Advances in Artificial Intelligence and Machine Learning; Research 4 (1) 1892-1924 Received 1-12-2023; Accepted 30-01-2024; Published 6-2-2024

Influence of Artificial Intelligence on Forecasting Net Asset Value and Return Volatility in Indian Mutual Fund

Jesmine Mary Antony jesminemary.a2021@vitstudent.ac.in
Research Scholar, Department of Commerce
School of Social Sciences and Languages
Vellore Institute of Technology
Tamil Nadu, 632014,
India

Sundaram Natarajan nsundaram@vit.ac.in
Professor, Department of Commerce
School of Social Sciences and Languages
Vellore Institute of Technology
Tamil Nadu, 632014,
India

Corresponding Author: Sundaram Natarajan

Abstract

Predicting the critical net asset value (NAV) in the financial market is difficult for investors and fund agencies. The present study introduces machine learning (ML) and deep learning (DL) models such as linear regression, deep long short-term memory recurrent neural network (DLSTM-RNN), and autoregressive integrated moving averages (ARIMA) for predicting the NAV. The five different equity sectoral technology mutual fund direct growth plans from January 2013 to December 2022 have been collected. The novelty of the current study is deeply examining, which ML or DL model devotedly predicts the NAV closing price. The major key findings of the experimental results proved that the DLSTM-RNN model makes statistically viable predictions, whereby the mean absolute percent error (MAPE) average prediction accuracy value is 0.02. Based on the accuracy of a superior model, we compute the annualized return volatility to compare the risk of investments with annual return periods over different time horizons. The Jarque-Bera statistics of the return volatility over time Gaussian distribution is rejected at the 0.01 level. Statistical paired t-test and Pearson correlation coefficient are used to compare the effects of the proposed three models. In addition, the benchmark portfolio strategy yields a Sharpe ratio of 7.0193 and the maximum drawdown is 0.3743. The AI performed deep LSTM neural network model simulation, especially when using a daily and monthly MAPE strategy giving 81% and 84% highest NAV prediction consistency than the linear regression and ARIMA models.

Keywords: Artificial intelligence, Technology sector mutual fund, Net asset value prediction, Return volatility prediction, Linear regression, Deep long short-term memory neural network, Autoregressive integrated moving average.

Citation: Jesmine Mary Antony and Sundaram Natarajan. Influence of Artificial Intelligence on Forecasting Net Asset Value and Return Volatility in Indian Mutual Fund. Advances in Artificial Intelligence and Machine Learning. 2024;4(1):109.

1. INTRODUCTION

A mutual fund is an investment vehicle that professionally succeeds and collects money from investors to buy securities [1]. Among mutual fund plans, technology (IT) funds capitalize on the assets of businesses engaged in the IT industry. India is one of the hotspot countries for supplying IT services. Through its entry into trading activity with a good artificial intelligence (AI) training data set, it offers precious investment decisions to trading investors and the funding agency [2]. Thus, it optimizes the trading performance and regulators to diversify the parameter risk among dissimilar IT stocks and shifts to buy or sell orders using the net asset value (NAV) price of the trading time [3]. The price at which investors purchase and sell shares of a mutual fund on a typical trading day is known as the net asset value per share. Every trading day, NAV is calculated using the closing market values of the securities in the portfolio. The NAV performance per share of a mutual fund refers to the price at which investors buy and sell shares on a regular trading day. However, the transaction price cannot be obtained until the next day. The NAV represents the company's asset and liability condition [4]. In consequence, rising NAV frequently designates business growth. Considering that NAV closing price data have a nonlinear character. Indeed, no shareholder would desire to invest more, and no new investor would want to take on risk when a NAV keeps declining. However, it is challenging to forecast how the NAV will change because of the frequent fluctuations in the value of a mutual fund's assets, cash on hand, liabilities, number of outstanding shares, and numerous other factors. This makes predicting NAV closing data an extremely challenging effort. When we apply AI models it will aim to address the particular challenges related to NAV prediction price which helps to make important decisions regarding trades in the mutual fund and to reduce risk, maximize return, or prevent insolvency. This study focuses on the performance of different machine learning models and the accuracy level for predicting the NAV closing prices and volatility returns in the securities market. Hence, the inclusive research question of this work is:

Research Question: *What is the performance of AI-influenced machine learning models for predicting the NAV closing price and return volatility in the security market?*

To answer this research question, the study employs two machine learning (ML) models and one deep learning (DL) classifier to predict the relative daily performance of the 11848 largest NAV per share of a mutual fund. In this connection, the study roofed five technology sector mutual fund spectrums such as ICICI-2517, Tata-1785, Aditya-2517, SBI-2511, and Franklin-2518 direct growth plan respectively. Furthermore, we analyzed the volatility return-based trading strategy by the outperformed deep learning model. This study has five main contributions:

First, we exhibit the strategized NAV prediction accuracy of all three machine learning models. Then, we compare the results of MAE, MAPE, MSE, and RMSE error metrics for evaluating the model performance. Here, our DLSTM-RNN model is highly accurate in that the average mean absolute percentage error (MAPE) arrived at 0.02 error in the loss function i.e., 98% the model is highly recommended. Third, we focus on each financial spectrum's annualized log return volatility and over time volatility for analyzing the historical risk-return strategy. Fourth, we evaluate the different portfolio indicators such as the Sharpe ratio, Sortino ratio, Modigliani ratio, Max drawdown, and Calmar ratio to extensively evaluate the risk-return performance of the AI-influenced deep neural network model. Finally, we used a paired t-test to hypothetically test the accuracy of the predicted closing price.

The remainder of this study is designed as follows: Section 2 presents the literature work, Section 3 projects the methodological strategy, Section 4 discusses the result analysis and inferences of the experimental results, and Section 5 concludes the work.

2. LITERATURE WORK

Most of the studies have used modern machine learning models for the prediction of stock price which enables investors to get high trading returns with fewer risks. In [5], the authors compared the stock market prediction by using the Moving Average (MA) with the Support Vector Machine (SVM) and LSTM neural network algorithms based on the Dow Jones Industrial Average (DJIA). The SVM with an MA p-value of 9.38e-25 and the LSTM with an MA p-value of 4.82e-97 are the two models whose performances are compared with p-values less than 0.05. Hence, the LSTM advanced model with moving averages performs better in predicting stock prices than the SVM. This study [6], uses two deep learning models, CNN and LSTM-neural network, that are based on the NIFTY-50 daily index values of the National Stock Exchange (NSE) of India, from December 29, 2014, to July 31, 2020. The findings of the research comparison demonstrated that the CNN model before one week of data as the input. Conversely, the encoder-decoder convolutional LSTM model finds that its forecasting results are most accurate when it takes the data from the prior two weeks as the input is the fastest in its execution. In [7], the authors created an ARIMA-LSTM hybrid model based on a 1260-daily NAV from June 2016 until July 2021. The hybrid ARIMA-LSTM model is the basis of this paper's fund prediction technique. In this work, the linear data characteristics were first eliminated by preprocessing the historical data and then applying the ARIMA model. After receiving the data, the LSTM model extracted the nonlinear features using the residual. To obtain the prediction results for the hybrid model, it inputs the corresponding prediction values from the two models at the end. Hence, the results demonstrated that the forecasting tool is more reliable as well as efficient for handling complex time series problems like the fund's NAV. In [8], the authors employ the potential of recurrent neural networks such as LSTM and gated recurrent units for predicting the NAV closing price based on a dataset ranging from January 1997 until December 2002. In the obtainable method, they analyzed the 1145 data samples and more than 1000 samples in each dataset to make trading predictions. Finally, they found that a single LSTM-RNN layer and bidirectional recurrent neural network dropout yield the best performance on their prediction accuracy. Hence, the result shows that the LSTM-RNN is a better-suited model to forecast the NAV.

In addition to the net asset value prediction, the present study analyzed the volatility price return strategy for measuring the dispersion of annualized returns. In [9], the authors applied deep LSTM-RNN and SVM models for forecasting the volatility return based on a dataset ranging from January 2000 to December 2011. In their study, they used two financial indices as S&P 500 and AAPL. They find that deep learning LSTM-RNN big data can be used to improve volatility prediction instead of SVM. In this case, the support vector machine did not predict well some financial stocks of a portfolio. In [10], the authors proposed a deep learning model such as the LSTM-RNN for predicting the copper price volatility in industrial applications. The study synthesized two models. One is the classic generalized autoregressive conditional heteroskedasticity (GARCH) model and the next one is the deep LSTM-RNN model. The results suggest that the LSTM-RNN model should consider the forecast horizon and time-varying copper price volatility to optimize the prediction results. Hence, through the literature review, it is very clear that the deep LSTM-RNN model is the most suitable model for predicting return volatility.

Different studies have suggested that the business cycle has nonlinear properties and economic variables display asymmetries throughout economic booms and busts [11–13]. The nonlinear models are interchangeable LSTM and RNN, whereas the LR and ARIMA are linear and sublinear models. Normally the NAV contains both linear as well as nonlinear variation concerning time. In [14], the authors compared the predictive abilities of LSTM-RNN with linear regression. They find that an LSTM-RNN model outperforms better returns than the linear and sub-linear models. These linear and ARIMA models have some significant flaws, one of which is their inability to account for the non-linearity in the data, which makes the forecast less accurate. Hence, analyzing the two-parameter differences in the literature of the work, the current study has considered the challenging task of both linear and non-linear models to recap the easiest way of arriving at an accurate prediction. These aforementioned non-linear models suggested that combining more than one forecast model delivers valuable guidance for superior time-stamped data prediction compared to a single forecasting model. By comparing these models, we arrived at a novel idea to analyze forecast accuracy and return volatility. Hence, the present study was established based on the previous study that employed the ML and DL models.

3. METHODOLOGY

The work encompasses five main stages and builds on methodological approaches [15, 16]. In the first phase, we obtain the relevant datasets composed for the study. Then, we designed the data analysis with statistics of different datasets accessible in all three models. The next step was individual training and testing datasets into different daily and monthly testing samples for comparing the performance error metrics with the individual model. In the fourth phase, we analyze the annualized volatility return using the deep learning neural network model to compare the risk of investments with annual return periods over the time horizons. In this case, we considered 252 trading days for calculating the annualized volatility return. In the final phase, the study calculated a paired t-test for all three models by comparing the different IT-sector dataset schemes.

3.1 Data Collection

For this study, we use daily NAV closing prices over the period from January 2013 to December 2022. We obtain the raw data through the AMFI India database to the subsequent technology schemes such as ICICI, Tata, Aditya, SBI, and Franklin. However, the Tata scheme inception period is 9^{th} December 2015. Hence, we took the Tata scheme study period from December 2015 to December 2022. The five different fund-wise direct growth plan NAV data is assessed by using both training and testing phases. The different volumes of datasets adopted for the study are ICICI-2517, Tata-1787, Aditya-2517, SBI-2511, and Franklin-2518.

The collected dataset, first we implemented to calculate the total market value of the IT sector mutual fund in the financial market. Net asset value (NAV) is the total asset value of a mutual fund less all of its liabilities. The total fund market value per share represents the NAV. FIGURE 1 shows the year-wise total market value of IT sector mutual funds that appeared in 2013 and gradually increased performance of the net asset value through 2022. Stockholders purchase the fund shares from a fund business at this price and then sell those shares back to the fund company. However, investors can be affected by economic factors, which affect the value of mutual funds in the IT industry.

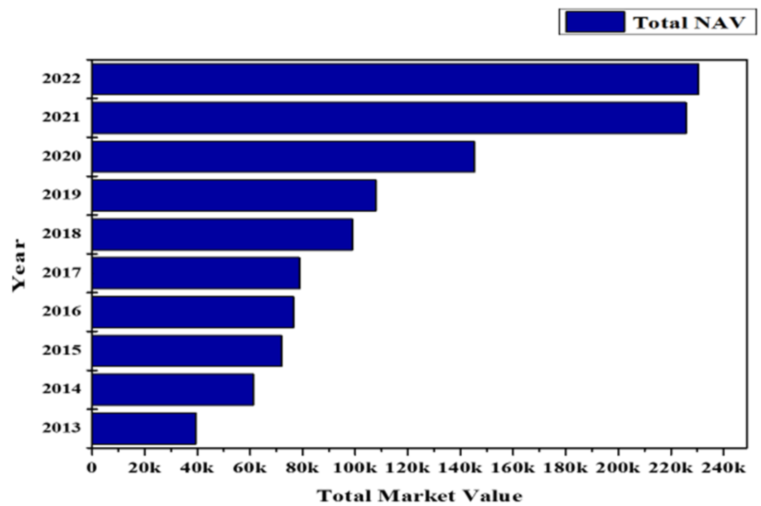

Figure 1: The total market value of the IT-sector mutual fund in the Indian financial market.

FIGURE 2, illustrates the date-specific actual NAV and volumes debuted from 2013 to 2022. There is a gradual increase in performance as the year passes (2013-2022). At the year-end of 2019, a sudden decrease in NAV performance can be seen. However, the annual year of 2020-2021 showed an excellent growth period and there is a negligible performance at the end of the year 2021. In the year 2022, the technology sectors showed a steady decline in their performance. Hence, the gap resulted in the historical market performance of the selected technology sectors such as SBI, Aditya, and TATA mounted in the highest positions of the current market.

3.2 Designing Data with Statistics

We used the trading day travel market value data of the five different technology funds on the trade floor exchange from January 2013 to December 2022 (Tata fund, December 2015 to December 2022). Every trading day contains two sets of opening and closing prices. The overall dataset offered for the study is 11848 and the yield produced by the model is 11846. This includes 9479 (80%) of training data and 2367 (20%) of validation and testing data. TABLE 1 describes the total data collection period, different tags of the scheme, number of training and testing patterns, and observed dataset used in the models. The term ”acquired data” in this case refers to all of the datasets that were gathered for the study between January 2013 and December 2022. The term ”produced data” describes all of the data that the model generated between January 2013 and December 2022 that did not contain null values. 20% of the data is used for testing, while the remaining 80% is used for training. The time frame for the data’s training and testing is shown in TABLE 2.

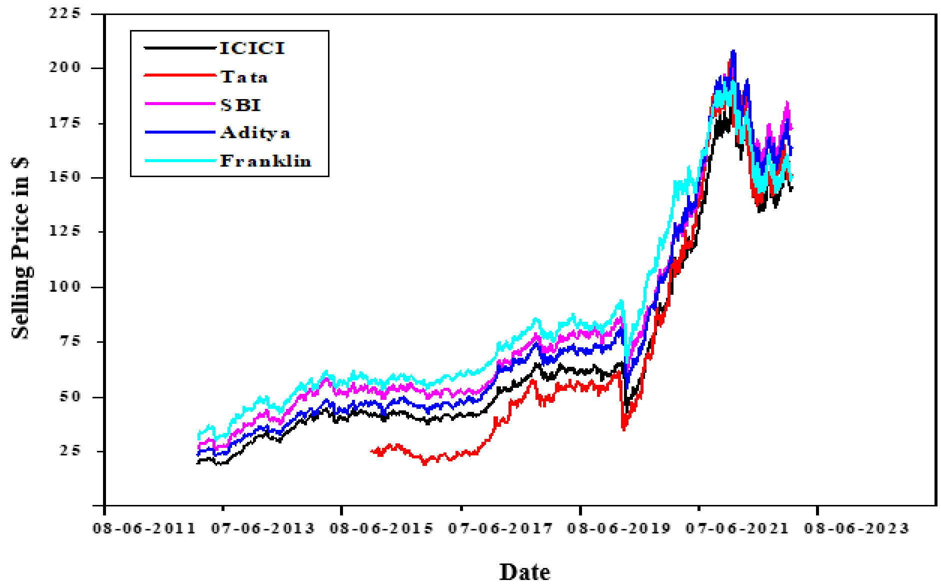

Figure 2: The comparison of date-specific actual and closing NAV in dollars.

Table 1: Summary of IT sector mutual fund datasets used for training and testing.

Name of the scheme	Dataset period	Acquired data	Produced data	Trained data	Tested data
ICICI prudential technology fund	02-01-2013 to 30-12-2022	2517	2517	2014	503
Tata Digital India funds	09-12-2015 to 30-12-2022	1785	1785	1428	357
Aditya Birla Sun Life Digital India fund	02-01-2013 to 30-12-2022	2517	2517	2014	503
SBI technology opportunities fund	10-01-2013 to 30-12-2022	2511	2510	2009	501
Franklin India Technology fund	01-01-2013 to 30-12-2022	2518	2517	2014	503
		11848	11846	9479	2367

Source: https://www.amfiindia.com

We obtained the trading data from the *AMFI* Association of *SEBI*-registered mutual funds in India. We gathered technology-wise mutual funds' direct growth spectrum, *viz.,* ICICI Prudential Technology Fund, Tata Digital India Fund, Aditya Birla Sun Life Digital India Fund, SBI Technology Opportunities Fund, and Franklin India Technology Fund, as the research sample. The reason for this selection is that all these five schemes are currently beating the benchmark return by a big margin and fully experienced the market cycles (bull and bear phases) since their dates of incep-

Table 2: The time frame of the data training and testing period.

Name of the scheme	Trained period	Tested period
ICICI prudential technology fund	02-01-2013 to 17-12-2020	18-12-2020 to 30-12-2022
Tata Digital India funds	09-12-2015 to 19-07-2021	20-07-2021 to 30-12-2022
Aditya Birla Sun Life Digital India fund	02-01-2013 to 17-12-2020	18-12-2020 to 30-12-2022
SBI technology opportunities fund	10-01-2013 to 16-12-2020	17-12-2020 to 30-12-2022
Franklin India Technology fund	01-01-2013 to 16-12-2020	17-12-2020 to 30-12-2022

Source: Data interpretation using ML and DL techniques.

tion. TABLE 3 and TABLE-5 summarize all five financial datasets benchmark returns, investment performance, and stock-fund holding strategies. FIGURE 3 shows the data processing architecture.

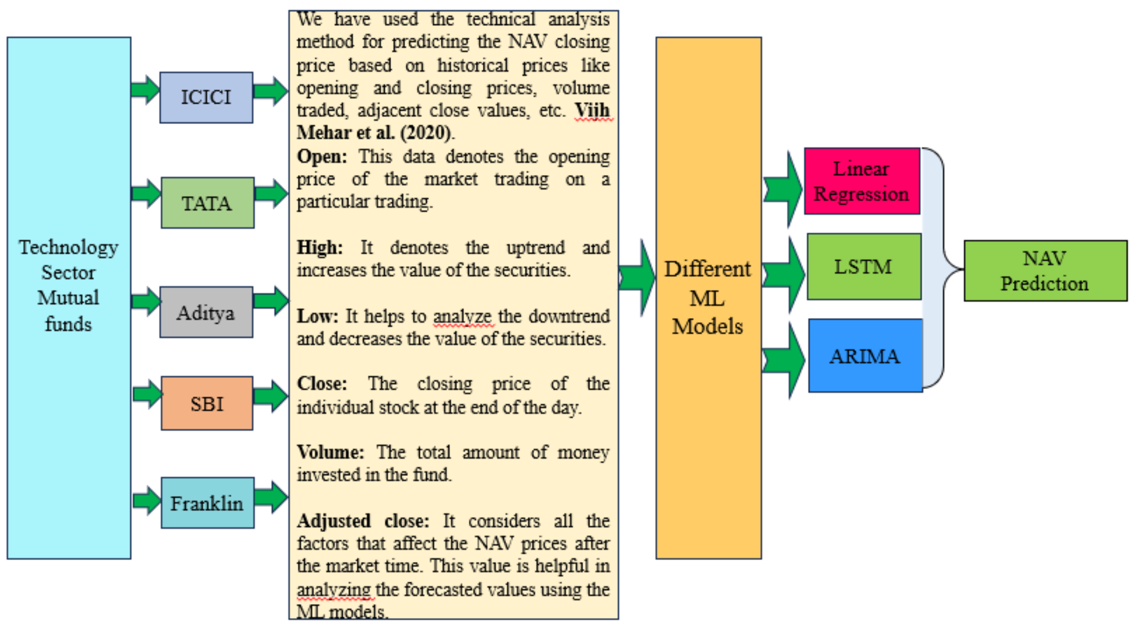

Figure 3: The data processing architecture

3.3 Software Database

We used *Anaconda3 (64-bit)* for data acquisition, processing, and analysis throughout the study. Here, Anaconda is an excellent option for those who are new to *Python* and data science. The *NumPy, pandas, Cython, urllib3, and joblib, etc.,* Python software packages are used for data processing and feature creation. Deep learning models are built using *Keras* with the *TensorFlow* backend and all other machine learning models are built and trained using the *matplotlib, pmdarima, and scikit* libraries.

Table 3: Summary of technology mutual fund investment performance and benchmark return.

Scheme	Genesis	Fund category	Benchmark category	Avg. annual return	AUM (₹Cr.)	Fund return (%)	Expense (%)	CAGR return (%)
ICICI	Jan. 2013	Equity sectoral technology	S&P BSE tech TRI	18.54	8772	78.95	0.71	32.50
Tata	Dec. 2015	Equity sectoral technology	Nifty IT TRI	18.84	3842	80.90	0.43	28.30
Aditya	Jan. 2013	Equity sectoral technology	S&P BSE tech TRI	17.61	3161	22.79	0.70	27.00
SBI	Jan. 2013	Equity sectoral technology	S&P BSE tech TRI	20.09	1890	24.90	0.98	26.70
Franklin	Aug. 1998	Equity sectoral technology	S&P BSE tech TRI	18.83	874	$711m	1.45	25.42

 # TRI- Total return index; ***Tables 2-4 sources:*** *amfiindia.com*

Table 4: Summary of technology mutual fund stock holding and fund holding.

	Stock holding (%)						Fund holding (%)					
Scheme	**Portfolio**	**Equity**	**Large cap**	**Mid cap**	**Small cap**	**Foreign**	**Infosys**	**HCL**	**Tech Mahindra**	**Wipro**	**TCS**	**Bharti Airtel**
ICICI	40	94.02	73.61	5.42	6.12	4.05	30.43	9.44	9.09	8.20	7.58	6.95
Tata	36	91.03	58.86	14.73	10.10	4.07	21.25	9.15	9.74	1.98	12.45	6.25
Aditya	23	95.71	70.97	18.95	5.78	55%	21.68	8.27	7.23	5.25	9.82	5.97
SBI	24	94.10	49.00	7.25	11.47	22.7%	22.47	8.43	7.11	5.91	14.22	10.11
Franklin	36	68.46	49.90	4.82	3.81	29.18%	16.57	9.42	3.65	14.84	6.24	-

Table 5: Summary of different trailing returns (%) performance in technology mutual funds over a specific time.

Fund	Since launch	5 year	3 year	1 year	6 months	3 months	1 month	1 week
ICICI	12.17	19.56	27.35	12.34	12.25	14.09	5.75	3.63
Tata	20.08	20.01	30.33	17.64	13.91	13.91	5.64	3.76
Aditya	11.70	19.87	27.41	19.11	17.09	16.30	6.93	4.00
SBI	21.09	20.74	29.76	22.18	12.54	13.33	5.54	3.45
Franklin	19.25	18.52	23.22	33.14	27.40	20.29	5.59	3.21

3.4 Models

Selecting models is the true challenge of applied machine learning. We scrutinize and compare various types of predictive models including linear regression, deep LSTM recurrent neural network, and ARIMA model as a simple and proficiently computed benchmark. The application of the deep LSTM-RNN model to time series prediction addresses a particularly challenging problem because of the presence of random noise, seasonal and cyclical oscillations, and long-term trends. The selected deep LSTM recurrent neural network model fitting criteria are relatively straightforward. For the outstanding stochastic character of the training process, we train residual models other than linear regression using various training datasets by averaging the cross-sectional ranks resulting from the predicted possibilities. The hyperparameter is separately optimized for each study period using the classification performance.

3.4.1 Linear Regression

In [17], the author outlined that linear regression is the first regression analysis method that underwent in-depth research and it is widely used in actual applications. The correlation between selling price movement and period predicted firms of the high-increase price of the stock over time, where the estimation browned by the trend linear regression metrics. Moreover, it is simpler to control the statistical features of the resulting estimators. Therefore, it is easier to predict trends and significances of any given dataset (NAV) more rapidly. Hence, these models with a linear dependence on their unidentified parameters are more modest to fit in than other models with non-linear dependence on their parameters.

We comprise the regression model of one independent variable and the dependent variables which are called simple linear regression. The independent variable signifies the time and the dependent variables make it difficult to predict the stock price in the best-fit line Y using the following equation [18].

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon \tag{1}$$

Where β_0 represents the NAV closing period and β_1 represents the daily selling price of the NAV. Hence, Y measures the predicted value of the dependent variable fund's assets between the purchase price and selling price beyond the capital investment and profit dividends had improved. Where X_1 is the independent variable that measures the daily selling price movement with the slope point. In the end, the outcome is constant between β_0 and $\beta_1 X_1$. Epsilon (ε) represents the errors, which measure the discrepancy between each point in the dataset. The linear system involves fitting a straight line to permit the greatest possible sum of the dataset's position points using the following equation [19].

$$y = mx + c \tag{2}$$

The ”Squared error function” or ”Mean squared error” are different terms for the error evaluation metrics. FIGURE 4 displays the actual and predicted value of different price movements with an intercept point.

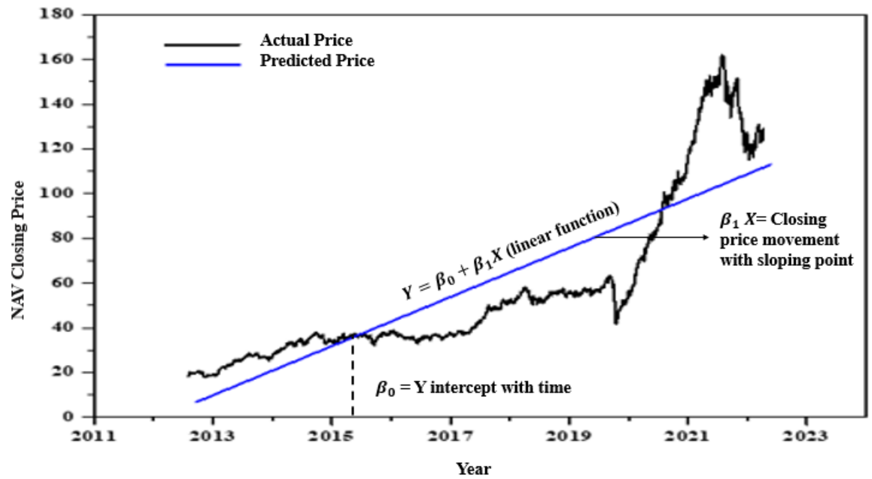

Figure 4: The linear regression architecture [20]

3.4.2 Deep LSTM Recurrent Neural Network

In [21], the author elaborated that LSTM is the recursive neural network method that can be used for the modeling of consecutive data in the normal recurrent neural network (RNN), also known as vanilla RNN. In [22], the authors explained the network delay recursion of RNN is a key characteristic, allowing it to depict the energetic routine of systems. The LSTM has feedback connections, unlike conventional feed-forward neural networks. It can process both individual data points and full data sequences. The LSTM ignores unrelated data and only retains the relevant information, using it to make predictions. Hence, the LSTM-RNN model recognizes only the fundamental information about a stock and disregards its outliers. FIGURE 5 shows how the LSTM-RNN architecture, an advanced variant of recurrent neural networks retains memory to handle data arrangements. The three gates in the LSTM-RNN cell, the forget (f_t), input (i_t), and output (o_t) gates are used to filter out information and modify the cell's state. Specifically, the forget gate establishes the appropriate threshold for removing information. The output gate determines how much data should be used as output, while the input gate determines how much data should be added. h_t stands for hidden layers, while x_t stands for the independent variable (new information) and c_t represents the memory cell's internal state. The gates may prudently control the cell state by eliminating or accumulating information. To limit the amount of information that is approved through the cell, gates have been applied. A cell, an input gate (i_t) layer chooses which values to be updated, an output gate and a forget gate make up the LSTM element. The three gates control the flow of information into and out of the cell, and the cell recalls values across random time intervals. The outcome usually ranges from 0 to 1, with 0 meaning "reject all" and 1 meaning "comprise all". The memory cell state, which resembles a transporter belt, is the central component of the LSTM structure. The gates deliver elective entry points for information obtained through [23].

FIGURE 6 shows a sigmoid neural net layer and a pointwise increase process. LSTM's first step is to choose information from the cell state that will be removed. The forget gate (f_t) makes the next decision using the following formula [24].

$$f_t = \sigma\left(X_t U^f + S_{t-1} W^f + b_f\right) \tag{3}$$

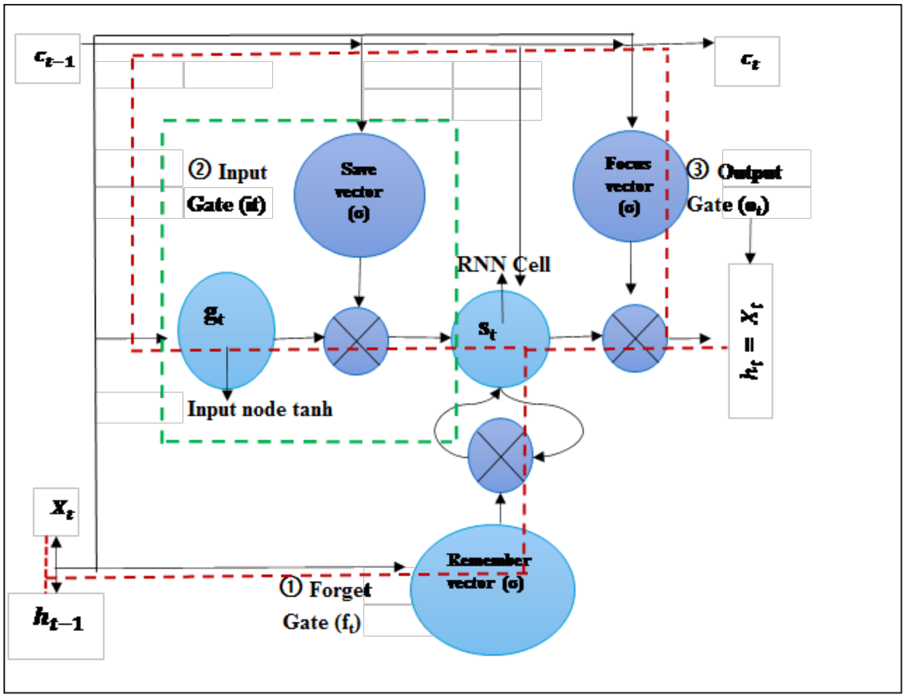

Figure 5: The assemble point in DLSTM - RNN architecture [25]

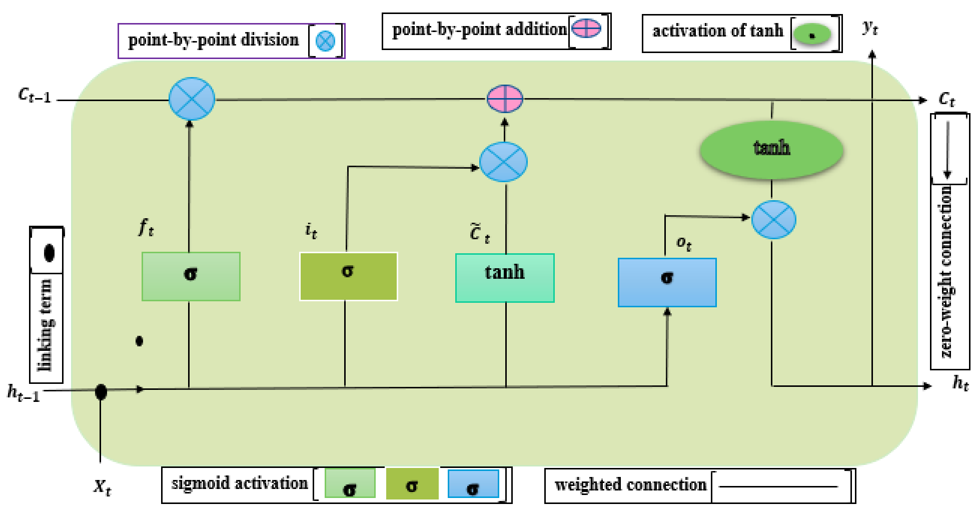

Figure 6: The DLSTM recurrent neural network block architecture [26]

Where X_t refers to the data input at the current time step, S_{t-1} refers to a cell state that can continuously update over time as information goes in or out. The forget gate uses a sigmoid function to calculate how much data should be removed from the cell state at the final timestep. Obviously, f_t ranges from 0 to 1, where 0 denotes removing all prior data and 1 denotes retaining all prior data.

LSTMs are made up of three logistic sigmoid gates and one tanh layer ($\widetilde{C_t}$), in contrast to RNNs which feature a single tanh layer. It will choose necessary information by the following cell and which should be ignored. The two-input gate and tanh layer are calculated using the mathematical formulas followed by [27].

$$i_t = \sigma\left(X_t U^i + S_{t-1} W^i + b_i\right) \tag{4}$$

$$\widetilde{C_t} = \tanh(X_t U^c + S_{t-1} W^c + b_c) \tag{5}$$

Where W^i, W^c represents weight parameters and b_i, b_c represents the bias parameters. The subsequent step is to change the previous cell state, C_{t-1}, and the new cell state C_t, is evaluated by using the formula [27].

$$C_1 = C_{t-1} \otimes f_t \oplus i_t \otimes \widetilde{C_t} \tag{6}$$

Where C_{t-1} represents the internal memory cell state and $\otimes, \oplus$ represents the elementwise division and addition operators. The final decision regarding the output will be generated. Although the filtered output will be produced on the cell status. An element of the cell state will be shaped as output in this stage is decided by the output gate (o_t). After passing through the tanh layer (which forces the values to be between -1 and 1), the cell state is multiplied through the output gate and is calculated using the following mathematical formulas given by [27].

$$(o_t) = \sigma\left(X_t U^o + S_{t-1} W^o + b_o\right) \tag{7}$$

$$S_t = o_t \otimes \tanh\left(C_t\right) \tag{8}$$

The above (3) – (8) formulas signified the following terms. Where U^f, U^i, U^c, and U^o represents the input weights, where W^f, W^i, W^c, and W^o represents the weight matrix, where b_f, b_i, b_c, and b_o represents the bias vectors, and σ refers to the sigmoid function.

Moreover, the capacity of LSTM and RNN to manage and learn from sequential data is what unites them. Basic sequential data tasks can be performed by RNN. The LSTM is capable of more complex sequential data openings. In this research, we suggest that all LSTM-RNN architecture can adapt to learning the complexity and nonlinearity of time-series data. The present model has several LSTM layers with numerous cells in each layer, which is a logical expansion of the basic LSTM model. This analysis showed an effective way to train the forecasting model by making better use of each LSTM layer's parameters. Each function of the LSTM layer processes a separate portion of the intended task and then passes it to the subsequent layer, processing the remainder of the task until the final layer creates the output.

3.4.3 ARIMA

An auto-regression (AR) model examines the dependency between observation and a positive number of lag comments. The process of differencing raw observations is rendering a time series stationary, removing an observation from a preceding time step. The performance evaluation is compared with the statistical model library to create the ARIMA program [28].

As shown in FIGURE 7, the ARIMA model functioned as the moving average residual error to protect the observations to control the relationship between observation and the error. The ARIMA model exactly specifies each of these elements as a parameter. It states the quantity of Y, delays to be functional as the predictor. The data is equipped with a degree of differencing to make it stationary, i.e., to remove trend and seasonal structures that unfavorably affect the regression, and linear regression models with the set of variables.

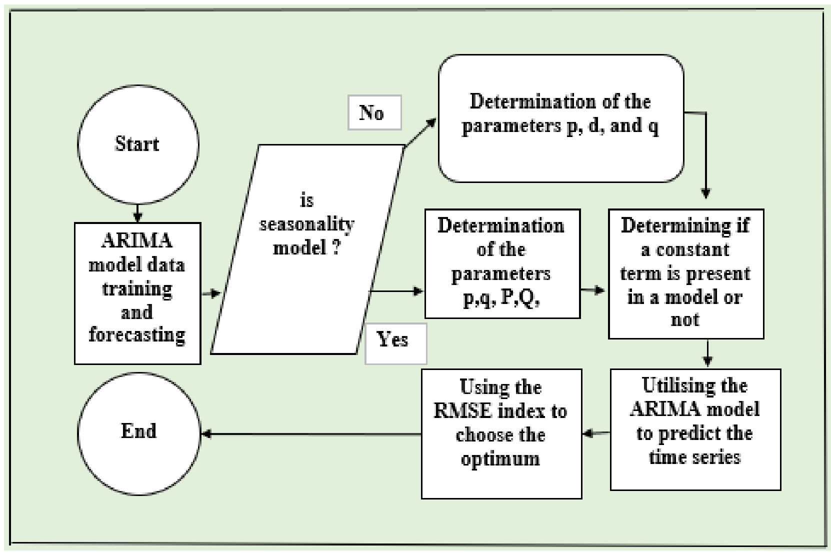

Figure 7: The ARIMA model architecture [29]

A parameter can have a value of 0, which signifies the model should not use a specific element. The ARIMA model can be set up to serve as an autoregressive moving average (ARMA) model or even a straightforward AR, I, or MA model.

The standard notation (p,d,q) is used rapidly to classify the ARIMA model in which the parameters are replaced by integer values. The typical representation of the p autoregressive model using the equation given by [30].

$$yt = c + \phi 1yt - 1 + \phi 2yt - 2 + \ldots + \phi pyt - p + \varepsilon_t, \tag{9}$$

where ε_t stands for background noise and yt depends entirely on its historical value ($yt - 1$, $yt - 2$, etc.) is a pure AR model.

A common depiction of a d in the model stands in for the I in ARIMA, which stands for Integrated. A non-seasonal ARIMA model is created when time series are different to make the stationary and, mixed with AR and MA models using the following equation by [30].

$$y't = c + \phi 1\mathrm{y}'\mathrm{t} - 1 + \ldots + \phi \mathrm{py}'\mathrm{t} - \mathrm{p} + \theta 1\varepsilon_t - 1 + \ldots + \theta q\varepsilon_t - q + \varepsilon_t, \tag{10}$$

where $y't$ denotes a series of differences made more than one, also the equation represented insulated values for yt from the AR model and lagged errors from the MA model. The order differencing (d) in the ARIMA model takes a value of 1 for data progress stationary.

However, q in the model is the Moving Average (MA) in the ARIMA equation. MA model depends on preceding forecast errors and reverses to the AR model. A general method for generating a moving average of q is using the following equation given by [30].

$$yt = c + \varepsilon_t + \theta 1\varepsilon_t - 1 + \theta 2\varepsilon_t - 2 + \ldots + \theta q\varepsilon_t - q \tag{11}$$

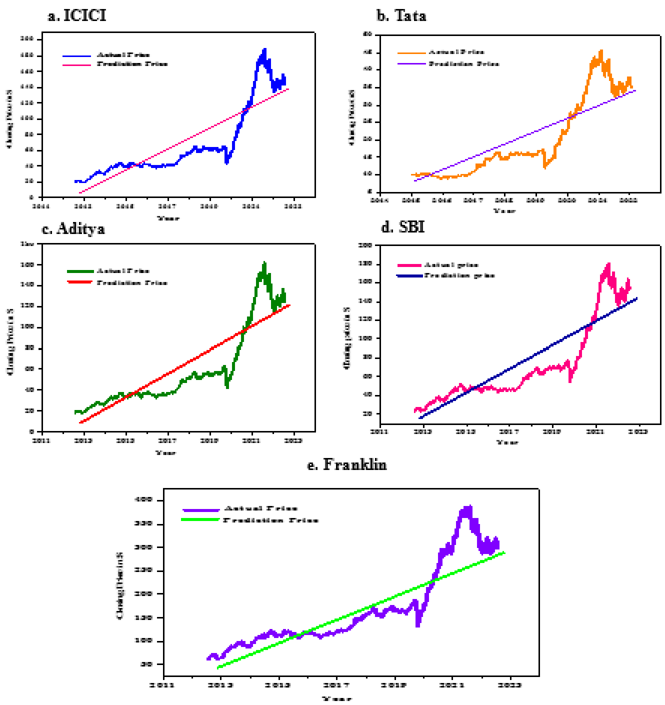

Figure 8: a. Comparison of actual and predicted values of proposed linear regression model during testing for 503 days ahead prediction using ICICI prudential technology fund direct plan growth. b. Comparison of actual and predicted values of linear regression model during testing for 357 days ahead prediction using Tata Digital India funds direct growth. c. Comparison of actual and predicted values of linear regression model during testing for 503 days ahead prediction using Aditya Birla Sun Life Digital India fund direct growth plan. d. Comparison of predicted and actual values of linear regression model during testing for prediction ahead 502 days using SBI technology opportunities fund direct growth. e. Comparison of actual and predicted values of linear regression model during testing for 504 days ahead prediction using Franklin India technology fund direct growth.

where ε_t represents white noise (spectral noise amplitude, η, for gravity measurements) and forecast errors in the future. The q is normally recognized as the moving average window size method as described by [31]. Subsequently, equations (9) – (11) explained the ARIMA process of the observations. However, assumptions are authenticated in the raw observations and outstanding predicting errors.

3.5 Training of the Proposed Prediction Metrics

For the simulation purpose, 80% of the feature datasets are randomly used for training of the models. The anticipated model of the current study fits the data as accurately as possible to assess the machine learning techniques. The linear regression, deep LSTM-recurrent neural network, and ARIMA are significant for analysis which fits the data by utilizing the training data as the test data set. The root-mean-square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and mean square error (MSE) are used to evaluate the errors in NAV forecasting. In the discussion part, error metrics were used to evaluate the predictive correctness of the current study. These indexes resulted precisely as expected and actual values were well coordinated. The training values are chosen according to the required number of days ahead for prediction. The learning principles specified in equations (1) and (2) state the linear regression model, and formulas (3) – (8) are used to train the deep LSTM model, and equations (9) – (11) are similarly used to train the ARIMA model. Each training pattern is systematically applied to the respective models, and the related error values are renowned as performance metrics. Moreover, each set of patterns yields the mean squared error (MSE). The different evaluations and prediction error is measured by the equations (12) – (15). The performance evaluation metrics are used to forecast in the present study by using scale-dependent and percentage errors.

3.5.1 Scale-Dependent Errors

In [32], the authors stated that the scale errors are the same as that of the raw data. Due to this boundary, it is not possible to compare series that are on different scales using accuracy measures that have uniquely built on these errors. Therefore, the mean absolute error (MAE), the mean squared error (MSE), and the root mean square error (RMSE) can be expressed by scale-dependent measures as given in equations (12) – (14) [33].

The mean absolute error (MAE) score is measured as the average of the absolute error values to confirm the following equation,

$$\frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y_i}| \tag{12}$$

The mean squared error (MSE) corresponds to the prediction error per square calculated by using the following equation,

$$\frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2 \tag{13}$$

The root-mean-squared error (RMSE) is regarded as an excellent all-purpose error metric for numerical forecasts by using the below equation,

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2} \tag{14}$$

The above equations (12) – (14) indicate, where $\mathrm{y_i}$ represent the actual closing selling price, $\widehat{\mathrm{y_i}}$ represent the prediction price, and $\frac{1}{\mathrm{N}}$ represent the total number of datasets utilized in the current study.

3.5.2 Percentage Errors

Percentage errors used to forecast the performance across multiple scaled datasets on the strength of scale independence. In addition, most of the statistics determine the Root Mean Square Percentage Error (RMSPE) in a triplicate manner to achieve accuracy using the following mathematical equation given by [33].

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \widehat{y_i}}{\widehat{y_i}}\right]^2 \times 100} \tag{15}$$

Where y_i represents the actual value, $\widehat{y_i}$ represents the closing value, and $\frac{1}{N}$ represents the total number of datasets used by the models.

While comparing the target values of the time series and associated expectations, we overcame several cost issues found in the regression machine learning practices. Although the results of applying the two metrics have different computed values, each metric's significance in terms of gauging the efficacy of the prediction models is identical. Notably, the production data often show distinct scales for the best price prediction. It is preferable to rely on RMSPE, or any other percentage error metrics, to determine the relative error between multiple models for the NAV prediction.

3.6 Evaluation of the Proposed Prediction Metrics

The residual 20% of the feature data, is used to assess the prediction performance of the model following the training phase. Comparing the training data with the actual (observed) data to determine the testing values of NAV predictions is the main objective of the performance evaluation. In [34], the authors stated that regression machine-learning techniques are extensively used by big data businesses to produce projections for a range of industries. The mean absolute percentage error (MAPE) and root of the mean squared error (RMSE) are considered for assessing the prediction accuracy.

3.7 Return Volatility and Performance Evaluation Metrics

In addition to the return volatility and performance evaluation metrics, we adopt the Sharpe ratio, Sortino ratio, Modigliani ratio, Max drawdown, and Calmar ratio to extensively evaluate the risk-return performance of the extreme deep learning model. The difference in closing prices from one day to the next as well as the fund's return from the previous day are used to compute return volatility. In this case, we considered 252 trading days for calculating the annualized return volatility. The specific evaluation indicators are as follows:

(1) ***Sharpe ratio.*** The Sharpe ratio which was introduced first in 1966 by Nobel laureate William F. Sharpe is a measure for calculating risk-adjusted return. The share ratio is the average return earned over the risk-free rate per unit of volatility. The formula is as follows:

$$S_i = \frac{E\left(r_i\right) - r_f}{\sigma_i} \tag{16}$$

Where $E\left(r_i\right)$ represents the expected annual return on investment of the fund during the investigation period, r_f is the annual risk-free return, σ_i annual volatility of investment of fund returns.

(2) ***Sortino ratio.*** The Sortino ratio is very similar to the Sharpe ratio. The only difference is that where the Sharpe ratio uses all the observations for calculating the standard deviation the Sortino ratio only considers the harmful variance.

(3) ***Modigliani (M2) ratio.*** The Modigliani ratio is similarly identified as the M2 ratio or Modigliani-Modigliani measure. The Modigliani ratio measures the returns of the portfolio, adjusted for the risk of the portfolio compared to that of some benchmarks. The formula is

$$M2 = SR * \sigma_{benchmark} + (rf) \tag{17}$$

Where SR represents the Sharpe ratio of the fund, $\sigma_{benchmark}$ is the standard deviation of the benchmark return, (rf) is the annual risk-free return of the investment.

(4) ***Max drawdown.*** The Max drawdown quantifies the steepest decline from peak to through observed for an investment. This is useful as an indicator of downside risk and volatility over a specific time. The main reason it does not rely on the underlying returns being normally distributed. The formula is as follows:

$$MDD = \frac{\max\left(p_i - p_j\right)}{p_i} \tag{18}$$

Where p_i and p_j represents the net value of the investment fund on a particular day and $\max\left(p_i - p_j\right)$represents the net value of the utmost decline.

(5) ***Calmar ratio.*** The Calmar ratio is used to analyze the performance of mutual fund investment. It demonstrates the amount of risk obligatory to realize the investment returns. It uses max drawdown in the denominator as opposed to standard deviation.

$$Calmar = R_p - R_f / Max\ D \tag{19}$$

Where R_p represents the portfolio return, R_f is the risk-free rate, $Max\ D$ is the Maximum drawdown value.

3.8 Pearson Correlations

Pearson correlation is a statistical indicator, usually used to measure the strength of correlation between two variables. We used to analyze the correlation between financial stock and the proposed model. The value of -1 and 1 indicates the meaning of total negative and positive linear correlation. The value of 0 specifies the zero correlation. The correlation between the two variables is given by equation (1):

$$r_{xy} = CovXY / \sigma_x * \sigma_y \tag{20}$$

Where $CovXY$ is the covariance between X and Y, σ_x is the standard deviation of X, and $\sigma_y$is the standard deviation of Y. Furthermore, the machine learning models are used to determine the correlation between the financial data.

4. RESULTS AND DISCUSSION

The results of the study show the compared prediction methods in terms of predictive accuracy and investment performance achieved with the machine-based trading strategy. The current study is contrasted with those of similar studies using linear and non-linear models. Simulation results of all three models *viz.* linear regression, long and short-term memory neural network, and autoregressive moving average predicting the accuracy level of the net asset value. First, we exhibit the strategized prediction accuracy of all three machine learning models. We then compare the results of MAE, MAPE, MSE, and RMSE error metrics for evaluating the model performance. We also focus on each financial spectrum's annualized log return volatility and over time volatility for analyzing the historical risk-return strategy. Additionally, we evaluate the different portfolio indicators such as the Sharpe ratio, Sortino ratio, Modigliani ratio, Max drawdown, and Calmar ratio to extensively evaluate the risk-return performance of the AI-influenced deep neural network model. For predictive accuracy, we then take a more granular look at the results for each model's performance over time with statistical metrics. Furthermore, we calculate the paired t-test to measure the hypothesis observed difference between each model and also conduct the statistical t-test to compare the predictive price fluctuation returns among the models. Finally, we shed light on the overall prediction performance of one individual model on the aggregate machine learning models.

4.1 Model Accuracy

An analysis of the various simulation findings produced in the previous part is provided in this section. Although simulation results for all NAVs, models, and entire estimates have been produced, only typical scenarios are shown here due to space restrictions. TABLE 6 summarizes the results of forecasting. As seen in TABLE 6, the mean absolute percent error (MAPE) deep LSTM neural network performed 99.87% better than the linear regression model in NAV forecasting and 99.88% better than the sub-linear ARIMA model. The MAPE between actual and forecasted price is 0.02 for the LSTM neural network model, 15.02 for the linear regression model, and 16.78 for the sub-linear ARIMA model. The LSTM neural network model outperformed the linear regression model on 473 of the time series (both models resulted in the same forecast for one technology mutual fund) and the sublinear ARIMA model on 474 of the time series of the 509 average series analyzed.

Table 6: Results summary of linear regression, ARIMA, and deep LSTM neural network models for technology sector mutual fund NAV prediction.

Performance metric	**Linear regression**	**Deep LSTM neural network**	**ARIMA**
Root mean squared error [RMSE]	21.296	2.418	16.558
Mean absolute error [MAE]	17.317	3.760	13.756
Mean absolute percent error [MAPE]	**15.02**	**0.02**	**16.78**
Mean squared error [MSE]	320.53	3.76	22.27

Note: The bold values indicate all three models' MAPE predictions.

The values of various performance measures for five different technology sector NAVs are listed and compared in Tables 7 and 8. As shown in the MAE performance comparison in Table 7, the ICICI, deep LSTM neural network performed 99.48% better than the linear regression model in NAV forecasting and 89.16% better than the ARIMA model. In Tata, the deep LSTM neural network performed 74% better than the linear regression model in NAV forecasting and 84.78% better than the ARIMA model. In Aditya, the deep LSTM neural network performed 75% better than the linear regression model in NAV forecasting and 80.37% better than the ARIMA model. In SBI, the deep LSTM neural network performed 63.37% better than the linear regression model in NAV forecasting and 70.41% better than the ARIMA model.

Table 7: Comparison of MSE and MAE for NAV prediction using three different models.

Models	Metrics	ICICI	Tata	Aditya	SBI	Franklin
Linear regression	MSE	490.6800	30.03243	337.97430	347.85996	396.10871
	MAE	18.49	3.49	15.51	17.17	31.92
Deep LSTM-RNN	MSE	0.00069	0.00073	0.00075	0.00125	0.00126
	MAE	**2.55**	**0.91**	**3.82**	**6.29**	**5.23**
ARIMA	MSE	0.00015	0.00017	0.00015	0.00013	0.00012
	MAE	23.52	5.98	19.46	21.26	41.11

Note: The bold values represent the best MAE for predicting the deep LSTM-RNN model accuracy.

Table 8: Comparison of RMSE and MAPE for NAV prediction using three different models.

Models	Metrics	ICICI	Tata	Aditya	SBI	Franklin
Linear regression	RMSE	22.98	4.84	18.86	20.87	38.94
	MAPE	23.52	4.13	15.10	16.18	16.18
Deep LSTM-RNN	RMSE	0.42	0.36	3.59	6.14	1.58
	MAPE	**0.02**	**0.02**	**0.02**	**0.03**	**0.03**
ARIMA	RMSE	30.91	6.93	25.12	28.49	46.81
	MAPE	17.73	5.98	19.49	21.26	19.46

Note: The bold values represent the best MAPE for predicting the deep LSTM-RNN model accuracy.

Finally, in Franklin, the deep LSTM neural network performed 83.62% better than the linear regression model in NAV forecasting and 87.28% better than the ARIMA model. Hence, the deep LSTM neural network model is a better candidate for the prediction of NAV compared to the other two models.

As the MAPE performance comparison of Table 8, the ICICI, deep LSTM neural network performed 99.91% better than the linear regression model in NAV forecasting and 99.88% better than the ARIMA model. In Tata, the deep LSTM neural network performed 99.52% better than the linear regression model in NAV forecasting and 99.67% better than the ARIMA model. In Aditya, the deep LSTM neural network performed 99.87% better than the linear regression model in NAV forecasting and 99.89% better than the ARIMA model. In SBI, the deep LSTM neural network performed 99.81% better than the linear regression model in NAV forecasting and 99.86% better than the ARIMA model. Finally, in Franklin, the deep LSTM neural network performed 99.81% better than the linear regression model in NAV forecasting and 99.85% better than the ARIMA model. Hence, the deep LSTM neural network model produced an excellent NAV prediction accuracy compared to the other two models. The exhaustive simulation results of the proposed linear regression prediction model for NAV prediction of various Indian technology-wise mutual funds during the testing era are shown in FIGURE 8(a) - (e). From the comparative results presented in the Figures, it is evident that the actual and predicted times series are skeptical. The exhaustive simulation results of the proposed deep LSTM neural network model for NAV prediction of various Indian technology-wise mutual funds during the testing era are shown in FIGURE 9(a) - (e). From the comparative results presented in the figures, it is evident that the actual and predicted time series precisely overlap, which demonstrates the potentiality of the deep neural network model.

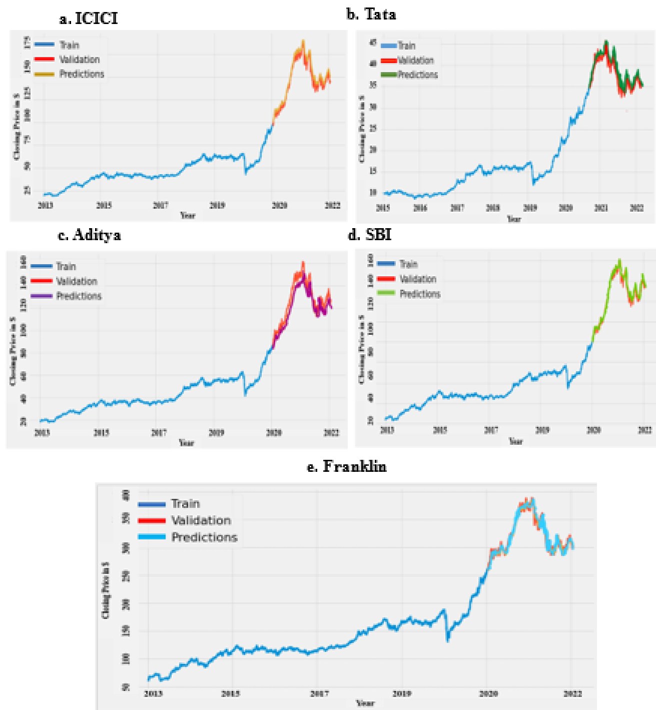

Figure 9: a. Comparison of actual and predicted values of proposed deep LSTM neural network model during testing for 563 days ahead prediction using ICICI prudential technology fund direct plan growth. b. Comparison of actual and predicted values of proposed deep LSTM neural network model during testing for 357 days ahead prediction using Tata digital India funds direct growth. c. Comparison of actual and predicted values of proposed deep LSTM neural network model during testing for 503 days ahead prediction using Aditya Birla Sun Life Digital India fund growth direct plan. d. Comparison of actual and predicted values of proposed deep LSTM neural network model during testing for 562 days ahead prediction using SBI technology opportunities fund direct growth. e. Comparison of actual and predicted values of proposed deep LSTM neural network model during testing for 563 days ahead prediction using Franklin India technology fund direct growth.

The exhaustive simulation results of the proposed sub-linear ARIMA model for NAV prediction of various Indian technology-wise mutual funds during the testing era are shown in FIGURE 10(a) - (e). From the comparative results presented in the figures, it is evident that the actual and predicted time series are modest. Hence, compared to the linear and sub-linear models, the deep LSTM neural network is highly powerful for predicting the large series of NAV closing prices.

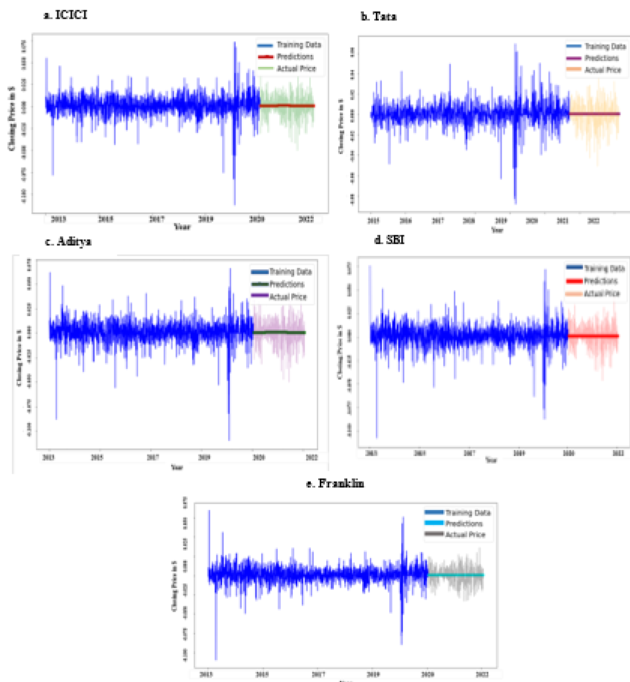

Figure 10: a. Comparison of actual and predicted values of proposed sub-linear ARIMA model during testing for 504 days ahead prediction using ICICI prudential technology fund direct plan growth. b. Comparison of actual and predicted values of proposed ARIMA model during testing for 357 days ahead prediction using TATA digital India funds direct growth. c. Comparison of actual and predicted values of proposed ARIMA model during testing for 504 days ahead prediction using Aditya Birla Sun Life Digital India fund growth direct plan. d. Comparison of actual and predicted values of proposed ARIMA model during testing for 503 days ahead prediction using SBI technology opportunities fund direct growth. e. Comparison of actual and predicted values of proposed ARIMA model during testing for 504 days ahead prediction using Franklin India technology fund direct growth.

Subsequently, the comparisons of various actual and predictive NAV daily performance measures for five different technology funds on the last day of December and June are listed in Tables 9-11. In the linear model, the main discrepancy between the actual and predictable price is shown in Table 9. The depicted linear model in FIGURE 8 reflects the main difference from Table 9. This is connected to squared errors having a greater effect on the outcome and the NAV closing price has a fundamentally non-linear landscape.

In [20], the authors specified a linear regression model's significant flaws are non-linearity datasets, resulting in inaccurate forecasts. Although linear regression is an excellent tool for examining the relationships between variables, it is not a complete description of relationships among other variables. Hence, the author directed the paper's audience to switch to the other two models for the NAV closing price prediction. Similar to the first model, the deep LSTM neural network model is used for NAV prediction. The actual and predicted price comparison is described in Table 10. In

Table 9: The experimental results of actual and prediction price using the Linear regression model.

Name of the scheme	Act. Price vs Pred. Price	June 28, 2013	Dec. 31, 2013	June 30, 2014	Dec. 31, 2014	June 30, 2015	Dec. 31, 2015	June 30, 2016	Dec. 30, 2016	June 30, 2017	Dec. 29, 2017	June 29, 2018	Dec. 31, 2018	June 28, 2019	Dec. 31, 2019	June 30, 2020	Dec. 31, 2020	June 30, 2021	Dec. 31, 2021	June 30, 2022	Dec. 31, 2022
ICICI	Actual Price	20.26	31.68	33.64	40.26	39.21	42.26	42.76	40.92	41.05	49.37	57.78	59.4	61.59	61.29	58.71	105.23	138.65	187.06	139.49	145.38
	Prediction Price	8.25	14.63	20.89	27.06	33.33	39.69	46.02	52.34	58.66	65.03	71.40	77.62	83.89	90.15	99.29	105.88	112.14	118.57	124.89	131.26
Tata	Actual Price				Not Commenced				9.57	9.71	11.65	14.67	14.79	16.10	16.18	15.07	25.47	33.49	45.16	33.79	35.23
	Prediction Price								7.97	10.24	12.52	14.81	17.04	19.29	21.53	24.37	27.17	29.42	31.72	33.99	36.27
Aditya	Actual Price	19.37	27.29	29.31	33.37	33.5	37.37	37.91	36.31	36.65	44.76	51.5	52.27	55.17	57.8	57.36	92.84	122.65	160.18	119.75	127.16
	Prediction Price	8.34	13.87	19.32	24.67	30.11	35.64	41.13	46.62	52.10	57.63	63.17	68.56	74.01	79.45	87.39	93.09	98.54	104.12	109.60	115.14
SBI	Actual Price	25.11	35.70	38.34	46.45	45.41	48.11	47.39	46.85	45.80	53.26	64.63	63.35	69.87	71.70	70.28	106.68	138.53	179.73	141.22	153.74
	Prediction Price	15.32	21.41	27.41	33.31	39.31	45.41	51.45	57.50	63.55	69.64	75.74	81.69	87.69	93.69	102.39	108.73	114.73	120.88	126.92	133.02
Franklin	Actual Price	66.58	94.45	96.54	110.78	111.99	115.67	117.66	113.38	120.25	135.80	158.64	152.93	166.39	173.05	180.47	273.89	323.22	384.18	293.09	301.50
	Prediction Price	50.05	63.13	75.99	88.65	101.52	114.59	127.57	140.54	153.51	166.59	179.67	192.43	205.29	218.16	236.94	250.44	263.31	276.49	289.46	302.54

Source: Linear regression model result interpretation

Table 10: The experimental results of actual and prediction price using deep LSTM neural network model.

Name of the scheme	Act. Price vs Pred. Price	June 28, 2013	Dec. 31, 2013	June 30, 2014	Dec. 31, 2014	June 30, 2015	Dec. 31, 2015	June 30, 2016	Dec. 30, 2016	June 30, 2017	Dec. 29, 2017	June 29, 2018	Dec. 31, 2018	June 28, 2019	Dec. 31, 2019	June 30, 2020	Dec. 31, 2020	June 30, 2021	Dec. 31, 2021	June 30, 2022	Dec. 31, 2022
ICICI	Actual Price	20.26	31.68	33.64	40.26	39.21	42.26	42.76	40.92	41.05	49.37	57.78	59.4	61.59	61.29	58.71	105.23	138.65	187.06	139.49	145.38
	Prediction Price				Trained Datasets							Trained Datasets					103.18	136.89	183.02	140.43	146.45
Tata	Actual Price				Not Commenced				9.57	9.71	11.65	14.67	14.79	16.10	16.18	15.07	25.47	33.49	45.16	33.79	35.23
	Prediction Price											Trained Datasets					26.24	34.56	43.88	34.16	35.71
Aditya	Actual Price	19.37	27.29	29.31	33.37	33.5	37.37	37.91	36.31	36.65	44.76	51.5	52.27	55.17	57.8	57.36	92.84	122.65	160.18	119.75	127.16
	Prediction Price				Trained Datasets							Trained Datasets					88.92	117.32	150.17	115.91	122.97
SBI	Actual Price	25.11	35.70	38.34	46.45	45.41	48.11	47.38	46.85	45.80	53.26	64.63	63.35	69.87	71.70	69.01	106.68	138.53	179.73	141.22	153.74
	Prediction Price				Trained Datasets							Trained Datasets					99.85	128.73	166.07	133.60	156.41
Franklin	Actual Price	66.57	93.68	96.54	110.78	111.99	115.67	117.66	113.38	120.25	135.80	158.64	152.93	166.39	173.05	180.47	273.89	323.22	384.18	293.09	301.50
	Prediction Price				Trained Datasets							Trained Datasets					270.61	319.19	376.08	296.55	301.69

Source: Deep LSTM model result interpretation

Table 11: The experimental results of actual and prediction price using the ARIMA model.

Name of the scheme	Act. Price vs Pred. Price	June 28, 2013	Dec. 31, 2013	June 30, 2014	Dec. 31, 2014	June 30, 2015	Dec. 31, 2015	June 30, 2016	Dec. 30, 2016	June 30, 2017	Dec. 29, 2017	June 29, 2018	Dec. 31, 2018	June 28, 2019	Dec. 31, 2019	June 30, 2020	Dec. 31, 2020	June 30, 2021	Dec. 31, 2021	June 30, 2022	Dec. 31, 2022
ICICI	Actual Price	20.26	31.68	33.64	40.26	39.21	42.26	42.76	40.92	41.05	49.37	57.78	59.15	61.59	61.29	58.71	105.23	138.85	187.06	139.49	145.38
	Prediction Price				Trained Datasets							Trained Datasets					99.56	110.55	124.69	139.36	154.14
Tata	Actual Price				Not Commenced					9.71	11.66	14.67	14.79	16.10	16.18	15.07	25.4	33.49	45.16	33.79	35.23
	Prediction Price				Trained Datasets							Trained Datasets						33.50	38.79	43.02	47.05
Aditya	Actual Price	19.37	27.29	29.31	33.37	33.5	37.37	37.91	36.31	36.65	44.76	51.5	52.27	55.17	57.8	57.36	92.84	122.65	159.58	115.75	127.16
	Prediction Price				Trained Datasets							Trained Datasets					88.18	97.66	109.56	122.01	134.49
SBI	Actual Price	25.11	35.70	38.34	46.45	45.41	48.11	47.38	46.85	45.80	53.26	64.63	63.35	69.87	71.70	69.01	110.25	138.53	179.73	141.22	153.74
	Prediction Price				Trained Datasets							Trained Datasets					107.29	117.81	131.79	145.64	159.95
Franklin	Actual Price	66.58	94.45	96.54	110.78	111.99	115.67	117.66	113.38	120.25	135.80	158.64	152.93	166.39	173.05	180.47	273.89	323.22	384.18	293.09	301.50
	Prediction Price				Trained Datasets							Trained Datasets					265.32	290.78	321.25	351.17	380.49

Source: ARIMA model result interpretation
Note: Act. = actual
Pred. = prediction

the same table, a 1 or 0.5 value difference is found between the actual price and the forecasted price. FIGURE 9 shows that 20% of the testing value has been parallel to the actual pricing. The final method used in the present study is the ARIMA model. Table 11, explains readers an understanding of actual and expected closing prices by providing the outcomes. FIGURE 10 directs the p, d, and q order which is made clear in the data description paragraph. Training and testing values have been distinguished in this plot. Although the dataset is stable, ARIMA still performs better in the minimum size of actual NAV. By comparison of all three models, the deep LSTM neural network model produced an excellent performance for predicting the NAV.

4.2 Prediction Bias

Bias forecast accuracy is the logical deviation between the actual values and the predicted values. To calculate the forecast accuracy, we divided the absolute error by the actual price. Tables 12 and 13 present the daily and monthly individual datasets testing samples of all three models MAPE which is less than 5% in the deep LSTM neural network model. The difference between the actual testing sample and all three models MAPE is taken into consideration for the prediction accuracy. In comparison, the other two linear and ARIMA model's MAPE values are more than 5%. Therefore, the daily forecast accuracy of MAPE in Table 12, the deep LSTM neural network performed roughly 90% better than the linear and ARIMA models in ICICI dataset NAV forecasting. In Tata, the LSTM neural network is 80% and 85% better than the linear and ARIMA model. In Aditya, the LSTM neural network performed roughly 80% better than the linear and ARIMA models. In SBI, the LSTM neural network is 70% and 60% better than the linear and ARIMA models. In the end, the deep LSTM neural network performed 88% better than the linear and ARIMA models in Franklin dataset NAV forecasting. Hence, the LSTM model of MAPE daily average prediction accuracy is considered 81% as an indication that the forecast is highly standard and accurate.

Table 12: The daily testing sample prediction for all three models.

Sl. No.	**Name of the scheme**	**Testing period**	**Total daily testing sample**	**Metrics**	**Model 1 Linear**	**Model 2 LSTM**	**Model 3 ARIMA**
1	ICICI	2020-12-18 to 2022-12-30	503	MAE MAPE	28.687 17.99%	**2.547** **1.74%**	23.522 14.84%
2	Tata	2021-07-20 to 2022-12-30	357	MAE MAPE	5.599 13.72%	**0.913** **2.36%**	5.960 16.23%
3	Aditya	2020-12-18 to 2022-12-30	503	MAE MAPE	23.849 17.41%	**3.824** **2.95%**	19.457 14.30%
4	SBI	2020-12-17 to 2022-12-29	502	MAE MAPE	26.049 16.57%	**6.288** **4.35%**	17.197 10.89%
5	Franklin	2020-12-18 to 2022-12-30	503	MAE MAPE	47.582 13.96%	**5.229** **1.64%**	41.114 12.66%

Note: The bolded column indicates the best MAPE values for daily testing sample prediction.

Table 13: The monthly testing sample prediction for all three models.

Sl. No.	Name of the mutual fund	Monthly testing samples	Metrics	Model 1 Linear	Model 2 LSTM	Model 3 ARIMA
1	ICICI	2022-12-30; 2022-11-30; 2022-10-31; 2022-09-30; 2022-08-30; 2022-07-29; 2022-06-30; 2022-05-31; 2022-04-29; 2022-03-31; 2022-02-28; 2022-01-31; 2021-12-31; 2021-11-30; 2021-10-29; 2021-09-30; 2021-08-31; 2021-07-30	MAE	35.43565	**2.312**	26.58596
			MAPE	21.70749	**1.46%**	15.93396
2	Tata	2022-12-30; 2022-11-30; 2022-10-30; 2022-09-30; 2022-08-23; 2022-07-25; 2022-06-24; 2022-05-25; 2022-04-23; 2022-03-24; 2022-02-22; 2022-01-24; 2021-12-25; 2021-11-23; 2021-10-24; 2021-09-23; 2021-08-24; 2021-07-25	MAE	5.43048	**0.667**	5.776801
			MAPE	13.43887	**1.71%**	15.74613
3	Aditya	2022-12-30; 2022-11-30; 2022-10-31; 2022-09-30; 2022-08-30; 2022-07-29; 2022-06-30; 2022-05-31; 2022-04-29; 2022-03-31; 2022-02-28; 2022-01-31; 2021-12-31; 2021-11-30; 2021-10-29; 2021-09-30; 2021-08-31; 2021-07-30	MAE	28.74116	**4.165**	21.37107
			MAPE	20.44374	**3.03%**	14.92469
4	SBI	2022-12-30; 2022-11-30; 2022-10-31; 2022-09-30; 2022-08-23; 2022-07-25; 2022-06-24; 2022-05-25; 2022-04-23; 2022-03-24; 2022-02-22; 2022-01-24; 2021-12-25; 2021-11-23; 2021-10-24; 2021-09-23; 2021-08-24; 2021-07-23	MAE	32.74737	**6.691**	20.23797
			MAPE	20.40609	**4.3%**	12.41173
5	Franklin	2022-12-30; 2022-11-30; 2022-10-31; 2022-09-30; 2022-08-30; 2022-07-29; 2022-06-30; 2022-05-31; 2022-04-29; 2022-03-31; 2022-02-28; 2022-01-31; 2021-12-31; 2021-11-30; 2021-10-29; 2021-09-30; 2021-08-31; 2021-07-30	MAE	49.33863	**4.967**	48.90677
			MAPE	13.9683	**1.49%**	14.97638

Note: The bolded column indicates the best MAPE values for monthly testing sample prediction.

Considering the monthly forecast accuracy of MAPE in Table 13, the deep LSTM neural network performed roughly 90% better than the linear and ARIMA models in ICICI dataset NAV forecasting. In Tata, the LSTM neural network is 87% and 89% better than the linear and ARIMA models. In Aditya, the LSTM neural network performed 85% and 79% better than the linear and ARIMA models. In SBI, the LSTM neural network is 79% and 65% better than the linear and ARIMA models. In the end, the deep LSTM neural network performed roughly 90% better than the linear and ARIMA models in Franklin dataset NAV forecasting. Hence, the LSTM model of MAPE monthly average prediction accuracy is considered 84% as an indication that the forecast is highly superior and accurate. Therefore, the conclusion is that the MAPE obtained 84% by the deep LSTM neural network is a highly recommendable model than the linear and sub-linear ARIMA models used in the study.

4.3 Return Volatility

The return volatility based on the difference in closing prices between the current day and the day prior and the return of yesterday's fund were calculated in this study. FIGURE 11 presents the annualized return volatility to extrapolate data over the sequence of a year. The annualized volatility return using the deep learning neural network model to compare the risk of investments with annual return periods over the time horizons was computed. In this case, we considered 252 trading days for calculating the annualized volatility return. Compared with all five technology funds, Tata's annualized return volatility volume produced the maximum level of 19.1 The results predicted by the model are highly precious to the investors for selecting the particular stock. FIGURE 12 represents the return volatility over 10 years of each financial dataset on the technology sector mutual fund return on investment. We can find that there are intense differences in both volatility over the period and the annualized volume of Gaussian distributions. Table 14 displays the descriptive statistics of the volatility of technology funds.

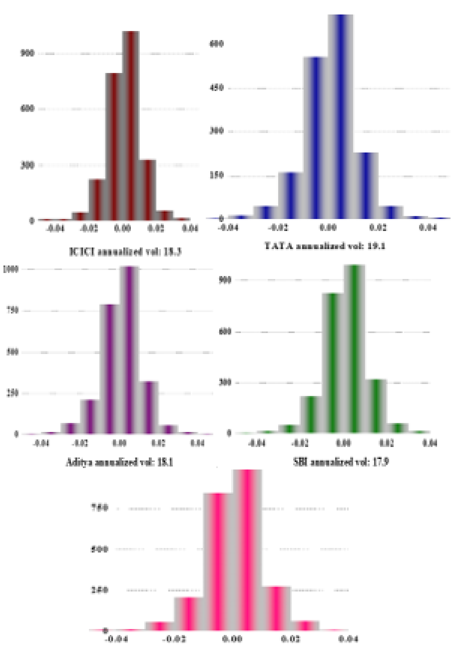

Figure 11: The frequency of log returns annualized volatility

From Figures 11, 12, and Table 14, it is evident that based on the Jarque-Bera statistics of the return of the five financial datasets, the assumption that the sample volatility over time follows the Gaussian distribution is rejected at the 0.01 level. Consequently, it is not appropriate to construct trading by using linear and sub-linear models.

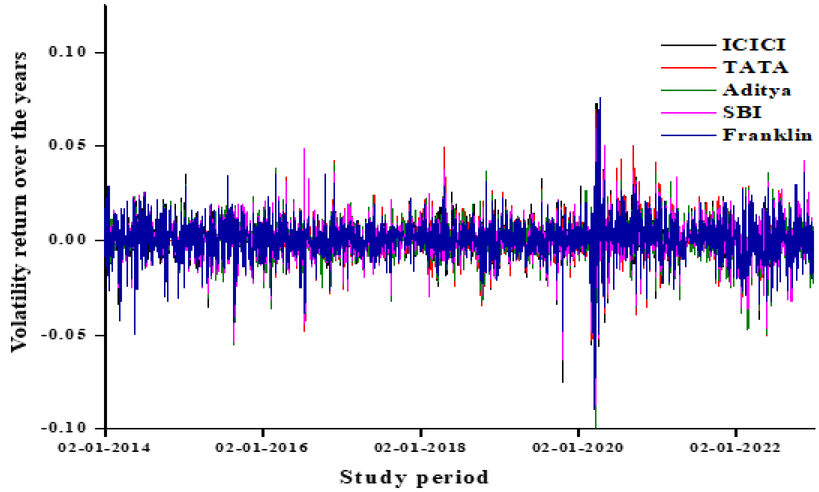

Figure 12: The volatility over time of different financial data series comparison

Table 14: Descriptive statistics of the volatility of IT-sector returns

Statistic item	ICICI	TATA	Aditya	SBI	Franklin
Mean	0.001	0.001	0.001	0.001	0.001
Max	0.076	0.072	0.069	0.074	0.088
Standard deviation	0.011	0.012	0.011	0.011	0.011
Skewness	-0.639	-0.435	-0.706	-0.651	-0.494
Kurtosis	9.455	6.895	7.648	7.949	11.234
J-B value ($\times 10^3$)	7.912	2.920	6.887	6.555	4.801
Sample size	2515	1783	2515	2509	2516

Note: The J-B value represents the Jarque-Bera test of normality.

Additionally, the AI-influenced deep learning neural network model gives a better performance in terms of Sharpe ratio, Max drawdown, and Calmar ratio. Table 15 shows the statistical summary of volatility return and performance evaluation metrics. In Table 14, an individual stock of ICICI prudential technology fund (0P0000XUZ6.BO) has a Sharpe ratio of 7.0193, an annual standard deviation of 0.1819, and an annual return of 22.27% is much highest of all portfolios. Usually, any Sharpe ratio is more than 3.0 is considered excellent by an investor to invest in the fund. Though its maximum drawdown is 0.3743, which is relatively high, it also partly reflects that the model can yield a higher return. The comparative advantages of a deep LSTM neural network with two outputs, exactly true output decrease the overfitting of the models and improve the hidden state performance.

As shown in Table 16, the five technology-wise mutual funds have low correlation and very good market representation specifically in the LSTM neural network model. The size of the correlation between the input and output variable of below 0.50 (-0.50) denoted the low positive (low negative) correlation of linear and ARIMA model forecasting. The LSTM neural network of correlation between input and output variable is 0.90 to 1.00 which signified a very high positive association.

Table 15: Descriptive statistics of volatility return and performance evaluation metrics.

Variable	Abbr.	ICICI	Tata	Aditya	SBI	Franklin
Risk free	R_f	1.5	1.5	1.5	1.5	1.5
Annual return	R_p	22.27%	19.70%	21.49%	20.91%	17.28%
Annual S. D	σ	0.1819	0.4208	0.1800	0.3641	0.1715
Sharpe ratio	S_i	7.0193	3.0967	7.1396	3.5454	7.7399
Max drawdown	MDD	0.0798	0.3743	0.0694	0.0788	0.0678

Source: Author's statistical calculations.

Table 16: Pearson correlation coefficient of IT sector mutual funds based on time series

Funds/Models	ICICI	Tata	Aditya	SBI	Franklin
Linear	0.44	-0.63	0.43	0.55	0.03
LSTM	0.99	0.96	0.99	0.99	0.98
ARIMA	0.40	-0.63	0.40	0.52	0.01

Source: Author's statistical calculations.

On the whole, Table 18 shows the last day's prediction performance comparisons received by all three models. Similarly, FIGURE 13 shows the LSTM neural network model is 90% relatively predicting the fund-wise NAV in terms of Franklin, Tata, ICICI, Aditya, and SBI respectively. The subsequent ARIMA model, tracked by the deep LSTM- RNN, is less accurate in terms of SBI, Aditya, ICICI, and Tata. Franklin's NAV predicted price is not coordinated with the actual price. Hence, ARIMA models are ineffective at long-term forecasts, comparable to other forecasting techniques. In terms of AI-influenced securities in the technology schemes, the current linear and ARIMA models are giving lesser predictions.

4.4 T-paired statistics

A statistical paired t-test is carried out to verify that the various random sample dataset's actual and predicted values are paired with the proposed three models. Out of each model, the deep LSTM neural network model is significantly different and superior to the other two linear and sublinear models. From the study, the equality in the mean is a null hypothesis. The difference between all three individual models' p-values is less than 0.05 or 5% which is only in the deep LSTM model. So, the deep LSTM neural network model is considered to be statistically different and superior when the null hypothesis is rejected. The remaining 95% or 0.95 are chosen to do the paired t-test. Hence, the values below the 5% for p-value and 0 for h display that it is not statistically possible to prove that the outputs of two compared algorithms are completely different from each model. Table 17 presents the calculated values of p and h values with random sample periods for different NAV datasets using paired t-tests. From the results, it is observed that the p-value is less than 0.05 and the h-value is 1 in Aditya (p-0.027 and h-1) and SBI (p-0.007 and h-1) technology mutual funds. The remaining two models such as ARIMA and linear financial dataset p-value comparisons are not less than 0.05. So, the h-value is 0 in ARIMA and Linear models.

Table 17: Calculated values of p and h values with random sample periods for different NAV datasets using paired t-test

S. No.	IT sector mutual fund scheme	Random sample period	ML models	h value	p-value
1	ICICI prudential technology fund direct plan growth	30/06/2021 31/12/2021 30/06/2022 31/12/2022	LR LSTM-RNN ARIMA	0 0 0	0.875 0.880 0.292
2	Tata Digital India funds direct growth	30/06/2021 31/12/2021 30/06/2022 31/12/2022	LR LSTM-RNN ARIMA	0 0 0	0.293 0.772 0.446
3	Aditya Birla Sun Life Digital India fund growth direct plan	30/06/2021 31/12/2021 30/06/2022 31/12/2022	LR **LSTM-RNN** ARIMA	0 **1** 0	0.091 **0.027** 0.306
4	SBI technology opportunities fund direct growth	30/06/2021 31/12/2021 30/06/2022 31/12/2022	LR **LSTM-RNN** ARIMA	0 **1** 0	0.060 **0.007** 0.337
5	Franklin India Technology Fund direct growth	30/06/2021 31/12/2021 30/06/2022 31/12/2022	LR LSTM-RNN ARIMA	0 0 0	0.190 0.461 0.782

Source: Author’s statistical calculations.

Table 18: Comparisons of the last one-day (31.12.2022) prediction performances of all three models.

Models	Category	ICICI	Tata	Aditya	SBI	Franklin
	Actual NAV	**145.38**	**35.23**	**127.16**	**153.74**	**301.50**
Linear regression	Prediction NAV	131.26	36.27	115.14	133.02	302.54
DLSTM-RNN	**Prediction NAV**	**146.45**	**35.71**	**122.97**	**156.41**	**301.69**
ARIMA	Prediction NAV	154.14	47.05	134.49	159.95	380.49

Note: In this table, for a clear comparison, the closest forecasted deep LSTM neural network and actual NAVs are highlighted with grey color.

As a result, the LSTM neural network model performs better than the other two machine learning models in terms of deep learning. By employing a deep learning model driven by artificial intelligence, investors can also profit from the discovery of market trends and patterns. The machine learning model’s main advantage is its ability to make investment judgments that are devoid of

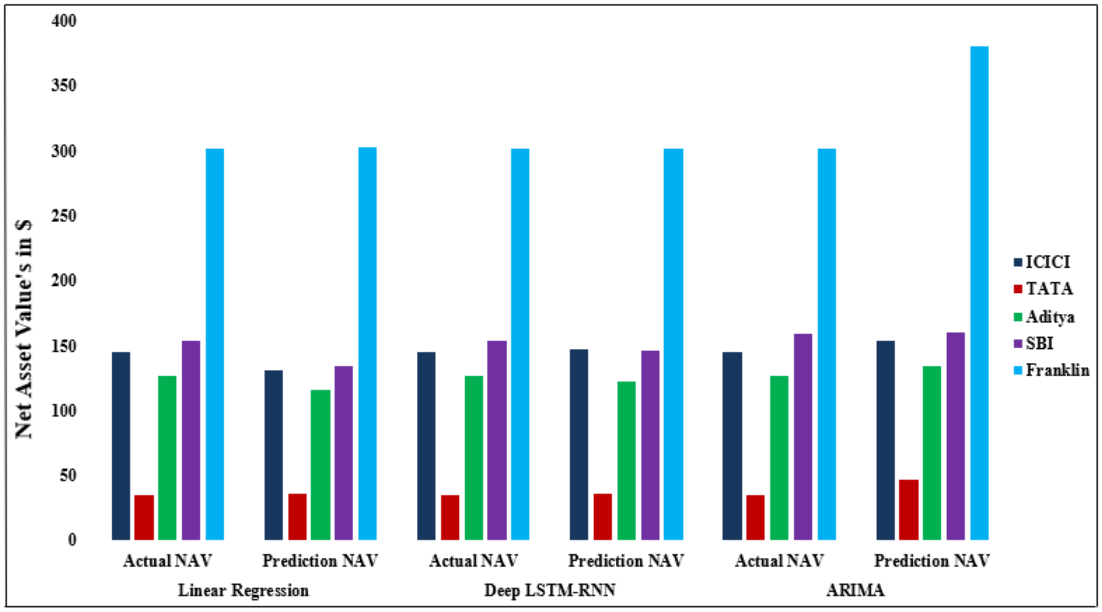

Figure 13: The **c**omparative analysis of the last one-day (31-12-2022) NAV prediction.

human emotional biases. Combining ML and DL models may also encourage the stock market to provide better investment warnings and financial guidance. Furthermore, deep learning techniques could reduce the possibility of losses and enhance investment strategies, enabling fund managers and investors to make prudent decisions regarding investments in the mutual fund sector. As a consequence, more accurate predictions will be provided by contemporary deep learning algorithms, which will improve investment decisions.

5. CONCLUSIONS AND FUTURE DIRECTIONS

The study model uses linear regression as the first model, a nonlinear deep LSTM neural network as a second model, and, the sublinear ARIMA model as a final model for predicting the daily NAV closing price (both independent and dependent). The study shows that the NAV's close price predictions are excellent, and changes in volatility are also better predicted by using the deep LSTM neural network model. The results of the study found the inability to predict NAV close price movement by using both linear and sublinear machine learning models. Our results suggest that the AI-influenced deep LSTM recurrent neural network model is statistically highly precious when comparing the performances of the other two models. This research focuses on the performances of artificial intelligence-influenced NAV close price predictions and return volatility measures which are statistically proven with a paired t-test, and Pearson correlation coefficient. From the results, the deep learning model observed that the p-value is less than 0.05 and the h-value is 1 in Aditya (p-0.027 and h-1) and SBI (p-0.007 and h-1) technology mutual funds. Hence, the LSTM deep learning neural network model is highly superior compared to the other two machine learning models i.e., Linear regression and ARIMA. Along with NAV forecasting, the study performs the volatility risk-return measures for analyzing the annual investment returns. In addition, different portfolio indicators like Sharpe, Max drawdown, M2, Sortino, and Calmar ratios are analyzed for evaluation purposes [16, 35]. The comparison should be made between the actual NAV's daily close price and the predicted price between the selected AI-influenced models and implied volatilities of return forecasted by the premier deep learning model. The research goal is to analyze the performance of an AI-influenced machine learning model for predicting the NAV closing price and volatility returns in securities trading. In comparison with machine learning and deep learning models, this study proved that the AI-influenced deep learning (deep LSTM-RNN) model showed excellent performance for predicting huge datasets i.e., more than 2500 for each financial spectrum [8]. We show that all employed models make predictions statistically, whereby the deep LSTM neural network MAPE daily and monthly average prediction accuracy is 81% & 84% superior to the linear and ARIMA models compared with all five financial datasets.

The study concludes with numerous implications for business forecasters, institutional investors, retail investors, funding organizations, academic researchers, and different practitioners they can use the deep learning network forecasting strategies and volatility returns and investments simultaneously to obtain the best performance of securities for making buy and sell investment decision according to the different market conditions. The sample evidence proves that daily volatilities are better predicted with a deep-learning neural network model. The research addresses the important machine learning and deep learning domain gaps, especially in the practical implementation of predictive accuracy and risk-return volatility measures for investment decision-making. There are still certain restrictions in place despite the greatest results and some advantages of the suggested deep learning and machine learning models. This research does not describe an in-depth analysis of the input selection features for volatility return forecasting. Future research works based on the deep learning model could expand this work by exploring deeper into historical price movements with risk-return measures, portfolio optimization, predicting financial instrument equity, bond price prediction, market trend analysis, and other decision-making domains. Furthermore, the deep learning LSTM neural network model used in this work may find application in other domains, such as agriculture production forecasting, risk insurance, marketing strategy, and the classification of large amounts of data for diverse tasks.

6. Conflict of interest:

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7. Acknowledgments

The authors would like to thank VIT, Vellore, for providing all the required facilities for this study.

References

[1] Galagedera DUA, Roshdi I, Fukuyama H, Zhu J. A New Network DEA Model for Mutual Fund Performance Appraisal: An Application to Us Equity Mutual Funds. Omega. 2018;77:168-

179.

[2] Bailey W, Kumar A, Ng D. Behavioral Biases of Mutual Fund Investors. J Financ Econ. 2011;102:1-27.

[3] Cushing D, Madhavan A. Stock Returns and Trading at the Close. J Financ Markets. 2000;3:45-67.

[4] Khan ZH, Alin TS, Hussain MA. Price Prediction of Share Market Using Artificial Neural Network (Ann). Int J Comput Appl. 2011;22:42-47.

[5] Lakshminarayanan SK, McCrae JP. A Comparative Study of Svm and Lstm Deep Learning Algorithms for Stock Market Prediction. AICS. 2019:446-457.

[6] Sen J, Mehtab S, Nath G. Stock Price Prediction Using Deep Learning Models. 2023. ATechRxiv Preprint: https://www.techrxiv.org/doi/full/10.36227/techrxiv.16640197.v1

[7] Zhou P, Li F. Prediction of Fund Net Value Based on Arima-Lstm Hybrid Model. 2021. Arxiv Preprint: https://arxiv.org/ftp/arxiv/papers/2111/2111.15355.pdf

[8] Koudjonou KM, Rout M. A Stateless Deep Learning Framework to Predict Net Asset Value. Neural Comput Appl. 2020;32:1-19.

[9] Liu Y. Novel Volatility Forecasting Using Deep Learning–Long Short-Term Memory Recurrent Neural Networks. Expert Syst Appl. 2019;132:99-109.

[10] Hu Y, Ni J, Wen L. A Hybrid Deep Learning Approach by Integrating Lstm-Ann Networks With the Garch Model for Copper Price Volatility Prediction. Phys A. 2020;557:124907.

[11] Murthi BPS, Choi YK, Desai P. Efficiency of Mutual Funds and Portfolio Performance Measurement: A Non-parametric Approach. Eur J Oper Res. 1997;98:408-418.

[12] Chung CY, Ryu D, Wang K, Zykaj BB. Optionable Stocks and Mutual Fund Performance. J Futures Markets. 2018;38:390-412.

[13] Rubanov D, Nnadi M. The Impact of International Financial Reporting Standards on Fund Performance. Acc Res J. 2018;31:102-120.

[14] Altay E, Satman MH. Stock Market Forecasting: Artificial Neural Network and Linear Regression Comparison in an Emerging Market. J Financ Manag Anal. 2005;18:18.

[15] Majumdar D, Bhattacharjee AK, Mukherjee S. Intelligent Investment Approaches for Mutual Funds: An Evolutionary Model. In: Machine learning applications in non-conventional machining processes. IGI Global; 2021:267-281.

[16] Chen J, Luo C, Pan L, Jia Y. Trading Strategy of a Structured Mutual Fund Based on Deep Learning Network. Expert Syst Appl. 2021;183:115390.

[17] Larose DT, Larose CD. Discovering Knowledge in Data: An Introduction to Data Mining. John Wiley & Sons, Inc. 2005.

[18] Long JS, Freese J. Regression Models for Categorical Dependent Variables Using Stata. Stata press. 2006.

[19] Maji G, Mondal D, Dey N, Debnath NC, Sen S, et al. Stock Prediction and Mutual Fund Portfolio Management Using Curve Fitting Techniques. J Ambient Intell Hum Comput. 2021;12:9521-9534.

[20] Narula A, Jha CB, Panda G. Development and Performance Evaluation of Three Novel Prediction Models for Mutual Fund Nav Prediction. Ann Res J Symbiosis Cent Manag Stud. 2015;3:227-238.

[21] Sherstinsky A. Fundamentals of Recurrent Neural Network (Rnn) and Long Short-Term Memory (Lstm) Network. Phys D Nonlinear Phenom. 2020;404:132306.

[22] Hüsken M, Stagge P. Recurrent Neural Networks for Time Series Classification. Neurocomputing. 2003;50:223-235.

[23] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Comput. 1997;9:1735-1780.

[24] Lin Y, Yan Y, Xu J, Liao Y, Ma F, et al. Forecasting Stock Index Price Using the Ceemdan-Lstm Model. North Am J Econ Fin. 2021;57:101421.

[25] Zhang W, Gong X, Wang C, Ye X. Predicting Stock Market Volatility Based on Textual Sentiment: A Nonlinear Analysis. J Forecasting. 2021;40:1479-1500.

[26] Li X, Tang P. Stock index prediction based on wavelet transform and FCD□MLGRU. J Forecasting. 2020;39:1229-1237.

[27] Sagheer A, Kotb M. Time Series Forecasting of Petroleum Production Using Deep Lstm Recurrent Networks. Neurocomputing. 2019;323:203-213.

[28] Seabold S, Perktold J. Stats Models: Econometric and Statistical Modeling With Python. Proceedings of the Python in Science Conference. Proceedings of the 9th python in science conference. 2010:92-96.

[29] Li Z, Han J, Song Y. On the Forecasting of High□Frequency Financial Time Series Based on Arima Model Improved by Deep Learning. J Forecasting. 2020;39:1081-1097.

[30] Harrop JW, Velicer WF. A Comparison of Alternative Approaches to the Analysis of Interrupted Time Series. Multivariate Behav Res. 1985;20:27-44.

[31] Cryer JD, Chan KS. Time Series Models of Heteroscedasticity. In: Time series analysis with applications in R. 2008:277-318.

[32] Hyndman RJ, Koehler AB. Another Look at Measures of Forecast Accuracy. Int J Forecasting. 2006;22:679-688.

[33] Vijh M, Chandola D, Tikkiwal VA, Kumar A. Stock Closing Price Prediction Using Machine Learning Techniques. Procedia Comput Sci. 2020;167:599-606.

[34] Xu S, Chan HK. Forecasting Medical Device Demand With Online Search Queries: A Big Data and Machine Learning Approach. Procedia Manuf. 2019;39:32-39.

[35] Pinelis M, Ruppert D. Machine Learning Portfolio Allocation. J Fin Data Sci. 2022;8:35-54.