IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Comparative Study Of Machine Learning (Ml) And Deep Learning (Dl) Techniques For Predicting Stock Prices

[1]Dr.Y.Sowjanya Kumari, [2]Dr.G.Prasuna, [3]M.Swapna

[1]Associate Professor, [2]Associate Professor, [3]Assistant Professor

[1]Computer Science & Engineering, [2]Computer Science & Engineering, [3]AIML,

[1]QISCET, Ongole, India,[2]SACET,Chirala, India, [3]QISCET, Ongole

Abstract: In order to forecast TATA Chemicals' stock prices over the next ten days, we use machine learning and deep learning techniques. The Root Mean Square Error (RMSE) is used to assess the model's accuracy. In order to achieve prediction accuracy, we employ time series forecasting techniques and historical stock price data, specifically using models like GRU, LSTM, Linear Regression(LR), Random Forest Regression(RFR), and XGBoost. To predict future patterns, these models are trained on historical stock values. We may evaluate the effectiveness of the prediction model by using RMSE as a statistic to quantify the difference between the actual and forecasted stock prices. The outcomes show how well the model captures transient changes in stock prices, making it a useful tool for analysts and investors.

Index Terms **- GRU, LSTM, XGboost, RMSE, RFR, LR**

I. INTRODUCTION

For a long time, analysts, researchers, and investors have all been interested in stock price prediction. Accurately predicting stock prices is a difficult but crucial undertaking since financial markets display complicated behaviours impacted by numerous factors. In this field, traditional statistical techniques like the Generalised Autoregressive Conditional Heteroskedasticity (GARCH) and Autoregressive Integrated Moving Average (ARIMA) have been used extensively. The non-linear and non-stationary features included in financial time series data, however, are frequently difficult for these models to capture .

As deep learning (DL) and machine learning (ML) technologies develop quickly, researchers are starting to investigate these cutting-edge methods to improve prediction accuracy. Support Vector Machines (SVM), Artificial Neural Networks (ANN), and ensemble approaches are among the machine learning (ML) techniques that have demonstrated promise in simulating the intricacies of stock price fluctuations [1]. Specifically, deep learning models—like Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks—have proven to be more effective at identifying complex patterns in time series data [1]. Additionally, incorporating data from other sources, such financial news and social media sentiment, has become a useful way to enhance stock price forecasts. Investor behaviour and market dynamics can be greatly impacted by the attitude conveyed in news stories and internet forums [1]. By incorporating sentiment analysis into predictive models, researchers aim to enhance the robustness and accuracy of stock price forecasts [1].

[2] seeks to present a thorough analysis of the state of stock price prediction approaches today, emphasising the shift from conventional statistical models to cutting-edge machine learning and deep learning methods. We will talk about the ramifications of adding sentiment analysis to stock price prediction frameworks and examine a variety of strategies, including hybrid models that incorporate several approaches. Through this investigation, we hope to draw attention to how stock price forecasting is always changing and the possibility

of further development of these predictive models in the future. The stock market is a dynamic and intricate ecology that is marked by erratic changes and shifting trends. Because financial time series data is inherently non-linear and very volatile, accurately predicting stock prices in the face of this volatility is still extremely difficult. Conventional methods, such technical indicators and statistical models, frequently fail to capture the complex patterns and abrupt shifts that characterise stock market movements [5,6].

Predicting the stock market has changed dramatically in recent years because to the development of deep learning models, especially deep neural networks. These models provide a promising answer to the problems with conventional approaches by using sophisticated data processing techniques to reveal hidden patterns in raw data [6].

Four deep learning models—Multilayer Perceptron (MLP), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN)—have become well-known among the several architectures investigated [6]. Studies have demonstrated that when it comes to stock price forecasting, these deep learning architectures routinely perform better than conventional linear models like ARIMA [6]. A comparison of neural network architectures on the New York Stock Exchange (NYSE) and the National Stock Exchange (NSE) of India, for example, showed that CNNs had the lowest Mean Absolute Percentage Error (MAPE) when compared to other models, indicating their capacity to identify complex relationships and non-linear trends in financial data [5,6]. The incorporation of textual analysis into stock price forecasting has become a novel approach that enables the incorporation of sentiment and market sentiment indicators into predictive models [8]. This interdisciplinary research, which combines machine learning and finance, not only advances stock market prediction methodologies but also creates opportunities for future research [5]. As the financial landscape continues to change, it will be essential to continuously refine and adapt deep learning techniques to improve their performance and reliability in forecasting market trends [5]. The purpose of this study is to investigate the effectiveness of different deep learning architectures in stock price prediction, with a focus on their capacity to capture the complexities of financial time series data while also taking into account the impact of textual sentiment analysis on forecasting accuracy.In order to improve model robustness, future research on stock price prediction will likely investigate alternative data sources, such as sentiment on social media and macroeconomic indicators [1]. Additionally, developments in real-time data analysis and quantum computing may further improve predictive capabilities [1]. As the field develops, combining traditional financial theories with state-of-the-art AI algorithms is likely to produce novel approaches to stock price forecasting [1].

II. RELATED WORK

With the advent of sophisticated machine learning (ML) and deep learning (DL) techniques, the field of stock price prediction has undergone a considerable evolution from conventional statistical methodologies. The purpose of this literature review is to present a summary of the many approaches used in stock price forecasting, emphasising their advantages, disadvantages, and new developments in the field.In the past, statistical models like the Generalised Autoregressive Conditional Heteroskedasticity (GARCH) and Autoregressive Integrated Moving Average (ARIMA) were widely used to predict stock prices. The non-linear and non-stationary nature of financial markets presents challenges for these models, which perform well for linear and stationary time series data [1]. The intricate dynamics and changing patterns in stock prices are frequently overlooked by ARIMA models, despite their ability to capture short- to medium-term trends [1].SVM has been shown to outperform traditional models in terms of prediction accuracy [4]. However, ML models can be sensitive to noise in the data and may require extensive computational resources for training [1]. The emergence of machine learning prompted researchers to investigate algorithms that could better handle the complexities of financial data.

With the advent of architectures like Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, deep learning has become a potent tool for stock price prediction. Complex temporal relationships and non-linear patterns in stock price fluctuations are well captured by these models [1].

One technique that has shown excellent success in stock price prediction is CNN-BiLSTM-AM, which combines CNN for feature extraction with BiLSTM for sequential data processing [2]. Furthermore, hybrid models that combine many deep learning methods have demonstrated encouraging outcomes, utilising each method's advantages to improve prediction accuracy [1].

The significance of using investor emotion in stock price prediction models has been emphasised by recent studies. Market dynamics can be greatly impacted by the attitude reflected in financial news and social media [4]. For example, the MS-SSA-LSTM model improves forecast accuracy by combining sentiment analysis with conventional financial indicators [4]. This strategy demonstrates the increasing acceptance of behavioural finance concepts, which imply that investor sentiment is a significant factor in market

fluctuations [4]. Due to their ability to reduce the danger of overfitting and effectively generalise to new data, hybrid models that incorporate multiple prediction techniques have become more and more popular [1]. This trend is best illustrated by the two-stage methodology based on Variational Mode Decomposition (VMD) and ensemble learning, in which the stock price time series is broken down into smaller series and several machine learning models are used to forecast these elements [3]. In addition to improving prediction performance, our approach tackles the difficulties caused by noisy and non-stationary data [3].Multilayer Perceptron (MLP) model has been utilized for its ability to capture non-linear relationships in data. It serves as a baseline for comparison against more complex architectures [5,6].

Time series forecasting can benefit from the use of Recurrent Neural Networks (RNN), including various variations such as Long Short-Term Memory (LSTM) networks, which are especially efficient with sequential data [5,7]. In capturing the intricacies of financial time series, they have been demonstrated to perform better than conventional models such as ARIMA [5,6]. CNNs have proven to be very effective at identifying trends and patterns in changes in stock prices, particularly when it comes to identifying seasonal patterns [5,6]. As demonstrated by their performance on both NSE and NYSE data, they have been recognised for their strong generalisation across many markets [5,6].

It has been discovered that conventional models, such as ARIMA, have trouble with the non-linearities present in stock market data. Neural networks, on the other hand, have demonstrated a notable decrease in Mean Absolute Percentage Error (MAPE) in stock price forecasting [5,6]. For example, across a range of stocks, CNNs' MAPE values were lower than ARIMA's [5,6].

For uniformity, normalising stock data is essential, particularly when working with datasets from several markets [5,6]. By bringing the data into a standardised range, this procedure aids in enhancing the model's performance.

It has been demonstrated that efficient feature extraction methods improve models' predictive power. For example, higher forecasting outcomes have been obtained by combining deep learning techniques with technical indicators [6].

It has been demonstrated that adding sentiment indices from news stories and social media increases forecasting accuracy. To illustrate how sentiment affects market movements, one study created a World Halal Tourism Composite Sentiment Index to predict stock prices [8]. Combining sentiment analysis and deep learning will improve prediction performance by enabling models to take irrational investor behaviour and market mood into consideration [8].

Some academics and practitioners may find the deployment of deep learning models prohibitively computationally demanding [5].Especially with complicated models, there is a risk of overfitting. To improve model reliability, methods must be continuously improved and adjusted [5].Additional difficulties arise when theoretical models are applied in real-world live trading settings, calling for more study on the resilience and flexibility of models [5].

The study highlights the drawbacks of relying only on accuracy as a performance parameter and suggests employing a variety of metrics to evaluate the validity of the model [10].The efficacy of Support Vector Machines (SVM) in text mining applications for stock prediction is highlighted by the authors [10]. The creation of a framework for assessing text-based prediction models, which incorporates information on model interpretability and validation, is a noteworthy contribution of this study [10].

In comparison to conventional algorithms, the authors show that adding a sentiment index gleaned from forum postings and social media greatly increases prediction accuracy [11]. In order to enable more precise sentiment analysis, the study highlights the significance of creating a sentiment lexicon specific to the financial context [11].With an average improvement of 10.74% in R2 values, the MS-SSA-LSTM model beats traditional LSTM models, demonstrating its resilience and suitability in erratic markets[11].

Using deep learning technology, this study suggests a novel approach to stock price prediction that combines conventional financial data with textual features from social media.- The technique uses a long short-term memory (LSTM) model for prediction after employing Doc2Vec for social media feature extraction and a stacked auto-encoder for dimensionality reduction. According to experimental results, this method is more effective than standard benchmark models in predicting stock prices across a range of evaluation metrics [12].

By accurately anticipating the COVID-19 recession six months before it was formally announced, the study demonstrates how supervised machine learning (ML) approaches may predict stock market collapses and economic recessions.- The labour market was found to be a strong predictor of both recessions and crashes by machine learning algorithms, highlighting the significance of financial considerations over economic indicators in stock market forecasts.According to the study, there is a need for better methods in recession forecasting because recession forecasts have greater false discovery rates than crash predictions [13].

The goal of this study is to employ machine learning techniques to forecast price fluctuations in the highly volatile and illiquid cryptocurrency market.- According to the study, cryptocurrencies' common characteristics can provide prediction models that are more accurate than asset-specific ones, consistently anticipating mid-price fluctuations with an accuracy of 78%. The difficulties of training models using real-time data are also covered, and a novel retraining technique is suggested to sustain model performance over time [14].The study uses evaluated linear regression as a machine learning technique to highlight how crucial it is to accurately predict stock market index movement for profitable trading strategies. It talks about the difficulties in predicting stock values and introduces the ELR-ML model as a way to increase prediction precision and dependability. These results indicate that the use of machine learning methods can greatly improve stock price forecasting [15].

By classifying them into traditional machine learning methods, deep learning, time series analysis, and graph-based approaches, this systematic study examines a variety of machine learning techniques used for stock price prediction.The study emphasises how combining several methods, such sentiment analysis and historical data, may increase forecast accuracy.It highlights the field's main obstacles, such as the requirement for bigger datasets and the use of behavioural finance knowledge to improve prediction models [16].By choosing characteristics, addressing missing values, and normalising data, the suggested stock market prediction system preprocesses data. With RMSE, MSE, MAE, and R2 values of 0.2883, 0.0831, 0.2099, and 0.9948, respectively, a combined LSTM and BiGRU model predicts the closing stock prices during the prediction phase, producing excellent results. On a variety of datasets, this model also exhibits good accuracy [17].

The volatile nature of stock valuations and the scarcity of previous data make it difficult to forecast stock market returns. To improve prediction accuracy, more variables were created, and an ANN model was used to forecast the closing price for the following day. This model outperformed the RF model in terms of accuracy. The ANN demonstrated improved performance with RMSE of 0.42, MAPE of 0.77, and MBE of 0.013 [18].Using a sliding window technique on OHLC values, this work uses GRU and LSTM architectures to identify intricate patterns and dependencies in stock market data. The suggested model performs better than current models, obtaining lower MAPE values (e.g., 0.630 vs. 1.748) according to performance comparisons across several datasets. To further improve model accuracy, technical indicators including SMA, EMA, RSI, MACD, and ADX are also incorporated [19].This technique evaluates 11 two-layer architectures for stock price prediction using CNN, GRU, LSTM, and GCN layer combinations. Based on criteria such as R2, MSE, RMSE, MAPE, and F1-score, the optimal model, TFGRU, was chosen. It performed best on 315 companies, with an average RMSE of 553.327, MAPE of 0.858, and F1-score of 0.456 [20].

III. PROPOSED WORK

3.1 Long short-term Memory (LSTM):

Long short-term Memory Neural Network (LSTM) : Long-term dependencies in sequential data can be modelled and captured using LSTM, a specialised kind of recurrent neural network (RNN). It gets beyond the drawbacks of conventional RNNs, especially the incapacity to manage long-range relationships because of issues like vanishing gradients. LSTM networks are another type of DL network that is specifically used for sequential data analysis. The advantage of LSTM networks is that both short - and long-term values in the network can be remembered. DL researchers are therefore primarily using translation, handwritten character recognition, time series data prediction, etc.). The LSTM network consists of LSTM units. LSTM cells consist of cells with input, output, and forget gates. These three gates regulate the flow of information. Using these features, each cell can remember the desired value at any time interval. LSTM cells merge to form the neural network layer

By introducing a distinct cell state and a number of gates that regulate information flow, LSTM makes it possible to keep crucial information over extended periods of time and eliminate irrelevant data. Memory State of the Cell serves as the LSTM's memory, accumulating data over time.Gates control the information flow in the cell state.GatesThe forget gate determines which data should be deleted from the cell state.The input gate determines what additional data should be added to the cell state. The output gate determines which data from the cell state should be sent to the following time step. . The following diagram illustrates the composition of an LSTM cell.

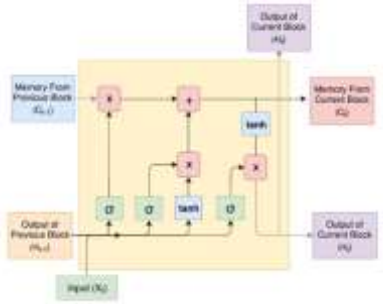

Figure 1: Architecture of LSTM [9].

3.2 Gated Recurrent Unit (GRU):

A recurrent neural network (RNN) architecture known as a Gated Recurrent Unit (GRU) was created to handle sequential input by solving the vanishing gradient issue that conventional RNNs frequently encounter. GRUs are more straightforward and have fewer characteristics than Long Short-Term Memory (LSTM) units, but they also feature gating mechanisms that regulate the information flow.There are two primary gates at GRUs.Update Gate: This gate determines which historical data should be transmitted to the future. It regulates the amount of the prior state that should be kept.Reset Gate: This gate helps the model adaptively choose when to disregard or reset previous states by regulating how much of the past should be forgotten.

Because GRUs are computationally lighter than LSTMs and are capable of efficiently capturing dependencies in data over time, they are frequently employed for jobs involving time-series data, language modelling, and other sequence-based applications.An improved RNN architecture called a Gated Recurrent Unit (GRU) controls information flow to identify long-term dependencies in sequential data while avoiding problems such as the vanishing gradient problem. With fewer parameters and less computational complexity, it was first presented as a quicker and easier substitute for Long Short-Term Memory (LSTM) networks. As a result, it gained popularity in applications like as time series analysis and natural language processing.GRUs are computationally more efficient than LSTMs thanks to their structure, which also enables them to efficiently capture sequential dependencies in data and provide the memory required to preserve context between time steps.

3.3 Linear Regression(LR):

A straightforward yet effective statistical technique for simulating the relationship between a dependent variable and one or more independent variables is linear regression. In its most basic form, linear regression fits a linear equation to observed data in order to assess the connection between two variables. Predictive modelling makes extensive use of it, particularly when attempting to forecast a quantitative result using continuous or categorical predictors. One way to evaluate a linear regression model is as follows: $y = mx+c$. Here, Y—also known as the dependent variable or target variable—is the result variable that we wish to forecast. In this case, variable X is referred to as an independent variable. Simple linear regression is used when there is only one independent variable; multiple linear regression is used when there are numerous independent variables. A linear relationship between the independent variable or variables and the dependent variable is assumed by linear regression. In other words, we suppose that when Y is plotted against X, it can be represented as a plane or straight line as follows, Finding the coefficient values that minimize the discrepancy between the observed values of Y and the values predicted by the linear equation is the aim of linear regression. Usually, the sum of squared residuals is minimized to quantify this difference. Linear regression determines the line that best fits the data by minimizing the sum of square error (SSE).Some of the drawbacks of linear regression include It may perform poorly if the true relationship is nonlinear, as it primarily captures linear relationships. The regression line may be significantly impacted by outliers. assumes that mistakes are homoscedastic and residuals are normally distributed.

3.4 Extreme Gradient Boosting (XGBoost)

For supervised learning tasks, XGBoost—short for eXtreme Gradient Boosting—is a sophisticated, effective, and scalable gradient boosting method. Because of its great efficiency, scalability, and flexibility, it has emerged as one of the most often used algorithms for structured or tabular data. By incorporating optimizations that make it faster and more accurate, especially for large datasets, XGBoost expands upon the gradient boosting framework.

Understanding gradient boosting is useful before delving into XGBoost. With the help of an ensemble technique called gradient boosting, several decision trees are constructed one after the other, each of which fixes the mistakes of the one before it. By iteratively adding models that forecast the residual errors (gradients)

of earlier models, the objective is to minimize a given loss function. The model's goal in XGBoost is to minimize the objective function. The total of the following is the objective function: Function of Loss evaluates the degree to which the target values and the model's forecasts agree. Term of Regularization penalizes big trees or overly complicated models in order to control the model's complexity and avoid over fitting.

By concentrating on mistakes made by earlier trees, XGBoost creates models in rounds. This is the comprehensive, step-by-step procedure: A base prediction is made at the beginning of XGBoost.The discrepancies between the actual values and the initial model's anticipated values are known as residuals. XGBoost calculates the gradient of the loss function for every data point in relation to the model's current predictions. The residuals, which show the error's magnitude and direction, are these gradients. In order to fit these residuals, XGBoost builds a new tree. It accomplishes this by determining which splits minimize residual errors for each feature.

The splits are selected with the goal of maximizing A specific value, frequently referred to as a weight or score, is represented by each leaf in the tree and will be added to the earlier forecast. This score helps fix the earlier prediction errors and is based on the average of the residuals assigned to that leaf. XGBoost penalizes high values on the leaf nodes by using L1 (Lasso) and L2 (Ridge) regularisation terms. By discouraging complex models, this ensures model generalization and lowers over fitting. XGBoost adds the output of the new tree to the prior forecast at the end of each round to update the predictions. By regulating the rate at which the model learns, the learning rate (also known as the shrinkage factor) helps to avoid over fitting by dictating how much of the output from the new tree should go towards the final prediction.

For a predetermined number of boosting rounds or until the loss function converges (stops dropping), this sequence of computation of residuals, construction of a new tree, updating of predictions, and application of regularization is repeated. A major enhancement over conventional gradient boosting, XGBoost incorporates both L1 and L2 regularization components in the goal function to avoid overfitting. By slowing down the learning process, XGBoost reduces the contribution of each tree after each boosting round, strengthening the model.Like Random Forests, XGBoost speeds up training by sampling a subset of features at each iteration to avoid overfitting.When working with datasets that contain missing values, XGBoost's ability to automatically handle missing data by determining the optimum path for missing values in each tree is advantageous.

Compared to other boosting techniques, XGBoost is incredibly fast on large datasets since it parallelizes the tree-building process.

3.5 **Random Forest Regressor (RFR):**

Based on decision trees, the Random Forest Regressor is an ensemble learning technique for regression applications. To arrive at a final forecast, it generates a number of decision trees, each based on a different collection of features and data. A more generalised model is produced, accuracy is increased, and overfitting is decreased because to this aggregation. Applications needing high interpretability and feature importance ranking frequently employ the Random Forest Regressor. Random Forest is an ensemble method that uses bagging(Bootstrap Aggregating), a technique where multiple trees are built independently, trained on random samples from the dataset. The process of combining several models, sometimes referred to as "weak learners," to create a stronger, more accurate model is known as ensemble learning.

A decision tree trained on a distinct random subset of the data makes up each tree in a random forest. Each node in a decision tree divides the data according to a feature in order to minimise error (in regression) or maximise the separation between classes (in classification). These trees are combined in Random Forests to lessen the inherent volatility in decision trees. Each tree in a Random Forest is trained using a random sample of the dataset, a process known as bootstrapping. It's possible to repeat some samples and leave out others. Random Forests choose a random subset of characteristics for node splitting for every tree. The overall variance is decreased and the association between the trees is further diminished by this feature selection unpredictability.

Every decision tree is expanded as deeply as possible without being pruned. Because Random Forest relies on mixing several decision trees to prevent overfitting, it does not trim trees like single decision trees do. Each tree's nodes are divided according to which feature minimises the residual error the most. After every tree has been constructed, the Random Forest Regressor averages the results (for regression tasks) or uses a majority vote (for classification tasks) to combine the predictions from every tree. The impact of each tree's prediction errors is lessened by this averaging procedure, which also helps to smooth out errors. A bootstrapped dataset (random sample with replacement) and a selection of features are used to train each decision tree separately.

Every tree offers a prediction for a certain input sample. The final regression output is calculated by averaging the predictions from each tree.

By analysing how frequently a feature is chosen for splits across all trees and how much it helps to lower the residual error, Random Forests are able to measure the significance of each feature. The most significant features in predicting the target variable are identified with the aid of the feature significance scores.

For real-world applications where interpretability is less important, Random Forest is a strong and adaptable model for regression problems. It offers excellent accuracy, resilience against overfitting, and the capacity to handle a range of data sources.

IV. RESULTS AND DISCUSSIONS

The results were computed in two stages.First, the Root Mean Square Error (RMSE) values were calculated to evaluate the performance of the models, including LSTM, GRU, Linear Regression (LR), XGBoost, and Random Forest Regression (RFR).Next, these models were used to predict the stock prices for the coming 10 days, providing a comparison of their predictive capabilities.

RMSE:A popular metric for assessing the precision of forecasting and regression models is RMSE. It provides information about how well the model's predictions match the observed data by calculating the average magnitude of the error between the predicted and actual values.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

n is Number of data points, yi Actual observed value at point i, $\hat{y}_i$Predicted value at point i.

A low RMSE suggests good accuracy because it shows that the model's predictions are fairly close to the actual values.A high RMSE indicates a less accurate model by implying greater differences between expected and actual values.

Table 1.ML/DL model and its RMSE value

Model	RMSE
LSTM	40.04
GRU	35.9749
LR	34.4285
XGBOOST	161.7445
RFR	161.1129

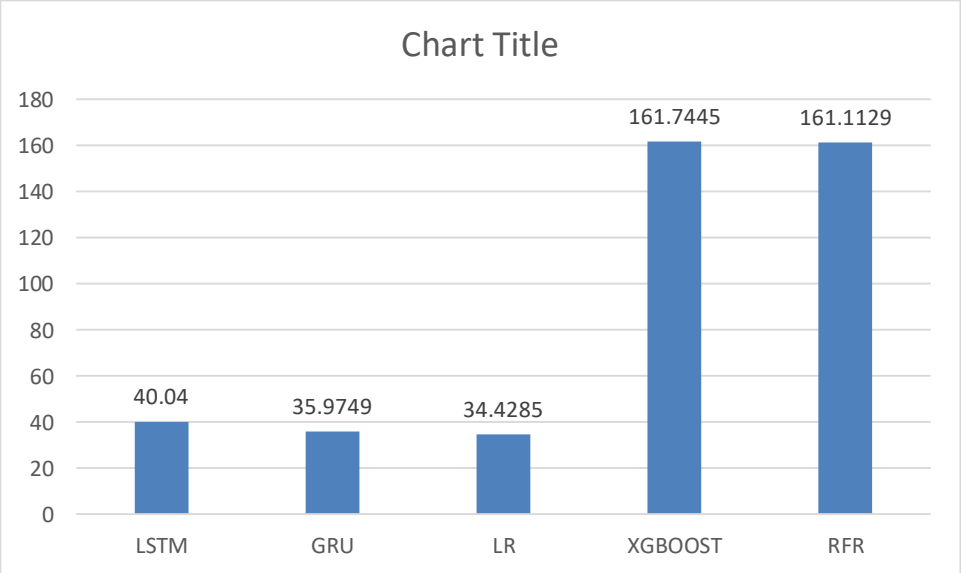

Figure 2: comparison of RMSE values for various ML,DL Models

Figure 2 shows various RMSE values that are obtained using different models. The RMSE values we acquired for TATA Chemicals' stock market price prediction using several models provide intriguing details about how well they performed. RMSE for Linear Regression (34.428) is performing the best, which may suggest that the stock price data exhibits rather straightforward linear patterns. This implies that, when employing linear trends, the stock prices are quite predictable. Neural network models GRU (35.974 RMSE) and LSTM (40.04 RMSE) are made to identify time dependencies and sequential patterns in data. These models' marginally poorer performance than Linear Regression may indicate that either: The stock price data lacks intricate time dependencies that these models can successfully use. These models might benefit from more fine-tuning (e.g., altering hyper parameters or architecture). GRU performs somewhat better than LSTM, most likely as a result of its simpler architecture and suitability for smaller datasets or situations with less time dependencies. Random Forest Regressor (161.112 RMSE) and XGBoost (161.7445 RMSE) the low performance of these tree-based models in comparison raises the possibility that the stock price data is not a good fit for tree-based techniques. The following are some potential explanations: XGBoost and Random Forest models are not built to capture time-dependent trends and seasonality, which frequently affect stock prices. Higher errors may result from these models' inability to handle the sequential nature of the data.

In the present scenario, sequential models (LSTM, GRU) could not be contributing much value because there aren't enough complex temporal interactions, but simple linear models (like Linear Regression) might be useful if the stock prices are generally stable with linear trends. If not appropriately adapted (e.g., utilising time-series specific adjustments), tree-based models (XGBoost, Random Forest) are probably having trouble with time series data.

Table2: Price Prediction for next 10 days by various Models

Model	DAY1	DAY2	DAY3	DAY4	DAY5	DAY6	DAY7	DAY8	DAY9	DAY10
LSTM	708.12	728.19	721.48	723.10	722.21	722.11	732.65	736.26	737.46	733.986
GRU	733.64	758.12	750.70	752.59	751.80	750.43	762.42	767.22	767.99	764.937
LR	757.31	768.90	754.23	766.17	758.67	762.80	775.46	779.01	775.35	776.063
XGBoost	323.36	322.11	325.06	325.83	327.36	322.40	325.67	320.33	320.12	319.016
RFR	326.70	322.44	327.67	321.70	327.21	321.28	327.21	320.72	320.65	321.259

Table 2 shows the predicted stock price for coming 10 days by various models. It is observed that the two ensemble models i.e XGBoost,RFR predicted the near values. The machine learning model linear regression performed inline with DL models.

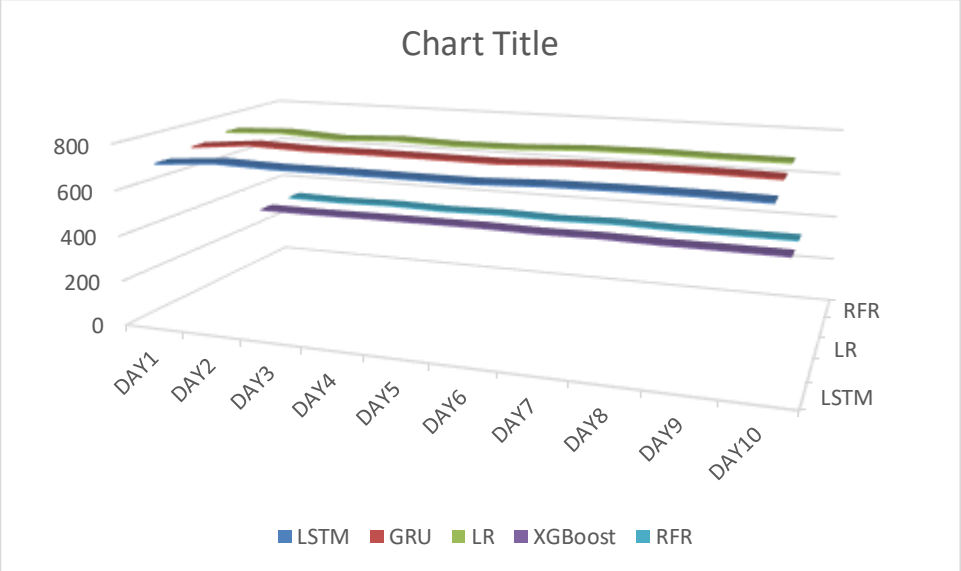

Figure 3: Price Prediction for next 10 days by various Models

The predicted stock prices, which range from 708.12 to 737.46, show a moderate degree of unpredictability. In comparison to LSTM, GRU forecasts exhibit a comparable but marginally greater range, ranging from 733.64 to 767.99. Compared to LSTM, GRU tends to overfit minor upward trends, but it performs competitively for sequential data.With a steady rising trend, Linear Regression (LR) predictions fall between 754.23 and 779.01. With only little variations, the XGBoostPredicted values are quite low, ranging from 319.02 to 327.36.It appears that XGBoost underestimated the stock prices, which could mean that its

feature engineering or hyperparameter tweaking did not adequately capture the behaviour of the stock. With minor daily changes, Random Forest Regression (RFR) predictions range from 320.65 to 327.67, which is similar to XGBoost.Similar to XGBoost, RFR seems to underestimate the stock price, most likely as a result of decision tree-based approaches' lack of temporal modelling.

V. CONCLUSION

The study employs machine learning and deep learning models, such as LSTM, GRU, Linear Regression (LR), Random Forest Regression (RFR), and XGBoost, to forecast the stock values of TATA Chemicals over a ten-day period. Root Mean Square Error (RMSE) was used as the assessment metric to evaluate the performance of these models, which were trained on historical stock price data. Among the models examined, Linear Regression (LR) had the highest performance, achieving the lowest RMSE of 34.4285, closely followed by GRU with an RMSE of 35.9749. The capacity of the LSTM model to capture temporal dependencies in stock price patterns was demonstrated by its strong performance, which included an RMSE of 40.04.

The RMSE values of 161.7445 and 161.1129 for XGBoost and RFR, on the other hand, were noticeably higher, indicating that these models were less successful in capturing the intricate, sequential character of stock price fluctuations.

All things considered, the analysis emphasises how crucial it is to choose the best model depending on the type of data and the issue at hand. Because of their ability to make accurate forecasts, GRU, LSTM, and LR are helpful tools for investors and financial analysts who want to predict short-term changes in stock prices.

REFERENCES

[1] Vuong, P. H., Phu, L. H., Van Nguyen, T. H., Duy, L. N., Bao, P. T., & Trinh, T. D. (2024). A bibliometric literature review of stock price forecasting: From statistical model to deep learning approach. Science Progress, 107(1). https://doi.org/10.1177/00368504241236557.

[2] Lu, W., Li, J., Wang, J., & Qin, L. (2020). A CNN-BiLSTM-AM method for stock price prediction. Neural Computing and Applications, 33(10), 4741–4753. https://doi.org/10.1007/s00521-020-05532-z

[3] Zhang, J., & Chen, X. (2023). A two-stage model for stock price prediction based on variational mode decomposition and ensemble machine learning method. Soft Computing, 28(3), 2385–2408. https://doi.org/10.1007/s00500-023-08441-0

[4] Mu, G., Gao, N., Wang, Y., & Dai, L. I. (n.d.). A Stock Price Prediction Model Based on Investor Sentiment and Optimized Deep Learning.

[5] Bishnu Padh Ghosh, Mohammad Shafiquzzaman Bhuiyan, Debashish Das, Tuan Ngoc Nguyen, Mahmud Jewel, Md Tuhin Mia, & Duc M Cao. (2024). Deep Learning in Stock Market Forecasting: Comparative Analysis of Neural Network Architectures Across NSE and NYSE. Journal of Computer Science and Technology Studies, 6(1), 68–75. https://doi.org/10.32996/jcsts.2024.6.1.8

[6] Md Salim Chowdhury, Norun Nabi, Md Nasir Uddin Rana, Mujiba Shaima, Hammed Esa, Anik Mitra, Md Abu Sufian Mozumder, Irin Akter Liza, Md Murshid Reja Sweet and Refat Naznin Deep Learning Models for Stock Market Forecasting: A Comprehensive Comparative Analysis Journal of Business and Management Studies. (n.d.). Al-Kindi Center for Research and Development. https://doi.org/10.32996/jbms.

[7] Wei, Y., Gu, X., Feng, Z., Li, Z., & Sun, M. (n.d.). Feature Extraction and Model Optimization of Deep Learning in Stock Market Prediction.

[8] Abdullah, M., Sulong, Z., & Ashraful Ferdous Chowdhury, M. (2024). Explainable deep learning model for stock price forecasting using textual analysis. Expert Systems with Applications, 249, 123740. https://doi.org/10.1016/j.eswa.2024.123740

[9] SONG, Jintong, CHENG, Qishuo,BAI, Xinzhu ,JIANG, Wei, SU, Guangze LSTM-Based Deep Learning Model for Financial Market. Journal of Economic Theory and Business Management ISSN 3006-4953 (Print) | ISSN 3006-4961 (Online) | Vol. 1, No. 2, 2024.

[10] Junqué De Fortuny, E., De Smedt, T., Martens, D., & Daelemans, W. (n.d.). Evaluating and understanding text-based stock price prediction models.

[11] Mu, G., Gao, N., Wang, Y., & Dai, L. I. (n.d.). A Stock Price Prediction Model Based on Investor Sentiment and Optimized Deep Learning.

[12] Ji, X., Wang, J., & Yan, Z. (2021). A stock price prediction method based on deep learning technology. International Journal of Crowd Science, 5(1), 55–72. https://doi.org/10.1108/ijcs-05-2020-0012

[13] Malladi, R. K. & California State University, Dominguez Hills, in California. California State University Dominguez Hills in California. (n.d.). Application of Supervised Machine Learning Techniques to Forecast the COVID-19 U.S. Recession and Stock Market Crash.

[14] Fang, F., Chung, W., Ventre, C., Basios, M., Kanthan, L., Li, L., & Wu, F. (2021). Ascertaining price formation in cryptocurrency markets with machine learning. The European Journal of Finance, 30(1), 78–100. https://doi.org/10.1080/1351847x.2021.1908390

[15] Sangeetha, J. M., & Alfia, K. J. (2024). Financial stock market forecast using evaluated linear regression based machine learning technique. Measurement: Sensors, 31, 100950. https://doi.org/10.1016/j.measen.2023.100950

[16] Soni, P., Tewari, Y., & Deepa Krishnan, P. (n.d.). Machine Learning Approaches in Stock Price Prediction: A Systematic Review.

[17] Warda M. Shaban,Eman Ashraf, Ahmed Elsaid Slama SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting, Neural Computing and Applications (2024) 36:1849–1873, https://doi.org/10.1007/s00521-023-09179-4.

[18] Mehar Vijh , Deeksha Chandola Vinay Anand Tikkiwal , Arun Kumar, Stock Closing Price Prediction using Machine Learning Techniques, International Conference on Computational Intelligence and Data Science (ICCIDS 2019).

[19] E. T. Sivadasan, N. Mohana Sundaram ,R. Santhosh, Stock market forecasting using deep learning with long short-term memory and gated recurrent unit, Soft Computing (2024) 28:3267–3282, https://doi.org/10.1007/s00500-023-09606-7.

[20] Agus Tri Haryono, Riyanarto Sarno, Kelly Rossa Sungkono, Stock price forecasting in Indonesia stock exchange using deep learning: a comparative study, International Journal of Electrical and Computer Engineering (IJECE),Vol. 14, No. 1, February 2024, pp. 861~869
ISSN: 2088-8708, DOI: 10.11591/ijece.v14i1.pp861-869.

Dr.Y.SowjanyaKumari working as Associate Professor in QISCET (Affiliated to JNT University, Kakinada), Chirala, India,and she has 20 years of teaching experience. she received her Ph.D.degree from JNT University, Kakinada in 2019. She has receivedM.Tech. in Computer Science & Engineering from JNTU,Kakinada A.P and B.Tech in Computer Science & Engineeringfrom N.B.K.R.I.S.T, Vidyanagar, Nellore(dt) A.P. Her areas of interesting includes Digital image processing and Artificial Intelligence, Machine Learning.

Dr Grandhi Prasuna received M.Tech in Computer Science & Engineering from Jawaharlal Nehru Technological University Kakinada, Kakinada and Ph.D in Computer Science & Engineering from Acharya Nagarjuna University in 2020. She has 25 years of experience in teaching Computer science related subjects. Currently, she is working as an Associate Professor in CSE department, St. Ann's College of Engineering & Technology, Chirala, Bapatla district. Her research interests include Software Engineering, Databases, Computer Networks, Artificial Intelligence, Internet of Things, Machine Learning and Deep Learning. She can be contacted at email: grandhiprasuna@gmail.com. ORCID: https://orcid.org/0009-0001-8953-4281