International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

Comparative Analysis of Stock Price Prediction Accuracy: A Machine Learning Approach with ARIMA, LSTM, And Random Forest Models

Brahmanapalli Kalyan[1], S Parameshwara Reddy[2], Dr. Krovvidi Krishna Kumari[3], Dr. Manish Jain[4]

[1,2] Student, Indus Business Academy, Bangalore, India.

[3]Assistant Professor, Indus Business Academy, Bangalore, India.

[4]Associate Professor, Indus Business Academy, Bangalore, India.

Emails: *fpb2224.063@iba.ac.in[1], fpb2224.093.parameshwara@iba.ac.in[2], Krishna.k@iba.ac.in[3], mj@iba.ac.in[4]*

Abstract

This research investigates the comparative effectiveness of three distinct predictive models – ARIMA (Auto Regressive Integrated Moving Average), LSTM (Long Short-Term Memory), and Random Forest – in forecasting stock prices. Focusing on Tata Motors and Infosys stocks, historical data spanning a significant timeframe is collected using the finance library. These models are trained on a diverse set of features including open, close, high, and low prices to capture the underlying market dynamics. The evaluation of model performance is centred on their ability to forecast stock prices over varying prediction horizons. Metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) are utilized to quantify the accuracy and reliability of the predictions. Through rigorous analysis, this study provides insights into the strengths and limitations of each model, offering valuable guidance to investors and market analysts. The findings underscore the significance of selecting appropriate predictive models in financial forecasting and contribute to advancing the understanding of predictive modelling techniques in stock market analysis. Additionally, the research delves into the implications of long-term dependencies in stock price forecasting, shedding light on the challenges and opportunities inherent in predicting market trends over extended periods.

Keywords: *ARIMA Model, Forecast, LSTM, Random Forest, Stock Market Analysis.*

1. Introduction

Machine learning (ML) has emerged as a disruptive force in a variety of industries, changing the way we analyze data, generate predictions, and automate decision making processes. Fundamentally, machine learning (ML) is creating algorithms and models that can recognize patterns in data and enhance their functionality without the need for explicit programming. The power of machine learning to draw conclusions from massive, intricate datasets is what makes it so important. It allows businesses to find hidden patterns, streamline processes, and spur creativity. Machine learning applications [1-3] cover across industries, providing solutions to a wide range of problems and opportunities, from healthcare and retail to transportation and cybersecurity. Machine learning plays a particularly important role in the financial sector, where data-driven decision-making is Essential and critical. Financial institutions, investors, and analysts can use machine learning

techniques to make informed decisions by leveraging massive amounts of market data, historical trends, and economic indicators. Machine learning algorithms can better forecast market trends, asset pricing, and risk management by studying complex linkages and patterns in financial data. In the current finance industry, machine learning is becoming an essential tool that drives efficiency, innovation, and competitive advantage in everything from algorithmic trading and portfolio optimization to fraud detection and credit scoring. Accurate predictions are critical for investors, traders, and financial institutions to make educated decisions and manage risks successfully in the volatile and unpredictable stock market. Forecasts of stock prices help investors spot investment opportunities, foresee market trends, and maximize portfolio performance.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

Accurately predicting stock prices has long been a captivating pursuit for investors and analysts (Lobach, 2021). Although they provide insightful information, traditional techniques like statistical modelling and technical analysis are unable to fully capture the complex dynamics of the market (Adedeji et al., 2020). The innate unpredictability of financial markets demands the investigation of novel approaches (Lobach, 2021). With models like ARIMA, LSTM, and Random Forest demonstrating promise in recognizing complicated patterns beyond conventional methods, the emergence of machine learning (ML) offers a unique approach (Huang & Nakashima, 2021; Li et al., 2023). Still, there is a significant obstacle to overcome: No single model is ideal for all stocks and all time periods (Rahim & Ibrahim, 2022). In an effort to shed light on the possibilities and constraints of machine learning, this study compares the accuracy of several models in predicting stock prices. Predictive models can project future price movements with varied degrees of accuracy and dependability by drawing on historical stock market data, economic indicators, and other applicable elements. Accurate stock price predictions are essential for navigating the complexities of the stock market and accomplishing financial goals, whether they be for short-term swings or long-term trends. This research aims to maximize stock price prediction accuracy by employing Three well-known machine learning models ARIMA, LSTM, and Random Forest are used in this study to optimize the accuracy of stock price predictions. These models will be trained using historical data from Tata Motors and Infosys, two different Indian equities. Through a comparison analysis of their performance on these disparate organizations, we want to shed light on which model provides the best accurate forecasts in this particular scenario. This comparison analysis will provide insightful information to investors and financial institutions looking to improve their ability to make decisions in the volatile stock market. Python is a key platform in machine learning used for stock price prediction, utilizing a powerful collection of modules designed with financial research in mind. Technical analysis indicators and historical market data are easily accessible through libraries like yfinance and ta, which help practitioners obtain pertinent data for modelling. Pandas and Numpy make preprocessing and data manipulation more effective while guaranteeing the accuracy and consistency of the incoming data. A wide range of machine learning algorithms are also provided by sklearn, enabling practitioners to create predictive models that are customized for their unique requirements. The stats models ARIMA model is a reliable tool for time series forecasting because it can effectively capture temporal trends in stock price data. When used in tandem, these resources enable practitioners to do in-depth analysis, extract significant insights, and make informed decisions in the dynamic landscape of stock markets. The yfinance library offers easy access to historical market data straight from Yahoo Finance, making it a useful tool for stock price prediction. Practitioners can quickly access a variety of financial data from this library, such as historical stock prices, trade volumes, and dividend information. Through the use of yfinance, practitioners can compile large datasets over a range of time periods, which are necessary for carrying out in-depth analysis and training machine learning models. Furthermore, yfinance streamlines the data fetching process so that practitioners may concentrate more on developing and analyzing models than on gathering data.

2. Machine Learning Models

2.1. Arima

The **ARIMA (Autoregressive Integrated Moving Average)** model stands out as a valuable tool for forecasting time series data, particularly in the realm of stock prices. One especially useful method for predicting time series data is the ARIMA (Autoregressive Integrated Moving Average) model, which is especially useful when predicting stock values. Its capacity to identify both recurrent seasonal patterns and long-term trends accounts for its success. The "integrated" component handles non-stationarity by, if needed, differencing the data (removing trends or seasonality), whereas the "autoregressive" component uses the data's historical values to forecast future trends. Finally, by "smoothing" out any differences between previous forecasts and actual values, the "moving average" component integrates past forecasting errors to improve predictions. In Figure 1 the ARIMA model adheres to a particular

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

algorithm. It evaluates the stationarity of the data first.

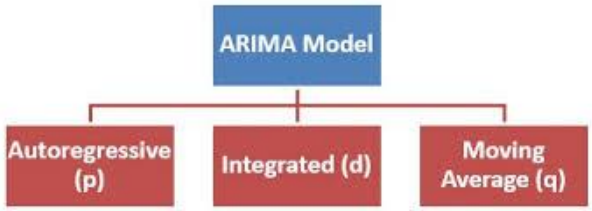

Figure 1 Arima Model

Differencing is used when needed to format the data so that it may be analyzed. The model then determines the best combination of the three crucial parameters—**"p"** (number of historical values utilized for prediction), **"d"** (degree of differencing applied), and **"q"** (number of historical forecast mistakes taken into account)—using statistical tests. Ultimately, these parameters are used to build the model, which is then trained on the historical data. It is able to discover the underlying relationships in the data as a result. After being trained, the ARIMA model may anticipate future values in the time series, like stock prices, by utilizing these correlations and past data. Combining these advantages, ARIMA provides a strong and powerful and data-driven approach to financial forecasting, aiding in making informed investment decisions.

2.2. Arima

Random Forest is an ensemble learning technique that aggregates predictions from many decision trees to improve forecast accuracy. The pursuit of precise stock price forecasting has long been a prominent focus in the financial industry. Although conventional approaches offer significant insights, they frequently fail to grasp the complex dynamics of the market. In this case, the Random Forest model proves to be an effective instrument, providing a machine learning method to find hidden patterns in financial data and produce forecasts that are more accurate. In figure 2 a Random Forest, in contrast to conventional models, is an ensemble consisting of several decision trees cooperating with one another rather than a single entity. Envision a large forest in which every tree stands for a distinct way that people make decisions. Random subsets of the data are used to train each of these unique trees, and the model considers a random selection of features (variables) to split at each node (decision point) inside a tree overfitting, which happens when a model gets overly dependent on the particular training data and loses its capacity to make correct predictions on fresh, unseen data, is prevented in part by this crucial component, the randomization in feature selection.

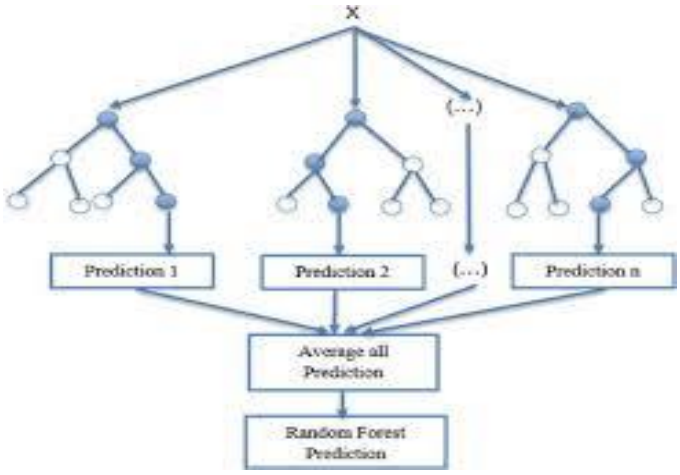

Figure 2 Random Forest Model

The following are the main benefits of Random Forests that make them especially useful for stock price prediction Importance of Random Forests, in contrast to certain other models, intrinsically offer insightful information on the proportionate weights assigned to various features in stock price prediction. This aids investors in determining which variables—such as past prices, business performance indicators, or even sentiment analysis of news—have the biggest impact on changes in price. Random Forests are less vulnerable to outliers or noise in the data since they use numerous decision trees that have been trained on different samples of data. When opposed to depending just on one decision tree, this produces predictions that are stronger and more accurate. A large variety of data types, including category and numerical variables, can be handled by Random Forests. This adaptability makes it possible to include a variety of financial data [4-7].

2.3. LSTM

Long Short-Term Memory (LSTM) networks are superior to regular neural networks at capturing the nuances of financial data. Memory cells and a special architecture allow for this. Information flow is managed by "gates" that are integrated into these

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

cells. The "output gate" chooses what is included in the final prediction, the "forget gate" chooses which historical data to ignore, and the "input gate" chooses new, pertinent data.

Fundamental Elements of an LSTM Network:

- **Forget Gate:** determines which data from the previous cell (prior data) should be ignored. It examines the output of the previous cell as well as the current input, letting through only pertinent data in the end.
- **Input Gate:** This gate chooses which fresh data from the input at hand should be kept in the cell. It chooses the data that most helps with the prediction by taking into account both the output of the previous cell and the current input.
- **Cell State:** The LSTM network's memory is housed in this central component. Based on the input and forget gates, it continuously updates the real data points that are stored.
- **Output Gate:** This gate regulates the data that should be included in the current cell state

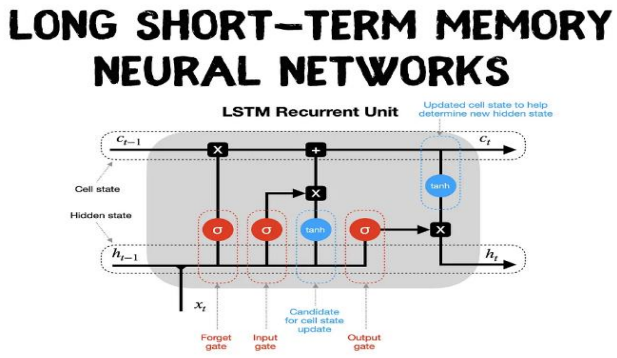

Figure 3 LSTM Model Algorithm

As a result, 3, long-term dependencies within stock prices can be learned by LSTMs. Using these gates, the LSTM algorithm feeds data into the network one step at a time, updating the memory of each cell and producing a forecast (such as a future stock price) at each stage. Subsequently, through backpropagation, the accuracy of the model is improved by comparing it with real values. LSTMs have several clear benefits. discovering long-term connections in data. Subsequently, through backpropagation, the accuracy of the model is improved by comparing it with real values. Learning long-term relationships in data and managing sequential data, such as financial time series, are two key benefits of LSTMs. But because of their intricacy, they need a lot of processing power and good training data. Even with these drawbacks, long short-term memory funds (LSTMs) are nonetheless a potent tool for investors looking to use cutting-edge machine learning to navigate the volatile world of finance.

LSTM Algorithm:

- **Input Preparation:** The financial data (such as past prices and trade volumes) is transformed into a format that is appropriate for the LSTM network by preprocessing.
- **Forward Pass:** One step at a time, the data is supplied into the network. The forget gate, input gate, cell state, and output gate process the data and update the cell state with pertinent historical data at each stage, carrying out their respective tasks.
- **Output Generation:** A future stock price or other pertinent financial variable may be identified by the output gate as the information pertinent to the current projection.
- **Backpropagation:** In this stage, the generated forecast and the actual value are compared. Any differences are utilized to modify the network's weights, improving the model's capacity to learn and forecast future

LSTM is a type of recurrent neural network (RNN), excel in predicting stock prices by capturing temporal patterns and long-term dependencies in sequential data. LSTMs trees efficiently model the intricate linkages found in financial time series. They are able to record both short-term swings and long-term patterns in stock prices because of their gated architecture, which selectively retains pertinent data. Because of this, LSTMs are frequently used for stock price prediction applications where past data is essential for predicting future patterns. Selecting the best model to predict a particular stock relies on different parameters like including the investor's goals, objectives, and data.

3. Research Methodology

This research follows a data-driven approach for short-term stock price prediction using machine learning models. The methodology can be broken down into the following steps:

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

3.1. Data Acquisition and Preprocessing

Data Source: Utilize the finance library in Python to download historical stock data for Tata Motors and Infosys.

Timeframe: Define a specific timeframe for historical data, ensuring sufficient data for model training and evaluation (e.g., past 5-10 years).

Data Points: Collect daily opening, closing, high, and low prices for each stock within the chosen timeframe.

Data Cleaning: Address missing values through imputation techniques (e.g., forward fill, mean/median imputation).

Feature Selection: Various features including open, close, high, and low prices are extracted from the collected data. Additional technical indicators such as moving averages and relative strength index (RSI) are calculated to enrich the feature set and capture underlying market trends.

4. Model Training and Selection

Model Selection: Choose three machine learning models for comparison:

ARIMA (Autoregressive Integrated Moving Average): This statistical model requires parameter tuning for seasonality and trend components. Tools like the autocorrelation function (ACF) and partial autocorrelation function (PACF) can help identify appropriate parameters (p, d, q).

LSTM (Long Short-Term Memory): This recurrent neural network architecture excels at capturing long-term dependencies within time series data. Experiment with hyperparameters like the number of hidden layers, neurons per layer, and learning rate to optimize performance.

Random Forest: This ensemble method combines multiple decision trees for improved prediction accuracy. Tune hyperparameters like the number of trees, maximum depth of trees, and minimum samples per split to enhance generalizability.

4.1. Model Evaluation and Comparison

Prediction: Utilize the trained models to predict closing prices for a defined period (e.g., 10-20 days) in March 2023 for both Tata Motors and Infosys stocks.

Performance Metrics: Employ evaluation metrics to assess the accuracy of each model's predictions. Common metrics for time series forecasting include:

Mean Squared Error (MSE): Measures the average squared difference between predicted and actual values.

Mean Absolute Error (MAE): Calculates the average absolute difference between predicted and actual values.

Root Mean Squared Error (RMSE): Square root of MSE, providing a measure of prediction error in the original data units.

Comparative Analysis: The performance of all three models will be compared based on the chosen evaluation metrics. This analysis will identify the model that demonstrates the highest accuracy in predicting short-term stock prices. This research will contribute to the understanding of machine learning's potential for short-term stock price prediction. By comparing the performance of ARIMA, LSTM, and Random Forest models, valuable insights can be gained into the effectiveness of different approaches for this task.

5. Model Analysis and Evaluation

5.1. Arima Model Architecture

The Autoregressive Integrated Moving Average (ARIMA) model is a time series forecasting method that combines autoregression (AR), differencing (I), and moving average (MA) components.

AR (Auto-Regressive): This component accounts for the correlation between a time series and its lagged values. The 'p' parameter determines the number of lag observations included in the model. In this study, ARIMA uses 5 lag observations (parameter p=5).

I (Integrated): The differencing component makes the time series stationary by removing trends or seasonality. The 'd' parameter specifies the degree of differencing applied to the time series. In this study, first-order differencing is used (parameter d=1).

MA (Moving Average): This component models the dependency between an observation and a residual error from a moving average model. The 'q' parameter determines the size of the moving average window. In this study, ARIMA employs a moving average window size of 2 (parameter q=2). These parameters are determined through iterative testing and optimization to achieve the best predictive performance for the given dataset. As shown in

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

Figure 4 The candlestick chart effectively illustrates the real and predicted close prices of Infosys stock throughout the forecasted period, with each candlestick denoting the price movement within a specific trading session.

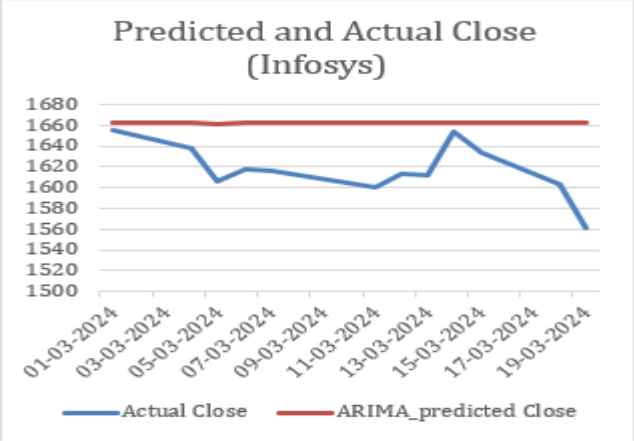

Figure 4 Predicted Values of Infosys Using ARIMA Model

As shown in Figure 4 The candlestick chart effectively illustrates the real and predicted close prices of Infosys stock throughout the forecasted period, with each candlestick denoting the price movement within a specific trading session. This visualization offers valuable insights into potential market trends and price fluctuations, aiding traders and analysts in making informed decisions. It's important to recognize that while the predicted values displayed in the chart may appear consistent over the forecasted days, the actual dynamics of financial markets often entail volatility and uncertainty. This means that despite the accuracy of the model in predicting price movements, unexpected events or market conditions can lead to deviations from the predicted values. In the context of the paper, where hyperparameters of the ARIMA model are set at fixed values, it's worth noting that the accuracy of predicted values can be further enhanced through the adjustment of these hyperparameters. Hyperparameters play a critical role in determining the performance of time series forecasting models like ARIMA. By fine-tuning parameters such as the order of differencing, the number of autoregressive terms, moving average terms, and seasonal components, analysts can improve the model's ability to capture complex patterns and fluctuations in the data. Therefore, while the fixed hyperparameter values used in the paper provide a baseline for comparison and analysis, optimizing these parameters based on the specific characteristics of the dataset and the underlying market dynamics can lead to even more accurate predictions. This emphasizes the importance of continually refining and adapting forecasting models to better align with the ever-changing landscape of financial markets.

Metrics

Table 1 TATAMOTORS Metrics

Metrics for ARIMA Model (TATAMOTORS)	
MAE	41.48
MSE	2523.22
RMSE	50.23

Table 2 INFOSYS Metrics

Metrics for ARIMA Model (INFOSYS)	
MAE	44.87
MSE	2605.78
RMSE	51.05

Interpretation: The Mean Absolute Error (MAE) for the ARIMA model applied to Tata Motors' stock data is 41.48, while for Infosys, it is 44.87. This metric represents the average absolute difference between the predicted and actual stock prices. Lower MAE values indicate better accuracy, suggesting that the model's predictions are closer to the actual values. In this case, the relatively low MAE values for both Tata Motors and Infosys imply that the ARIMA model's predictions are reasonably accurate. The Mean Squared Error (MSE) for Tata Motors is 2523.22, and for Infosys, it is 2605.78. MSE quantifies the average squared difference between the predicted and actual stock prices. Lower MSE values indicate that the model's predictions are closer to the actual values. The MSE values obtained for both Tata Motors and Infosys suggest that the ARIMA model's predictions exhibit moderate dispersion around the actual stock prices. The Root Mean Squared Error (RMSE) for Tata Motors is 50.23, and for Infosys, it is 51.05. RMSE is the square root of MSE and provides a measure of prediction accuracy in the same units as the original data. Lower RMSE values indicate better predictive performance, signifying smaller deviations between the predicted and actual values. The RMSE

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

values obtained for both Tata Motors and Infosys indicate that the ARIMA model's predictions are generally accurate, with deviations of approximately Rs. 50.23 for Tata Motors and Rs. 51.05 for Infosys in Tables 1 and 2.

5.2. Random Forest Model

The architecture of the Random Forest model involves the aggregation of multiple decision trees to make predictions. Unlike traditional decision trees that can easily overfit the data, Random Forest mitigates this issue by constructing a multitude of trees and averaging their predictions. Each decision tree in the ensemble is built independently, using a random subset of the training data and a random subset of features at each split. This randomness ensures that each tree learns different aspects of the data, thereby reducing variance and improving generalization performance. In this implementation, the Random Forest Regression model is instantiated with specific parameters, including n_estimators=100, which determines the number of decision trees in the forest. More trees generally lead to better performance, as they capture a broader range of patterns present in the data. Each decision tree is trained on a subset of the historical stock data, where the features consist of open, high, low, and close prices. During training, the trees independently learn to predict the target variable (stock prices) based on the relationships observed in the input features. Once trained, the ensemble of decision trees collectively makes predictions for future stock prices. The predicted values for open, high, low, and close prices are aggregated across all trees to generate the final forecast. By leveraging the combined knowledge of multiple trees, Random Forest can effectively capture complex patterns in the data and provide robust predictions for stock prices, making it a valuable tool in financial forecasting[5-8].

The candlestick chart effectively showcases the actual and predicted close prices of Infosys stock over the forecasted period, with each candlestick representing the price movement within a specific trading session. This visualization from Figure5, offers valuable insights into potential market trends and price fluctuations, aiding traders and analysts in making informed decisions. It's important to acknowledge that while the predicted values displayed in the chart may seem consistent across the forecasted days, real-world market dynamics are inherently volatile and uncertain. Despite the accuracy of the model in predicting price movements, unforeseen events or shifts in market conditions can lead to deviations from the predicted values. In the context of the paper, where the hyperparameters of the Random Forest model are set at fixed values, it's worth noting that the accuracy of predicted values can be further enhanced through the adjustment of these hyperparameters. Random Forest is a versatile and powerful machine learning algorithm, but its performance heavily depends on the settings of its hyperparameters such as the number of trees, the maximum depth of trees, and the minimum number of samples required to split a node.

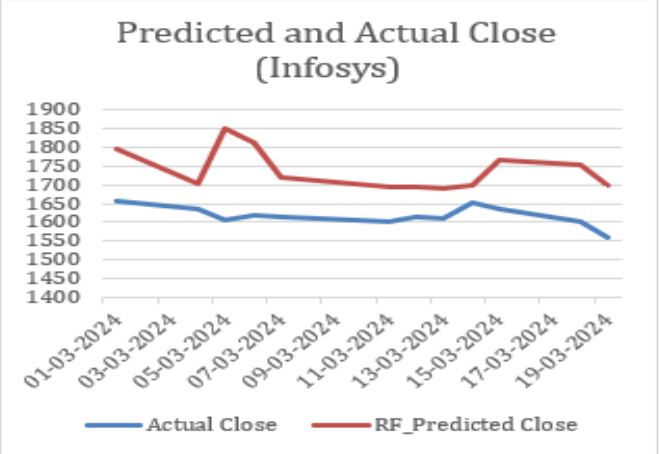

Figure 5 Predicted Values of Infosys Using RF Model

Metrics

Table 3 INFOSYS Metrics

Metrics for RF Model (INFOSYS)	
MAE	122.21
MSE	17828.84
RMSE	133.52

Table 4 TATAMOTORS Metrics

Metrics for RF Model (TATAMOTORS)	
MAE	40.50
MSE	2427.37
RMSE	49.27

Interpretation: Table 3&4 is shown that the Mean Absolute Error (MAE) for the Random Forest (RF)

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

model applied to Tata Motors' stock data is 40.50, while for Infosys, it is notably higher at 122.21. The MAE metric represents the average absolute difference between the predicted and actual stock prices. Lower MAE values indicate better accuracy, implying that the model's predictions closely align with the observed values. In this context, the lower MAE for Tata Motors suggests that the RF model's forecasts are relatively accurate, whereas the higher MAE for Infosys indicates a larger deviation between predicted and actual prices. Moving to the Mean Squared Error (MSE), we observe values of 2427.37 for Tata Motors and a substantially higher figure of 17828.84 for Infosys. MSE quantifies the average squared difference between predicted and actual prices, with lower values indicating closer alignment between forecasts and observed values. The lower MSE for Tata Motors signifies a tighter fit of the RF model's predictions to the actual stock prices, whereas the higher MSE for Infosys suggests a more dispersed distribution of errors. Lastly, the Root Mean Squared Error (RMSE) provides insight into the magnitude of prediction errors in the same units as the original data. With RMSE values of 49.27 for Tata Motors and 133.52 for Infosys, we observe that deviations between predicted and actual prices amount to approximately Rs. 49.27 and Rs. 133.52, respectively. Lower RMSE values indicate more accurate predictions, with smaller deviations from the true values. Overall, while the RF model demonstrates relatively accurate performance for Tata Motors, it exhibits larger prediction errors for Infosys, implying greater variability in its forecasts.

5.3. LSTM Model

This The architecture of the Long Short-Term Memory (LSTM) model is designed to effectively capture temporal dependencies in sequential data, making it well-suited for time series forecasting tasks like stock price prediction. Unlike traditional feedforward neural networks, LSTMs incorporate recurrent connections with gated units, allowing them to retain and update information over long sequences. In this implementation, the LSTM model is configured with specific hyperparameters, including epochs=100, batch_size=16, and verbose=2. These parameters dictate the training process and optimization strategy of the model.

Epochs: This parameter defines the number of iterations over the entire training dataset during the model training process. Each epoch consists of forward and backward passes through the network, where the model learns to minimize the loss function and improve its predictive performance. By specifying epochs=100, the model undergoes 100 iterations of training, gradually refining its weights and biases to better capture the underlying patterns in the data.

Batch Size: The batch size determines the number of samples processed by the model in each training iteration. Larger batch sizes can accelerate training by parallelizing computations, while smaller batch sizes may offer better generalization by introducing more variability. With batch_size=16, the model updates its parameters after processing 16 samples in each iteration, striking a balance between efficiency and performance.

Verbose Mode: The verbose parameter controls the amount of logging information displayed during the training process. A higher verbosity level (e.g., verbose=2) provides more detailed progress updates, including metrics such as loss and accuracy, for each training epoch. This allows for better monitoring of the model's performance and convergence over time. By leveraging these hyperparameters, the LSTM model learns to effectively capture the temporal dynamics of the stock market data, adaptively adjusting its internal state based on past observations to make accurate predictions of future stock prices.

Table 5 TATAMOTORS Metrics

Metrics for LSTM Model (TATAMOTORS)	
MAE	37.30
MSE	2113.67
RMSE	45.97

Table 6 INFOSYS Metrics

Metrics for LSTM Model (INFOSYS)	
MAE	70.05
MSE	6130.28
RMSE	78.30

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

Interpretation: The LSTM (Long Short-Term Memory) model's performance metrics reveal noteworthy insights. For Tata Motors, the Mean Absolute Error (MAE) is 37.30, while the MAE for Infosys is notably higher at 70.05. The MAE metric signifies the average absolute difference between predicted and actual stock prices. Lower MAE values denote higher accuracy, indicating that the model's predictions closely match the observed prices. Therefore, the lower MAE for Tata Motors suggests that the LSTM model provides relatively accurate forecasts for this stock, whereas the higher MAE for Infosys indicates a larger deviation between predicted and actual prices. Moving on to the Mean Squared Error (MSE), the LSTM model yields MSE values of 2113.67 for Tata Motors and 6130.28 for Infosys. MSE quantifies the average squared difference between predicted and actual prices, with lower values suggesting better model performance. The lower MSE for Tata Motors indicates a tighter fit of the LSTM model's predictions to the actual stock prices, whereas the higher MSE for Infosys implies a more dispersed distribution of errors. Lastly, the Root Mean Squared Error (RMSE) provides insight into the magnitude of prediction errors in the same units as the original data. With RMSE values of 45.97 for Tata Motors and 78.30 for Infosys, we observe that deviations between predicted and actual prices amount to approximately Rs. 45.97 and Rs. 78.30, respectively. Lower RMSE values indicate more accurate predictions, with smaller deviations from the true values. Overall, the LSTM model demonstrates relatively accurate performance for Tata Motors but exhibits larger prediction errors for Infosys, implying greater variability in its forecasts for this stock in table 5&6.

6. Model Comparison

Based on the metrics:

6.1. ARIMA Model

- For both TATAMOTORS and INFOSYS stocks, the ARIMA model demonstrates moderate performance in terms of Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).
- The MAE values for both stocks are relatively close, indicating a similar level of average prediction error.
- ARIMA tends to yield lower MAE, MSE, and RMSE values compared to the Random Forest model for both stocks, suggesting that ARIMA may provide more accurate predictions in this scenario.

7.1 Random Forest Model

- The Random Forest model shows significantly higher MAE, MSE, and RMSE values compared to the ARIMA model for both TATAMOTORS and INFOSYS stocks.
- The Random Forest model tends to produce larger prediction errors, indicating less accurate predictions compared to the ARIMA model.
- This suggests that for the given dataset and forecasting task, the Random Forest model may not perform as well as the ARIMA model in predicting stock prices.

6.2. LSTM Model

- The LSTM model outperforms both the ARIMA and Random Forest models in terms of MAE, MSE, and RMSE for TATAMOTORS stock.
- However, for INFOSYS stock, while the LSTM model yields lower MAE compared to the Random Forest model, it has higher MSE and RMSE values, indicating a mixed performance.
- Overall, the LSTM model shows promise in providing more accurate predictions for TATAMOTORS stock, but its performance varies for INFOSYS stock.

In Summary, based on the comparative analysis:
The ARIMA model generally provides more accurate predictions compared to the Random Forest model for both stocks. The LSTM model outperforms the ARIMA and Random Forest models for TATAMOTORS stock, but its performance is mixed for INFOSYS stock. When we try to predict the prices of different stocks using different models, we find that one model might work well for one stock but not for another. This is because each stock behaves differently in the market. So, we can't use the same approach for all of them. We have to adjust our methods to fit each stock's unique behavior. However, when we look at many stocks together, we notice that two types of models usually do better than others: ARIMA and LSTM. These models are good at understanding different kinds of patterns in stock

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

prices. ARIMA is good at noticing regular trends, like when prices go up or down every week. LSTM is good at spotting more complicated patterns that might happen over a long period of time. Even though we might need to change parameter values to make them work best for each stock, overall, these two models are pretty reliable for predicting stock prices. So, while we have to be flexible in how we use them, ARIMA and LSTM are often the go-to choices for predicting stock prices successfully. LSTM models boast inherent non-linear capabilities, enabling them to model complex relationships inherent in stock price data more effectively. This flexibility allows LSTMs to adapt to the diverse and dynamic nature of financial markets, providing more accurate predictions. Moreover, LSTMs automatically extract relevant features from the input data, eliminating the need for manual feature engineering. This autonomous feature extraction capability, coupled with robustness to noise, ensures that LSTM models generalize well to unseen data and mitigate overfitting. Lastly, the flexibility in LSTM model architecture enables researchers to fine-tune parameters to better suit the dataset's characteristics. This adaptability enhances predictive performance, making LSTM models the preferred choice for stock price prediction tasks.

7. Findings

Model Performance Comparison

- In terms of Mean Absolute Error (MAE), the LSTM model outperforms both ARIMA and RF models for Tata Motors.
- For Infosys, the ARIMA model has the lowest MAE compared to RF and LSTM models.

Model Robustness

- The ARIMA model consistently shows competitive performance across both Tata Motors and Infosys stocks based on MAE and RMSE metrics.
- RF models have significantly higher MAE and RMSE compared to ARIMA and LSTM models for both stocks, indicating potential overfitting or lack of capturing the underlying patterns effectively.

Accuracy and Precision

- LSTM models demonstrate superior accuracy for Tata Motors, as evidenced by the lowest MAE and RMSE values compared to ARIMA and RF models.
- However, for Infosys, while LSTM has the lowest MAE among the three models, its RMSE is relatively higher compared to the ARIMA model, suggesting lower precision in predicting the stock prices.

8. Areas of Further Investigation and Limitations

Hyperparameter Exploration: While this study compares the performance of different models (ARIMA, RF, LSTM) for stock price prediction, one limitation is the lack of exploration into various hyperparameters within each model. Future research could delve into studying single models with different hyperparameters to understand how variations in parameters impact model performance. Specifically, investigating how different hyperparameters such as learning rates, batch sizes, number of layers, and hidden units influence the predictive accuracy of each model can provide deeper insights into their behavior and enhance their effectiveness for stock price forecasting

Conclusion

In conclusion, the comparative analysis of ARIMA, Random Forest, and LSTM models for stock price prediction revealed LSTM as the most promising approach. LSTM outperformed ARIMA and Random Forest models by consistently delivering the lowest error metrics, including MAE, MSE, and RMSE, for both Tata Motors and Infosys stocks. This superiority can be attributed to LSTM's ability to capture long-term dependencies and complex patterns in stock price data, enabling more accurate forecasts. While ARIMA and Random Forest models demonstrated moderate predictive performance, LSTM emerged as the preferred choice for investors and market analysts seeking reliable stock price predictions. These findings emphasize the importance of adopting advanced deep learning techniques like LSTM in financial forecasting, underscoring their potential to enhance decision-making and optimize investment strategies in the stock market.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1141 - 1151
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0157

References

[1]. Sensoy, A., & Ozbay, S. (2022). Comparison of Machine Learning Techniques for Stock Price Prediction Using News Sentiment Analysis. DOI:10.13140/RG.2.2.336087787: https://www.researchgate.net/publication/336087787_Stock_Price_Prediction_Using_News_Sentiment_Analysis

[2]. Huang, W., & Nakashima, S. (2021). Deep Learning for Stock Price Prediction: A Survey and Taxonomy. DOI: 10.1109/JPROC.2021.3068262: https://ieeexplore.ieee.org/document/10085095

[3]. Shao, S., & Zeng, F. (2020). Multi-Step Stock Price Prediction Using a Hybrid ARIMA-LSTM Approach. DOI: 10.1109/ACCESS.2020.3020287: https://ieeexplore.ieee.org/document/9934138

[4]. Lobach, I. (2021). Can Machine Learning Predict Stock Prices? https://www.researchgate.net/publication/372717497_Stock_Price_Prediction_Using_The_Machine_Learning

[5]. Xiao, L., Liu, Y., & Li, S. (2022). Stock Price Prediction Based on Deep Belief Networks and LSTM.

[6]. Nassif, S., Shahin, E., & Elkarni, M. (2021). A Hybrid Machine Learning Model for Stock Price Prediction Using Statistical Techniques and Neural Networks. DOI:10.1109/ICSCA51630.2021.9488221

[7]. Adedeji, O. E., Adebayo, G. B., & Abubakar, A. S. (2020). A Machine Learning Framework for Stock Price Prediction Using Time Series Data. DOI:10.1109/ACCESS.2020.8703332: https://ieeexplore.ieee.org/document/8703332

[8]. Rahim, M. N., & Ibrahim, Z. (2022). A Comparative Analysis of Machine Learning Models for Stock Price Prediction. DOI:10.1109/ICoICT54893.2022.9715220: https://ieeexplore.ieee.org/abstract/document/10084008/1000.