Ballini, et al. “A comparative analysis of neurofuzzy, ANN and ARIMA models for Brazilian stock index forecasting,” *SCE-Computing in Economics and Finance, 1995*.

[83] J. H. Wang, and J. Y. Leu, “Stock market trend prediction using ARIMA-based neural networks,” In *Proceedings of International Conference on Neural Networks (ICNN'96)* vol. 4, pp. 2160-2165, 1996.

[84] H. Tang, K.C. Chiu, X. Lei, “computational intelligence in economics and finance, North Carolina, USA, pp 1112–1119 17, 2003.

[85] T. Duan “Auto regressive dynamic Bayesian network and its application in stock market inference,” In: IFIP international conference on artificial intelligence applications and innovations. Springer, Berlin 18, 2016.

[86] G. Hinton, L. Deng, and D. Yu “Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups,” IEEE Signal Process Mag 29(6):82–97 ,2019.

[87] D. Zhang, H. Song, and P. Chen, “Stock market forecasting model based on a hybrid ARMA and support vector machines,” In: International conference on management science and engineering. IEEE 20, 2008.

[88] S. H. Xingjian et al., “Convolutional LSTM network: A machine learning approach for precipitation nowcasting,” In: Advances in neural information processing systems 2015, pp. 802–810, 2015.

[89] X. Ma, Z. Tao, and Y. Wang, “Long short-term memory neural network for traffic speed prediction using remote microwave sensor data,” Trans Res Part C Emerg Technol 54:187–197 23, 2015.

[90] H. Liu X. Mi, and Y. Li, “Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM,” Energy Convers Manag 159:54–64 24, 2018.

[91] C. Mala, et al., “A hybrid artificial bee colony algorithmic approach for classification using neural networks,” In *2nd EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing* (pp. 339-359), Springer, Cham, 2021.

[92] A. Yoshihara, K. Fujikawa, K. Seki, and K. Uehara, “Predicting stock market trends by recurrent deep neural networks,” In Proceedings of the Pacific Rim International Conference on Artificial Intelligence, Gold Coast, Australia, 1–5 Springer: Berlin/Heidelberg, Germany, 2014; pp. 759–769. 9, 2014.

[93] E. W. Saad, D. V. Prokhorov, and D. C. Wunsch, “Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks,” IEEE Trans. Neural Network, vol. 9, pp. 1456–1470, 1998.

[94] Y. Chen, A. Abraham, J. Yang, and B. Yang, “Hybrid methods for stock index modeling,” In Proceedings of the International Conference on Fuzzy Systems and Knowledge Discovery, Changsha, China, Springer, pp. 1067–1070, 2005.

[95] H. Yu, R. Chen, and G. A. Zhang, “A SVM stock selection model within PCA,” Procedia Computer Science, vol. 31, pp. 406–412, 2014.

[96] Y. Chen, and Y. Hao, “A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction,” Expert System Application, vol. 80, pp. 340–355, 2017.

[97] X. Glorot, A. Bordes, and Y. Bengio, “Domain adaptation for large-scale sentiment classification: A deep learning approach,” In Proceedings of the 28th International Conference on Machine Learning (ICML-11), Bellevue, WA, USA, pp. 513–520, 2011.

[98] V.E. Balas, R. Kumar, and R. Srivastava, “Recent trends and advances in artificial intelligence and internet of things,” pp. 389-425, Springer, 2020.

[99] Y. Guo et al., “Deep learning for visual understanding: A review,” Neurocomputing, vol. 187, pp. 27–48, 2016.

[100] E. Chong, C. Han, and F. C. Park, “Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies,” Expert System. Application, vol. 83, pp. 187–205, 2017.

[101] J. Lee, D. Jang, and S. Park, “Deep Learning-Based Corporate Performance Prediction Model Considering Technical Capability,” Sustainability, vol. 9, pp. 8-99, 2017.

[102] O. B. Sezer, M. Ozbayoglu, and E. Dogdu, “A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters,” Procedia Computer. Science. 2017, 114, 473–480, 2017.

[103] X. Li, P. Wu, and W. Wang, “Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong,” *Information Processing & Management*, *57*(5), 102212, 2020.

[104] H. Yasin, R. E. caraka, “Stock price modeling using localized multiple kernel learning support vector machine,” *ICIC express letters. Part B, Applications: an international journal of research and surveys*, *11*(4), 333-339, 2020.

[105] S. Rikukawa, H. Mori, and T. Harada, “Recurrent neural network based stock price prediction using multiple stock brands,” *International Journal of Innovative Computing, Information and Control*, vol. *16*(3), 1093-1099, 2020.

[106] M. Mahan, C. Chorn, and A. Georgopoulos, “White Noise test: Detecting autocorrelation and nonstationarities in long time series after Arima modeling,” *Proceedings of the 14th Python in Science Conference*, 2015.

[107] N. H. An, and D. T. Anh, “Comparison of strategies for multi-step-ahead prediction of time series using Neural Network,” *2015 International Conference on Advanced Computing and Applications (ACOMP)*, 2015.

[108] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, “Multiple-output modeling for multi-step- ahead time series forecasting,” *Neurocomputing*, vol. 73, no. 10-12, pp. 1950–1957, 2010.

[109] S. Ben Taieb, G. Bontempi, A. F. Atiya, and A. Sorjamaa, “A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition,” *Expert Systems with Applications*, vol. 39, no. 8, pp. 7067–7083, 2012.

LIST OF PUBLICATIONS

Journal (s)

1. R. Kumar, P. Kumar, Y. Kumar, “Three Stage Fusion for Effective Time Series Forecasting using Bi-LSTM-ARIMA and Improved DE-ABC Algorithm,” Neural Computing and Application, Springer (2022). **(SCIE Indexed, IF=5.606).**

2. R. Kumar, P. Kumar, Y. Kumar, “Integrating big data driven sentiments polarity and ABC-optimized LSTM for time series forecasting,” Multimedia Tools and Applications, 1-20, 2021. **(SCIE Indexed, IF=2.757).**

3. R. Kumar, P. Kumar, Y. Kumar, “Two Phase Hybridization using Deep Learning and Evolutionary Algorithms for Stock Market Forecasting,” International Journal of Grid and Utility Computing, *12*(5-6), 573-589 (2021). **(Scopus, ESCI Indexed).**

4. R. Kumar, P. Kumar, Y. Kumar, “Multi-step Time Series Analysis using Hybrid Model of ARIMA and Evolutionary Algorithms,” International Journal of Information Technology, 2021 **(Scopus Indexed).**

5. R. Kumar, P. Kumar, Y. Kumar, “hybrid time series data model to enhance forecasting accuracy of stock market trends using optimized DE-LSTM,” International Journal of Operational Research (*in press*) (**Scopus Index).**

6. R. Kumar, P. Kumar, Y. Kumar, “Time Series Data Prediction using IoT and Machine Learning Technique,” Procedia computer science, Vol.167, pp. 373-381, 2020. **(Scopus Indexed)**

Conference (s)

1. R. Kumar, P. Kumar, Y. Kumar, "Analysis of Financial Time Series Forecasting using Deep Learning Model," In 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence) pp. 877-881, (2021). IEEE.

2. R. Kumar, P. Kumar, Y. Kumar, "Time Series Data Prediction using IoT and Machine Learning Technique," Procedia computer science, Vol.167, pp. 373-381, 2020.