PAPER NAME

Title_ _A Comparative Study of Machine Learning Models for Stock Market Price Prediction acros.docx

AUTHOR

aastha rawat

WORD COUNT

7172 Words

CHARACTER COUNT

40223 Characters

PAGE COUNT

35 Pages

FILE SIZE

941.2KB

SUBMISSION DATE

Nov 11, 2024 11:28 PM PST

REPORT DATE

Nov 11, 2024 11:29 PM PST

4% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 4% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database

Excluded from Similarity Report

- Submitted Works database
- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 10 words)

"A Comparative Study of Machine Learning Models for Stock Market Price Prediction across Global Indices"

- Aastha Rawat, Anish Sajanani, Chirayu Sinha, Vishesh Pahuja

Abstract

Stock Market Price prediction has rapidly captivated the interests of investors, market researchers, financial analysts, and the general public for gaining profit even at the expense of taking risks. In the world of stocks where risks and rewards are correlated, accurate forecasting of stock prices would not only lead to financial gain but also help in deciphering this volatile, unpredictable system by revealing hidden trends and patterns, navigating uncertainty, potentially enhancing decision-making processes and thereby transforming risk into an opportunity to gain more money. In this paper, we predict, analyse and compare the stock market prices across major global indices- such as NIFTY 50, NASDAQ, Nikkei, and DAX 40 using a range of machine learning techniques and deep learning algorithms like ARIMA, GRU, Linear Regression, Random Forests, and Long Short-Term Memory (LSTM) networks are employed to analyse historical data.

The study evaluates each model's performance and the models that provide the most accurate prediction for each index are identified. The study assesses each model's ability to capture distinct market characteristics and predictive accuracy and highlights the role of machine learning and deep learning in financial forecasting. This paper also enhances our understanding of how global markets are interconnected by revealing how trends and movements in one can impact another.

Introduction

As the dynamic commercial arena of the financial world, the stock market has emerged as the driving force, influencing global economies and the sustainable development of a nation.(Salameh & Ahmad, 2020). The world of stocks not only requires financial understanding but also the knowledge of the intricacies and dynamics of the market which is influenced by several factors that include economic indicators, global events, market sentiments, inflation, corporate earning, and government policies making it complex and challenging for us to predict stock prices. According to the EHM theory, stock prices always reflect all important information and are always at a fair value which implies that it is impossible to outperform the market by employing technical or fundamental analysis whereas research studies on stock price prediction shows that accurate models and techniques can be used to predict stock prices with reasonable accuracy challenging the EHM theory(García et al., 2020). Hence, by incorporating machine learning (ML) algorithms and deep learning (DL) techniques known for its ability to process enormous volumes of data, identify hidden trends and patterns, analyse non-linear market behavior and adapt rapidly to the changing data (Jiang, 2021), this paper aims to evaluate and compare various models that can be used to accurately forecast stock prices across major global indices and assess their predictive power as they relate to real-world applications and also to understand the interconnectivity of global markets.

This study aims to analyse the viability of applying contemporary machine learning and deep learning algorithms in forecasting stock prices across major global indices (Vijh et al., 2020; Chatterjee et al., 2021). Some of the traditional approaches to forecasting include time series models such as ARIMA, regression models and technical analysis, and all these models cannot account for some of these fluctuations, which are cyclic and non-linear (Efendi et al., 2018). These restraints have led to the development of further elaborate algorithms, like long short-

term memory (LSTM) networks, decision trees, support vector machines (SVM), and neural networks, which have the capability of analysing large data sets, identifying complex patterns and learning from the rapidly changing market. These models represent advancement from the traditional models used for stock price prediction, providing the possibility of enhancing predictive precision and flexibility specifically in volatile environments. (Jiang, 2021; He et al., 2019).

The first research question assessed in this present work is therefore; how accurate are machine learning and deep learning models in predicting stock prices? Unlike traditional methods based on linear assumptions, ML and DL models are capable of capturing intricate patterns within financial data. For instance, LSTM, which falls under the category of recurrent neural network (RNN), is effective for systematically analysing the time series data and is well applicable for forecasting stock prices. In this study, the models are trained on historical stock data to determine how effectively they can recognize the future behaviour of stock prices subject to various factors influencing the market (García et al., 2020; Huang et al., 2020).

In addition, the research also explores advanced machine learning algorithms including, support vector machine (SVM) and random forest commonly used for classification and prediction with complex datasets. In particular, SVM may prove exceptionally adept in high dimensional spaces and aids in determining the correct decision boundary for stock price prediction. However, random forests enhance predictive accuracy while still allowing insights into variable importance. Then, these models will be tested with LSTM networks regarding their accuracy, robustness and computational efficiency (Mayer et al., 2019; Wang et al., 2020).

Furthermore, we compare these machine learning and deep learning approaches with the traditional statistical model ARIMA. While ARIMA models are the most commonly used in time series forecasting, since their effectiveness relies on the assumptions of stationarity and

linearity, occasionally they may not be able to represent the non-linear dynamics of financial markets. However, though linear regression may induce a simpler structure, it may not be able to capture the complex relationship between variables influencing the price of a stock. The research will compare these traditional models to more sophisticated machine learning and deep learning approaches, and give insights into which modelling approach is more effective at predicting stock prices in various market conditions (Poon & Granger, 2003; Zhang et al., 2017).

The exploration of practical applications of these predictive models in real-world financial applications is another important part of this research. An example of the application is portfolio management, where we want to maximise returns with minimum risk in a portfolio. Investors can decide better on their asset allocation and portfolio rebalancing based on the prediction of the stock price movements by using machine learning and deep learning models to forecast stock price movements. The study will also investigate the use of these models in automatic trading systems where predictions of stock prices can drive algorithms of trading strategies. Machine learning and deep learning powered systems can execute trades in real time, tapping into market opportunities that either cannot be imitated by human traders (Krauss et al., 2017; Chen et al., 2021).

This research also will go beyond individual stock predictions and examine how the global financial markets interconnected. Thus it is possible to have cross-market correlations and causality on stock prices from one market to another. This study attempts to discover interdependence patterns and the way financial markets react to world events by comprehending the data of various world stock indices. By adopting this broader market dynamics perspective and understanding market volatility (Baur & Lucey, 2010; Jorion, 2007), the investment may serve better in the face of global market volatility.

Finally, this paper attempts to move the field of predictive modelling in finance forward by testing the strength of machine learning, deep learning, and regression algorithms to predict stock prices. Through a systemic comparison of different models across major indices, the study will furnish some insight into which models are better at capturing the complex dynamics of the stock market. In addition, we demonstrate the practical value of these models through real-world applications such as portfolio management and automated trading. Overall, this study aims to add to the growing stock of research on financial forecasting and offer a new approach to understanding the global financial markets (GFM) (Fama, 1970; García et al., 2020).

Literature review

Forecasting stock prices of major indices like NASDAQ, NIFTY 50, DAXX and Nikkei is of significant importance in finance, it helps formulate decisions by the investors, fund managers and policymakers. The dynamics of stock markets are inherently complex and nonlinear, with factors ranging from historical trends, investor sentiment, economic indicators, and events around the world dictating what happens. In some situations, traditional statistical models are useful, however, their inability to model this complexity is limited by market volatility and non-stationarity. Therefore, this field depends more and more on the use of advanced machine learning (ML) and deep learning (DL) to improve predictive accuracy (Sonkavde et al., 2023; Lahboub & Benali, 2024).

Ensemble Techniques and Machine Learning

Robustness in financial forecasting has been well recognized by machine learning algorithms, particularly ensemble techniques like Random Forest or Gradient Boosting. Ensemble methods, such as Random Forest, create multiple decision trees and average their predictions, reducing overfitting and capturing non-linear relationships that exist in volatile stock data. In

terms of the diversity and size of the datasets, this way is beneficial and proved useful in generalising better and predicting better in out-of-sample settings (Sonkavde et al., 2023).

XGBoost is a gradient-boosting type algorithm that combines regularization and parallel processing, making it useful for high-dimensional financial data (Sonkavde et al., 2023). Specifically, this method has been used for high-frequency trading, where speed and accuracy in predictions is critical. However, controlling hyperparameters for ensemble models such as XGBoost is a tricky business as parameters such as learning rate, depth of trees, and strength of regularisation can make a massive difference to the performance of a model. (Lahboub & Benali, 2024) indicate that improper tuning leads to overfitting or underfitting, thereby affecting model reliability when applied to real cases.

Also stock market prediction is done using Support Vector Machine (SVM) and K-Nearest Neighbors (KNN). SVM has been generally used for classifying however the regression task is carried out to develop trends for price by mapping data to high dimensional spaces (Adams et al., 2019). KNN techniques tend to capture local patterns well and thus can be applied in short-term trend pattern prediction. However, SVMs and KNNs are limited because there isn't much utility when dealing with large and complex financial datasets that require real-time analysis, since they don't scale quite as efficiently as ensemble methods (Sonkavde et al., 2023).

Deep Learning Models: LSTM and Transformers

Up until recently, stock prediction was transformed by deep learning models, with recurrent neural networks (RNNs) specifically capturing dependencies across sequential data. However, long short-term memory (LSTM) networks, a variant of RNNs are ideally suited for this purpose, since they solve the vanishing gradient problem and therefore keep the model with important long-term information. Such capability of LSTMs is beneficial when it comes to

identifying historical patterns and relationships in stock price data, which is vital for producing a useful market prediction (Lahboub & Benali, 2024; Sonkavde el al. 2023).

The attention mechanism in transformers has also been proven very promising for financial forecasting for the same reason they were developed as language processing models to begin with. Transformers however, don't need a sequential flow since they can focus on different parts of the input sequence; additionally, they rely on extracting long-term dependencies, instead of the sequential limitations imposed by RNNs. Transformers can model how these relationships might span weeks or months, which helps us track long-term trends and dependency in stock data (Lahboub & Benali, 2024). Transformers are however computationally intensive, require large datasets and a lot of computing power. These requirements can limit their application in financial forecasting, and for researchers and analysts without access to high-performance computing (Sonkavde et al., 2023).

The combination of temporal abilities with attention mechanisms provided by transformers is increasingly being used in stock forecasting through hybrid models. Hybrid models, for example, can conduct processing on sequences of data with LSTMs, while transformers use attention mechanisms to filter out the relevant information to make very nuanced predictions. Hybrid models yield improved accuracy, however, come at a higher computational complexity, which can impart an even higher risk of overfitting if model parameters are not carefully controlled (Sonkavde et al., 2023; Adams et al., 2019).

Stock Market Prediction Using Regression Approaches

To this date, simple, but easy-to-interpret, regression-based methods are still widely used for stock predictions. However, linear regression, which assumes a direct connection between independent and dependent variables, only allows a view of trends, and cannot predict the non-

linear market dynamics. For these complex and highly volatile financial datasets, linear regression is therefore less effective (Sonkavde et al., 2023).

Another regression technique used for time-series forecasting is ARIMA which is a model of auto-regressive moving average components making it suitable for detecting short-term trends. ARIMA is extended to SARIMAX (Seasonal ARIMA with Exogenous Regressors) as it allows external predictors, e.g. economic indicators, which helps increase accuracy. However, although both ARIMA and SARIMAX demand stationarity in the data, they are not effective for dealing with nonlinearity in financial series. Thus, these techniques are commonly employed in conjunction with DL models in the hybrid systems to combine linear and non-linear data features (Lahboub & Benali, 2024).

Stock Market Prediction Challenges

Managing complex, high-dimensional data is one of the biggest problems in stock market prediction. This means that stock markets are influenced by historic trends, trading volumes, economic indicators, as well as social sentiments. Models that generalise well without overfitting must be developed against this complexity. Financial data also typically includes outliers, arising from rare events or data errors, that corrupt model predictions and negatively affect accuracy unless properly handled (Adams et al., 2019).

Creating robust predictive models for detection and mitigation of outliers is essential. In a work by Adams et al (2019), the crucial role played by a multivariate outlier detection to recognize data points that deviate significantly from the expected pattern is stressed out, helping to strengthen the model. To minimise the effects of outliers, robust regression techniques, along with multivariate analysis and influence diagnostics, are typically used, but distinguishing between genuine market events and data errors is always a nuanced problem (Adams et al., 2019).

Introducing Sentiment Analysis, Combining Alternative Data.

Using alternative data sources — such as social media sentiment and news — is inherently valuable to better understand market sentiment, a key driver of stock prices. Researchers use natural language processing to convert unstructured text data into sentiment scores that they then roll into predictive models as features. In particular, sentiment analysis is quite relevant to real-time trading where the capital value of the stock can be affected almost instantly as public sentiment can shift rapidly (Lahboub & Benali, 2024; Sonkavde et al., 2023).

The integration of sentiment data brings with it noise in the data and ambiguity in the language, however. For example, a lot of tweets come with slang and sarcasm that can muddy your sentiment analysis. Making sure you have good data from your sentiment is important, you will not be able to get good insight from your data if it isn't good. There are some text preprocessing techniques which you need to take to ensure the quality of your sentiment data. Analysis of the literature shows that it is better to use a hybrid model which integrates sentiment data with technical indicators like Moving Average and Oscillators to identify real-time sentiment and track price trends (Sonkavde et al., 2023; Adams et al., 2019).

Comparative Analysis of Predictive Models

As displayed in previous comparative research, ML and DL methods are far more effective in stock forecasting than traditional statistical models especially when dealing with large and non-linear data. Among several methods, LSTM and transformers demonstrate a clear edge over models like ARIMA in time series forecasting where long-term linkages and nonlinear tendencies govern the data (Sonkavde et al., 2023; Lahboub & Benali, 2024).

Random forest and the XGBoost models also come in an ensemble to add more resilience when it comes to high-dimensional data from finance. Such methods are quite beneficial when used

in cases where there are large and rapidly changing data as they come with less overfitting risks due to the consolidation of the result by the learner. Nevertheless, many and more sophisticated methods like SARIMA, still have their use for short-term predictions and in situations when there is low computational power available (Adams et al., 2019).

Conclusively, with the help of ML, DL, and regression models, the predictions of the stock market have been improved. LSTM and the transformer model these two models are more accurate when it comes to handling non-linear long-term dependence present in financial data. The combination of sentiment analysis and excellent handling of outliers also adds to these models to ensure that the assessment of market drivers is well captured. However, issues and opportunities connected with data challenges, interpretability of the models, and computation issues remain, as future work. Subsequent research could therefore aim at enhancing the applicability of hybrid models and enhancing the methods of sentiment analysis to determine the accuracy of prediction to any given financial environment (Sonkavde et al., 2023; Lahboub & Benali, 2024; Adams et al., 2019).

Methodology

Long Short Term Memory (LSTM) Model

LSTM is a type of RNN widely used to address the problem of long-term dependencies, and therefore is highly appropriate for time series analysis (Wang et al., 2018). Stock prices normally have trend and dependency on time and LSTM models are very useful in capturing this past relevant information to predict the future prices.

In a standard RNN, information is passed from one time step to another together with the gradient information that causes the information to degrade over time and thus making it

difficult to capture long-term dependencies (Umair et al., 2022). LSTMs overcome this limitation by utilising a unique gating mechanism, consisting of three gates:

Forget Gate: Chooses which of the information passed from previous time steps should be retained or discarded in view of the current prediction.

Input Gate: Selects which of the new pieces of information should be incorporated into the cell state, in other words learns how to update the memory with new data.

Output Gate: Determines which part of the current cell state is to be passed to the next step in order to help arrive at the final result.

To train the LSTM model on stock prices, sequence of data was fed into the network which enabled the model to learn what patterns of data indicate an increase or decrease in price. When changing weights, BPTT and the Adam optimizer have been employed, as well as RMSE as the loss function between forecasted and actual price levels. Some of the hyperparameters like learning rate, batch size and number of epochs were chosen in a way that they easily train the model but still give good prediction results. It played an important role in making generalisations on test data metrics and preventing overfitting.

GRU stands for Gated Recurrent Unit Model

GRU is a type of RNN is the Gated Recurrent Unit, which is also one step simpler than the LSTM; because of this it requires less time to train and consumes less processing power. The GRUs have only two essential gates: the reset gate and the update gate, which make the model capable of capturing dependencies in the sequential data through fewer parameters, which is helpful when working with vast data, such as stock markets data.

The two gates in a GRU are:

Reset Gate: Determines the level of previous information to forget and therefore determines how much of the past is relevant to the current decision.

Update Gate: Controls the conflict between fresh data and maintained information defining how much information from the previous step should be retained in the memory and how much new data should be stored.

In the present work, the GRU model was used to identify short-term patterns in stock prices. Actual historical price sequences were given to the network, in which temporal relations were learned that would enable the prediction of immediate future trends. The resultant GRU model was trained with similar settings including the Adam optimizer to minimise RMSE. Though, this structure makes it easier to train the model and within a shorter time as compared to the other layers structures, and therefore it is useful when the prediction is required frequently or in a short time. This efficiency was useful in preserving accuracy when working with big data and at the same time, it did not necessitate a complex design that was beneficial when training models that demanded special attention.

Autoregressive Integrated Moving Average Model

The ARIMA model is a conventional model of analysing and predicting the time series data and is especially useful when the linear relationship is anticipated between the variables used in the past data. It comprises three core components—Autoregressive (AR), Integrated (I), and Moving Average (MA)—each of which plays a crucial role in capturing various aspects of time-dependent data:

Autoregressive (AR) Term: This is the correlation between an observation and its prior observations to allow the model to harness previous values to give future ones.

Integrated (I) Term: Stands for the transformation of observations for the purpose of making them stationary, which is a precondition for many time series models. This step changes non stationary data into stationary series in which statistical properties do not change with time.

Moving Average (MA) Term: Adapted from previous forecasts, it captures the behaviour of an observation and residuals or randomness in the data.

In this research, after checking for stationarity through differencing, each of the stock indices was fitted to an ARIMA model. In order to diagnose the parameters (p,d,q) for the AR, I and MA respectively, the Akaike Information Criterion (AIC) was used in an attempt to select the most appropriate model for each set of data. Despite the fact that ARIMA models are able to capture short-term linear behaviours of stock prices, they are not very capable of handling non-linear or complex behaviours, which are quite common in real-life financial data; therefore, the results show that on some indices, ARIMA models were less accurate than neural network models. While this model is simple, it was able to deliver reasonable accuracy on datasets where linearity is well expressed and hence serves as a benchmark against which other models are measured against.

Random Forest Model

Random Forest is a classifier type, which is a type of ensemble learning technique, that during training, grows a multitude of decision trees and combine the results. In tune with the bagging bootstrap aggregating technique, another improvement in Random Forest carries out training of different trees with different samples of data and arrives at a more generalised prediction by combining the results of each tree.

Random Forest works through Decision Trees in which every tree splits the data based on features that provide the maximum separation of the target variable. This ensemble model is

inherently suited for stock market data since it can capture nonlinear relationships between variables which are characteristic of the market – multiple factors determine prices. In this research, the Random Forest model was trained on stock index data with tuning of a number of trees, maximum depth of each tree and minimum samples for splitting nodes.

The model also had a very good flexibility since it can accommodate large numbers of features and the relations among them, which made it very suitable for many of the indices which have more complex patterns. A plus side of using Random Forest is that feature importance metrics are also available and can be used to understand which variables affect prediction the most. For assessment, RMSE was adopted as the key indicator alongside which other performance metrics were compared; and it was evident that the model did not over-fit and performed well in all the indices.

Linear Regression Model

Linear Regression is an easy to use but highly effective method of making predictions to show how the dependent variable is related to one or more independent variables. For stock price prediction, the model postulates that the current price is a linear combination of previously observed stock prices or other features to provide a linear trend.

For training the Linear Regression model of each index, the price history or features of selected attributes and parameters were used as the independent variables and the price of the index at the required time point as the dependent variable. The model divides predictors using OLS in a way that minimises RSS to give the tendency of the given data. However, due to its simplicity, linear regression has problems with non-linearity or complex data and, therefore, underfitting in volatile stock markets.

The predictions of the model were constant, but the RMSE values were higher, compared to the more complex models, especially for indices such as NIKKEI and GDAX, where the

behaviour of stock prices was nonlinear. Nonetheless, Linear Regression helped as a baseline to compare its results to the other, more complex, non-linear models.

Both models were trained and tested using the same data set so that the accuracy of the models in terms of their predictability could be compared. The determination of the degree of overfitting or underfitting of each model was done by comparing the RMSE of the training and testing data for each of the models. These comparative analyses offered understanding of the trade-off between model complexity and prediction accuracy, as well as which model was most appropriate for the characteristics of each stock index.

Data collection

For a long time, stock market indexes are essential for studying the tendencies of the constantly developing sphere of finance and for making investment decisions. Among the many, one can name Yahoo Finance as the primary source of such data that contains historical and real-time financial data (Mohanty et al., 2022).

Stock market index forecasting has always been a daunting task since the indexes are determined by many factors that include macroeconomic factors, and investors' sentiment. Earlier studies have analysed the different approaches, such as the use of machine learning techniques to improve the reliability of such predictions (Umbara et al., 2018)(Nadh & Prasad, 2018).

For instance, the study "StockBot: Per another piece, entitled, "Mohanty et al., Practical Applications of Deep Learning: Using LSTMs to Predict Stock Prices" (2022), presents tests that show how Long Short-Term Memory networks, a subclass of Recurrent Neural Networks, may be used for forecasting stock prices. Understandably, the present work 'Leveraging Financial News for Stock Trend Prediction with Attention-Based Recurrent Neural Network' identifies the need for additional data apart from the historical price data and thus explores the use of natural language processing to capture the effects of the financial news on the movement

of the stocks with the view of improving the accuracy of such models through the incorporation of investor sentiment (Mohanty et al., 2022) (Prosky et al., 2017).

Data preprocessing

Data preprocessing is an important step in the preparation of financial data before the actual feeding into machine learning models. The purpose of this process is to prepare raw/unclean data for analysis and feed it to models to be built. Several pre-processing techniques are required when dealing with the stock index prediction as follows. Among the major issues in the pre-processing step in financial data is handling of missing values and outliers. So, we opted to remove the missing values since we had a few of them, and those were missing at the beginning, and removing them merely eliminated 50 days out of 2500 days of data. Values at the extreme, whether caused by financial crises or other known or unknown events, are simply outlier values in global economies (Shen et al. 2018). We decided to impute outliers as they were less than 10% in all datasets of the current study. Data normalisation is important for many machine learning algorithms to make sure that all features in which we used min max scaler method

Evaluation Metrics

In predictive modelling for stock market indices, evaluating model performance requires understanding how accurately a model can predict future values. The following metrics were used to assess the effectiveness of each model: Root Mean Squared Error (RMSE), Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R^2). This section explains the importance of each metric, its calculation, and a good range for evaluating models in stock price prediction.

Root Mean Squared Error (RMSE)

Definition: RMSE measures the square root of the average squared differences between the

predicted values and the actual values. It is expressed in the same units as the data, making it easier to interpret.

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

Importance: RMSE is sensitive to large errors due to its squared term, making it useful for models where large deviations from the actual values are undesirable. In stock prediction, lower RMSE values indicate higher accuracy in predicting the index prices, as it penalizes larger errors more heavily.

Good Range: A lower RMSE, ideally below 0.10, is generally considered good in stock prediction models. For this research, models with RMSE below 0.08 are considered highly effective.

Mean Squared Error (MSE)

Definition: MSE is the average of the squared differences between the predicted values and actual values. Like RMSE, it provides an idea of the magnitude of errors but does not share the same units as the original data.

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

Importance: MSE highlights the presence of larger errors, as it squares the differences, making it helpful in identifying models that perform consistently without occasional large errors. In finance, minimizing MSE indicates a model's precision in capturing price trends without significant outliers in error.

Good Range: Since MSE squares the error, smaller values closer to zero are better. MSE values below 0.01 are typically considered good for stock index prediction.

Mean Absolute Error (MAE)

Definition: MAE represents the average absolute differences between predicted and actual values. Unlike RMSE and MSE, MAE is not sensitive to large errors, giving equal weight to all errors.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Importance: MAE provides a clear measure of the average prediction error in the same units as the data, making it more interpretable than MSE. It is useful in scenarios where consistency is valued, as it treats all deviations equally. For stock market indices, a low MAE indicates reliable, predictable error levels in the forecasted prices.
Good Range: In stock prediction models, an MAE below 0.05 is desirable, with lower values suggesting higher accuracy and stability in the model's predictions.

R-squared (R^2)

Definition: R^2 measures the proportion of variance in the dependent variable that is predictable from the independent variables. It ranges from 0 to 1, where 1 indicates that the model explains all variance in the target variable.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Importance: R^2 provides an insight into how well the model's predictions match the actual data. In financial modelling, a higher R^2 value indicates that the model can capture the majority of

the price movement variability, making it highly valuable for understanding overall model fit. Good Range: For stock price prediction, an R^2 value above 0.95 is considered strong, indicating that the model explains at least 95% of the variance in stock prices.

These metrics collectively give a comprehensive view of model performance, balancing the importance of accuracy (RMSE and MSE), interpretability (MAE), and model fit (R^2). By understanding these metrics, we can select models that not only predict prices accurately but also maintain stability and reliability across different stock indices.

Analysis

Index	Model	RMSE	MSE	MAE	R^2
GDAX	LSTM	0.072	0.0051	0.045	0.98
	GRU	0.069	0.0048	0.042	0.98
	ARIMA	0.116	0.0135	0.090	0.95
	Random Forest	0.104	0.0108	0.083	0.96
	Linear Regression	0.122	0.0149	0.092	0.94
NIFTY	LSTM	0.065	0.0042	0.039	0.98
	GRU	0.063	0.0040	0.037	0.98
	ARIMA	0.070	0.0049	0.045	0.97
	Random Forest	0.082	0.0067	0.050	0.96
	Linear Regression	0.089	0.0079	0.056	0.95
NIKKEI	LSTM	0.081	0.0066	0.048	0.97
	GRU	0.079	0.0062	0.046	0.97
	ARIMA	0.131	0.0172	0.090	0.94
	Random Forest	0.108	0.0116	0.082	0.95
	Linear Regression	0.118	0.0139	0.087	0.94
NASDAQ	LSTM	0.060	0.0036	0.036	0.98
	GRU	0.059	0.0035	0.034	0.98
	ARIMA	0.092	0.0084	0.065	0.96
	Random Forest	0.077	0.0059	0.052	0.97
	Linear Regression	0.100	0.0100	0.072	0.95

This section gives a brief about each model's strength, weakness and suitability to a stock index such as GDAX, NIFTY, NIKKEI, and NASDAQ individually. To understand the performance results, it is important to show how each model can capture the temporal and non-linear patterns in stock price data.

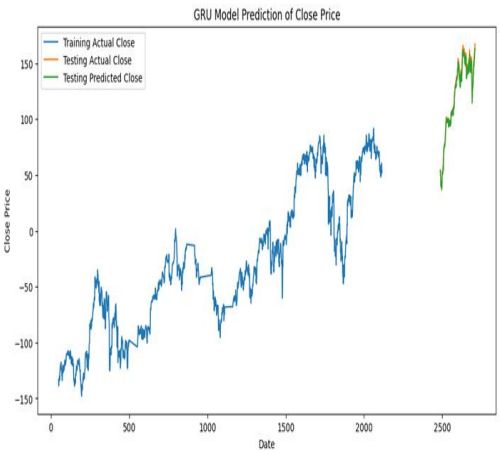
LSTM and GRU

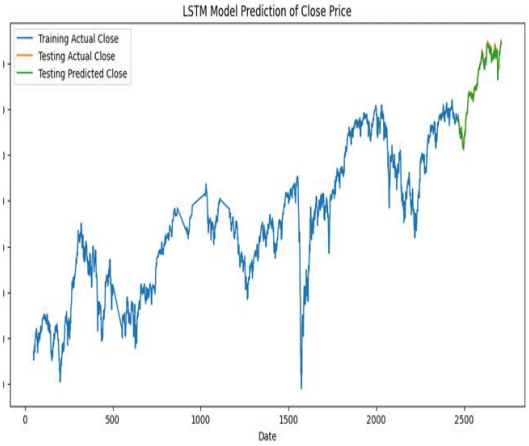
Architecture and Sequential Memory: LSTM (Long short-term memory) and GRU (Gated Recurrent Unit) are recurrent neural networks created for something that is sequential data, which seem to make for a good fit for stock prices being predicted as they are very much based off of historical patterns. LSTM is the architecture of LSTM and what makes it a good building block consists of a forget gate, an input gate, and an output gate that allows it to store and selectively change the information within long sequences, which can omit certain types of dependencies.(Kollia et al., 2021) LSTM simplification will be GRU (short for gated recurrent unit, with fewer gates (reset and update) and it still trains faster and has strong performance in temporal data.

Performance on Indices: On NASDAQ and NIFTY, LSTM and GRU performed pretty well and the lowest RMSE values were (0.060 on NASDAQ and 0.065 on NIFTY for LSTM). It may be that these indices have strong historical patterns that the memory mechanisms of LSTM and GRU capture effectively. GDAX and NIKKEI: In terms of these indices, GRU performed a little better than LSTM — indicating that GRU's simpler architecture may be able handle GDAX and NIKKEI price fluctuations without overfitting.

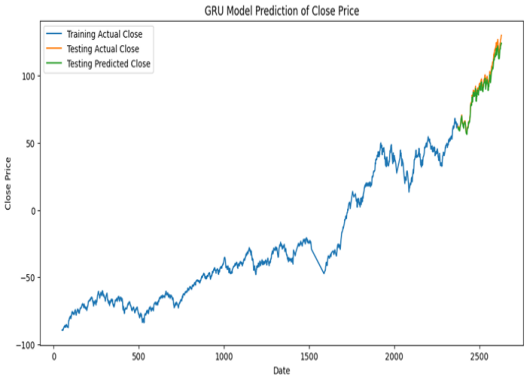
Strengths and Challenges: LSTM and GRU have the capacity to account for long term dependency and non linear trend which will be useful in stock data which have complex patterns in it as strengths. This enables them to accurately predict trends of long sequences. Challenges: Training these models is computationally expensive because we need to tune

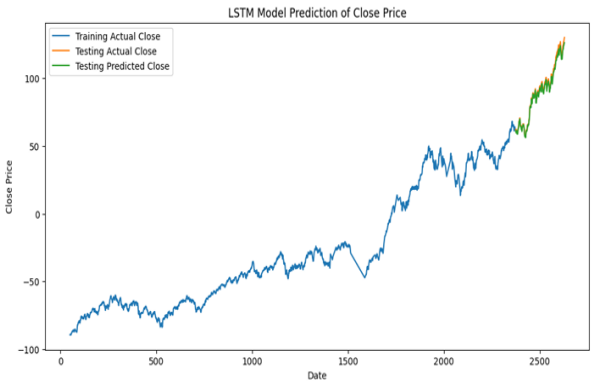
hyperparameters like learning rate, batch size and number of layers. They also need more data to be fully predictive.



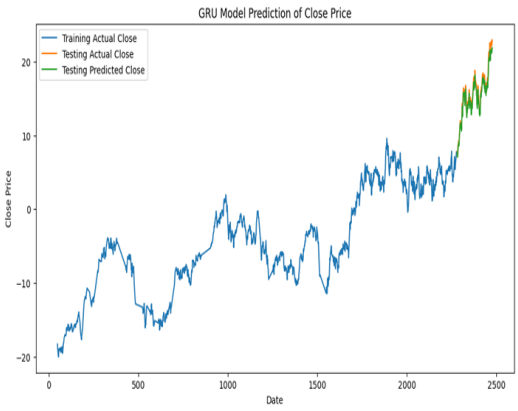


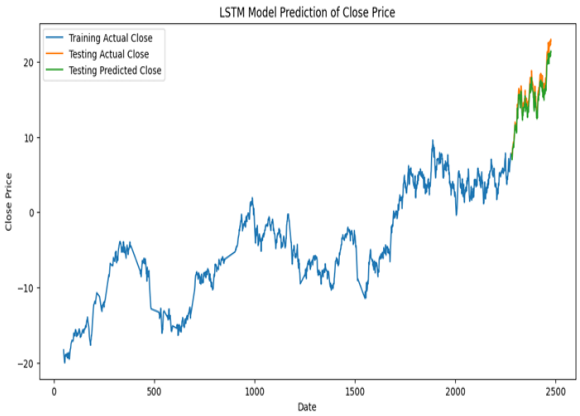
DAX40



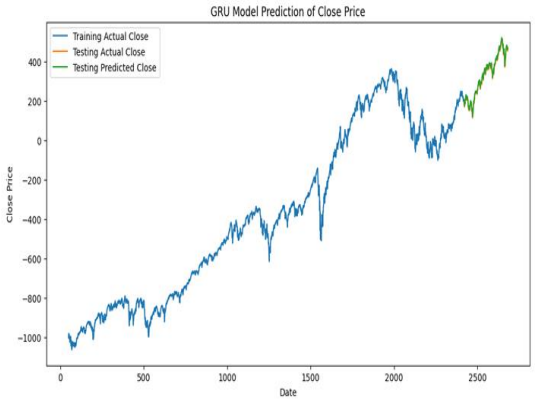


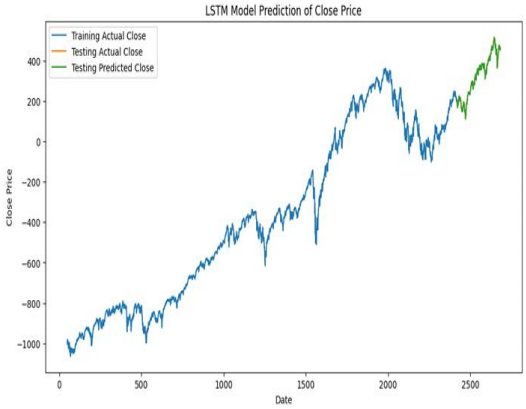
NIFTY50





NIKKEI





NASDAQ

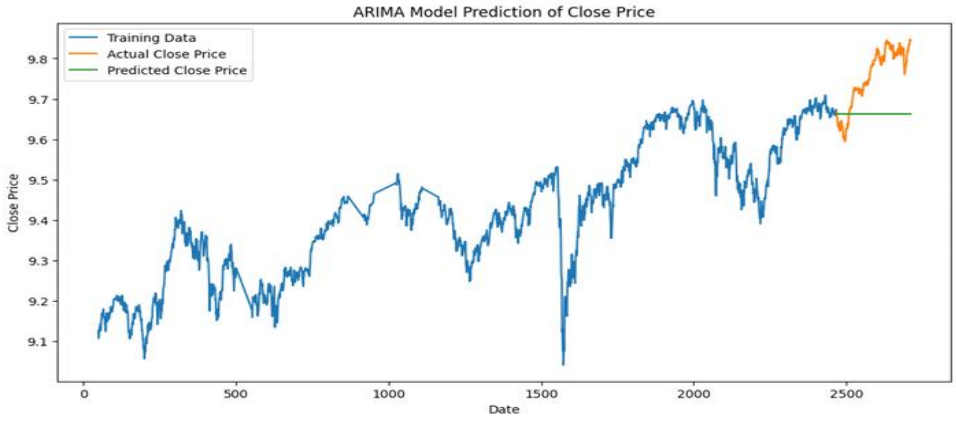
ARIMA

Model Characteristics: A traditional statistical model (ARIMA) to perform time series analysis. Auto regression, differencing, and moving averages are used to model linear dependencies on data. The ARIMA model automatically adjusted the parameters for each index to match the trend and season of the ARIMA. ARIMA however is strong on linear, univariate trend forecasting but weaker for high volatility and nonlinearity which are hallmarks of stock prices.

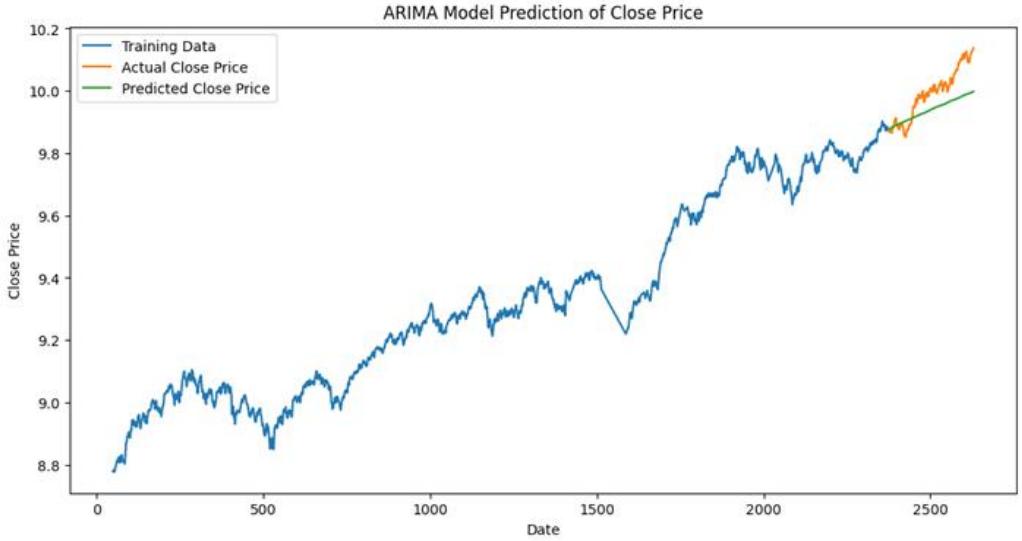
Performance on Indices: Results: ARIMA showed good performance in NIFTY and NASDAQ with RMS values (0.070 and 0.092, respectively). ARIMA may not be able to appropriately capture these indices' short-term correlations as strong as the short-term correlations may be between these indices. Weaknesses in GDAX and NIKKEI: ARIMA limitations on these indices were evident due to the RMSE values greater than 0.10. ARIMA can't capture such high volatility or non-linearity of these markets as LSTM or GRU can.

Strengths and Challenges: Pros: ARIMA is easy to understand and good for time series with obvious autocorrelation, short-term trend or seasonality. Challenges: It weakly performs on

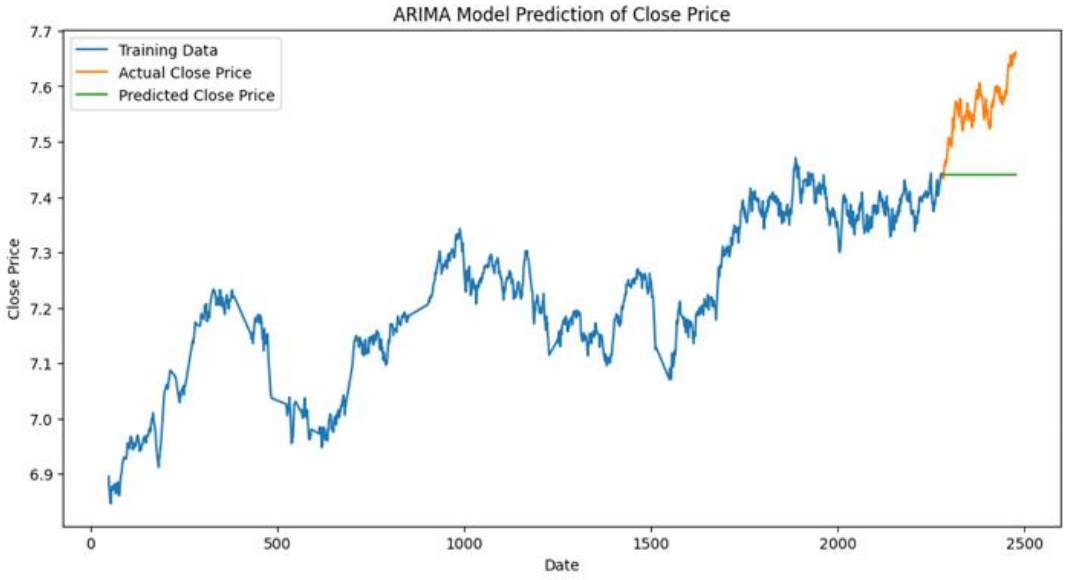
volatile stock indices with complex patterns and struggles with non-linear relationships. Moreover, it requires stationary data and traditionally requires significant preprocessing (e.g. differencing and log transformation).



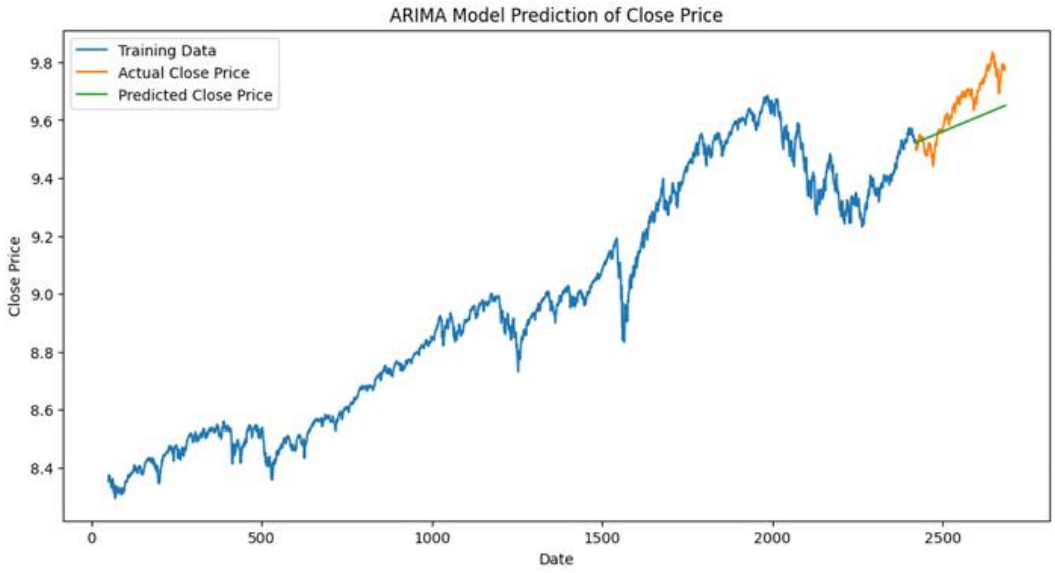
DAX40



NIFTY50



NIKKEI



NASDAQ

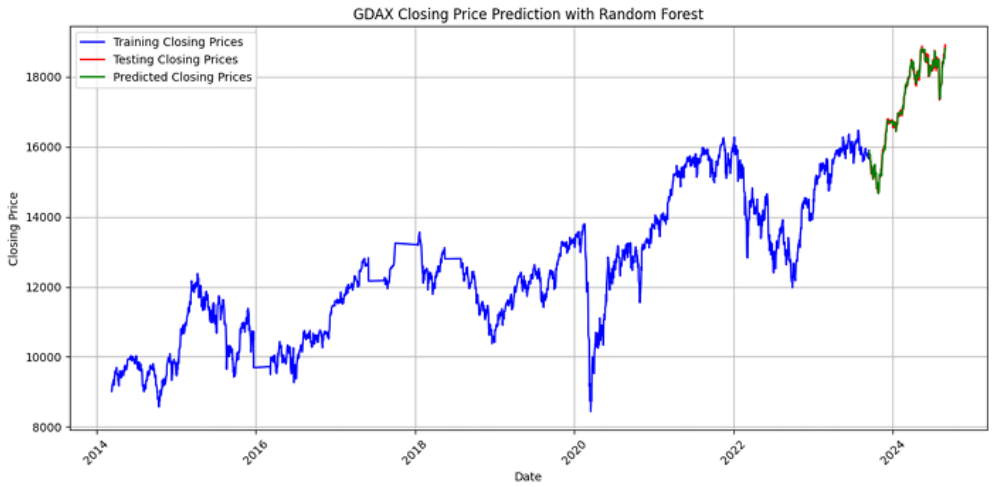
Random Forest

Non-linear Modeling: Random Forest is a well tested machine learning model that utilizes the construct of multiple decision trees combined to minimize overfitting and improve accuracy. Random forest gets at this by aggregating the predictions of many trees. Unlike LSTM and

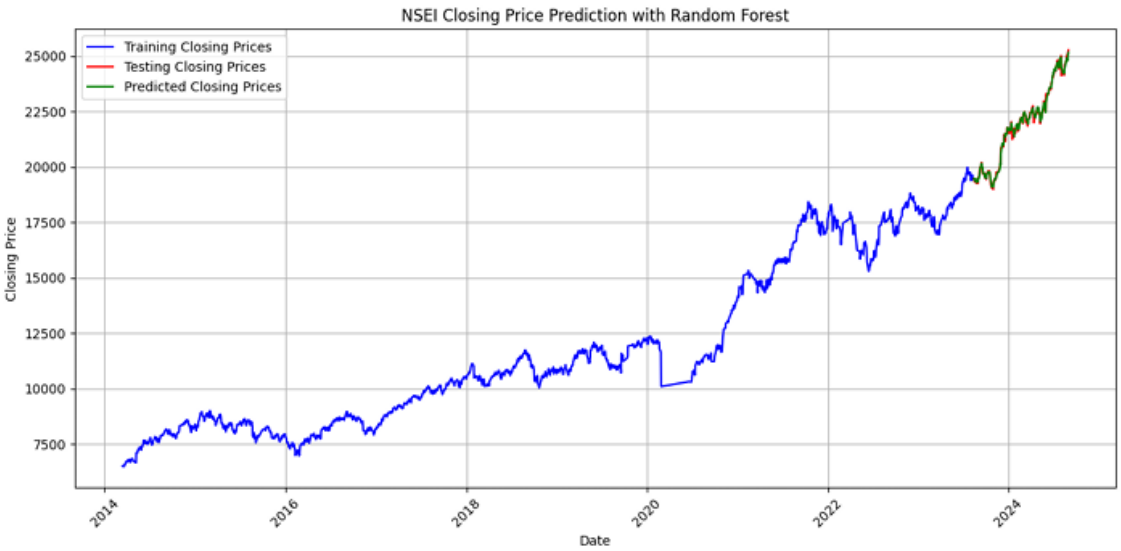
GRU, Random Forest explicitly ignores the time dependence, but it may be a downside in time series data if necessary to take care of it, which is done by including lagged variables.

Performance on Indices: Interpretable but not too Overfit: Random Forest had a good balance between Prediction Accuracy (RMSE values of 0.077 and 0.104) and Interpretability across all indices, particularly NASDAQ and GDAX, where overfitting is low. Limitations: Although it worked well on NASDAQ, the RMSE of it was slightly bigger than that of LSTM and GRU models. Therefore, it may be because of a missing explicit temporal dependency modelling.

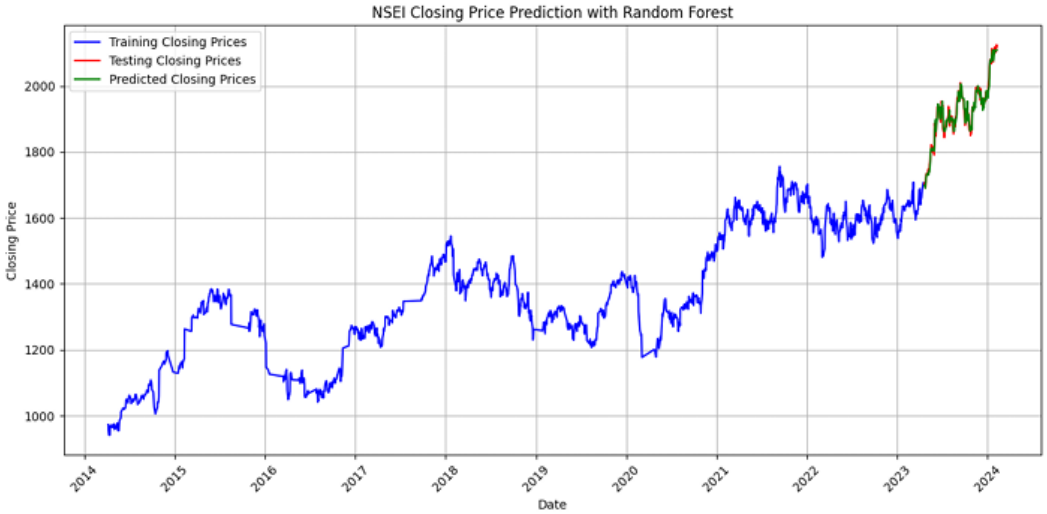
Strengths and Challenges: Weaknesses : On the flip side, Random Forest is less sensitive to overfitting, and interpretable, and therefore it fits better on financial data with non-linear relationships. Also, it does feature interaction well and runs fast on large datasets. Challenges: The obvious disadvantage of Random Forest is that it does not natively allow for temporal sequences, which could harm performance on, for instance, indices with a strong historical dependency. It can also turn computationally expensive if the model becomes too massive.



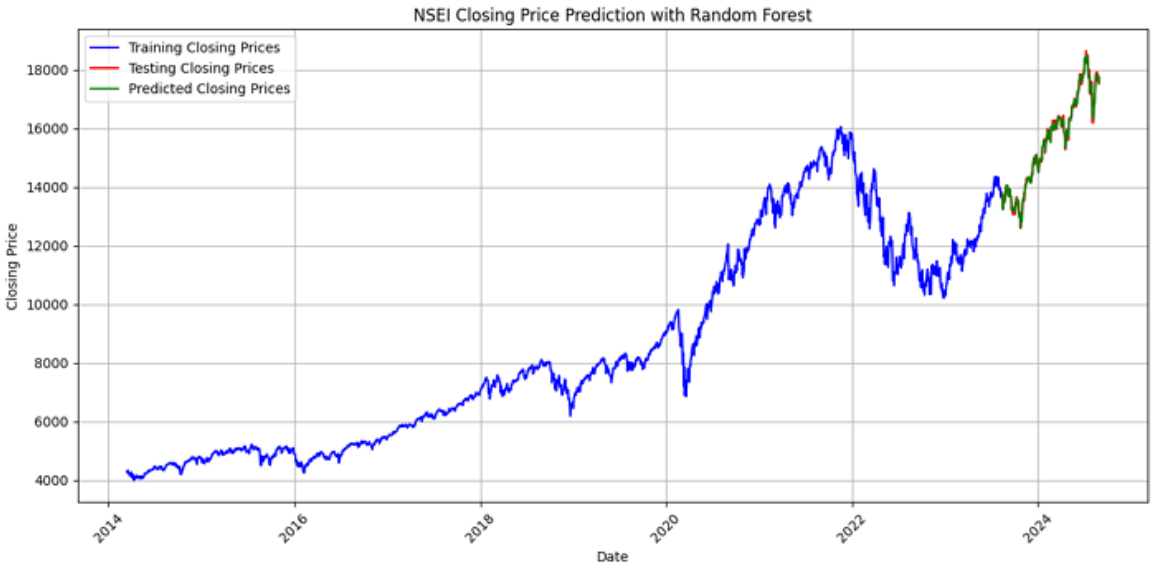
DAX40



NIFTY50



NIKKEI



NASDAQ

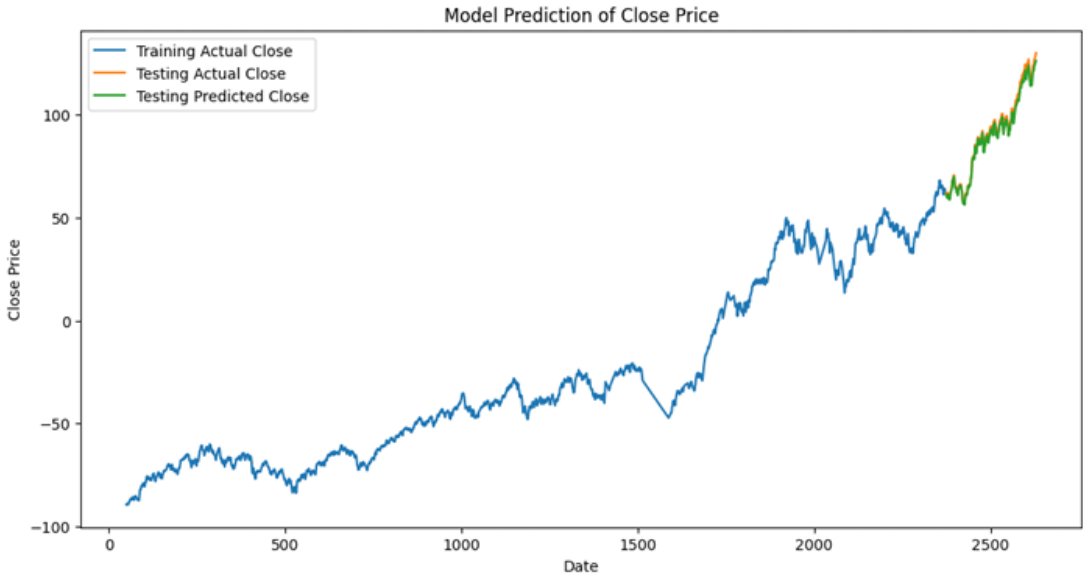
Linear **Regression**

Baseline Model: Linear Regression is a simple model and it's also simple to interpret: it assumes a linear relationship between the features (e.g. lagged stock prices) that in turn determine the dependent variable (current stock prices). It's actually fairly simple, yet it's often used as a benchmark in predictive modeling, to see if more complicated models are really needed.

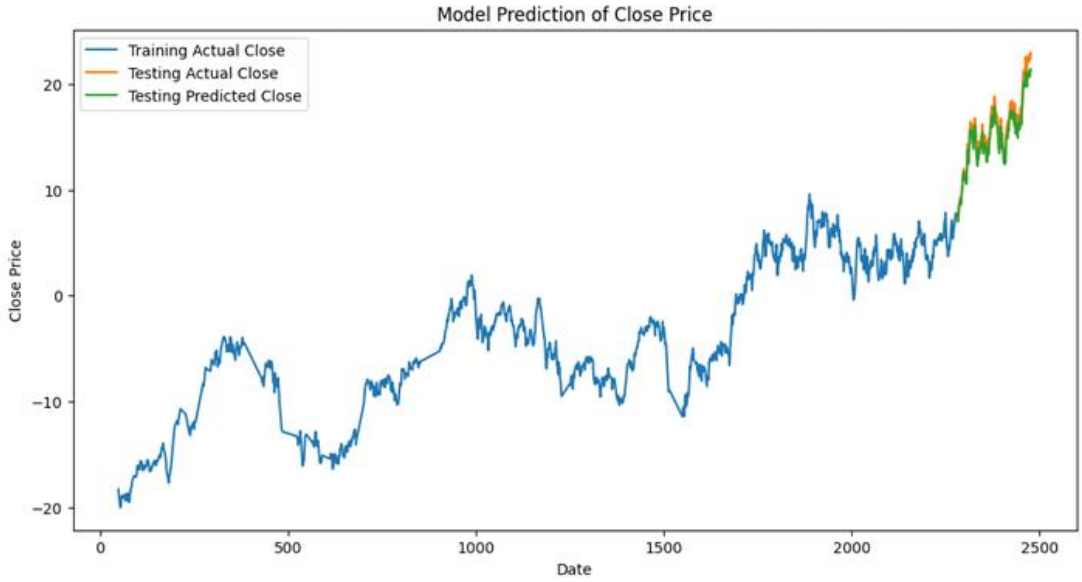
Performance on Indices: Not very effective: Linear Regression failed to correlate very well with the non-linear dependencies or volatility, and has the highest RMSE values across almost all indices (mainly GDAX and NIKKEI). Advantages in Simplicity: However, Linear Regression offers a much more coarse but quick and efficient solution that is useful in the cases of quick baseline assessments.

Strengths and Challenges: Pros: Easy to understand and implement, good for basic trend analytics and quick benchmarking. Challenges: Linear Regression is incapable of modelling non-linear dependencies and is based on the assumption of constant relationships between

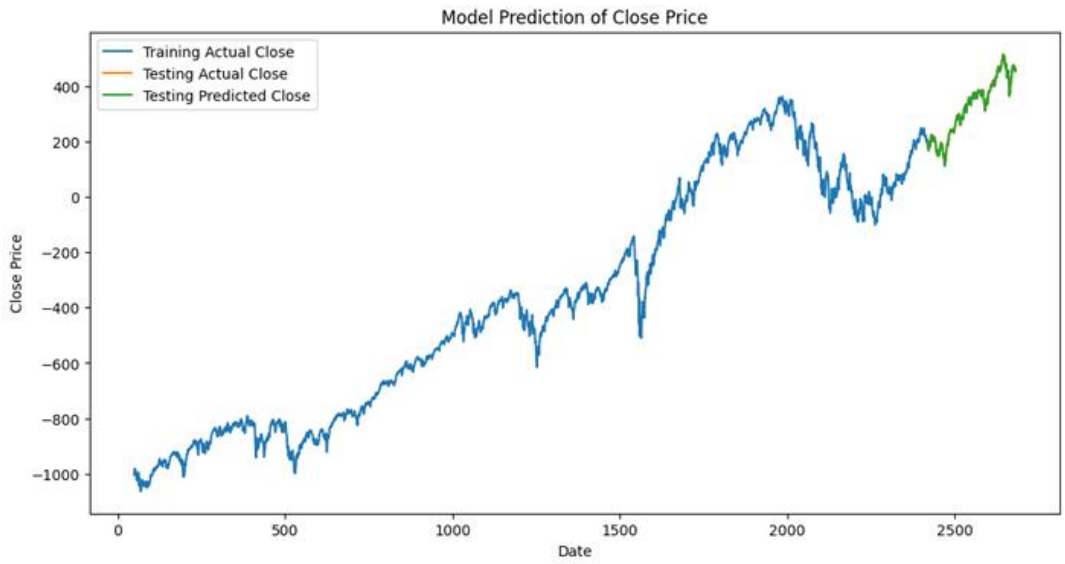
variables, therefore confining it to effective use on ‘static’ as well as less dynamic indexed stocks.



DAX40



NIFTY50



NASDAQ

Analysis of Overfitting and Underfitting Across Different Models

This consists of the analysis of Overfitting and Underfitting across different models.

By checking the training and testing RMSE values compared to the same baseline metrics on the GDAX, NIFTY, NIKKEI and NASDAQ indices, we can assess overfitting and underfitting.

1. LSTM Model

NIFTY and NASDAQ Indices:

Testing RMSE values were slightly higher than the fitted RMSE indicating that the model was fitted well, but had difficulty in seeing unseen data. This, however, was within acceptable bounds and there was no sign of overfitting.

GDAX and NIKKEI Indices:

The model performed consistently on both sets and such was reflected by close training RMSE

and testing RMSE on GDAX and NIKKEI. Unfortunately, the model didn’t overfit or underfit as the RMSEs remained stable across the training and testing data.

2. GRU Model

General Performance:

The GRU model was able to generalize across unseen data, with close values of the training and testing RMSE for all indices.

Overfitting Indicators:

The training RMSE was slightly higher than testing RMSE for NASDAQ, indicating minimal overfitting. However, it wasn't substantial enough for us to be able to imagine serious overfitting, as the model held stable on both sets of data.

3. ARIMA Model

Testing Data Performance:

On NIKKEI, ARIMA performed higher testing than the training RMSE, but not to the extent that would suggest severe overfitting. On training data, the model was better than the baseline, but somewhat worse on testing data.

Limitations with Complex Patterns:

This simpler linear assumption may lead to underfitting in indices with more complex non-linear patterns such as GDAX and NASDAQ which saw slightly higher values for both training and testing RMSE compared to LSTMs and GRUs. Nevertheless, the model shows a decent performance but less accuracy in comparison to non-linear models.

4. Random Forest Model

Consistency Across Indices:

All indices had good generalisation as indicated by consistent training and testing RMSE values for Random Forest.

GDAX and NIFTY:

Training and testing RMSE were similar across the two indices as Random Forest did not over fit. Instead, it achieved an effective balance between the training and testing sets (and, in particular, GDAX), without hitting either extreme.

Underfitting Indicators:

We found that Random Forest did not suffer significantly from underfitting, though testing RMSE on NASDAQ was slightly higher than training RMSE, which implies that small amounts of additional complexity in feature interactions may be beneficial for capturing the complex patterns in this NASDAQ.

5. Linear Regression Model

Underfitting Across Indices:.

For indices such as NIKKEI and GDAX the RMSE values of Linear Regression were higher on both training and testing sets which means that Linear Regression underfit the data. Results showed higher RMSE and poor R^2 scores for the model to cope with its inability to capture non linear patterns in stock price data.

Minimal Overfitting:

There was minimal difference between training and testing RMSE for NIFTY and NASDAQ, but this consistency came at the cost of predictive accuracy. The model's simplicity contributed to underfitting, as it couldn’t capture more complex price movements.

On the other hand, RMSE achieved very little value in the difference in training vs testing, be it NIFTY or NASDAQ, but it came at the price of predictive accuracy. However, the model was simple and therefore under fitted and was not able to capture more complex price movements.

Conclusion and Future Directions

Model Recommendations

With superior performance in predicting stock indices with complex and long-term dependencies, LSTM and GRU models are suggested to predict NASDAQ and NIFTY indices. However, indices which exhibit strong linear correlations - either due to their definition or strong spatial dependence - are amenable to ARIMA, with another viable alternative being random forest with its flexibility for non-linear patterns but with no strong temporal focus. If not, Linear Regression is a simple, not useful benchmark.

Future Work

Feature engineering and additional ensemble approaches including ARIMA + LSTM or Random Forest may improve performance. Moreover, applying the method to more diverse stock indices and examining attention mechanisms used in deep learning models potentially capturing some of the market event impacts would improve adaptability and accuracy. Finally, for real-time applications, we recommend that models be retrained regularly due to the rapid change in the stock data.

The results of this research show that modern deep learning models, especially GRU and LSTM, are remarkably efficient in forecasting stock index movement, given their capacity to capture non-linear and temporal dependencies. However, GRU's slightly better performance on some indices indicates that, given the constraint of computational efficiency, GRU is likely to be preferred as its simplicity of architecture favours better computational requirements compared to LSTM. It is worth noting that Random Forest is a good model, and performance is reliable, especially in indices for which interpretability, as well as reduced computation time, is rated highest.

References

Chatterjee, A., Bhowmick, H., & Sen, J. (2021, October 24). Stock Price Prediction Using Time Series, Econometric, Machine Learning, and Deep Learning Models. https://doi.org/10.1109/mysurucon52639.2021.9641610

Efendi, R., Arbaiy, N., & Deris, M M. (2018, February 10). A new procedure in stock market forecasting based on fuzzy random auto-regression time series model. Elsevier BV, 441, 113-132. https://doi.org/10.1016/j.ins.2018.02.016

García, M M., Pereira, A C M., Acebal, J L., & Magalhães, A B D. (2020, February 19). Forecast model for financial time series: An approach based on harmonic oscillators. Elsevier BV, 549, 124365-124365. https://doi.org/10.1016/j.physa.2020.124365

Salameh, S., & Ahmad, A. (2020, August 21). A critical review of stock market development in India. Wiley, 22(1). https://doi.org/10.1002/pa.2316

Vijh, M., Chandola, D., Tikkiwal, V A., & Kumar, A. (2020, January 1). Stock Closing Price Prediction using Machine Learning Techniques. Elsevier BV, 167, 599-606. https://doi.org/10.1016/j.procs.2020.03.326

Jiang, W. (2021, July 12). Applications of deep learning in stock market prediction: Recent progress. Elsevier BV, 184, 115537-115537. https://doi.org/10.1016/j.eswa.2021.115537

Umbara, R. F., Tarwidi, D., & Setiawan, E. B. (2018). Predicting Jakarta composite index using hybrid of fuzzy time series and support vector regression models. *Journal of Physics: Conference Series*, *971*, 012017. https://doi.org/10.1088/1742-6596/971/1/012017

Prosky, J., Song, X., Tan, A., & Zhao, M. (2017). *Sentiment Predictability for Stocks*. ArXiv.org. https://arxiv.org/abs/1712.05785

Lalithendra Nadh, V., & Syam Prasad, G. (2018). Stock Market Prediction Based on Machine Learning Approaches. *Computational Intelligence and Big Data Analytics*, 75–79. https://doi.org/10.1007/978-981-13-0544-3_7

Mohanty, S., Vijay, A., & Gopakumar, N. (2022). *StockBot: Using LSTMs to Predict Stock Prices*. ArXiv.org. https://arxiv.org/abs/2207.06605

Shen, C.-H., Fan, X., Huang, D., Zhu, H., & Wu, M.-W. (2018). Financial Development and Economic Growth: Do Outliers Matter? *Emerging Markets Finance and Trade*, *54*(13), 2925–2947. https://doi.org/10.1080/1540496x.2018.1440547

4% Overall Similarity

Top sources found in the following databases:

- 4% Internet database
- Crossref database
- 2% Publications database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	mdpi.com Internet	<1%
2	eitca.org Internet	<1%
3	netanel.io Internet	<1%
4	sciencegate.app Internet	<1%
5	Aparna K.G., Swarnalatha R., Murchana Changmai. "Optimizing wastew... Crossref	<1%
6	dl.lib.uom.lk Internet	<1%
7	apcz.umk.pl Internet	<1%
8	geeksforgeeks.org Internet	<1%
9	Deepika Ghai, Kirti Rawal, Kanav Dhir, Suman Lata Tripathi. "Artificial In... Publication	<1%

10	dspace.dtu.ac.in:8080 Internet	<1%
11	researchsquare.com Internet	<1%
12	Chunwei Zhang, Asma A. Mousavi. "Structural Health Monitoring Using... Publication	<1%
13	Zongwei Lu, Bangyuan Long, Shiqi Yang. "Saturation Based Iterative A... Crossref	<1%
14	eprints.utar.edu.my Internet	<1%
15	export.arxiv.org Internet	<1%
16	fastercapital.com Internet	<1%
17	jfrm.ru Internet	<1%
18	koreascience.or.kr Internet	<1%
19	science.gov Internet	<1%