C O V E N T R Y
U N I V E R S I T Y

Faculty of Engineering, Environment and Computing

School of Computing, Mathematics and Data Science

MSc. Data Science

7150CEM
Data Science Project

ENHANCE THE STOCK MARKET PRICE PREDICTION USING MACHINE LEARNING TECHNIQUES

Author: PANKAJ YADAV

SID: 13578147

Supervisor: Dr. MOHAMMED AHMED

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in DATA SCIENCE

Academic Year: 2023/24

Declaration of Originality

I declare that this project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Statement of copyright

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialise products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see www.coventry.ac.uk/ipr or contact ipr@coventry.ac.uk.

Statement of ethical engagement

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking)

Signed: Pankaj Yadav Date: 31/07/2024

Please complete all fields.

First Name:	PANKAJ
Last Name:	YADAV
Student ID number	13578147
Ethics Application Number	P177746
1st Supervisor Name	Dr. MOHAMMED AHMED
2nd Supervisor Name	Dr. PENGPENG HU

This form must be completed, scanned and included with your project submission to Turnitin. Failure to append these declarations may result in your project being rejected for marking.

Abstract

Stock price prediction is an active research area in the field of finance and trading that assists traders and analysts in decision making, which further enhances trading strategies and profits. In an attempt to make predictions for the closing prices of the NSE Bank Index, this project looks at the efficacy of several machine learning and deep learning models. The major focus of the work is to identify suitable models which can capture the complicated structures and the temporal dependencies characterizing the financial time series. Due to the reasons of achieving high model accuracy and ensuring the reliability, feasibility and broad applicability of the model, the project uses data preprocessing, cross validation and hyperparameters tuning. It is evident that the employed Linear Regression as well as LightGBM algorithms are the most accurate followed by Random Forest and Ridge Regression. Despite the high computational cost, DL models are useful in capturing the temporal dependencies and their learning is relatively stable if set-up properly. The project also satisfactorily solves issues related to reproducibility by utilizing the seeds in random processes and defining the setting of a computing environment, which assures that the model returns similar performances in multiple runs. The qualitative measures that are applied in measuring the models' performances include Root Mean Squared Error (RMSE), R-squared (R^2) as well as the bias. The analysis and representation of the forecast help decision-makers for proper decision-making sit. The project proves the applicability of the state-of-the-art machine learning algorithms in the domain of financial prediction and offers a comprehensive approach that can be successfully used in future projects, dealing with real-time prediction of the stock prices, for enhancing of the financial analytic field with useful knowledge and effective tools.

Keywords : Random Forest, Decision Tree, Linear Regression, Ridge Regression, Support Vector Regression, LightGBM, Recurrent Neural Network, Long Short-Term Memory, Hyperparameter Tuning, Cross Validation, Bias, Root Mean Squared Error, R-square.

TABLE OF CONTENTS:

Table of Tables:

Table of Figures:

Acknowledgements

I would like to extend my deepest appreciation to everyone who helped me to successfully finished this project.

Firstly, I am deeply thankful to my project supervisor, Dr. Mohammed Ahmed, for his continuous guidance, invaluable feedback, and unwavering support throughout this project. His expertise and insights have been instrumental in shaping the direction and quality of this work.

I also extend my gratitude to Library of Coventry University for providing the resources and facilities required for this project. The access to the necessary tools has been crucial in enabling the comprehensive analysis conducted in this study.

Lastly, I would like to acknowledge the authors and researchers whose work has significantly contributed to the foundation of this project. Their pioneering studies and publications have been a vital source of knowledge and inspiration.

1. Introduction

Stock market price forecasting has been one of the most significant issues of finance study all throughout these years because of the possible immense pecuniary gains and implications to the stability of the market. In this regard, most of the conventional statistical methods and models are often rather limited in their ability to identify the long, non-linear structures inherent in financial data. The introduction of machine learning gives a disruptive solution to these problems through use of advanced algorithm to enhance stock price forecasts.
To make choices quickly, a financial research analyst must have access to all relevant and meaningful data. There are many variables that affect stock performance, and they should all be carefully investigated. Because of the market's volatility and the numerous elements that either directly or indirectly affect stock value, stock market prediction is challenging. In this research, we want to investigate three distinct methodologies that employ varying machine learning algorithms to determine the same outcome—that is, whether the stock price will increase or decrease. (Deshpande et al., 2022)
Neural networks, decision trees, and ensemble methods are the common ML approaches that enable analysis of massive amount of historical data and pattern that can easily go unnoticed by the conventional models. Deep learning algorithms and specially the LSTM networks are capable of handling sequential data thus being able to efficiently employ time series analysis which is vital for the stock trading.

1.1 BACKGROUND TO THE PROJECT

Prediction of returns or any other aspect is very challenging when it comes to the stock market because of its high-risk nature and is influenced by changeable factors of the market, the behaviour of investors and factors such as economies. In the earlier time market investors have relied on technical and fundamental analysis in order to make proper judgments. However, these techniques often do not collect a large amount of information generated in financial markets and subtle, nonlinear interactions.

Recent advances in the fields of deep learning (DL) and machine learning (ML) present new possibilities to enhance the specificity and reliability of the stock prices' prediction. Thus, these methods can help credit rating agencies analyse large volumes of history data, trends, patterns, and derive various lessons from previous experiences to make accurate forecast that will serve as a basis for formulating data-driven recommendations. The performance of applying machine learning and deep learning in the stock prediction has enormous potential; nonetheless, there are several challenges such as overfitting of the data, poor quality of data, and large demand of processing resources.

1.2 PROJECT OBJECTIVES

Specifically, the objective of this stock price prediction project is to develop and evaluate several deep learning and machine learning algorithms to accurately forecast the closing values of the NSE Bank Index. In this study, Random Forest, Decision tree, Linear regression, Ridge regression, SVR , Light GBM, Recurrent Neural network, Long short term memory networks will be employed in order to determine the predictive efficiency of the models. The paper aims at comparing different models since the authors seek to establish which techniques are most effective in modelling the complex patterns and dynamic dependencies commonly associated with financial time series data.

Maintaining is ensuring that the models are robust, reliable and transportable is one of the major objectives. This involves having to perform a rigorous cleaning of the data to remove issues surrounding normalisation, missing values, among other issues on the quality of the given data. Also, it was planned to use more high-level methods, including cross-validation and hyperparameters' tuning, to increase the efficiency of the models and avoid overfitting.

The work also seeks to address some of the common problems of model reproducibility commonly linked with deep learning models wherein the models' results are demonstrated on different runs of code rather than being consistent across the runs through using random seeding and fine-tuning the computing environment to ensure that the outcome streams are consistent across the several runs. Furthermore, the project is going to consider such as requirements as the legislation in privacy and data protection, ethical concerns, and ensure that the mechanisms of accountability and transparency have been met at the stage of predictive modelling.

Another objective is to provide comprehensive performance evaluation measures so that the validity and reliability of the different models can be compared. Some of these measures are bias, R-squared (R^2) score or efficiency measure and Root Mean Squared Error (RMSE). The endeavour aims to help the stakeholders make a good decision by presenting the forecasts and their results in a form of a picture.

The idea is to create a system that is capable of real-time stock price prediction, and beneficial to the traders, financial analysts and others who form the stakeholder's population in this field. By fulfilling these goals, the project hopes to advance the area of financial analytics by offering insightful analysis and useful resources.

1.3 OVERVIEW OF THIS REPORT

This study is structured into 11 primary sections, each of which focusses on a distinct set of tasks, as outlined under. Also, at last Appendix is there for extra work and requirements.

Section 1: This section is consisting of Introduction, Background of Project, Project Objective and Overview of this Report.

Section 2: This Section consisting of Literature Review, which provides previous research on the project topic.

Section 3: This section consisting of Methodology, which discusses about the methods used to conduct this research.

Section 4: This section provides the requirements of this project.

Section 5: This section illustrates the design of this project.

Section 6: This section deals with the explanation of results which findings from the project.

Section 7: This section covers project management of this research.

Section 8: This section is critical appraisal which is detailed discussion and analysis of the work and its positive and negative outcome.

Section 9: Conclusions of this project, having achievements and future work, is included in this section.

Section 10: This section is consisting of student reflection.

Section 11: References for this project are included in this section.

Also, there are 2 appendices included in which one consists of Interim progress report and Meeting records and second consist of Ethics Certificate.

2. Literature Review

The analysis of the extensive existing literature on machine learning approaches to the improvement of the accuracy of stock price forecast during the last several years shaws that this is a rather dynamics rapidly progressing field with significant achievement, discussions, and challenges. Scholars have discussed various classes of architectures, input data, and models that can be used for deep learning, all have their pros and cons.

Investors have traditionally used traditional stock market analysis techniques, such as technical and fundamental analysis, to help them make well-informed judgements. Technical analysis comprises statistical examination of market activities, such as price and volume, whereas fundamental analysis looks at a company's financial statements, health, and economic considerations. Nevertheless, these conventional approaches frequently fail to capture the complex and dynamic character of stock markets, which prompts the use of machine learning techniques. (Gangthade, 2024).

Numerous algorithms have demonstrated the ability to increase prediction accuracy in the context of financial markets, where machine learning has been widely investigated. Massive volumes of historical data may be processed by ML models, which can also spot intricate patterns that are invisible to conventional approaches. Stock prices have been predicted using methods including neural networks, decision trees, support vector machines (SVM), and linear regression based on sentiment analysis, technical indicators, and historical data. (Chen et al., 2023, Deshpande et al., 2022)

Utilisation of Recurrent Neural Networks (RNNs), more specifically Long Short-Term Memory (LSTM) networks, has been popular over the past few years. RNNs are perfect for the sequential information, such as stock price time series (Zhong & Enke, 2019; Hoseinzade & Haratizadeh, 2019). Compared to shallow neural networks and conventional predictive models, long-term relationships and temporal patterns in stock data may be effectively captured by LSTMs. But according to Siami-Namini et al. (2019), LSTMs may have trouble capturing very long-term relationships and experience the issue of gradients vanishing.

Since these strategies have the limitations, scholars have looked into attention processes and transformer models, such as the Transformer model and its derivatives (Zhang et al. , 2022; Feng et al. , 2018). Such models have been applied to the stock price prediction problems and show rather promising outcomes in terms of learning long-range dependencies. Transformer-based models, then again, can be technically expensive and requires a vast amount of training information, which is not consistently available in the efficient world.

To create reliable machine learning models for stock price prediction, feature engineering and selection are essential processes. Finding the most pertinent factors, such as past prices, trading volumes, and financial indicators, that affect stock prices is essential to effective feature selection. In feature engineering, new variables like moving averages, relative strength indices, and emotion ratings generated from news articles and social media postings may be created from the raw data. These designed elements aid in identifying the fundamental patterns and attitudes influencing the market. (Varaprasad et al., 2022, Chen et al., 2023)

For a better understanding of the prediction ability of deep learning models on stock prices, many scholars have also looked into different feature engineering and data pre-processing methods. To capture a larger number of features that might affect the prices of the stocks, Long et al. (2019) proposed a feature engineering method that combines sentiment analysis of text data and added

news data along with the technical indicators. Focusing on the accuracy of their algorithms with using only price data for the forecast, they succeeded.

There has been also another approach to address the problem of noise and high variability, which is typical for stock market data. Hong et al (2020) proposed a denoising autoencoder based pre-processing technique to utilize in pre-processing of the stock data, whereby the method helped to minimize the impact of noise and outliers on the forecasting model. Similarly to this, Sun et al. (2017) applies the k closest neighbour choosing data strategy to screen out the noisy data which enhance the ability of deep learning model to detect the short-time and long-time trends. While Sun et al. (2017) researched features like wavelet modifications to capture different frequencies more assertive strategies to address the abrupt and non-stationary changes in stock market data is still necessary.

Another field of research is combination with other kinds of knowledge including domain information and experience by integrating it into deep learning models. To integrate the economic indicators' expert information into the attention layer of the transformer-based model, Zhao et al. (2023) proposed a knowledge-guided attention mechanism. It was proven that compared to the models that does not use the same domain knowledge integration this method increases both vastness and openness.

Some other studies have elaborated on the ways to increase the interpretability/ Explainability of DL models for stock price prediction in addition to enhancing the forecast accuracy. as the method to obtain the interpretable rules from the trained model, Park et al. (2022) suggested the technique which utilized the random forest framework and LSTM networks. This approach makes it is possible to have a high flexibility on the input variables and does not over-fit the model. This makes it possible to achieve high accuracy in prediction while at the same time come up with significant information regarding the decision maker; a factor that is very relevant in fiscal applications . However, much research needs to be done to enhance the understandability and explainability of the deep-learning methods for the prediction of stock prices and its compliance with existing laws. This is the case even though, there have been efforts made such as the knowledge-guided attention mechanism proposed by Zhao et al. (2023) by Park et al. (2022, hybrid approach.

Researchers have also explored the potential of deep reinforcement learning (DRL) for stock trading and portfolio management. Hao et al. (2023) proposed a DRL-based framework that learns to make trading decision based on market conditions and portfolio dynamics. The framework demonstrates promising results in generating portfolio trading strategies while managing risk effectively.

Another area of active research is the integration of auxiliary data sources, such as news article, social media sentiments, and macroeconomic indicators, into the stock price prediction models (Li et al., 2023). These multimodal approaches aim to capture a broader range of factors that influence stock prices, potentially improving prediction accuracy. However, the effective fusion of heterogenous data sources and the interpretation of the resulting models remain challenging. The majority of current research is on daily or lower- frequency stock price prediction; however, Li et al. (2023) points out that there is an increasing need for real- time or high-frequency trading applications. To create models that can analyse and react in almost real-time to quickly changing market situations, more research is required.

Therefore, in the light of presented literature one can summarize that there are still certain unsolved problems and further research opportunities that can be considered; at the same time, it is possible to state that utilization of machine learning approaches has provided greater level of predicted

accuracy in the field of stock prices. There are other aspects that are being improved which include incorporation of domain knowledge, features extraction/pre-processing, and model explainability Other challenges that face the developer include noise, availability of labelled data and real time prediction.

3. Methodology

In this section, the detailed process of the forecasting of a stock's closing price using several machine learning methods is outlined. There are various methods which are used and some of them are Random Forest (RF), Linear Regression (LR), Support Vector Regression (SVR), Decision Tree (DT), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), LightGBM Regressor (LGBMR). All techniques have been explained in detail including why they have been chosen, strength and limitations, how and to what extent they were applied and evaluated for this project.

3.1 OVERVIEW OF DATASET

The dataset used to examine the historical stock prices of a stock index that presumably is NSE Bank Index is taken from Yahoo Finance (Yahoo Is Part of the Yahoo Family of Brands, 2024). This dataset typically includes several key columns: In the dataset, the headings include Date, Open, High, Low, Close, Adj Close which represent the stocks opening price, the highest price, the lowest price, the closing price, and the adjusted closing price of the stocks respectively while the Volume represents the total number of shares traded in the market. Each of these columns means something and they are all aimed at recording the activities of stock trading that took place in each day. Any operations, analysis or predictive modelling that must be performed on this table has to consider these columns.

Explaining the Key Columns in the Field

Table 1: Dataset features

Attribute	**Datatype**
Date	Float64
Open	Float64
High	Float64
Low	Float64
Close	Float64
Adj. Close	Float64
Volume	Float64

1. Date:
The Date column measures trading day. Every entry relates to a particular date of the calendar when the stock exchange was operational for trading. This column is very useful for the time series as it contains the time reference from which the chronological order of the data is derived from.
2. Open:
The Open column concerns the price levels that prevailed when the shareholders or investors first began trading the stock or index on a specific day in the market. This tends to be the first price used to affect the exchange of a commodity in the market.

3. High:
The High column contains the stock's highest worth achieved on the trading day. It offers information concerning the most extreme operation of daily changes in price.

4. Low:
The Low column shows which is the lowest price of the stock for the day. Just like High value,

this is fundamental in establishing variations in the price as well as the level of fluctuation during the trading day.

5. Close:
The Close column indicates the value of the equity at which the stock price for the stock in question closed after the market was closed for the day. This is thought to be the most crucial of all prices because it illustrates the final trading, with the help of which people compare further results.
6. Adj Close:
The Adj Close (Adjusted Close) line modifies the CPC by incorporating information regarding corporate events involving the stock including splits, dividends and rights issues. This actual value makes it easier in evaluating the stock as it has been adjusted to take the changes in the number of stocks outstanding into consideration.

7. Volume:
The Volume column shows the total number of any share or contract that changed hands in a day. Large volumes of trades usually depict higher activities in the stock either by buyers or by sellers.

Importance of the Dataset

This dataset is crucial for various financial analyses, including:

1. Historical Analysis:
Understanding past prices, the structure, trend and rhythms of the stock's performance are easily explained by the price analysts find on the stock. Hence, this historical perspective is properly informative to the investment decision making.

2. Volatility Measurement:
The High and Low columns are useful in determining the higher and the lower price range recorded on a particular day and as such measures the volatility of the stock. Higher levels of volatility mean that price changes are more variable meaning that they stand for potential greater risk and potential greater returns.

3. Market Sentiment:
In the same manner, the Volume column is considered one of the raw reflections of the sentiment in the market. Volumes can therefore show powerful trends prevailing in the market whether a bullish or a bearish one.

4. Technical Analysis:
They are used directly in technical analysis which is a fundamental technique of conducting an analysis of past trading activity and movements to predict the future. This data is used in indicators like Moving Average, Bollinger Bands and Relative Strength Index (RSI).

5. Predictive Modelling:
LSTM networks, as well as other machine learning models, are trained on prior price data to predict much higher prices. It is a fact that features from this dataset such as Open, High, Low, Close and Volume are used to feed these models.

The NSEBANK stock is one of the informative datasets available for analysis and stock market prediction. Hence, by using and comprehending each of the mentioned column, analysts and data

scientist can successfully analyse stock market behaviour and incorporate suitable models to make relevant investments. It includes data preparation, data visualization, and steps that use sophisticated forms of statistical physics to make sufficient and precise predictions.

3.2 DATA PREPROCESSING

The first step of the technique was to clean the data so that it was suitable for modelling. In the dataset, corrected formats of sequential historical stock prices were used in which the names of the columns included Date, Open, High, Low, Close, Adj Close, and Volume. To do this the 'Date' column was coerced to a datetime format, then ordered to display the information in ascending order of time. Information missing in the Close price column was imputed with the forward fill technique. The reason behind using forward fill technique is that market on Monday starts according to the close price on that preceding weekend. The control variable, which was most indicative, from which the forecasts were made, was the 'Close' price. It was necessary to standardise the data, so 'Close' was scaled between 0 and 1 using MinMaxScaler. The normalised data was then used to create sequence of the required length for time-series model like RNN and LSTM.

EDA is also performed on the dataset to explore insights of this dataset.

Seasonal Trend Decomposition: The seasonal decomposition of this time series is performed using the **'seasonal_decompose'** function of **'statsmodels'** library. The original time series is divided into three primary parts using this technique: trend, seasonal, and residual. These elements offer various perspectives on the time series data. The graph 1 below shows the output of this decomposition:

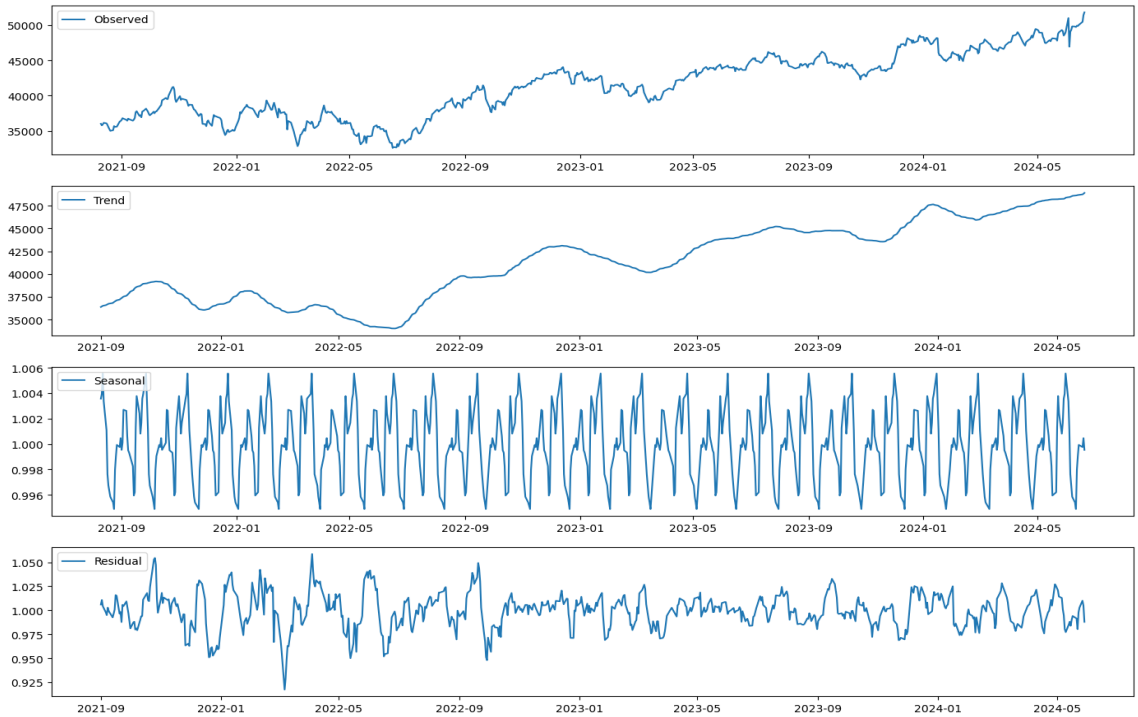


Figure 3.1: Seasonal trend Decomposition.

There are four components in graph 3.1, which are as follows:

1. Observed Component:

   The observed values, or real data points of the time series, are represented in the top panel of the graph. Here, it displays the NSEBANK stock index closing prices over a period.

- The stock price movements throughout the given period are depicted in this component.
- With a few noticeable peaks and troughs, the data shows an overall rising tendency.
- Additionally, there are noticeable short-term swings that might be related to daily or weekly changes in the market.

2. Trend Component:

   The trend component, which depicts the time series' long-term movement, is shown in the second panel. This is obtained by removing short-term variations from the data by smoothing.

   - The long-term growth tendency of the NSEBANK index is indicated by the trend line, which depicts a general upward movement in stock prices during the time.
   - The trend exhibits phases of acceleration and slowdown, which are indicative of shifts in the economic environment or market circumstances that affect the stock index. For example, the decline in the early part of 2022 may be related to again rise in corona virus cases.

3. Seasonal Component:

   The seasonal component, which depicts the data's recurring short-term cycle, is seen in the third panel. This is the portion of the data that repeats itself on a regular basis, such as weekly, monthly, quarterly, or daily trends.

   - There is an obvious cyclical pattern in the seasonal graph that repeats throughout time.
   - This might represent market behaviours in the context of stock prices, which are impacted by things like trade volumes, investor mood, or macroeconomic cycles.
   - The data exhibits persistent seasonal trends, as indicated by the rather constant amplitude of the seasonal component.

Seasonal trends in stock markets are impacted by several factors:

Monthly/Quarterly Reports: Businesses publish their quarterly earnings reports, which can regularly cause stock values to fluctuate.

Investor Behaviour: Predictable patterns can be produced by end-of-month or end-of-quarter investor actions like portfolio modifications.

Economic Cycles: Seasonality in stock prices can be influenced by broader economic cycles, such as fiscal year-end activity, tax concerns, and holidays.

4. Residual Component:

The residual component, which is displayed in the fourth panel, is the amount of variability in the data that remains after the trend and seasonal components have been eliminated. It records the erratic, haphazard variations.

- If the trend and seasonal components are well recorded, the residuals should ideally seem like white noise.
- Periods where the model falls short of adequately capturing the underlying patterns in the data are shown by variability in the residuals.
- Greater deviations from the trend or seasonal components in the residuals may indicate the existence of outliers or market oddities.

Autocorrelation:

In time series analysis, the Autocorrelation Function (ACF) plot is an essential tool. It shows the correlation between the time series' values and their historical values over various time delays. The x-axis displays the lag in terms of time intervals, while the y-axis indicates the correlation coefficient, which is a number between -1 and 1. (Smith, 2023)

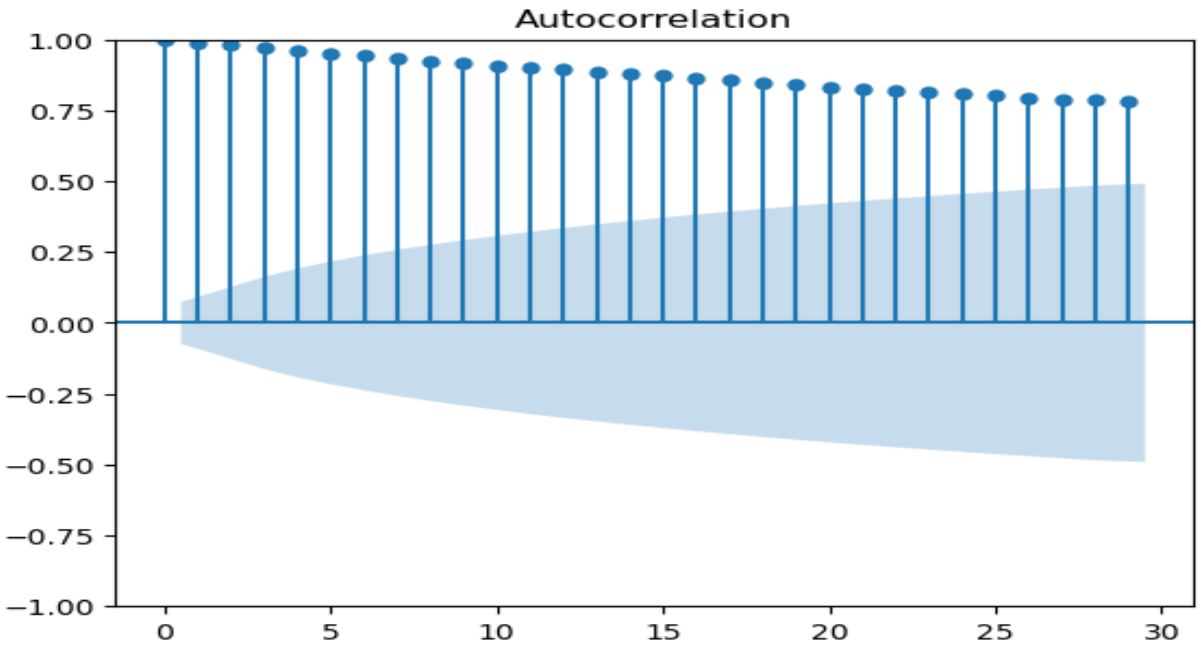


Figure 3.2: Autocorrelation plot

Figure 3.2 shows the correlation coefficients for various delays as vertical bars, with the confidence interval shown by the blue-shaded region. A bar indicates that the association is statistically significant at that specific lag if it goes beyond the shaded region.

Figure 2 reveals a high positive autocorrelation and virtually does not differ from it at almost all lags up to 25, as is evident from the graph. The other coefficients with the first few lags stand at almost 1 and thereafter, they are highly positive though constantly reducing. This mean that series has got ability to remember or has got persistence and this implies that past values of the series do influence the future values.

Such a pattern means that the given series has a high degree of dependence in time and does not go back to the mean shortly. It is typical for time series which contain trends or seasons. Understanding the structure of autocorrelation is critical to choosing the proper forecasting models since it draws attention to these relationships to boost the forecasting accuracy.

Distribution of Daily Returns:

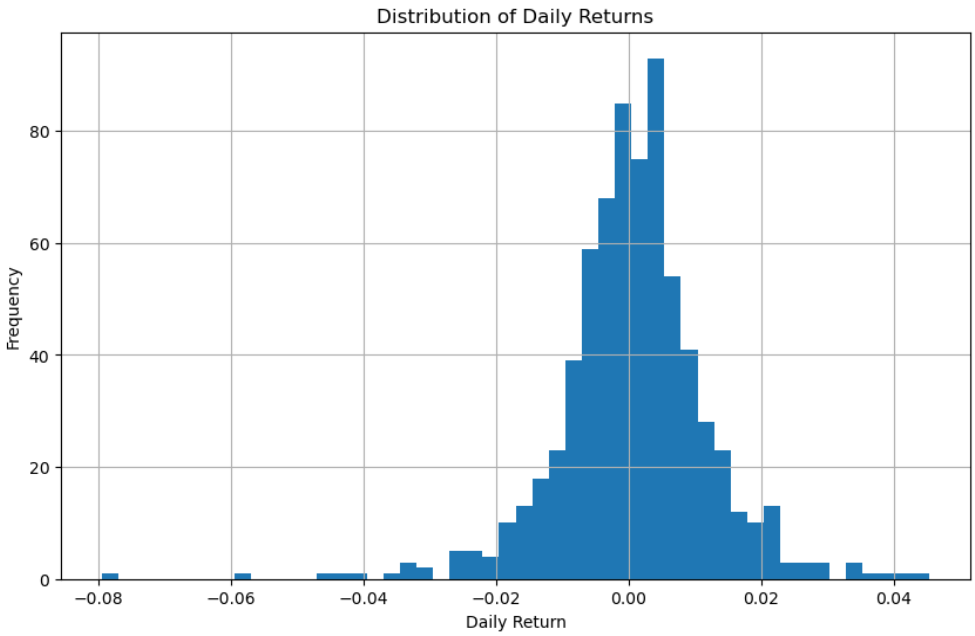


Figure 3.3: Distribution of daily return

Stock return data of NSEBANK is available for daily returns and the histogram depicting its return distribution is given in Figure 3.3. Daily returns are calculated using the percentage change of the closing price of the accumulated stock from the previous day's closing price. Hence, the x-axis depicting a range of daily returns and the y-axis depicting the frequency of occurrences within said range.

Important notes:

- Neither is the distribution significantly skewed towards a positive or negative performance because the distribution is roughly symmetrical.
- Most of the results are grouped in the range of -0.02 to 0.02, with a few anomalies on each extreme.
- Although there may be a little leftward tilt, the distribution seems to be quite symmetrical overall and may occasionally have higher negative returns than positive ones.

Candlestick chart:

Figure 3.4: Candlestick Chart

A candlestick chart showing the price changes of the NSEBANK stock is shown in Figure 3.4. With the green and red bodies designating days where the stock ended higher or lower than it opened, respectively, each candlestick symbolises a single trading day. The day's highest and lowest prices are displayed by the lines (wicks) above and below the body.

Important notes:

- During this time, the stock has seen both rising and declining movements.
- Longer candlesticks and wicks imply several instances of notable price moves, indicating times of increased volatility.
- By the end of the session, the general trend appears to be favourable, with a string of green candlesticks signifying a bullish movement.

Training and Testing Data:

The next step will be to divide the dataset into training and testing datasets in an 80:20 format for the stock price prediction project. Several methods are applied to train the model, the training set is formed by 80% of total data. In this way, it is ensured that the models catch for the underlying trend, patterns and seasonal aspects observed in the previous prices of the stock. During the training phase, we utilize several algorithms working on this portion of the data. After the models are trained, they are tested using the testing set that is 20 percent of the data.

3.3 MODELS USED

Linear Regression:

One of the most basic and popular statistical techniques for predictive analysis is linear regression. By fitting a linear equation to the observed data, it represents the connection between a dependent variable and one or more independent variables. Because of its ease of use and interpretability, Linear Regression was the baseline model employed in this study. The target variable and the input characteristics are assumed to have a linear connection by the model, although this may not always be the case for stock prices. Nonetheless, it's a useful beginning point due to its cheap computing cost and ease of implementation.

The equation for simple linear regression is:

$$y=\beta 0+\beta 1x+\epsilon$$

where y is the dependent variable, x is the independent variable, $\beta 0$ is the intercept, β1 is the coefficient, and ϵ is the error term.

In multiple linear regression, the equation is extended to:

$$y = \beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta pxp + \epsilon$$

where $x1, x2, \ldots, xp$ are the independent variables. (Draper & Smith, 1998)

Linear Regression models the relationship between the variables by fitting a line that minimizes the sum of squared differences between the observed values and the values predicted by the line.

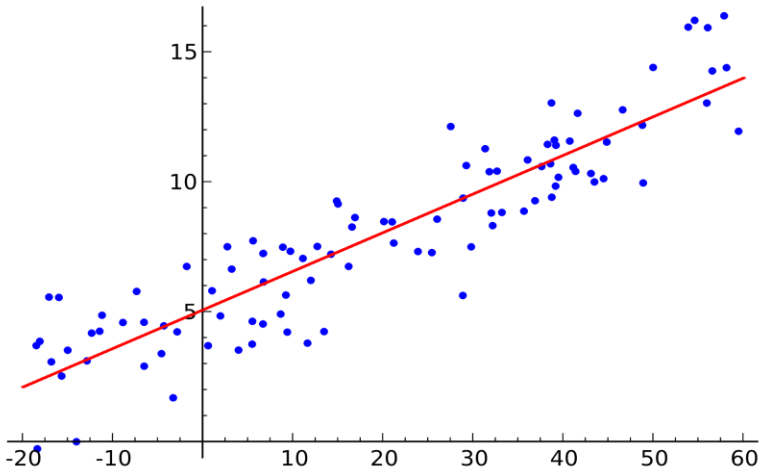


Figure 3.5: Fitting of Linear Model (*Papers With Code - Linear Regression Explained*, n.d.)

Ridge Regression:

To avoid overfitting, Ridge Regression is an extension of Linear Regression that incorporates a regularisation component. By penalising big coefficients, this regularisation term enhances the generalisation of the model. For this project, Ridge Regression was selected as a solution to the overfitting problem that frequently arises with Linear Regression, particularly in the case of noisy financial data. Ridge Regression reduces variance and stabilises the model by using a regularisation term, which produces predictions that are more trustworthy. (Ak, 2023)

The Ridge Regression equation modifies the linear regression cost function by adding a penalty term:
Cost Function= $\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2$

where λ is the regularization parameter, which controls the degree of shrinkage applied to the coefficients. (Residentmario, 2017)
Ridge Regression seeks to strike a compromise between minimising overfitting by maintaining modest model parameters and providing a good fit to the training set.

The Ridge Regression model performed admirably, displaying a high R-squared value and a low RMSE. Because the regularisation term was included, overfitting was less likely to occur, producing minimal bias and accurate predictions.

Decisions Tree (DT):

Decision trees are non-parametric methods in supervised learning for regression as well as classification types. They develop a decision tree like—a structure based on the subsets of the data that are generated based on input feature values. The decision trees can analyse both nominal and continuous data and they are highly understandable. Nevertheless, they are rather sensitive to overfitting, especially when it comes to some random phenomena like stock prices.

For a regression tree, the prediction is:

$$\hat{y} = \frac{1}{|R_m|}\sum_{i \in R_m} y_i$$

where R_m is the region (leaf node) containing the sample and y_i are the actual values of samples in R_m. (Quinlan, 1986)

A Decision Tree method divides nodes according to parameters such as Mean Squared Error, Information Gain, and Gini Impurity. Until a stopping criterion—such as the maximum depth of the tree or a minimum number of samples per leaf—is satisfied, the algorithm keeps splitting. (Quinlan, 1986)

For evaluating the performance of the developed model and tuning its parameters, the techniques of cross validation and hyperparameters tuning were applied. The Decision Tree model was easily understandable, but the scores of R-squared values were lower along with the higher RMSE than the other counterparts, but the model was overfitting.

LightGBM Regressor (LGBMR):

There is a gradient boosting system known as LightGBM that works on tree-based learning processes. This has a basic, powerful architecture that doesn't take very long to train and gives good accuracy. This work focuses on LightGBM due to its capability to handle large amounts of data and its capacity to pick complex features.

The objective function in LightGBM for regression is:

$$min_F \sum_{i=1}^{n} l(y_i, F(x_i)) + \sum_{k=1}^{K} \Omega(T_k)$$

where l is the loss function, $F(x_i)$ is the prediction, and $\Omega(T_k)$ is the complexity of tree T_k.

LightGBM reduces more loss and produces more accurate models by building tree's leaf-wise using the best fit. Additionally, it employs methods based on histograms to speed up training and consume less memory. (Ke et al., 2017)

The LightGBM Regressor was refined with the help of hyperparameter tuning for the adjustment of the model parameters implying the boosting of its efficiency. The performance of the LightGBM Regressor was high in stock price prediction it is evident from low RMSE and high R-squared values.

Random Forest (RF):

Classification and regression problems are the main applications for Random Forest (RF), an ensemble learning technique. During training, a large number of decision trees are built, and the mean prediction (regression) of each tree is produced. The main idea of RF is to use averaging to decrease overfitting and increase the accuracy of the model. (Breiman, 2001)

The prediction for a new instance x using a Random Forest is given by:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(x)$$

where:

- T is the number of trees in the forest.
- $h_t(x)$ is the prediction of the t-th tree for the input x. (Hastie et al., 2013)

Different bootstrap samples are used to train each decision tree (a sample of the same size as the training set but obtained with substitution). To guarantee that every tree is unique and captures a variety of patterns in the data, a random subset of characteristics is also considered when splitting nodes.

The Random Forest model's performance was further improved using cross-validation and hyperparameter adjustment. By refining the model's parameters, these methods helped to

increase accuracy and decrease overfitting. Strong performance was shown by the Random Forest model, which had modest RMSE and high R-squared values.

Support Vectors Regression (SVR):

Linear and non-linear regression are supported by Support Vector Regression (SVR), a kind of Support Vector Machine. By fitting the error under a predetermined threshold, it effectively deals with noise and outliers. For this study, SVR was chosen to investigate its potential for identifying intricate patterns in stock price data.

The SVR algorithm solves the following optimization problem:

$$min_{\beta,\beta_0} \frac{1}{2}||\beta||^2$$

Subject to :

$$|y_i - (\beta^T x_i + \beta_0)| \leq \epsilon + \xi_i$$
$$\xi_i \geq 0$$

where ϵ is the margin of tolerance, and ξ_i are slack variables. (Wikipedia contributors, 2024)

In order to effectively generalise to unseen data, SVR aims to minimise the coefficients while making sure that the error for each training point is within the margin of tolerance. (Wikipedia contributors, 2024)

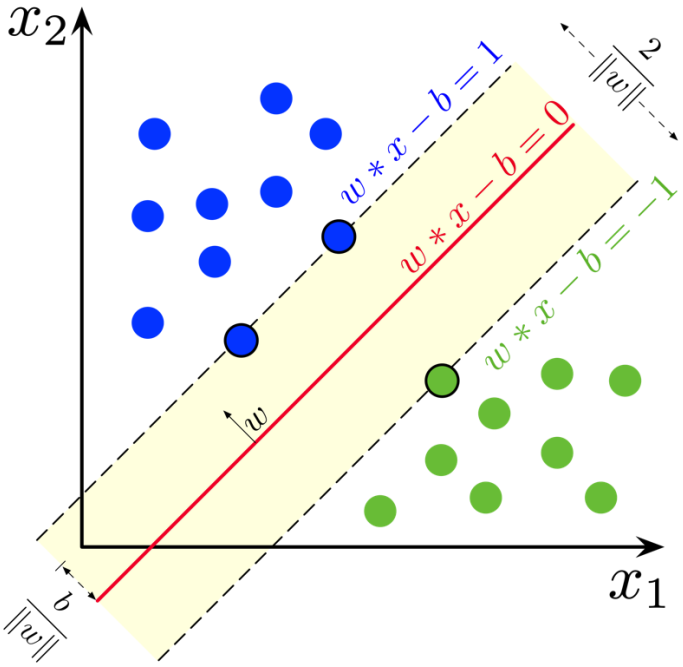


Figure 3.6: Margin hyperplane for an SVM Model(Wikipedia contributors, 2024b)

In contrast to previous models, the SVR model displayed lower R-squared values and greater RMSE, despite its promise. This suggests that while support vector machines (SVR) are capable of handling non-linearity, they might not have been as successful in capturing the complex relationships seen in the stock price data for this specific dataset.

Recurrent Neural Network (RNN):

As for the neural networks that work well with sequential data, there exists Recurrent Neural Network, or shortly RNN. They are equipped to hold first order relationships between the current forecast and prior inputs because of their internal state. In this study, we have benefited from the temporal nature of RNNs in analysing temporal features of stock prices.

The hidden state h_t in an RNN is updated using the following equations:

$$h_t = \tanh(W_h x_t + U_h h_{t-1} + b_h)$$
$$y_t = W_y h_t + b_y$$

where W_h and U_h are weights, x_t is the input, h_{t-1} is the previous hidden state, b_h and b_y are biases, and y_t is the output. (Wikipedia contributors, 2024c)

RNNs are capable of capturing temporal dependencies by maintaining a hidden state that gets updated at each time step. (Wikipedia contributors, 2024c)

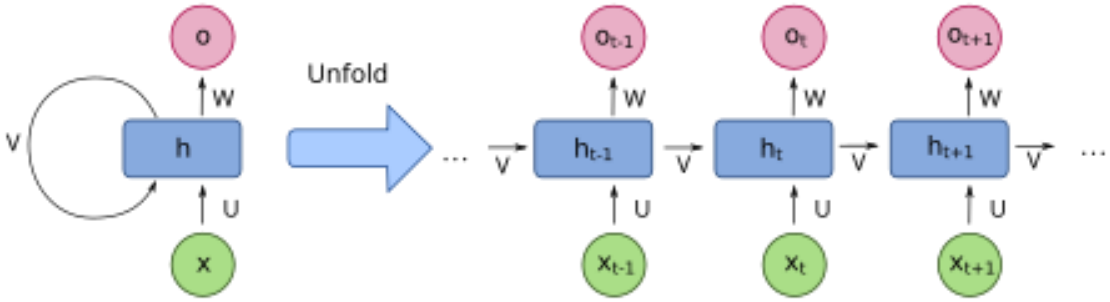


Figure 3.7: Compressed and Unfolded basic RNN(Wikipedia contributors, 2024c)

Here it was shown how the employed RNN model was trained using the sequences of past prices to make a prediction of the closing price. However, vanishing gradients and other issues may pose a threat on the performance of RNNs. If one compares the RMSE and R-Squared of the RNN model with the other models in this paper, then this model yielded moderately better results, but not as well as expected. The given model was found to offer high bias, the effect of which was that the model over-estimated the prices of stocks.

Long-Short Term Memory (LSTM):

There is a type of RNN that works with dependencies in sequential data called Long Short-Term Memory (LSTM) networks. They can overcome the vanishing gradient problem because the transfer of information between cells is regulated by gates. Among various recurrent neural networks, LSTM networks were chosen in this case to enhance the prediction of stock prices by addressing the LTSMs. (Wikipedia contributors, 2024c)

The LSTM cell is defined by the following equations:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$
$$\widetilde{C_t} = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C_t}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$
$$h_t = o_t \odot \tanh(C_t)$$

where f_t is the forget gate, i_t is the input gate, $\widetilde{C_t}$ is the candidate cell state, C_t is the cell state, o_t is the output gate, σ is the sigmoid function, and $\odot$ denotes element-wise multiplication. (Wikipedia contributors, 2024c)

Because LSTMs are better at updating and maintaining long-term states than ordinary RNNs, they are a good choice for time series prediction. (Wikipedia contributors, 2024c)

The LSTM model was trained using sequences of historical price since the next price would have to be predicted based on price forecasts. Again, the proposed LSTM model offered lower R-squared and higher RMSE compared to tree-based models and linear regression despite its complexity. Nevertheless, it provided informative thing for temporal relations on the data set. This means that on the average the model overestimated the stock prices, and the forecast was always on the high side as suggested by the bias .

EVALUATION MATRICES

Different models have been used to do this project and to evaluate these models, three different model evaluation metrics have been used, which are as follows:

RMSE: The measure used for determining the square root of the average of the squares of the difference between the actual and the expected value is called the Root Mean Squared Error or RMSE. RMSE is used to evaluate the models wherein, smaller values of RMSE leading to better performance models. The formula of RMSE is:

$$RMSE = \sqrt{\frac{\sum(y_i - \hat{y_i})^2}{N - P}}$$

Where:

- y_i is the actual value for the i^{th} observation.
- $\hat{y}_i$ is the predicted value for the i^{th} observation.
- N is the number of observations.
- P is the number of parameter estimates, including the constant. (*RMSE*, 2024)

R-squared (R^2): This statistical measure indicates the degree to which the changes in the dependent variable can be explained by the changes in the model's independent variables. Better match can therefore be expected when the values are closer to one.

Forecast Bias: Forecast bias is defined as the average of the differences between the expected and actual numbers. If the bias is close to zero, the prediction made by the model is unbiased, over estimation equals to high positive bias while low positive bias indicates high level of under estimation. The formula for forecast bias is:

$$\textbf{Forecast Bias} = \frac{\textbf{forecast} - \textbf{actual value}}{\textbf{No of values}}$$

Also, some methods like Hyperparameter tuning and Cross validation are applied to enhance the results of different models.

3.5 HYPERPARAMETER TUNING AND CROSS-VALIDATION:

Hyperparameter tuning is the process of determining which set of parameters will optimise the model's performance. To investigate various hyperparameter configurations for models like Random Forest, Decision Tree, and LightGBM, methods like Grid Search and Random Search were employed. By dividing the data into a collection of folds, cross-validation—specifically, K-Fold cross-validation with K=5 was utilised to evaluate the models' performance. To guarantee robustness and generalizability, the models were evaluated on the remaining fold or folds after being trained on a subset of them.

4. Requirements

Collection of significant data is mandatory for this stock price prediction type of work. The historical stock price of the NSE Bank Index is obtained from the Yahoo finance website. The coding and the visualisations are done using Jupyter Notebook and the whole analysis and the modelling is made based on it. Sklearn involves both the machine learning procedures and the preprocessing steps while TensorFlow and Keras are used in building the deep learning models LightGBM in providing for the gradient boosting models NumPy package provides for the numerical computations and last but not the least the Pandas in the data manipulation. Further for the data visualization other libraries are used such Matplotlib and Seaborn, while for the interactive graphs there is the Plotly. Combined, all these technologies ensure the analysis of data, training, and assessment of models to ensure maximum result is achieved.

Version of Different Libraries and Laptop configuration is as follows:

Selected Jupyter core packages...
IPython : 8.12.0
ipykernel : 6.19.2
ipywidgets : 8.0.4
jupyter_client : 7.4.9
jupyter_core : 5.3.0
jupyter_server : 1.23.4
jupyterlab : 3.6.3
nbclient : 0.5.13
nbconvert : 6.5.4
nbformat : 5.7.0
notebook : 6.5.4
qtconsole : 5.4.2
traitlets : 5.7.1

Laptop Configuration:- **Hardware Overview:**

Model Name:	MacBook Air
Model Identifier:	Mac14,2
Model Number:	MLY33ZS/A
Chip:	Apple M2
Total Number of Cores:	8 (4 performance and 4 efficiency)
Memory:	8 GB
System Firmware Version:	10151.101.3
OS Loader Version:	10151.101.3
Serial Number (system):	D4DXRH7VGX
Activation Lock Status:	Enabled

5. Design

The aim of this project is to use multiple machine learning and deep learning models to forecast the closing values of stock indices, particularly the NSE Bank Index. Preprocessing the data, extracting features, training, evaluating, and predicting the model are all part of the project. The models used include Random Forest, Decision Tree, Linear Regression, Ridge Regression, Support Vector Regression (SVR), LightGBM (LGBM), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks.

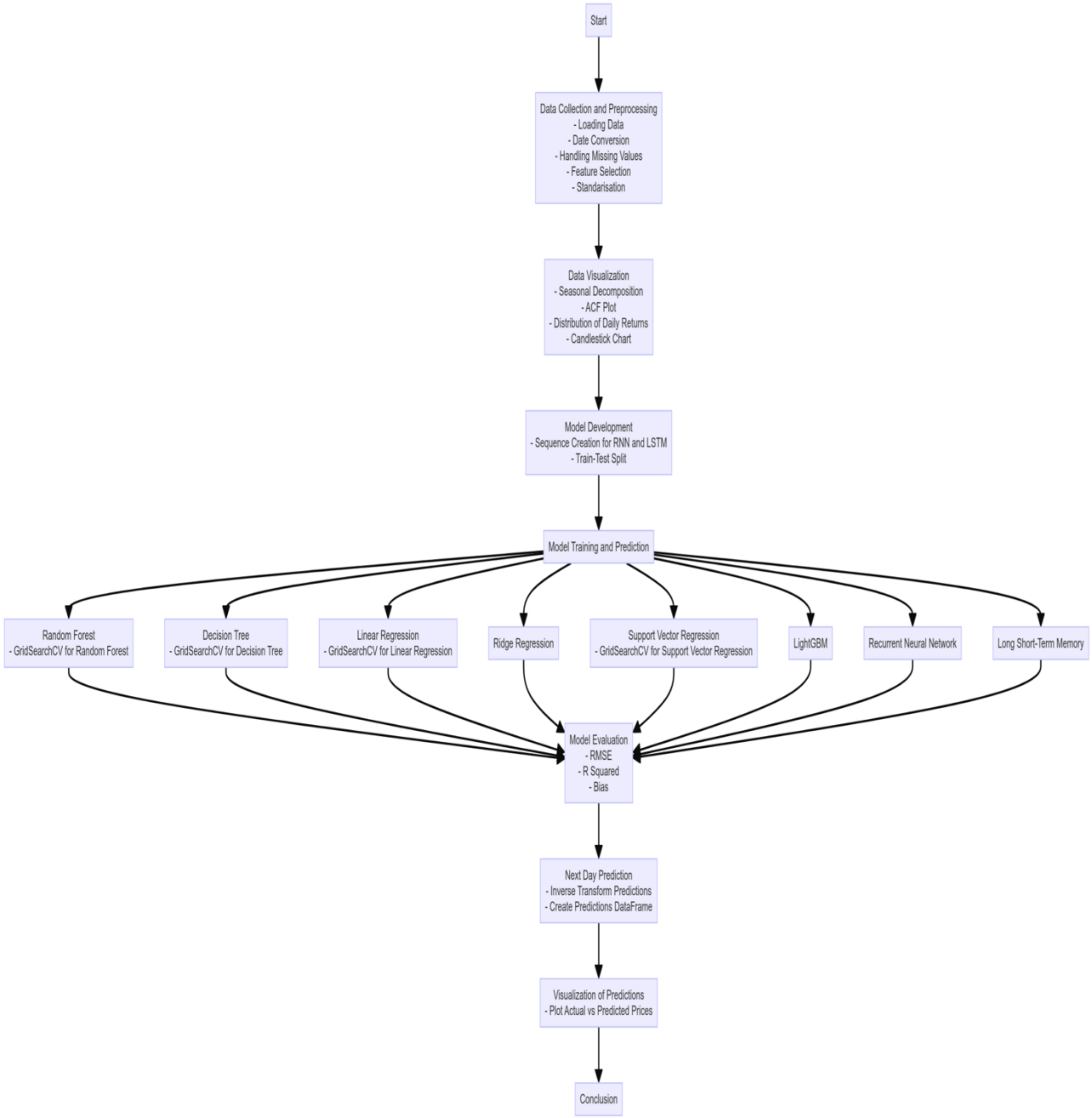


Figure 5.1: Design of the Project

The stock price prediction project's system design is made up of many modularly organised components. Because each module oversees a single job, there is a clear division of responsibilities, allowing for scalability and maintainability.

There are various steps involved in design of this project:

1. Data Collection and Preprocessing:

Data Source: The NSE Bank data used in this project is obtained from the Yahoo Finance website, which consists of historical stock prices.

1. **Loading Data**: The data is loaded from a CSV file containing the historical prices.
2. **Date Conversion**: The 'Date' column is converted to datetime format to facilitate time series analysis.
3. **Handling Missing Values**: Missing values are handled using forward fill to ensure continuity in the data.
4. **Feature Selection**: Only the 'Close' price is selected for prediction as it represents the end-of-day stock value.
5. **Normalization**: The data is normalized using MinMaxScaler to scale the 'Close' prices between 0 and 1, which is crucial for training models that are sensitive to the scale of input data.

2. Data Visualization:

1. **Seasonal Decomposition**: The data is decomposed into its seasonal, trend, and residual components to understand its underlying patterns.
2. **Autocorrelation Function (ACF) Plot**: The ACF plot is generated to visualize the correlation between the 'Close' prices at different lags.
3. **Distribution of Daily Returns**: A histogram with a kernel density estimate (KDE) plot is used to visualize the distribution of daily returns.
4. **Candlestick Chart**: A candlestick chart is created to represent the open, high, low, and close prices of the stock over time.

3. Model Development

Sequence Creation: For RNN and LSTM models, sequences of a fixed length (60 days) are created to serve as input features. The target variable is the 'Close' price for the next day following each sequence.

Train-Test Split: The dataset is split into training and testing sets with an 80-20 ratio.

4. Model Training and Prediction:

1. **Random Forest (RF)**:
   - A RandomForestRegressor is trained on the flattened input data.
   - GridSearchCV is used to optimize hyperparameters such as the number of estimators, max depth, and min samples split.
2. **Decision Tree (DT)**:
   - A DecisionTreeRegressor is trained similarly, with hyperparameters optimized using GridSearchCV.
3. **Linear Regression (LR)**:
   - A simple LinearRegression model is trained on the flattened input data.

4. **Ridge Regression**:
    - A Ridge model is trained to provide a regularized linear regression solution.
5. **Support Vector Regression (SVR)**:
    - An SVR model with an RBF kernel is trained to capture non-linear relationships.
6. **LightGBM (LGBM)**:
    - A LightGBMRegressor is used for its efficiency and speed in handling large datasets.
7. **Recurrent Neural Network (RNN)**:
    - A SimpleRNN model is built with 50 units in the RNN layer followed by a dense layer.
8. **Long Short-Term Memory (LSTM)**:
    - An LSTM model is built with two LSTM layers followed by a dense layer to capture long-term dependencies.

5. Model Evaluation:

The models are evaluated using Root Mean Squared Error (RMSE), R-squared (R^2) score, and bias. These metrics provide insights into the models' accuracy and reliability.

Next Day Prediction: The trained models are used to predict the closing prices for the next day. The predictions are inverse transformed to get the actual price values and are displayed in a DataFrame.

6. Visualization of Predictions

The predicted prices are plotted against the actual prices to visually assess the models' performance.

This design ensures a robust, systematic approach to developing a predictive model for stock prices, covering all critical aspects from data handling to model evaluation. Each phase is essential to the overall success of the project, ensuring that the final model is both accurate and reliable.

6. Results

6.1 Evaluation Metrics Results:

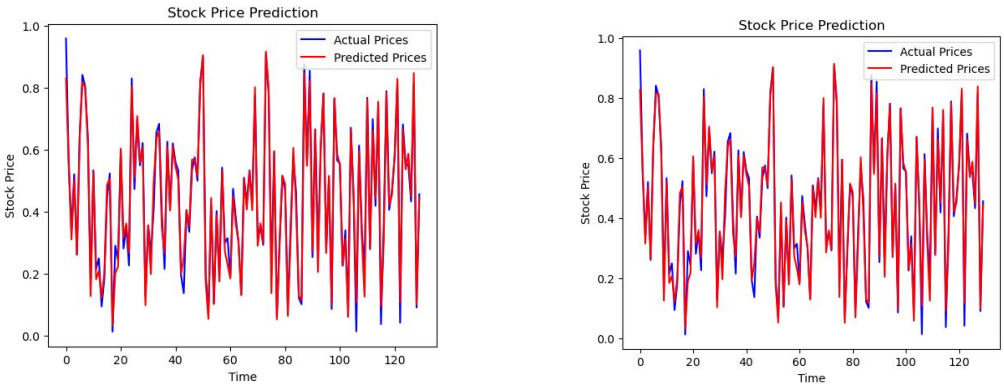
Table 2: Evaluation Metrics of Models employed

	RMSE	R Square	Bias
RF	629.9699	0.98	-80.5071
RF+Cross+Ht	634.4852	0.9797	-84.0136
LR	542.2110	0.9852	-24.0777
LR+Cross+Ht	545.4350	0.9850	-39.4618
DT	821.0825	0.9660	-61.3542
DT+cross+Ht	744.4926	0.9721	-79.1317
Ridge Regression	700.4165	0.9753	-47.6023
SVR	1785.5804	0.8395	-184.1174
SVR+Cross+Ht	523.4919	0.9862	-34.6146
LGBMR	609.9427	0.9812	-47.6396
LSTM	904.1031	0.9588	-134.5523
RNN	813.2214	0.9667	230.0317

In this project, various machine learning and deep learning models employed to predict the closing prices of the NSE Bank Index. The models included Random Forest, Decision Tree, Linear Regression, Ridge Regression, Support Vector Regression (SVR), LightGBM, Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM). The performance of each model was evaluated using metrics such as Root Mean Squared Error (RMSE), R-squared (R^2) score, and bias, which is shown in Table 2.

1. Random Forest: The Random Forest model performed well with a high R^2 score of 0.98, indicating that the model explained 98% of the variance in the stock prices. The RMSE of 629.9699 shows a relatively low error in predictions. The negative bias suggests that the model tends to underestimate the stock prices. After cross-validation and hyperparameter tuning, the performance slightly decreased, which might indicate overfitting in the initial model.

   Visual Representation:



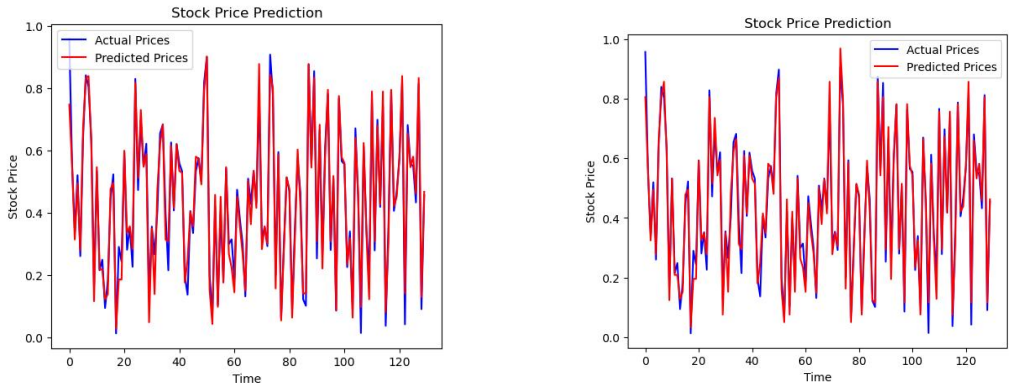
a. Random Forest b. Random Forest with CV and GS

Figure 6.1: Random Forest Actual vs Predicted

Figure 6.1 show the predicted stock prices (red line) closely following the actual prices (blue line), indicating a good fit.

2. Decision Tree: The Decision Tree model showed a good performance with an R^2 score of 0.9660 and an RMSE of 821.0825 before tuning. The negative bias indicates a tendency to underestimate stock prices. After cross-validation and hyperparameter tuning, the RMSE improved to 744.4926, and the R^2 increased to 0.9721, showing that the tuning helped in improving the model's performance by reducing overfitting and providing a more generalized model.

   Visual Representation:



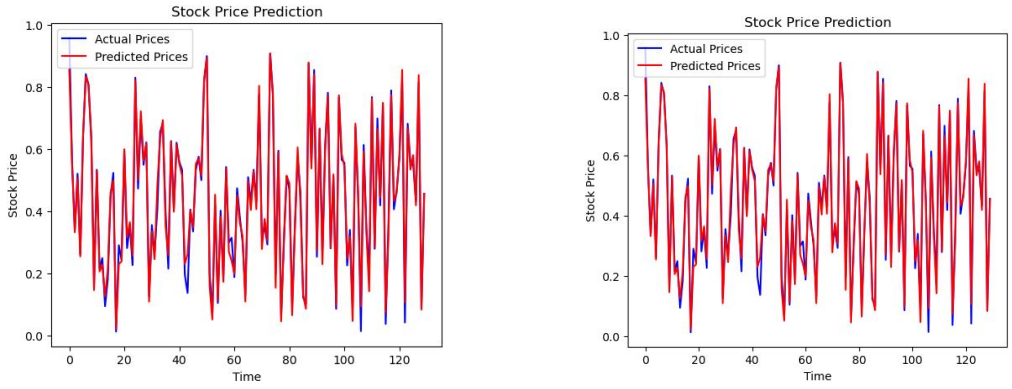
a. Decision Tree b. Decision Tree with CV and GS

Figure 6.2: Decision Tree Actual vs Predicted

The improvement in predictions after tuning is evident in Figure 6.2, where the predicted prices align more closely with the actual prices.

3. Linear Regression: The Linear Regression model performed exceptionally well, with the lowest RMSE of 542.2110 among all models and a high R^2 score of 0.9852. The low bias of -24.0777 indicates minimal systematic error, showing that the model's predictions are very close to the actual values. After Tuning and Cross validation, the evaluation metrics remain same which shows that there is no overfitting in the model.

   Visual Representation:



a. Linear Regression b. Linear Regression with CV and GS

Figure 6.3: Linear Regression Actual vs Predicted

Figure 6.3 shows that the predicted prices (red line) are very close to the actual prices (blue line), indicating the model's high accuracy. Also, there is no change after cross validation and Grid search in the graph.

4. Ridge Regression: Ridge Regression, a regularized version of linear regression, showed good performance with an RMSE of 700.4165 and an R^2 score of 0.9753. The model's bias is relatively low, indicating a slight tendency to underestimate the stock prices. Ridge Regression helps prevent overfitting by adding a penalty to the regression coefficients, which can be beneficial in handling multicollinearity in the data.

Visual Representation:

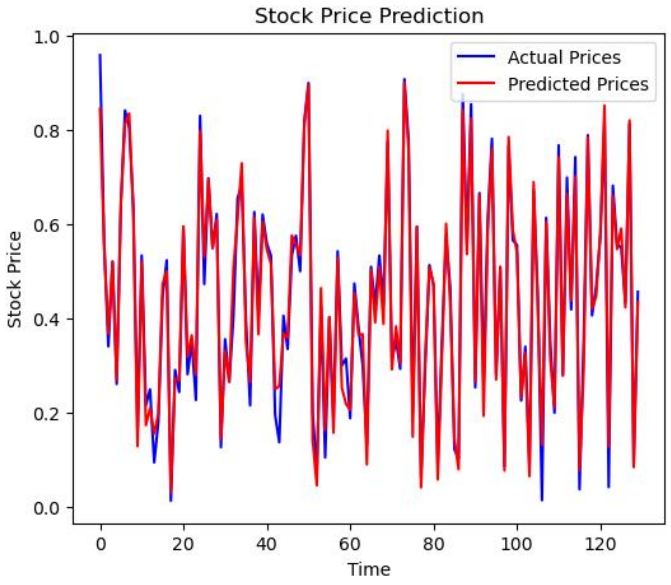


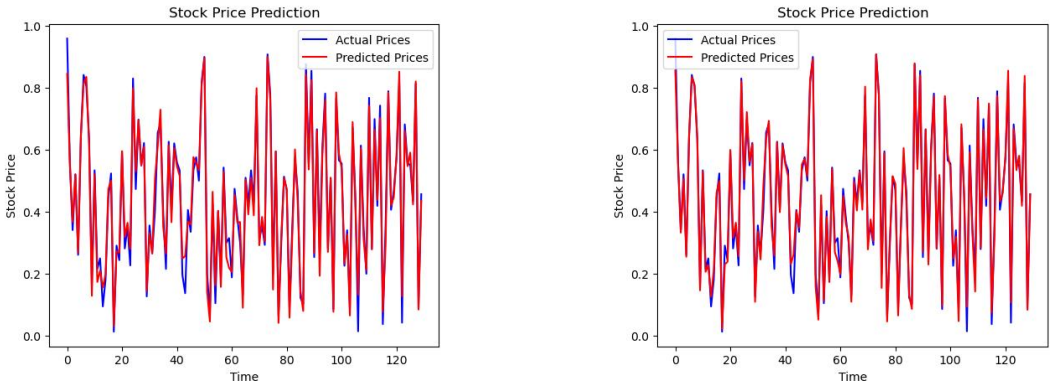
Figure 6.4: Ridge Regression Actual vs Predicted

Figure 6.4 shows that the predicted prices closely follow the actual prices, demonstrating the effectiveness of regularization.

5. Support Vector Regression (SVR): The SVR model performed the worst among all the models, with a high RMSE of 1785.5804 and a relatively low R^2 score of 0.8395. The high negative bias of -184.1174 indicates a significant underestimation of stock prices. SVR, particularly with the RBF kernel, is sensitive to the choice of hyperparameters and the scaling of data. The poor performance suggests that the model may not have captured the underlying patterns in the stock price data effectively. After Tuning and Cross validation, the evaluation metrics shows a significant change and SVR performed good which shows that there is no overfitting in the model.

Visual Representation:

Figure 6.5 shows that the predicted prices deviate significantly from the actual prices, highlighting the model's inadequacy in this context. Whereas, after applying tuning it captures significant data, and the predicted prices closely follow the actual prices.



a.. SVR b.. SVR with CV and GS

Figure 6.5: Support Vector Regression Actual vs Predicted

6. LightGBM (LGBMR): The LightGBM model showed strong performance with an RMSE of 609.9427 and an R^2 score of 0.9812. The relatively low bias indicates that the model's predictions are close to the actual values. LightGBM is known for its efficiency and accuracy in handling large datasets and complex patterns, which is reflected in its good performance.

Visual Representation:

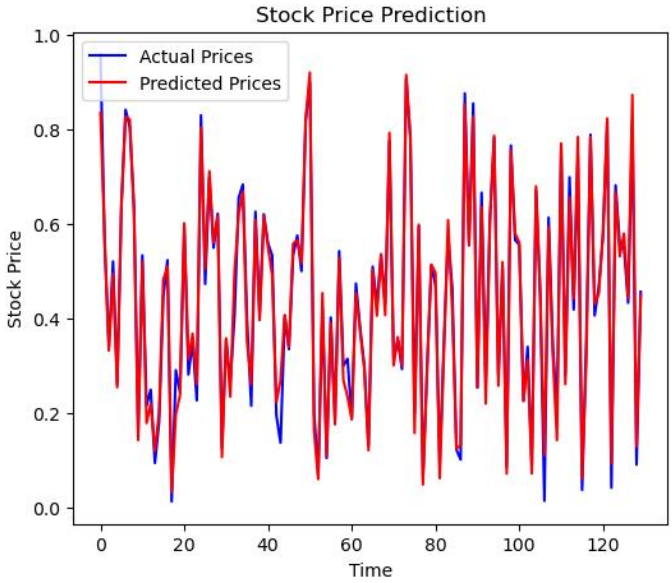


Figure 6.6: LightGBM (LGBMR) Actual vs Predicted

In Figure 6.6, the predicted prices align well with the actual prices, indicating the model's capability to capture the underlying trends in the data.

7. Recurrent Neural Network (RNN): The RNN model showed good performance with an RMSE of 813.2214 and an R^2 score of 0.9667. However, the high positive bias of -230.0317 indicates a significant overestimation of stock prices. RNNs are less complex than LSTMs but can still capture temporal dependencies. The high bias suggests the model might need further tuning to improve its accuracy.

Visual Representation:

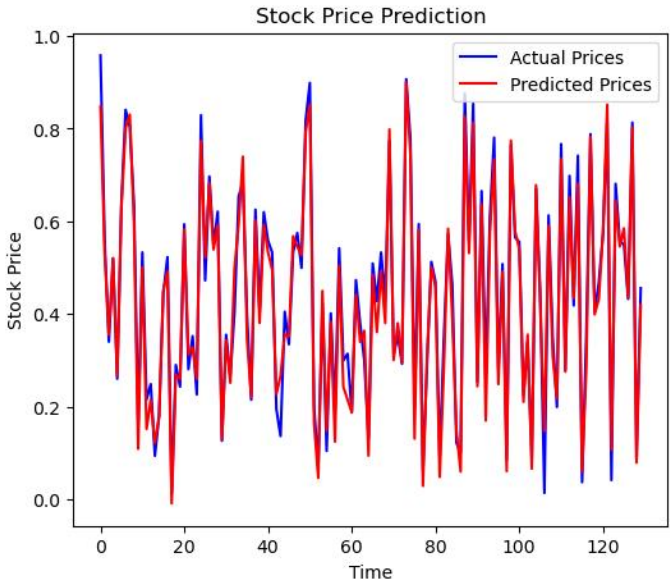


Figure 6.7: RNN Actual vs Predicted

In Figure 6.7, the predicted prices show a general alignment with the actual prices but with noticeable underestimations, indicating the need for model adjustments.

8. Long Short-Term Memory (LSTM): To overcome the overestimation of RNN, LSTM is used. The LSTM model, designed for time series data, performed well with an RMSE of 904.1031 and an R^2 score of 0.9588. Interestingly, the negative bias of 134.5523 indicates a underestimation of stock prices. LSTMs are effective in capturing long-term dependencies in sequential data, and this performance demonstrates their capability in stock price prediction despite the higher complexity. LSTM's ability to model temporal sequences makes it suitable for stock price prediction, although it requires more computational resources and careful parameter adjustments.

Visual Representation:

The Figure 6.8, show the LSTM model's predictions following the actual prices closely, with some deviations indicating areas for further tuning.

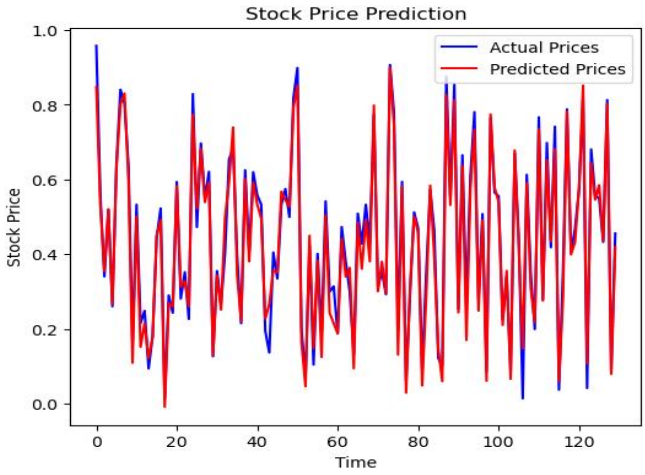


Figure 6.8: LSTM Actual vs Predicted

6.2 Next Day Prediction:

These models have also been deployed for real time prediction to predict close price of NSE Bank Stock of next day whose results are shown in Table 3 below:

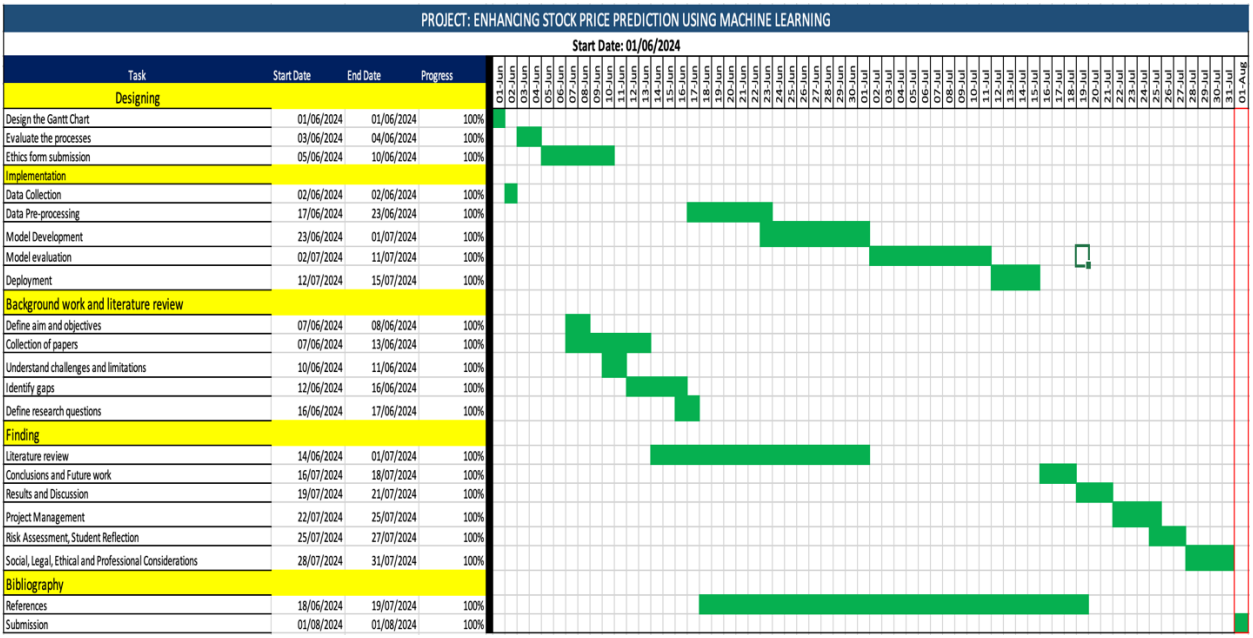
Table 3: Actual and Predicted Next day stock price

Model	Actual	Predicted
Random Forest	51,661.45	50441.70
Decision Tree	51,661.45	49061.73
Linear Regression	51,661.45	51707.43
Ridge Regression	51,661.45	51046.1
Support Vector Regression	51,661.45	50863.98
LightGBM	51,661.45	50164.32
Recurrent Neural Network	51,661.45	50768.64
Long Short-Term Memory	51,661.45	50501.6

Table 3 shows the actual stock price for the NSE Bank Index as well as the corresponding forecasts of different models for one day. The two models that were closest to the actual stock price are Linear Regression Model and Ridge Regression Model as they predict the linear relationship very well. Random Forest, SVR and LightGBM models were also good in the given dataset, exploiting the fact with regards to the inherent capability of handling intricate relationship and interaction within the data. Nonetheless, the Decision Tree formed part of the worst-performing model, probably because it offered highly overfitting solutions and because its performance had high sensitivity to hyperparameters, respectively. Also, the models RNN and LSTM demonstrated moderate deviation from the actual price levels, and this especially emphasized the task of tuning the deep learning models for the financial time series predictions.

7. Project Management

7.1 PROJECT SCHEDULE



Task	Start Date	End Date	Progress
Designing			
Design the Gantt Chart	01/06/2024	01/06/2024	100%
Evaluate the processes	03/06/2024	04/06/2024	100%
Ethics form submission	05/06/2024	10/06/2024	100%
Implementation			
Data Collection	02/06/2024	02/06/2024	100%
Data Pre-processing	17/06/2024	23/06/2024	100%
Model Development	23/06/2024	01/07/2024	100%
Model evaluation	02/07/2024	11/07/2024	100%
Deployment	12/07/2024	15/07/2024	100%
Background work and literature review			
Define aim and objectives	07/06/2024	08/06/2024	100%
Collection of papers	07/06/2024	13/06/2024	100%
Understand challenges and limitations	10/06/2024	11/06/2024	100%
Identify gaps	12/06/2024	16/06/2024	100%
Define research questions	16/06/2024	17/06/2024	100%
Finding			
Literature review	14/06/2024	01/07/2024	100%
Conclusions and Future work	16/07/2024	18/07/2024	100%
Results and Discussion	19/07/2024	21/07/2024	100%
Project Management	22/07/2024	25/07/2024	100%
Risk Assessment, Student Reflection	25/07/2024	27/07/2024	100%
Social, Legal, Ethical and Professional Considerations	28/07/2024	31/07/2024	100%
Bibliography			
References	18/06/2024	19/07/2024	100%
Submission	01/08/2024	01/08/2024	100%

Figure 7.1: Gantt Chart

Figure 7.1 displays a Gantt chart which offers a thorough project timetable with tasks from June 1, 2024, to August 1, 2024. To provide an organised method for task achievement, the assignment is organised in to five primary phases: designing, implementation, background work and literature review, and result. Every step is further subdivided into certain tasks.

Designing Phase: The initial phase focuses on establishing the project's framework. It begins with designing the Gantt chart on June 1, followed by evaluating the processes on June 3. The ethics form submission was completed by June 10. This phase ensures that the project starts with a clear plan and necessary ethical approvals.

Implementation Phase: This phase is the core of the project, spanning several weeks. Data Collection occurs on June 2 , where relevant historical stock price data is gathered. Data Pre-processing, from June 17 to June 23, involves cleaning and preparing the data for analysis. Model Development takes place from June 23 to July 1, focusing on creating and training various machine learning models. Model Evaluation follows from July 2 to July 11, assessing the models' performance. The phase concludes with Deployment from July 12 to July 15, finalizing the implementation of the predictive models.

Background Work and Literature Review: This phase involves essential preparatory work. From June 7 to June 8, the project's aims and objectives are defined. Collection of Papers occurs from June 7 to June 13, gathering relevant academic literature. Understanding Challenges and Limitations is addressed on June 10 and 11, followed by Identifying Gaps from June 12 to June 16, to spot areas lacking in research. Defining Research Questions on June 16 and 17 establishes specific questions guiding the research. This phase ensures a solid theoretical foundation and identification of research gaps.

Finding Phase: The final phase focuses on synthesizing and presenting the project's results. A comprehensive Literature Review is conducted from June 14 to July 1. Conclusions and Future Work, summarized from July 16 to July 18, provides insights and suggestions for future research directions. Results and Discussion occur from July 19 to July 21, interpreting the findings from the model evaluations. Project Management from July 22 to July 25 ensures the smooth execution of the project. Risk Assessment and Student Reflection from July 25 to July 27 evaluate potential risks and reflect on the learning process. Social, Legal, Ethical, and Professional Considerations from July 28 to July 31 address the broader implications of the project.

Bibliography: This final section ensures all sources are properly referenced and documented. References are compiled from July 19 to August 1, leading up to the final Submission on August 1. This comprehensive and structured approach ensures thorough research and robust model development, addressing all critical aspects of the project methodically.

7.2 RISK MANAGEMENT

Any project may encounter unforeseen circumstances, therefore anticipating these occurrences as possible risks and having a suitable plan of action in place for them is crucial to effective project management. As a result, various risks have been recognised during this project, and some solutions are given in table

Table 4: Risk and corresponding solution

Sr no	Risk	Impact	Severity	Mitigation Plan	Materialise
1	Health Issue	Delay in Submission	High	Apply for Extension	No
2	Technical issue	Long Processing Time	Medium	Use systems having high processors	Yes
3	Loss of Data	Delay in Submission	High	Regularly Backup of data on one drive or external Hard disk	No
4	Loss of Laptop	Delay in Submission	High	Use University systems	No
5	Reproducibility Issues	Inconsistent Results for RNN and LSTM	Medium	Set random seeds for all libraries involved in the computation to ensure consistent results. Document the entire modelling process meticulously to allow replication of experiments.	Yes

During this project, I encounter with two of the above risks. One is Technical Issue. While performing tuning of random forest and linear regression model, my laptop hangs, and it took nearly one day to overcome from this situation. Then, I decided to use university systems. Although, I already have backup of my work on one drive which helps me to complete my work.
Second is Reproducibility impact. The possibility of reproducibility problems emerged during the process. Prediction reliability was impacted by the inconsistent outcomes of the RNN and LSTM models' initial runs. To ensure predictable behaviour, this was handled by specifying environment variables and creating random seeds for TensorFlow, NumPy, and Python. This resulted in consistent results across several runs. Because of the successful mitigation technique, the project was able to go on with consistent and repeatable model outputs.

7.3 QUALITY MANAGEMENT

To guarantee the precision, dependability, and resilience of the predictive models in the stock price prediction project, quality management is essential. To preserve data integrity, the procedure starts with rigorous data validation and preparation, which addresses problems like missing values and abnormalities. To reduce overfitting and improve model generalisation, extensive cross-validation and hyperparameter adjustment are used. Setting random seeds in all computational libraries and carefully recording the modelling process guarantee reproducibility of outcomes. Model accuracy is evaluated on a regular basis to pinpoint areas for improvement using measures like bias, R2 and RMSE. By integrating the advantages of different models, ensemble techniques like Random Forest and LightGBM significantly improve prediction dependability.
A culture of quality and responsibility is fostered via weekly code reviews and collaboration with the supervisor.

7.4 SOCIAL, LEGAL, ETHICAL AND PROFESSIONAL CONSIDERATIONS

In the process of creation and using the system of the stock price prediction, several professional, social, legal and ethical issues must be taken into consideration as a result of observing the working professionalism and the legal requirements-oriented acknowledgement of the society. To overcome these issues several actions have been taken:

Data Protection and Privacy Laws: Of those key ones is following the data protection laws such as the CCPA and the GDPR. All these rules establish very stringent procedures of collecting the data, of storing the data as well as disposing of the data. Based on the discussion in the previous section, all the historical stock data employed in this study has been off-label, the dataset has also been posted online, thus, there is compliance with the privacy acts and protection of individuals' identity.

Ethical Position and Code of Conduct: In this regard, such a policy can be lauded as desiring to be held accountable together with a policy that is open and just in terms of the ethical standards. Consequently, the prediction system that is incorporated in the design requires documentation of the models and the algorithms that are used in the system consequently, leading to more transparency of the approaches that are being used. Besides, first of all, ethical requirements are used during training in order to exclude prejudice. Making sure that the data taken reflects all the possibilities in the market and does not incline towards a particular end or a particular player is part of it.

Social Implications: Since the use of stock price prediction system has a social importance it is therefore deserving to use it. Such systems are capable of affecting investment programmes and the trends in the markets. Thus, the forecasts are aimed at the elimination of any convenient fictions that influence a given person; that is why it is possible to report accurate forecasts only.

In summary, an all-encompassing strategy that incorporates legal compliance, ethical integrity, social responsibility, and professional scrutiny is necessary for the effective use of stock price prediction models. Through a holistic approach to these problems, the initiative may accomplish its goals and promote justice, trust, and the good of society.

8. Critical Appraisal

The paper on the stock price prediction provided a fairly good knowledge of how to apply multiple machine learning and deep learning techniques. That the procedure of dealing with missing values and normalizing the data was fully implemented was one of the shinning features of the project. These are critical in ensuring high reliability of the prediction models that would be developed. On the accuracy front, the models with low value of RMSE and high R^2 which included LightGBM as well as Linear Regression were clearly excellent. These models' repeatedly high performance showed how vital the use of ensemble methods and the parsimonious approach are for providing highly accurate forecast. Furthermore, hyperparameter tuning and-cross validation ensured that the models could not be over fit onto the training dataset to increase the model's accuracy.

However, they encountered some challenges and limitations on the agenda of the initiative. Thought possibilities of using Principal Component Analysis (PCA) were discussed it did not appear to have any measurable influence on results, implying that starting characteristics for prediction were rather optimal. Further, the most commonly used time series forecasting model, namely, the ARIMA as well as the integration of different techniques in the hybrid models were also considered. Unfortunately, these methods provided results that were off the pace with the main models, at a higher error rate. Flaws in the ARIMA model were thus experienced in the form of poor performance in forecasting due to failure in capturing all the other features of the data namely stock prices.

The variances in the RNN and LSTM models were significantly reduced as a result of setting the random seeds, but these models were stochastic anyway and hence were not constants at all and showed considerable variations. Information sub-set Depending on the differentiated vectors, the fine-tuning of deep learning models used in the analysis of financial time series data posed several challenges. The SVR model's poor accuracy also stressed the importance of preprocessing and regularization as well as the sensitivity of some algorithms to the choice of hyperparameters. In summary, the study effectively showcased the utilisation of diverse predictive models; yet, it also exposed the constraints and intricacies associated with stock price forecasting, furnishing significant perspectives for forthcoming research in this field.

9. Conclusions

9.1 ACHIEVEMENTS

The experiment on stock price prediction indicates how various deep learning and machine learning models do exceptionally well in predicting data of financial nature. Random Forest, Decision Tree, Linear Regression, Ridge Regression, SVR, LightGBM, RNN, and LSTM models were chosen and tested in order to accentuate the slight differences in their performance and the key variables that affect the prediction accuracy. This meant that the project had a solid plan, which entailed data pretreatment, cross-validation process, hyperparameters tuning, and proper model assessment. Some of the best performers that found the complex patterns were Ensemble methods such as Random Forest and LightGBM, whereas linear methods made decent benchmarks such as Ridge Regression and Linear Regression. Although challenges were experienced with repeatability as well as the possibility of fine-tuning the networks, the use of deeper learning algorithms like RNN and LSTM gave meaningful data pertaining to temporal dependency in spite of the high demand for processing power. The paper illustrated the importance of managing overfitting, ensuring the data's cleanliness and legitimacy, and utilizing computational resources efficiently. Moreover, ethical issues and data protection regulation that also needed to be addressed when it comes to the application of methods based on artificial intelligence were also considered as increasing the project's focus on responsible Artificial Intelligence methods. The research contributed to the enhancement of the understanding of predictive modelling in finance and introduced a solid framework as a basis for the future expansions of the approach with the combination of technical methodologies and the perspective of the problem's potential uses. Thus, this project essentially contributes to the field of financial analytics and paves the way for more complex and flexible approaches to prediction in the field of fluctuant markets. The experience that is acquired through it ranges from model optimization to ethical utilization.

9.2 FUTURE WORK

The promising findings of this stock price prediction study reveal a plethora of directions to future research in an effort to enhance the existing model's robustness, precision, and readiness for practical use. First, macroeconomic variables' inclusion and the data set extension to encompass a broader index of financial indicators might improve the model's ability to perform under different market conditions. As the above study shows including interest rates, inflation rates, and worldwide market indices as inputs might give a wider input feature set which can capture the effect of the other macroeconomic variables on stock prices.

The future studies also require to learn complex deep learning structures. It is also possible to consider the application of the losing strategy which might help in time series forecasting; the models that can be adapted include the Transformer based networks which produced great results in natural language processing. These models are more capable of incorporating long range dependencies than the conventional-RNNs and LSTMs.

Another exciting method is to incorporate sentiment from the business data, social media, and news article sentiment. It could be worthwhile by applying techniques of natural language processing the sentiment ratings are measured and added to the prediction models as extra features. This could enhance the project's ability to improve its forecasts based on data and the different factors that compel changes to its rates in the market. To speed up the model selection and hyperparameter tuning and to ensure that the best settings of the models are found within a reasonable amount of time, automated machine learning (AutoML) systems might be applied too.

Another important issue that needs more exploration is enhancing the models' interpretability. Techniques for explaining the models' predictions, such as SHAP (SHapley Additive exPlanations), may increase the level of trust within stakeholders and contribute to enhanced decision-making by indicating aspects contributing to the predictions. This is particularly crucial in the financial application since, at times, the services provided must have a high level of accountability.

Finally, getting rid of actual deployment challenges is crucial if the idea is to be transformed into a research prototype yielding a system ready for deployment. This relates to establishing a real-time data processing system to handle such volumes, the need to ensure that this ecosystem can handle large volume and integrating the system with current financial systems for proper functioning. Keeping accuracy consistent over the future, will necessitate the implementation of clear structures for tracking and updating that will help with detecting model shift and readjust the models at regular intervals.

The project might have a far-reaching impact by incorporating the following paths of future work; these methods will provide enhanced and more dependable stock price forecasts valuable to numerous levels of the financial industry.

10. Student Reflections

It has been insightful and transforming to start the machine learning-based stock price prediction project. My comprehension of predictive modeling's theoretical and applied components has increased as a result of this adventure. I faced a variety of difficulties, including managing huge datasets, preventing overfitting, and guaranteeing the repeatability of outcomes. Each of these obstacles forced me to improve my analytical and problem-solving abilities. The need to regulate and comprehend the stochastic components of machine learning algorithms was brought to light by the requirement to set random seeds in RNN and LSTM models in order to obtain consistent results. Furthermore, even if they had little effect in enhancing predictions, using Principal Component Analysis (PCA) and investigating hybrid models emphasised the value of rigorous experimentation and the iterative process of model creation. The experiment also shown how important it is to preprocess and validate data because the calibre of input data directly affects the performance of the model. Engaging in cooperative dialogues and code reviews with colleagues yielded a variety of viewpoints and strengthened the resilience of the created models. Furthermore, this initiative has made me aware of the moral and societal obligations that come with making financial projections, which has helped me to see the bigger picture of my profession. Looking back, this research has improved my technical proficiency and given me a greater understanding of the careful and moral approach needed in machine learning applications. It has strengthened my will to keep studying and growing while preparing me for obstacles in the industry in the future. Also, this project has been critically important to my academic and professional growth because of the invaluable practical insights I have learnt and the experience I have gained in navigating through complications.

11. Bibliography and References

Ak, A. (2023, October 8). Ridge Regression - Abel AK - medium. *Medium*. https://medium.com/@abelkuriakose/ridge-regression-98c2a65cb3b1

Breiman, L. (2001). Random Forest. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Chen, K., Lee, P., & Liu, S. (2023). Poster:Stock Price Prediction Using Machine Learning. *Educational Administration: Theory and Practice*. https://doi.org/10.1109/icdcs57875.2023.00123

Draper, N. R., & Smith, H. (1998). Applied Regression analysis. pg no 21-26 In *Wiley series in probability and statistics*. https://doi.org/10.1002/9781118625590

Deshpande, R., Lunkad, D., Kunjir, M., & Ingle, S. (2022). Stock Price Prediction and Analysis using Machine Learning Techniques. *International Journal for Research in Applied Science and Engineering Technology*, *10*(5), 2092–2098. https://doi.org/10.22214/ijraset.2022.42747

Feng, F., Chen, H., He, X., Ding, J., Sun, M., & Chua, T. (2019). *Enhancing Stock Movement Prediction with Adversarial Training*. IJCAI. https://www.ijcai.org/proceedings/2019/0810

Faraz, M., Khaloozadeh, H., & Abbasi, M. (2020). Stock Market Prediction-by-Prediction Based on Autoencoder Long Short-Term Memory Networks. *IEEE*. https://doi.org/10.1109/icee50131.2020.9261055

Gangthade, R. A. (2024). Stock price prediction using machine learning. *International Journal for Research in Applied Science and Engineering Technology*, *12*(4), 3472–3477. https://doi.org/10.22214/ijraset.2024.60725

Hastie, T., Tibshirani, R. J., & Friedman, J. (2013). *The elements of statistical learning: data mining, inference, and prediction*. http://catalog.lib.kyushu-u.ac.jp/ja/recordID/1416361

Hao, Z., Zhang, H., & Zhang, Y. (2023). Stock portfolio management by using Fuzzy Ensemble Deep Reinforcement Learning Algorithm. *Journal of Risk and Financial Management*, *16*(3), 201. https://doi.org/10.3390/jrfm16030201

Hoseinzade, E., & Haratizadeh, S. (2019). CNNpred: CNN-based stock market prediction using a diverse set of variables. *Expert Systems With Applications*, *129*, 273–285. https://doi.org/10.1016/j.eswa.2019.03.029

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *HAL Open Science*. https://hal.science/hal-03953007

Long, W., Lu, Z., & Cui, L. (2019). Deep learning-based feature engineering for stock price movement prediction. *Knowledge-based Systems*, *164*, 163–173. https://doi.org/10.1016/j.knosys.2018.10.034

Li, A., Wei, Q., Shi, Y., & Liu, Z. (2023). Research on stock price prediction from a data fusion perspective. *Data Science in Finance and Economics*, 3(3), 230–250. https://doi.org/10.3934/dsfe.2023014

Park, H. J., Kim, Y., & Kim, H. Y. (2022). Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. *Applied Soft Computing (Print)*, *114*, 108106. https://doi.org/10.1016/j.asoc.2021.108106

Papers with Code - Linear Regression Explained. (n.d.). https://paperswithcode.com/method/linear-regression

Quinlan, J. R. (1986). Induction of decision trees. Machine Learning. *Symposium on the Theory of Computing*, *1*, 81–106. https://ci.nii.ac.jp/naid/10013159129

Residentmario. (2017, October 31). *Ridge regression cost Function*. Kaggle. https://www.kaggle.com/code/residentmario/ridge-regression-cost-function

RMSE. (2024, February). Statistics by Jim. Retrieved July 21, 2024, from https://statisticsbyjim.com/regression/root-mean-square-error-rmse/

Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019, November 21). *A comparative analysis of forecasting financial time series using ARIMA, LSTM, and BILSTM*. arXiv.org. https://arxiv.org/abs/1911.09512

Sun, H., Rong, W., Zhang, J., Liang, Q., & Zhang, X. (2017). Stacked denoising autoencoder based stock market trend prediction via K-Nearest Neighbour Data selection. In *Lecture Notes in Computer Science* (pp. 882–892). https://doi.org/10.1007/978-3-319-70096-0_90

Smith, T. (2023, March 20). *Autocorrelation: what it is, how it works, tests*. Investopedia. https://www.investopedia.com/terms/a/autocorrelation.asp

Varaprasad, B. N., Kanth, C. K., Jeevan, G., & Chakravarti, Y. K. (2022). Stock Price Prediction using Machine Learning. *2022 International Conference on Electronics and Renewable Systems (ICEARS)*. https://doi.org/10.1109/icears53579.2022.9752248

Wikipedia contributors. (2024, July 17). *Support vector machine*. Wikipedia. https://en.wikipedia.org/wiki/Support_vector_machine

Wikipedia contributors. (2024b, July 17). *Support vector machine*. Wikipedia. https://en.wikipedia.org/wiki/Support_vector_machine#/media/File:SVM_margin.png

Wikipedia contributors. (2024c, July 20). *Long short-term memory*. Wikipedia. https://en.wikipedia.org/wiki/Long_short-term_memory

Zhong, X., & Enke, D. (2019). Predicting the daily return direction of the stock market using hybrid machine learning algorithms. *Financial Innovation*, *5*(1). https://doi.org/10.1186/s40854-019-0138-0

Zhang, Q., Qin, C., Zhang, Y., Bao, F., Zhang, C., & Лю, П. (2022). Transformer-based attention network for stock movement prediction. *Expert Systems With Applications*, *202*, 117239. https://doi.org/10.1016/j.eswa.2022.117239

Zhao, L., Feng, X., Feng, X., Qin, B., & Liu, T. (2023). Length Extrapolation of Transformers: A Survey from the Perspective of Position Encoding. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2312.17044

Appendix A – Interim Progress Report and Meeting Records

Table 4: Meeting Records with Supervisor

Date	Meeting no	Discussion	Time
05/06/2024	1	Ethics Process and Submission of Ethics application	10-12 min
10/06/2024	2	Discussion Regarding process for Literature Review	15 min
17/06/2024	3	Initial coding and data preprocessing	17-20 min
24/06/2024	4	Regression model implementation	15 min
01/07/2024	5	Neural Network Model implementation and Methods to improve results of Regression models	20 min
08/07/2024	6	Hybrid Model implementation and methods to improve neural network models results	15 min
15/07/2024	7	PCA and initial writing overview	12 min
22/07/2024	8	Feedback on Half report	15 min
29/07/2024	9	Final Report overview and suggestion of some changes	15 min

Appendix B – Certificate of Ethics Approval

Certificate of Ethical Approval

Applicant:	Pankaj Yadav
Project Title:	Enhance the Stock Market Price prediction using machine learning techniques.

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Low Risk

Date of approval:	05 Jun 2024
Project Reference Number:	P177746

Appendix C- Source Code

The code for this project is at https://github.com/Pankaj1582/Stock-Price-Prediction-NSEBank.git