A

Mini Project Report

On

“Stock Price Prediction”

Submitted for partial fulfillment of requirement for the award of degree

Of

Master of Business Administration
(Artificial Intelligence and Data Science)

GRAPHIC ERA (DEEMED TO BE UNIVERSITY)
DEHRADUN (UTTARAKHAND)

Session: 2024-2025

Supervision by:
Supervisor Name: Mr. Chandra Prakash
Designation:

Submitted by:
Name: Vanshik Narula
Roll No.: 36
Enrollment No.: GE – 24144943

DEPARTMENT OF MANAGEMANT STUDIES
GRAPHIC ERA (DEEMED TO BE UNIVERSITY) DEHRADUN

DECLARATION

I hereby declare that the Mini Project entitled *"Stock Price Prediction"* submitted for the Degree of Master of Business Administration in Artificial Intelligence and Data Science, is my original work and the Mini Project has not formed the basis for the award of any degree, diploma, associateship, fellowship or similar other titles. It has not been submitted to any other University or Institution for the award of any degree or diploma.

(Signature of Student)

Vanshik Narula
Name of the Student

CERTIFICATE BY SUPERVISOR

I have the pleasure of certifying that Mr.Vanshik Narula is a student of Graphic Era (Deemed to be University) of the master's degree in business administration (MBA) in AI&DS. His/her University Roll No is. 1404943

He has completed his Mini Project titled as *"Stock Price Prediction"* under my guidance.

I certify that this is his original effort & has not been copied from any other source. This project has also not been submitted in any other university for the purpose of award of any Degree.

This project fulfils the requirement of the curriculum prescribed by Graphic Era (Deemed to be University), Dehradun, for the said course.

I recommend this Mini Project for evaluation & consideration for the award of Degree to the student.

Signature: Name of the Guide:

Signature: Name of the Area Chair/ HOD:

ACKNOWLEDGEMENT

I express my sincere thanks to my project guide, Mr. Chandra Prakash Designation ________________, Department ____________________ for guiding me right form the inception till the successful completion of the project.

I also record my indebtedness to my supervisor, Prof. /Dr. /Mr. /Ms. __________________________, for his / her counsel and guidance during the preparation of this Mini Project. I am grateful to (Director, Head or any other faculty) …………………………………………………….

I wish to record my sincere thanks to (your family members or friends etc.) .... for their help and cooperation throughout our project. My thanks are due to (those who have helped in collecting data or analysis or typesetting etc.).,

(Signature of Student)

Vanshik Narula

Name of the Student

Table of Content

ABSTRACT

Stock Price Prediction tends to be a challenging task due to market volatility and influence of various external factors including environment, politics and the regulations made by SEBI. The project is made to develop an ML model for predicting the closing stock prices of Google (GOOGL) using various ML models. I have utilized financial news, social media, and Kaggle for key inputs such as Historical Stock Data, technical indicators, and sentiment analysis. The various ML models used for stock price prediction are ARIMA, SARIMA, ETS, and Facebook Prophet. Multiple models are being tested for higher accuracy of the results. Data pre-processing, feature selection, and model optimization techniques have been implemented for performance enhancement.

The results demonstrate that advanced ML/DL models can provide valuable insights, assisting investors in making informed trading decisions.

However, the project acknowledges the limitations of unpredictability in stock markets and the need for continuous model refinement.

1. CHAPTER 1: INTRODUCTION

Stock Market has captured the interests of almost all the firms operating in a system. India's stock market exchange has become the world's 7^{th} largest exchange exceeding $5 trillion market capitalization. NSE has been ranked the 10^{th} largest unlisted company in the world by Burgundy Private Hurun India 500 list according to their report in 2024.

1.1. Domain Specific Knowledge

Stock Market has become more prominent in recent times as it provides the companies most easiest way to raise funds from investors. Stock Market is the environment where investors and fund raisers interact with each other to exchange the financial securities being regulated by SEBI (Securities and Exchange Board of India) in India. It allows a platform for those who are willing to take high risks to earn profits.

Securities are of various types such as Stocks, Mutual Funds, Futures, Options, Cryptocurrencies, etc. They all are traded on a daily basis in the two national stock exchanges namely BSE (Bombay Stock Exchange) & NSE (National Stock Exchange).

Stock Market is influenced by many external factors including, Political influences, Market conditions, Economic Conditions, and Environmental conditions along with certain other factors such as Frauds, Leakages in SEBI, P&L of companies, and the funds required by the companies.

Thus study of stock markets is prominent and required so as to provide the interested parties an idea on how to invest and when to sale & purchase the securities in the market.

1.2. ARTIFICIAL INTELLIGENCE & MACHINE LEARNING TECHNOLOGY

1.2.1. Role of AI & ML in Stock Price Prediction

AI and machine learning technologies examine past data to forecast *future closing prices of the stock* and optimize the sale and purchase of the stock. The dataset indicates that various factors influencing the stock prices.

1.2.1.1. *Economic Factors:*

Economic Indicators:

GDP growth, inflation rates, interest rates, and unemployment all play a significant role.

Monetary Policy:

Central bank actions, like interest rate adjustments, can influence borrowing costs and investment decisions, impacting stock prices.

Fiscal Policy:

Government spending and tax policies can affect economic growth and corporate profitability, ultimately impacting stock prices.

Global Economic Conditions:
International trade, economic crises, and geopolitical events can create uncertainty and volatility in stock markets.

1.2.1.2. *Company specific Factors:*

Company Performance:

Earnings reports, revenue growth, profit margins, and debt levels are key indicators of a company's financial health and can significantly impact stock prices.

Corporate Actions:

Dividends, stock splits, mergers, acquisitions, and new product launches can all influence investor sentiment and stock prices.

Industry Trends:
Changes in consumer preferences, technological advancements, and regulatory shifts can affect the profitability and prospects of companies within specific sectors.

1.2.1.3. *External Factors:*

Geopolitical Events:

Political instability, wars, or other geopolitical events can create uncertainty and volatility in financial markets, impacting stock prices.

Natural Disasters:

Major natural disasters can disrupt businesses and economies, potentially leading to lower stock prices.

Currency Fluctuations:

Changes in exchange rates between currencies can affect the profitability of companies that operate internationally, impacting their stock prices.

1.2.2. *ML Models for Stock price prediction:*

1.2.2.1. ARIMA:

- Used in case of time series data and helps in forecasting future closing prices of data based on previous values.

1.2.2.2. SARIMA:

- Seasonal AutoRegressive Integrated Moving Average helps in finding out the seasonal forecasts on the basis of previous periodical values.

1.2.2.3. ETS:

- Exponential Smoothening is a statistical model used for capturing the trend and seasonality of the dataset and the noise in the time-series dataset .

1.2.2.4. Regression Models:

- The Regression models like XGBoost, Linear Regression & Random Forest Regressors are being used to predict the future prices based on the historical patterns.

1.2.3. *Real World Insights from the dataset:*

1.2.3.1. Risk Management:

- Prediction of stock prices prevents losses and supports an investor by providing him with future possibilities so that they may decide upon the risks involved and take an early step to minimize their risk.

1.2.3.2. Portfolio Optimization:

- It provides the investor with market insights giving them an upper hand to invest cautiously in a variety of stocks and earn maximum returns and eventually optimize their portfolio.

1.2.3.3. Corporate Decision Making:

- By accessing possible future prices corporate decision making becomes a bit easier as major decisions such as of Mergers & Acquisitions are being taken by the companies if there is a crash in prices of the opposite party (the company which will be dissolved).

2. CHAPTER 2: LITERATURE REVIEW

1. Introduction

The prediction of stock prices has remained a vital aspect of study in finance owing to its effect on investment decisions, strategies, and risk management. The time-proven methods of forecasting, such as Exponential Smoothing (ETS) and Seasonal Autoregressive Integrated Moving Average (SARIMA), have used traditional time-series models based statistical frameworks while emerging machine learning techniques, especially Long Short-Term Memory (LSTM) networks, have begun to harness more sophisticated nonlinear relationships arising in market dynamics and temporal dependencies.

2. Traditional Time-Series Models: ETS and SARIMA

ETS models rely on weighted averages of past observations and assign exponentially decreasing weights to older data points (Hyndman et al., 2008).

In its most basic form, an ETS model averages out each observation, placing greater weight on more recent data, trend, and seasonality factors. Still, these methods capture, at least in basic form, each component and capture trends and seasonality.

SARIMA extends the ARIMA model by incorporating seasonal effects making it particularly useful for financial series with cyclical behaviour (Box et al., 2015).

The literature argues that SARIMA has a decent performance benchmark for short-term forecasts, especially when looking at the univariate time series data with regular seasons (Ahmed et al., 2010; Zhang et al., 2017).

However, both models assume linearity and stationarity, which restricts them in a more volatile market (Tsay, 2005).

3. Emergence of Deep Learning: LSTM Networks

LSTM, a type of recurrent neural network (RNN), was designed to overcome the vanishing gradient problem in traditional RNNs (Hochreiter & Schmidhuber, 1997). Due to its gated architecture, LSTM can capture long-range dependencies, making it ideal for financial time series prediction.

Fischer and Krauss (2018) demonstrated that LSTM outperforms traditional models and even some shallow neural networks in directional stock prediction on the S&P 500. Similarly, Nelson et al. (2017) integrated technical indicators with LSTM, significantly improving predictive accuracy.

4. Hybrid and Comparative Approaches

Several researchers have compared LSTM with statistical models like ARIMA and SARIMA. Chong et al. (2017) found LSTM models to be more robust in volatile markets, while ARIMA showed better performance in stable conditions.

Zhang et al. (2020) proposed a hybrid LSTM-SARIMA model, showing improved results by leveraging SARIMA's short-term linearity and LSTM's non-linear sequence learning. Similarly, Nayak et al. (2021) integrated LSTM and ETS for Bitcoin price prediction, achieving lower error metrics compared to individual models.

5. Challenges and Future Directions

Despite LSTM's success, challenges remain. Overfitting, lack of interpretability, and sensitivity to hyper parameters are critical issues (Zhang et al., 2018). For ETS and SARIMA, their reliance on stationarity assumptions may underperform in real-world, non-linear market dynamics (Hyndman & Athanasopoulos, 2018).

Future directions include hybridization, attention mechanisms, and reinforcement learning-based strategies.

References:

- Ahmed, N. K., Atiya, A. F., Gayar, N. E., & El-Shishiny, H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29(5-6), 594–621.
- Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). Wiley.

- Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83, 187–205.
- Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Hyndman, R. J., Koehler, A. B., Snyder, R. D., & Grose, S. (2008). *Forecasting with Exponential Smoothing: The State Space Approach*. Springer.
- Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice* (2nd ed.). OTexts.
- Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *International Joint Conference on Neural Networks*, 1419–1426.
- Nayak, S. C., Mishra, M., & Rath, A. K. (2021). Hybrid LSTM and exponential smoothing model for Bitcoin price prediction. *Journal of Ambient Intelligence and Humanized Computing*, 12(1), 669–679.
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. *Expert Systems with Applications*, 42(1), 259–268.
- Tsay, R. S. (2005). *Analysis of Financial Time Series* (2nd ed.). Wiley.
- Zhang, G., Eddy Patuwo, B., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35–62.

- Zhang, L., Aggarwal, C. C., & Qi, G.-J. (2017). Stock price prediction via discovering multi-frequency trading patterns. *Proceedings of the 23rd ACM SIGKDD*, 2141–2149.
- Zhang, X., Qu, Y., & Zhu, Y. (2018). Stock market prediction via multi-source multiple instance learning. *Information Fusion*, 50, 37–44.
- Zhang, Y., Zohren, S., & Roberts, S. (2020). Deep learning for portfolio optimization. *The Journal of Financial Data Science*, 2(4), 8–20.
- Shen, S., Jiang, H., & Zhang, T. (2012). Stock market forecasting using machine learning algorithms. *Stanford CS229 Report*.
- Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90, 106181.
- Kim, K. J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1-2), 307–319.
- Dixon, M. F., Halperin, I., & Bilokon, P. (2020). *Machine Learning in Finance: From Theory to Practice*. Springer.
- Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques – Part II: Soft computing methods. *Expert Systems with Applications*, 36(3), 5932–5941.
- Kara, Y., Acar Boyacioglu, M., & Baykan, Ö. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, 38(5), 5311–5319.
- Fischer, M., Krauss, C., & Horn, M. (2021). Deep learning in stock price prediction: A review. *Big Data*, 9(4), 278–293.

- Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, 42(20), 7046–7056.
- Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. *International Conference on Advances in Computing, Communications and Informatics*, 1643–1647.
- Qiu, M., & Song, Y. (2016). Predicting the direction of stock market index movement using an optimized artificial neural network model. *PLoS ONE*, 11(5), e0155133.

3. CHAPTER 3: METHODOLOGY

3.1 Overview

This project aims to analyse past data and predict the stock price of Google Inc. (GOOGL). Predicting a stock price has always posed a complicated problem in stock market analysis because of the constantly changing and chaotic behaviour of the stock market.

To achieve this, the project implements an RNN (Recurrent Neural Network) architecture using an LSTM model (Long Short-Term Memory). An LSTM unit is more convenient for sequence prediction since it can learn long-term dependencies which combine temporally ordered data, like stock prices over time.

This project pipeline consists of:

Cleaning and pre-processing data

LSTM modelling for sequential modelling

ETS Model

A Hybrid Model of ETS + LSTM

A stacked model Ensemble.

Tuning of hyperparameters

Evaluation and visualization of the model

Save the model for future use

3.2 Data Description

The data has been compiled from datasets of GOOGL which include daily stock prices from the year 2020 till 2025 and as seen in the following attributes:

- Date
- Open
- High
- Low
- Close
- Volume
- Adj Close

Close was selected among others as the prime target variable for predicting purposes which represented the market consensus of price for the final trading day.

3.3 Data Pre-processing

Key Steps:

1) **Date & Time Conversion:**

   The Date column is converted into datetime format to allow for chronological processes.

2) **Sort the Data:**

   Before Model preparation, it is necessary to sort the data chronologically as it becomes an essential step for Time - Series Data.

3) **Feature Selection :**

   The LSTM model is processed using only the 'Adjusted Close' column. This gives the model a focus on the most relevant price for investors.

4) **Scaling :**

   Since the LSTMs are very sensitive to the scale of input features, MinMaxScaler definitely normalizes values between 0 and 1.

   Training convergence is enhanced if such mechanisms are implemented.

5) **Sequence Generation:**

   In LSTM, the input to the model should be three-dimensional, for example, [samples, time_steps, features].

   In the custom function, 60 days' worth of prices are generated to predict the 61st day's price.

   For example:

   Input (X): Price from day 1 to day 60.

6) **Train Test Split:**

   The data is split into training and testing in a ratio of 70-30, i.e. ,
   Train Data – 70%
   Test Data – 30%

Along with preservation of the data sorted in earlier steps.

3.4 Model Implementation

3.4.1 Exponential Smoothing (ETS)

I have applied the ETS Model on the data to find the linear trend & seasonality in the stock prices using classical statistical techniques. It is implemented using the `ExponentialSmoothing` method from the statsmodels library and configured with additive trend and seasonal components. It is then used as a base model for further forecasting in the hybrid model.

3.4.2 Long Short-Term Memory (LSTM)

The LSTM model is implemented to capture the non-linear trends & long term dependencies in the stock prices.

The LSTM neural network was implemented using TensorFlow & Keras. The architecture included:

- **Input Layer:** Sequences of lagged close prices
- **Hidden Layers:** 1–2 LSTM layers with 50–100 units
- **Dropout Layers:** Added to mitigate overfitting
- **Output Layer:** Single neuron for next-day price prediction
- **Epochs:** 50–200
- **Batch Size:** 32–64
- **Optimizer:** Adam
- **Loss Function:** Mean Squared Error (MSE)

A sliding window approach was used to create sequences for supervised learning.

3.4.3 Hybrid Model (LSTM + ETS) :

The hybrid model has been implemented to capture the strengths of both LSTM & ETS providing a better evaluation of the model.

ETS is used to capture the forecasted prices using linear trends. Residuals are calculated by differencing the actual prices from the predicted prices.

Residuals are then used to capture the remaining non-linear components using LSTM.

The final forecast is the sum of the ETS forecast and the LSTM-predicted residuals, integrating both statistical and deep learning insights.

3.4.4. Stacked Model:

The stack model uses ensembling to combing the results of both the models. In this project, both LSTM & ETS are used for ensembling and preparing the stacked model. Their individual predictions are used as inputs to a meta-learner, a Linear Regression model. It is trained to learn best combinations of base model to improve the model accuracy.

3.4.5. Hyperparameters Tuning:

It is conducting using Keras-tuner library and a RandomSearch strategy. It is a deep learning technique to autotune the parameters to find the best value from the trained data minimising the validation MSE, also includes early stopping to prevent data from overfitting.
Several parameters optimized are:

- No. of LSTM units (Ranging 50-128)
- Dropout Rates (Ranging 0.2 to 0.5)
- Learning Rates (Ranging 0.001 to 0.01)
- Batch Size (16,32,64)

3.5 Model Evaluation

For model evaluation all metrics used are listed below, these are :

- **Mean Absolute Error (MAE):**

  It is a measure used for getting the absolute deviations average in the data and is considered useful in financial data to get the reasonable errors involved.

- **Root Mean Squared Error (RMSE):**

  It is used for penalising large errors as it squares the mean errors and used the root of squares for the evaluation. It makes errors more sensitive to outliers and volatility in stock price data. It plays a major role in forecasting where deviations play a major role in results. The lower RMSE indicates a better model fit.

- **Mean Absolute Percentage Error (MAPE):**

  MAPE is particularly useful for investors or financial analysts looking to understand forecast accuracy in proportional terms.

- **Directional Accuracy (for trend prediction):**

  This metric compares the direction of change in the predicted values to the actual direction of movement from one day to the next. A high directional accuracy indicates that the model is not just getting the numbers close but also correctly forecasting market sentiment or momentum, which is crucial for trading strategies.

The statistical significance of performance differences has been tested using paired t-tests and residual analysis.

3.6 Tools and Technologies

- **Programming Language:** Python 3.10, Google Colab
- **Libraries:**
  - `statsmodels` for ETS
  - `sci-kit learn` for pre-processing
  - `TensorFlow` and `Keras` for LSTM
  - `pandas`, `numpy`, `matplotlib`, and `Seaborn` for data manipulation and visualization
  - ta for technical indicators

4. CHAPTER 4: DEVELOPMENT OF MODEL

4.1. Data Loading & Inspection:

- Loaded historical stock price data of Google from a cleaned CSV file.
- Focused on the ‘Adjusted Close’ column for modeling.
- Verified data consistency and structure before processing.

4.2. Data Preprocessing:

- Converted the ‘Date’ column to datetime format and sorted data chronologically.
- Normalized the ‘Adjusted Close’ prices using MinMaxScaler to scale values between 0 and 1.
- Created time-series sequences: 60 days of past data were used to predict the next day’s price.
- Split the dataset into training and testing sets while maintaining the sequence order (no shuffling).

4.3. LSTM Model Development:

- Built the LSTM using Keras with 1–2 layers and 50–100 units.
- Added dropout layers to prevent overfitting.
- Used a Dense output layer with one neuron to predict the next day's price.
- Compiled the model with Mean Squared Error (MSE) loss and Adam optimizer.

4.4. ETS Model Development:

- Used statsmodels’ `ExponentialSmoothing` to model trend and seasonality.
- Configured the model with additive components.
- Served both as a baseline forecast model and a component of the hybrid model.

4.5. Hybrid Model (ETS + LSTM):

- Generated baseline forecasts using the ETS model.
- Calculated residuals (actual - ETS prediction).
- Trained a separate LSTM model on these residuals to capture nonlinear patterns.
- Combined ETS forecasts and LSTM-predicted residuals for final predictions.

4.6. Stacked Model (Ensemble Approach):

- Collected predictions from both ETS and LSTM models.
- Trained a Linear Regression model as a meta-learner to combine base model predictions.
- Produced more robust and accurate forecasts by learning optimal weights.

4.7. Hyperparameter Tuning:

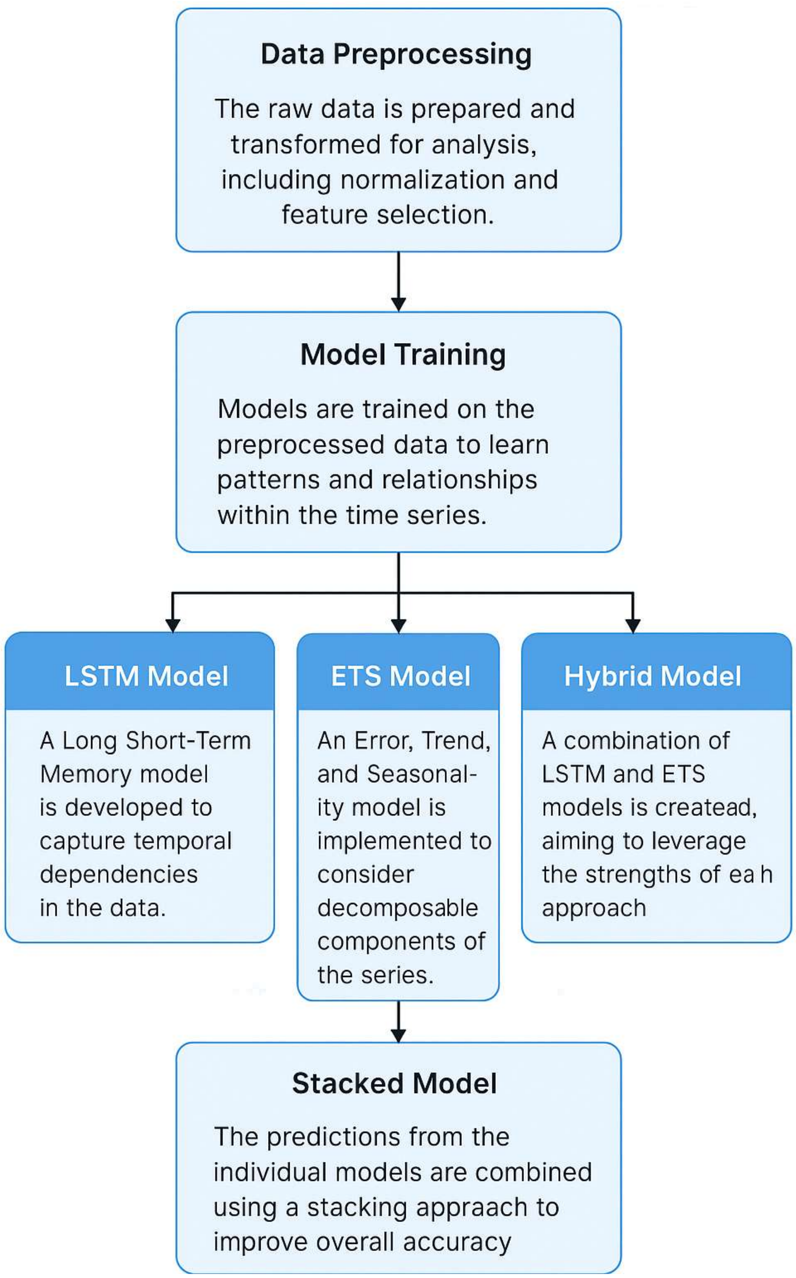
- Used Keras Tuner with RandomSearch to explore various configurations.
- Tuned parameters like LSTM units, dropout rate, learning rate, and batch size.
- Applied early stopping to halt training when validation performance stopped improving.

4.8. Model Evaluation & Visualization:

- Evaluated models using MAE, RMSE, R^2 Score, MAPE, and Directional Accuracy.
- Plotted predicted vs actual prices for visual comparison.
- Used training and validation loss plots to diagnose overfitting or underfitting.

4.9. Model Saving & Export:

- Saved trained models and scalers using `joblib` for easy reuse or deployment.
- Ensured reproducibility and potential for integration into applications.



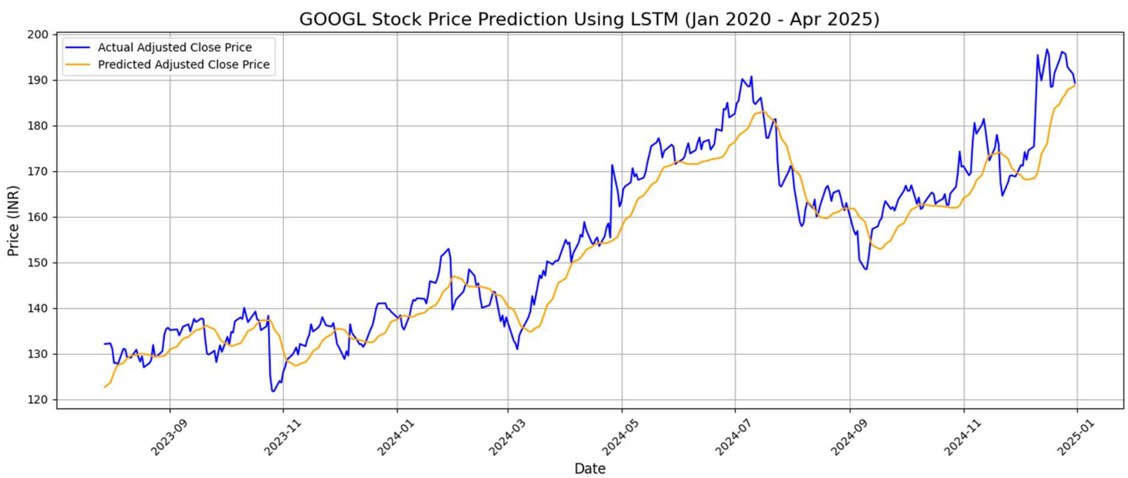
5. CHAPTER 5: RESULTS & DISCUSSION

5.1. Results Summary:

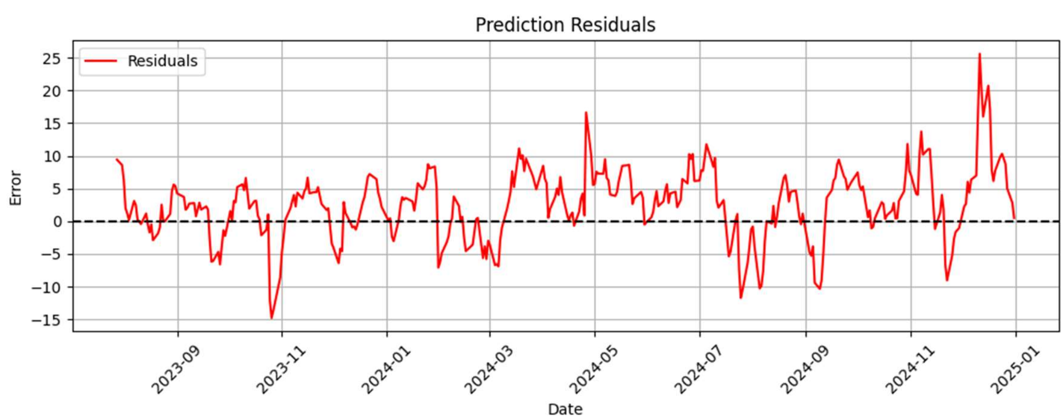
5.1.1. LSTM:

- MSE: 34.9942
- RMSE: 5.9156
- MAE: 4.6377
- Directional Accuracy: 49.37%

- Graphical Prediction -



- Residual Predictions (Showing volatility) :

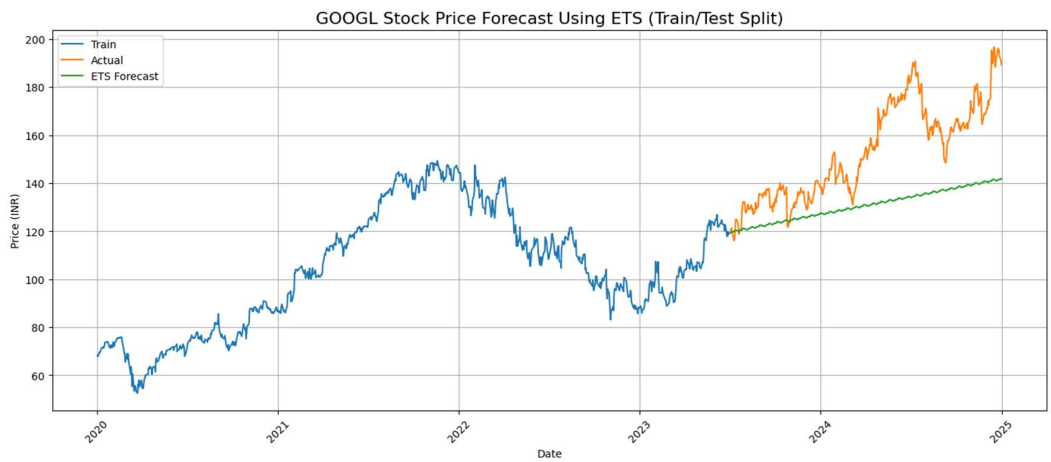


The LSTM Model summary shown above shows the benefits and advantages of this model to be used. 49.37% directional accuracy is not upto the mark and thus this model can not be solely used in results prediction.

5.1.2. ETS :

- MSE: 700.5763
- RMSE: 26.4684
- MAE: 22.1460
- Directional Accuracy: 51.88%

- Graphical Representation –

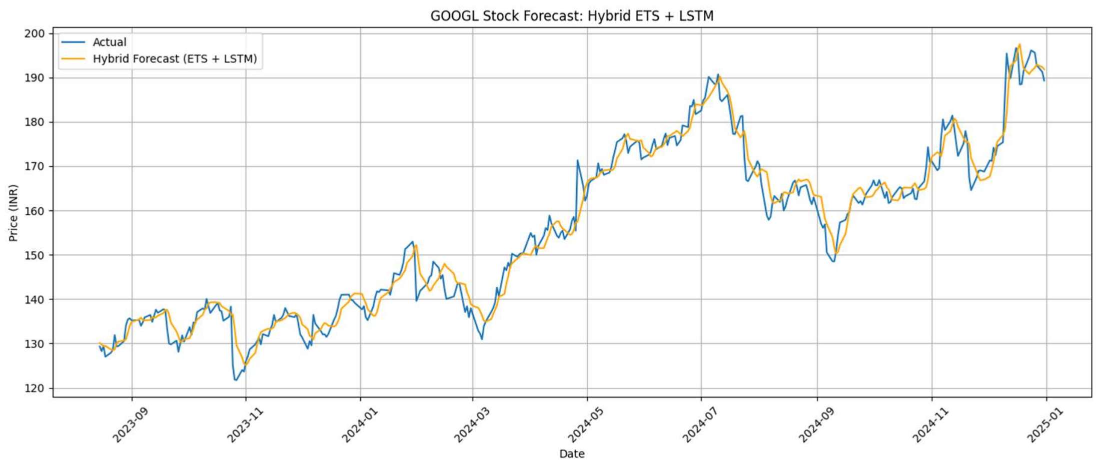


ETS model plays a very less role due to its insignificant values, i.e. , a very high MSE and lower directional accuracy but is forming the baseline for the hybrid and stacked models thus becoming significant for the model development.

5.1.3. *Hybrid Model (ETS + LSTM) :*

- MSE: 27.2878
- RMSE: 3.3220
- MAE: 3.9756
- Directional Accuracy: 55.32%

- Graphical Representation –

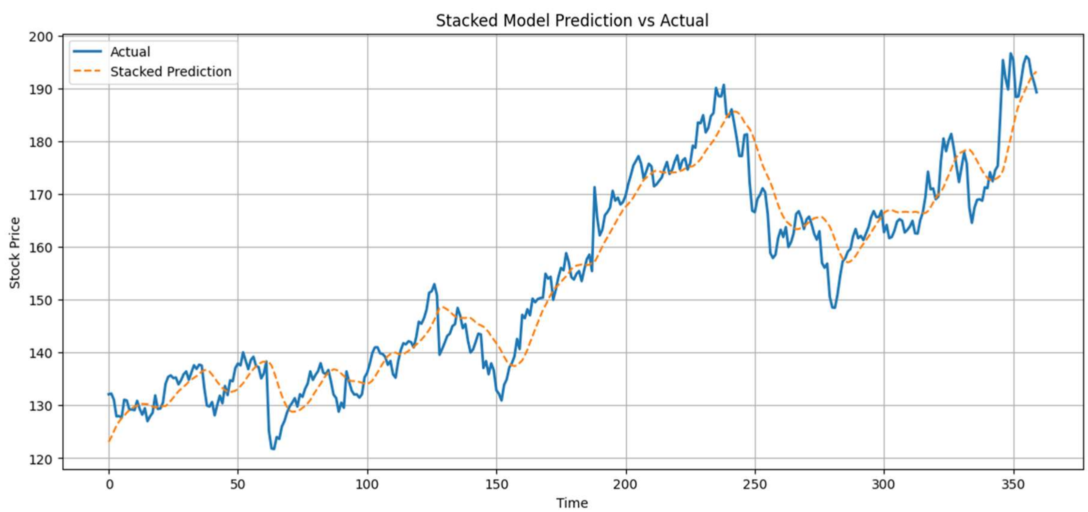


The hybrid model takes both LSTM and ETS models as the baselines and takes the best of them to prepare a model including both the models. ETS on one hand provides linear predictions while LSTM, a deep learning model, provides the non linear predictions. Thus hybrid model is better than the two models but lacks the confidence with only 55.32% directional accuracy.

5.1.4. *Stacked Model :*

- MSE: 27.2878
- RMSE: 5.2238
- MAE: 3.9756
- Directional Accuracy: 87.87%

- Graphical Representation –



The Stacked Model is considered to be the best among all and performs well with the accuracy of 87.87%. This model is also developed by LSTM & ETS and is thus significant for further predictions.

- A summary of All models :

Table1

Model	MSE	RMSE	MAE	Directional Accuracy
LSTM	34.9942	5.9156	4.6377	49.37
ETS	700.5763	26.4684	22.146	51.88
LSTM (Return-based)	92.5039	9.6179	8.1424	52.23
LSTM + ETS Hybrid	27.2878	3.322	3.9756	55.32
Stacked LSTM + ETS	16.1213	4.051	2.8862	87.87

As seen with the metrics shown in the summary table, we can easily identify that best fitted model is stacked thus showing the greatest directional accuracy of 87.87% thus being the most prevalent model for predicting the closing stock prices of google.

ETS model, on the other hand, is of least significance with MSE of above 700 shows the volatility effect implying to very less efficiency.

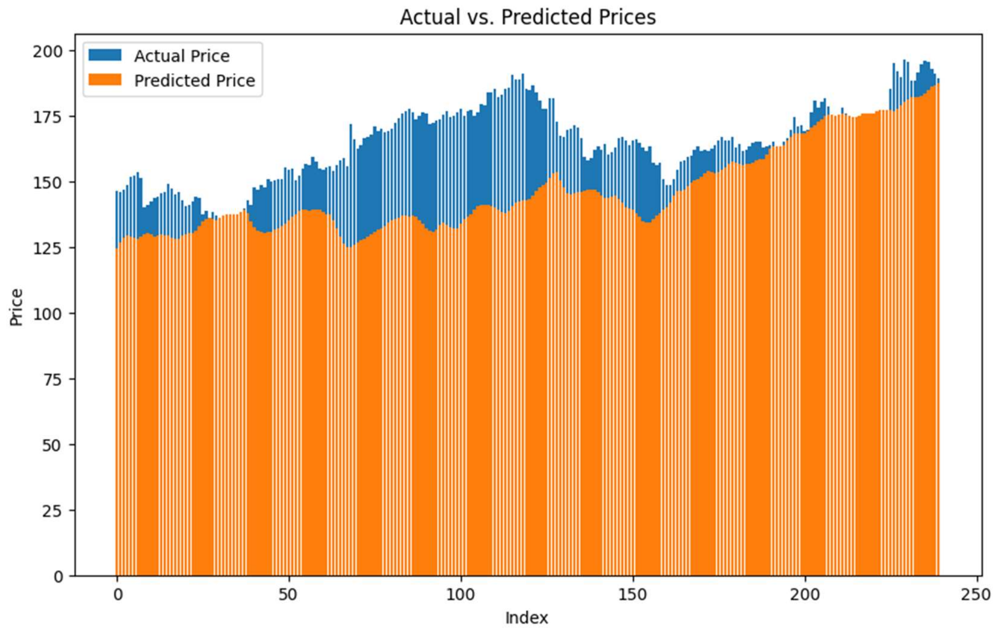
Hybrid Model and Return based models are moderately valuable and plays a significant role in development of other models.

5.2. Prediction Table & Plot :

.2.1. *Prediction table (1st Five Rows) :*

	Actual Price	Predicted Price	Error
0	146.380005	124.758125	21.621880
1	145.990005	126.908675	19.081330
2	147.039993	128.702593	18.337400
3	148.699997	129.597177	19.102820
4	151.869995	129.065844	22.804151

.2.2. *Prediction Plot (Bar plot) :*



The above table and plot shows the actual and predicted values and the difference between the two which is the error value.

This signifies the variation in the values implying to stock price volatility and also affecting the results of the model.

These 3 values namely Actual Price, Predicted Price, and The Error Values plays a significant role in evaluation of the results and further predictions to be made.

.3. Limitations of the Model :

The major limitation I felt is that the data is limited.
It Lacks Various other factors influencing the stock prices such as the news sentiments, technical indicators, Seasonality and a little bit of trend.

The data shows only a little trend affecting the model implementation results significantly showing different outputs outplaying the role of ARIMA & SARIMA models in Time – Series Data.

These lacking of information limits the number of models that can be applied and also the variety of results that can be interpreted from the data.

.4. Future Work :

- The model is now developed and saved. This will be thus be used for forecasting and can be deployed in real world after some configurations which are to be done.
- There can be addition of some information which it lacks right now to perform various other models and interpret the data in a different manner.
- Trying various other Deep Learning Models for further results and predictions to be made overcoming the disadvantages of the models which are being used and developing a further more strong model.

6. CONCLUSION

- The project implemented an LSTM model, ETS Model, A hybrid model, A reutrn Based LSTM Model and A Stacked model for predicting the stock prices of Google using historical data and technical indicators. This was done for analyzing stock market data because its design enables the capture of long-term temporal dependencies, linear and non-linear predictions and a model overcoming the cons of the other models which outperform traditional machine learning methods.

- I conducted 20 training epochs for the model before assessing its performance through regression metrics. The final results indicated that the Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) both were significantly low demonstrating the minimum error. The model demonstrated its ability to track overall market trends through visual analysis of predicted versus real stock prices which proved especially accurate during stable market intervals.

- The model demonstrated acceptable performance levels but revealed noticeable limitations. Lack of seasonality and Trend made it difficult to find the price changes occuring frequently. The model is limited to predict to an extent because of sole dependency of close prices on historical data and some basic technical indicators. This model ignores numerous real-world factors like news sentiment and macroeconomic indicators that impact stock prices.

- The LSTM model has established itself as a significant approach for financial forecasting despite having certain limitations. Future research needs to investigate additional methods to boost both accuracy and robustness and also making the predictions more reliable taking into consideration the other factors being ignored currently..

- Future research should implement advanced hyperparameter tuning using tools such as Keras Tuner for better model performance.

- The model can achieve better precision by incorporating external data sources such as news articles and social sentiment analysis.

- Future studies should include performance evaluations against other methods such as GRU and Transformer models.

- Model interpretability helps reveal the decision-making process behind model outcomes.

- This project demonstrates that LSTM-based models work well for predicting stock prices while laying groundwork for more advanced financial prediction systems in the future.

References

My Project has been uploaded on Github for reference

Link - https://github.com/Vanshik18/Googl-Stock-Price-Prediction