



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: VIII Month of publication: Aug 2023

DOI: <https://doi.org/10.22214/ijraset.2023.55450>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Unveiling Stock Market Trends Implementing LSTM and KNN Algorithms

Rasagna T¹, Kankanala L S V S Nagamani Charan²

Kalinga Institute of Industrial Technology, India

Abstract: *The objective of this study is to enhance and assess the efficacy of Long Short Term Memory (LSTM) and k-Nearest Neighbors (KNN) algorithms in the context of predicting stock prices. This study entails the acquisition of historical data from credible sources pertaining to pricing, as well as other significant variables such trade volume and market mood. Subsequently, the gathered data undergoes a process of cleansing, preprocessing, and refining to ensure its compatibility with the training model. Subsequently, we constructed a forecasting model utilizing Long Short-Term Memory (LSTM) and another forecasting model employing K-Nearest Neighbors (KNN).*

LSTM models are trained utilizing past data, and their performance is assessed through the utilization of diverse measures, including mean squared error and mean error. The aforementioned statement holds true for the K-Nearest Neighbors (KNN) model, as it employs a competitive approach to determine the optimal value for k. Ultimately, the training model is employed to forecast forthcoming stock prices, and the projected price is juxtaposed with the real stock price in order to assess the efficacy of the model. The performance evaluation and comparison of the two models have been conducted, revealing that the LSTM model outperforms the KNN model in terms of accuracy. This study makes a valuable contribution to the domain of stock price prediction by showcasing the efficacy of the LSTM and KNN algorithms in this endeavor.

Keywords: *LSTM, KNN, Stock Market Trends, Forecasting, Comparison, and Accuracy*

I. INTRODUCTION

The stock market is a multifaceted endeavor that is influenced by a multitude of elements, including political events, economic data, and public sentiment. Accurately predicting stock prices is a multifaceted undertaking that necessitates extensive research spanning several decades. With the advent of recent advancements in machine learning and deep learning methodologies, the feasibility of constructing models capable of properly forecasting stock values has become a reality.

The objective of this study is to construct and evaluate the predictive capabilities of two distinct models, namely Long Short Term Memory (LSTM) and K-Nearest Neighbors (KNN), in forecasting future stock values. The Long Short-Term Memory (LSTM) model is a variant of the Recurrent Neural Network (RNN) architecture that is particularly well-suited for prediction tasks. On the other hand, the K-Nearest Neighbors (KNN) approach is a straightforward yet effective machine learning technique mostly used for regression problems. The project will encompass multiple stages, comprising data collection and prioritization, design and training, and performance evaluation. The collection of historical data from reputable sources pertaining to prices, trade volume, and market mood will be undertaken. The initial step in our methodology involves the processing of the data to ensure its reliability and appropriateness for the training model. In the subsequent step, we shall proceed to generate and train Long Short-Term Memory (LSTM) and K-Nearest Neighbors (KNN) models utilizing the Keras and scikit-learn libraries correspondingly. In conclusion, an assessment will be conducted to gauge the efficacy of the models by employing diverse metrics, including mean squared error and mean error. By comparing the outcomes, it will be possible to ascertain the superior performing model. The findings of this study hold significant implications for market traders and investors, since precise price forecasting can enable them to make well-informed choices and optimize their financial gains.

II. BASIC CONCEPTS AND LITERATURE REVIEW

A. Pandas

The Python Pandas library is widely recognized and highly regarded for its robust capabilities in the realm of data manipulation and analysis. The software provides a range of data structures and operations that facilitate efficient manipulation of data. Pandas, which is constructed upon the foundation of the NumPy library, is regarded as a fundamental instrument for professionals in the field of data science and analysis.

B. Numpy

The NumPy package, often known as Numerical Python, is widely utilized for conducting scientific computations within the Python programming language. The software offers a very effective and user-friendly method for manipulating arrays and matrices, while also providing an extensive array of mathematical operations, including those related to array manipulation and linear algebra.

C. Matplotlib

Popular Python data visualization toolkit Matplotlib offers many capabilities for creating high-quality plots, charts, and figures. Its many customization possibilities let users produce professional visuals for scientific and technical applications. Matplotlib lets users construct line, scatter, bar, histogram, and other charts.

D. Scikit

Scikit-learn, alternatively referred to as sklearn, is a widely utilized machine learning package that is constructed upon Python's scientific computing stack.

The platform offers a diverse selection of algorithms for both supervised and unsupervised learning tasks, encompassing classification, regression, clustering, and dimensionality reduction techniques. Scikit-learn has been developed with a primary focus on ensuring user-friendly functionality and enhanced comprehensibility. The application programming interface (API) of this system has a high degree of consistency and is thoroughly documented, hence facilitating its acquisition and utilization by individuals at all levels of proficiency, ranging from novices to experts.

E. Keras

Keras is widely recognized as a prevalent open-source framework for deep learning. The software has a high degree of usability and offers robust capabilities for the construction and refinement of complex neural networks. Keras is commonly employed by researchers, data scientists, and engineers due to its straightforward yet robust application programming interface (API) and its ability to seamlessly integrate with popular deep learning frameworks such as TensorFlow, Theano, and CNTC.

F. Tensorflow

TensorFlow is a machine learning library that has been developed by the Google Brain team and is available as an open source software. The platform offers a diverse range of tools and frameworks that facilitate the construction and implementation of machine learning models across a multitude of applications, including but not limited to image recognition, natural language processing, and speech recognition.

TensorFlow has been specifically engineered to possess a high degree of adaptability and expansiveness, rendering it well-suited for the purposes of scholarly investigation and advancement.

III. EVALUATION METRICS USED

A. MAE (Mean Absolute Error)

MAE is another use of the standard deviation measurement. It is the mean of the difference between the predicted and observed results. The MAE is a linear score, meaning that all independent variables have equal weight on the mean.

B. MSE (Mean Squared Error)

MSE is also a statistical measure for regression models. It is the mean of the difference between the predicted and observed values. The MSE is a quadratic score, meaning larger deviations are given more weight than the mean.

C. R^2 Score

The R^2 score is a statistic that shows how well the model fits the data. The closed view evaluates data points based on values predicted by the model. The higher the R^2 score, the better the model meets our requirements.

D. Explained Variance Regression Score

The variance score is a measure of how well the model can explain the variance in the data. This number shows the percentage of the variance in the response variable that can be explained by the model's predictor variable(s).

IV. REQUIREMENT SPECIFICATIONS

A. Analysis

This research aims to reliably anticipate stock values using Long-Term Memory (LSTM) and K-Nearest Neighbors (KNN) algorithms. Data collection, planning, design, and performance assessment comprised the project. Data quality, model validity, and forecast accuracy affect project success. In order to reach project goals, we must thoroughly analyze data, models, and projections. Data analysis helps uncover influential aspects like trade volume, market mood, and corporate news that impact product prices.

Model parameters, hyperparameters, and performance measurements like square of error and mean error will be measured for model analysis. The model will be cross-validated to ensure it doesn't fit the training data. We will visualize projected occurrences and compare them to prior generations. We will also examine project constraints, including data availability, model complexity, and product effects from unanticipated occurrences.

Predictive algorithms to seek funds for fair and lawful enterprises are ethical. Testing will entail running data samples to check accuracy and dependability. We'll also compare LSTM and KNN results to see which model works better. This project uses LSTM and KNN algorithms to accurately anticipate inventories. To reach our aims, we give data and analytical models to investors to improve insights. The project will analyze more product price components and enhance the user interface for forecast information.

B. Design Constraint

Design constraints are constraints that must be considered when developing a forecasting model for stock price forecasting. Some of the appropriate design factors for this task are:

Data Availability: The availability of the best historical data on price is important to the accuracy of the forecast model. The project should consider the limitations of the available data, such as missing or incorrect data.

Model Complexity: The complexity of the LSTM model should be limited to ensure that it can be trained in a reasonable amount of time with the hardware included. If the model is too complex, it takes a lot of time and resources to train and may not be applicable in practice.

Regulatory restrictions: The project must comply with ethical and legal requirements such as data privacy and financial regulations. The plan should ensure that the model is not used for insider trading or other illegal activities.

Comment: The results of the model should be interpreted in a way that stakeholders such as traders, investors, and regulators can understand. The project should consider design that facilitates interpretation, such as the use of visualization techniques and explaining the decision-making pattern.

Scalability: The predictive model must be scalable so that it can handle large amounts of data and be used in real-time environments. The project should consider design parameters such as the optimization model to ensure the model is efficient and robust. By considering design criteria, planners can ensure that the forecasting model is accurate, ethical, interpretable, measurable, and actionable.

C. System Architecture

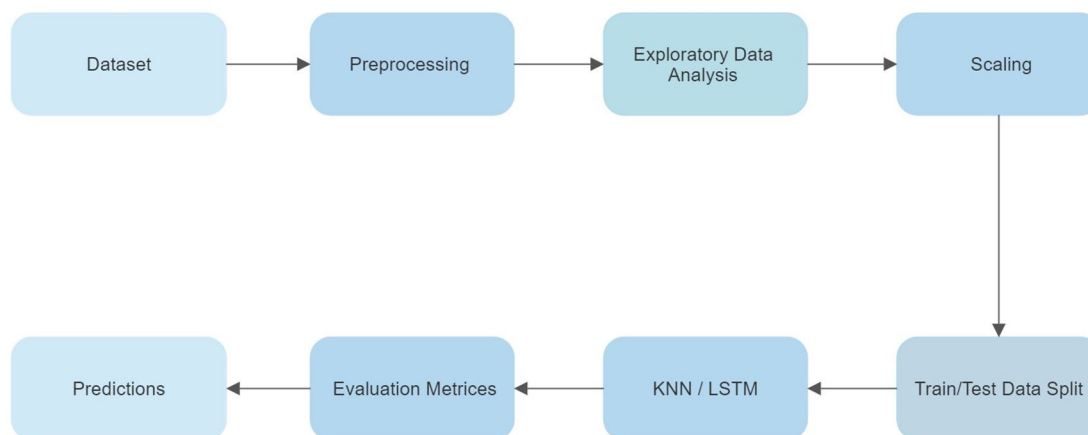


Fig.1: System Architecture OR Block Diagram

V. IMPLEMENTATION

The stock prediction project used Python and a Jupyter Notebook. This project relied on NumPy, Pandas, Scikit-Learn, Keras, and TensorFlow. Yahoo Finance used Pandas DataReader to collect historical market prices for chosen companies. The file was created after first using NumPy and Pandas. Scikit-learn's MinMaxScaler scales training data, which is separated into test and training data. Prediction uses LSTM and K-Nearest Neighbors. The Keras and TensorFlow libraries employ LSTM. A three-layer LSTM and linked processes with one output make up the model. Show the model on training data using MSE loss and the Adam optimizer.

The Scikit-learn library uses KNN. The model is trained on training data using the KNeighborsRegressor class with $k = 5$. Both models are assessed using MSE, RMSE, MAE, explained variance regression score, and R^2 Score metrics on test data. The MSE and RMSE of the LSTM model are lower than the KNN model and better at forecasting stock prices.

Overall, the LSTM and KNN models estimated the project successfully, demonstrating the LSTM's stock price accuracy. The project uses conventional coding and testing for code and control.

VI. METHODOLOGY

The approach to project estimation using LSTMs often involves several steps to ensure the accuracy and reliability of the estimation. An example of the approach that can be taken for this project is:

- 1) *Data Collection*: Gather historical data on prices from various sources, such as Yahoo Finance, Alpha Vantage, or Quandl.
- 2) *Data Preparation*: cleaning, formatting, and converting raw data into a format that can be used for training and assessment models. This includes techniques such as data normalization, data scaling, and feature selection.
- 3) *LSTM Modeling*: Use Keras or TensorFlow libraries to build LSTM models to predict future market prices based on historical data. This includes determining the number of LSTM layers, neurons, processing functions, and optimization methods best suited to the data.
- 4) *Model Training*: Train the LSTM model and KNN on a small dataset and evaluate its performance using metrics such as MAE, MSE, and RMSE.
- 5) *Model Tuning*: Adjust the hyperparameters of the LSTM model based on the results of the model evaluation to improve the accuracy of the prediction.
- 6) *Model Testing*: Test the accuracy and reliability of the LSTM model using a subset of data not used for training.
- 7) *Visualization*: Visualize event patterns and use charts, bar charts, and histograms to analyze stock prices.
- 8) *Deployment*: Deploy the LSTM model to predict future stock prices in real time.

VII. RESULTS AND ANALYSIS

A. LSTM

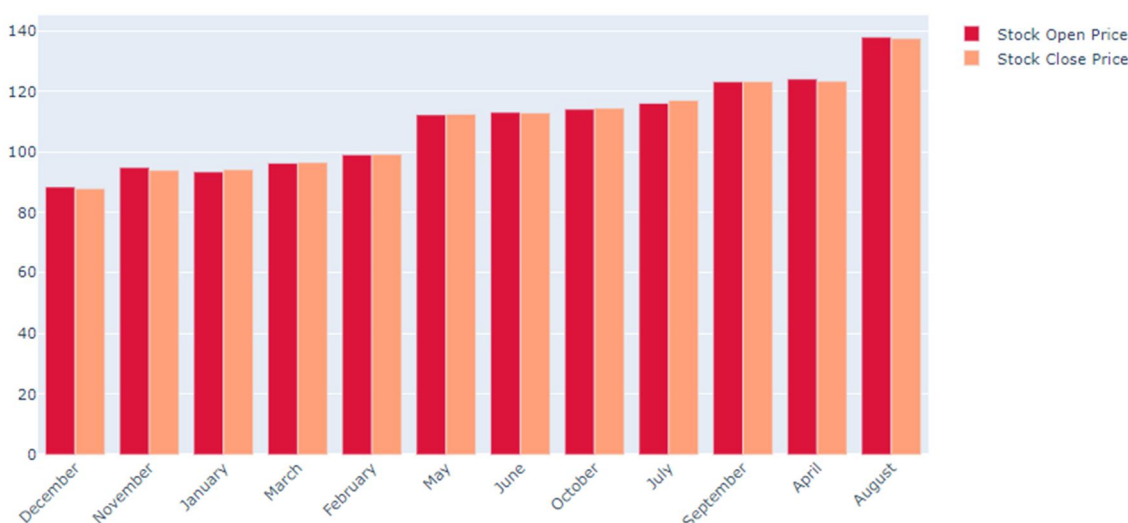


Fig. 2: Month Wise comparison between Stock actual, open and close prices

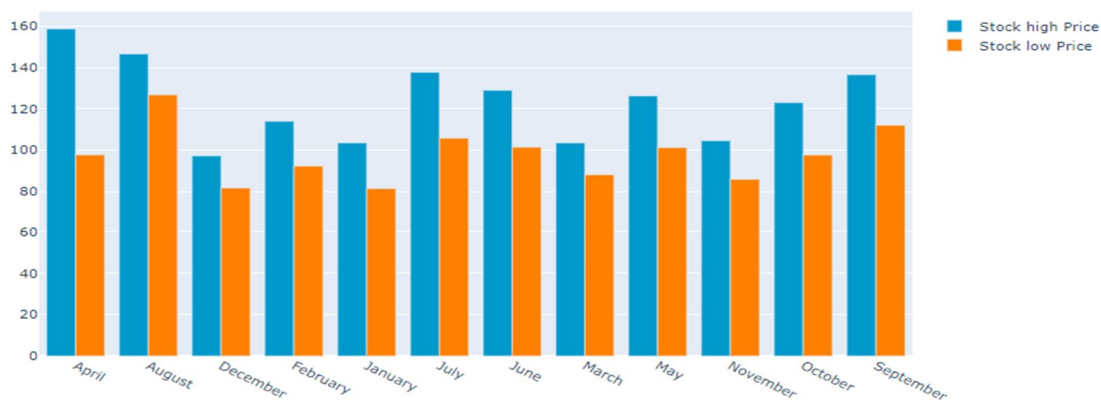


Fig. 3: Month Wise High and Low stock prices

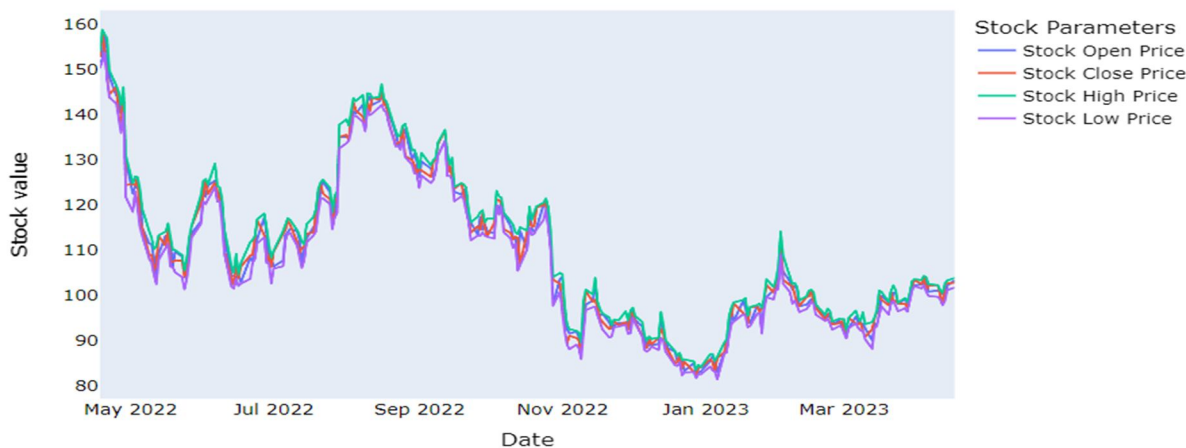


Fig. 4: Stock analysis chart



Fig. 5: Stock close price chart

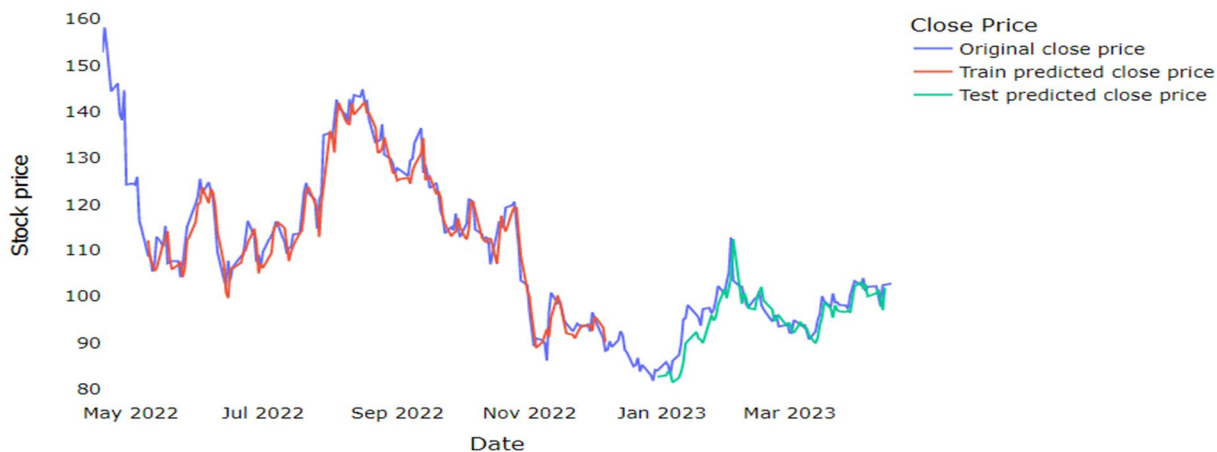


Fig. 6: Comparison between original close price and the predicted close price

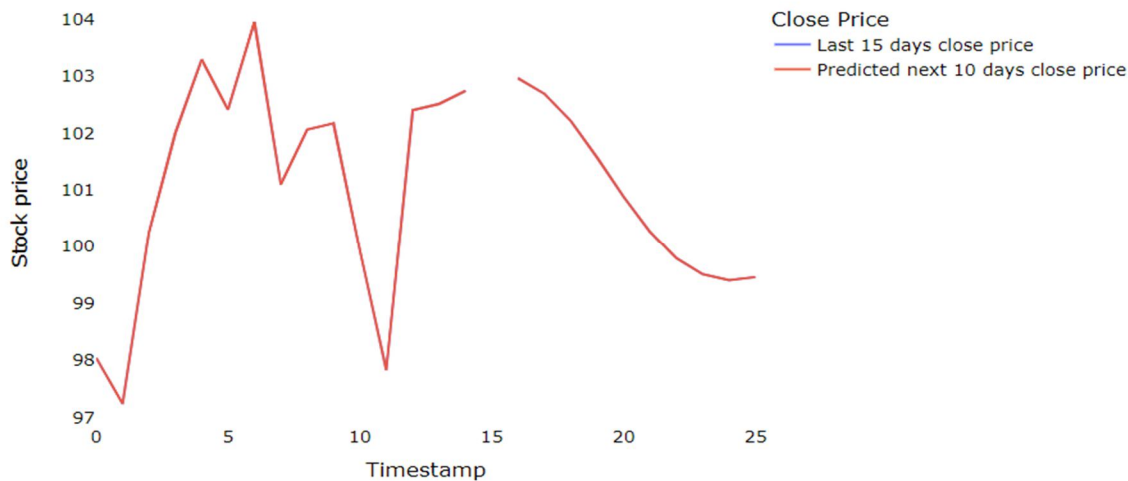


Fig7: Compare last 15 days vs next 10 days

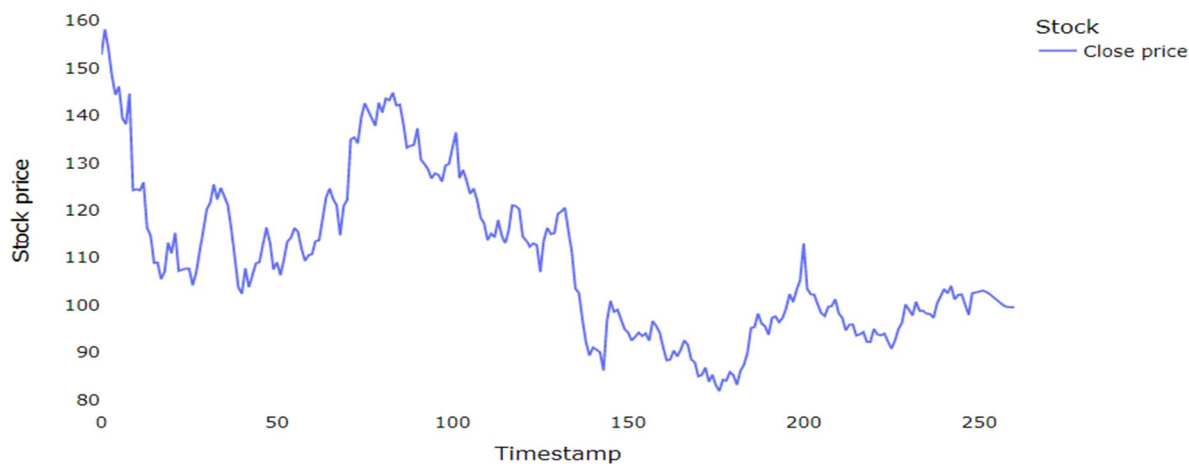


Fig8: Plotting whole closing stock price with prediction

B. KNN

Monthwise comparison between Stock actual, open and close price

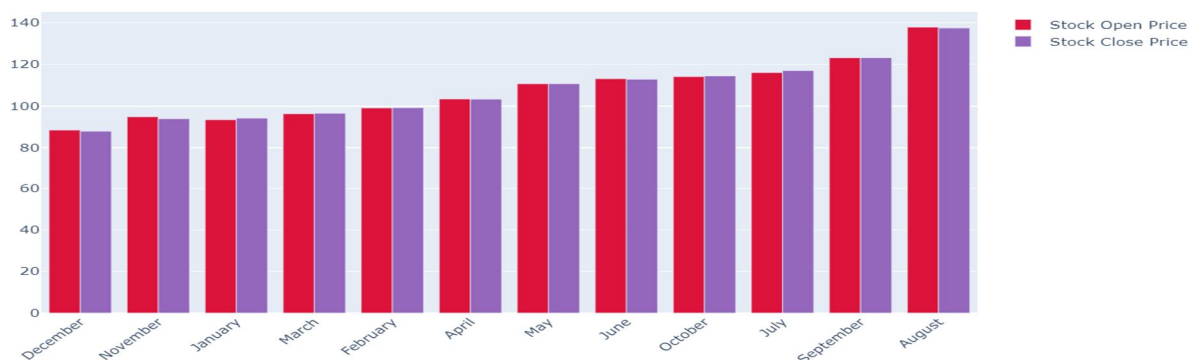


Fig. 9: Month Wise comparison between Stock actual, open and close prices

Monthwise High and Low stock price

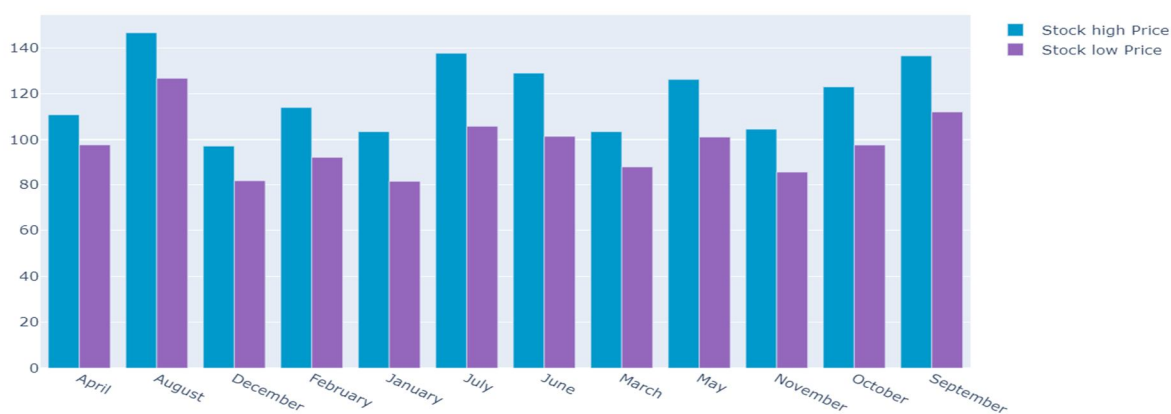


Fig. 10: Month Wise High and Low stock price

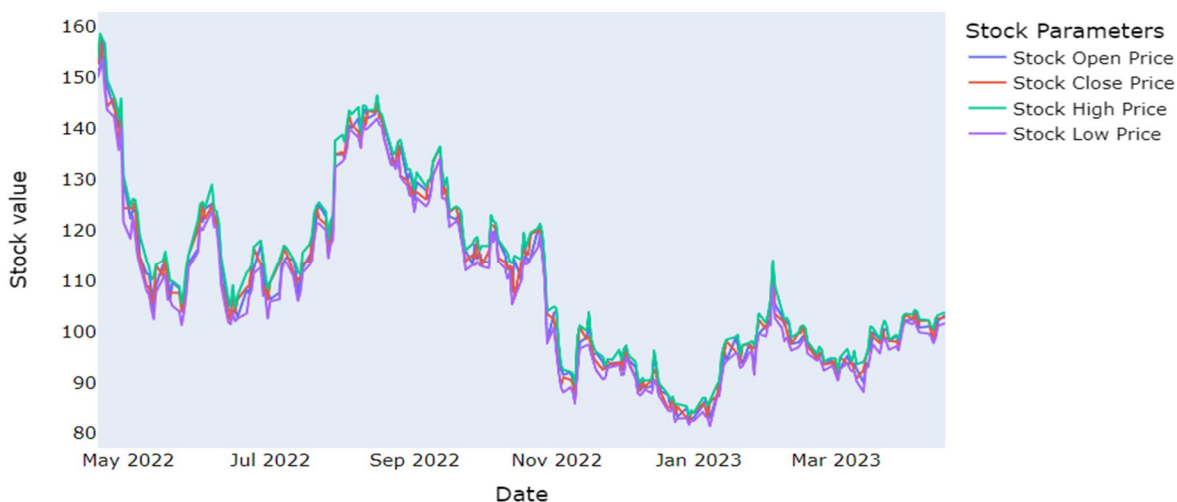


Fig. 11: Stock analysis chart

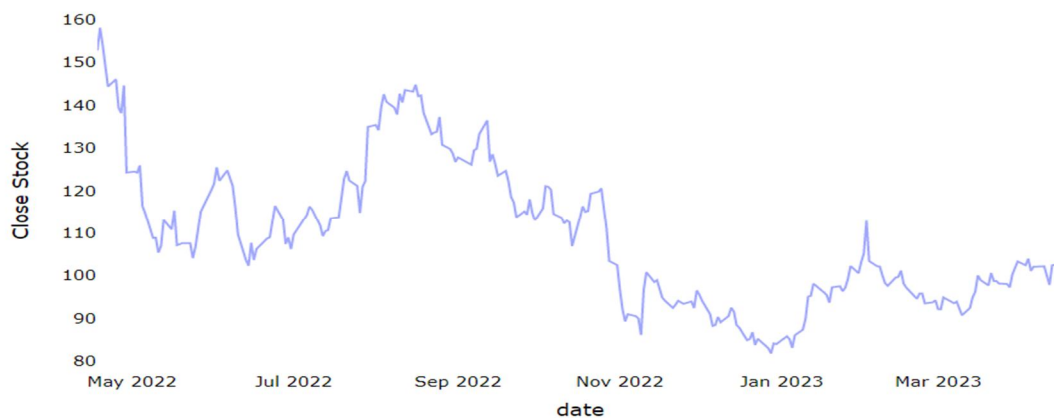


Fig12: Stock close price chart

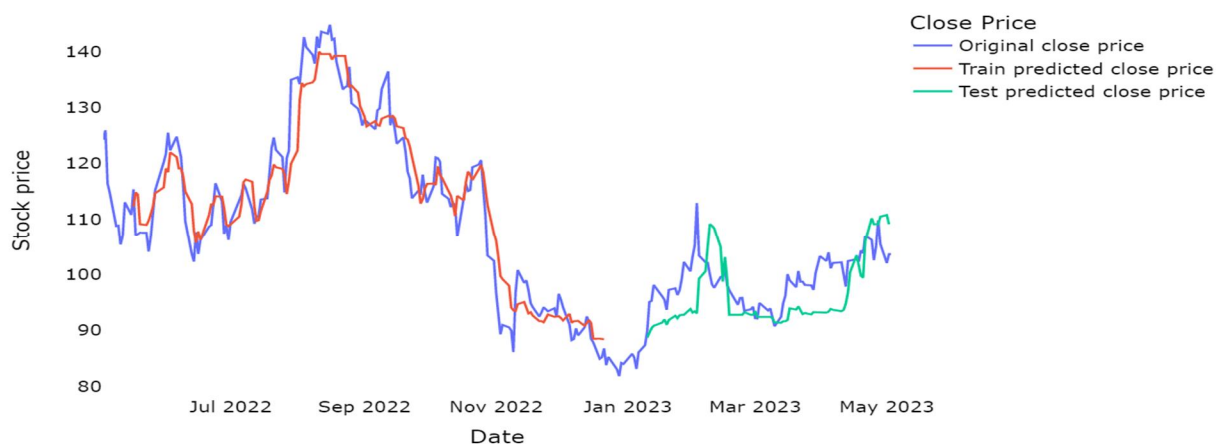


Fig. 13: Comparison between original close price and the predicted close price

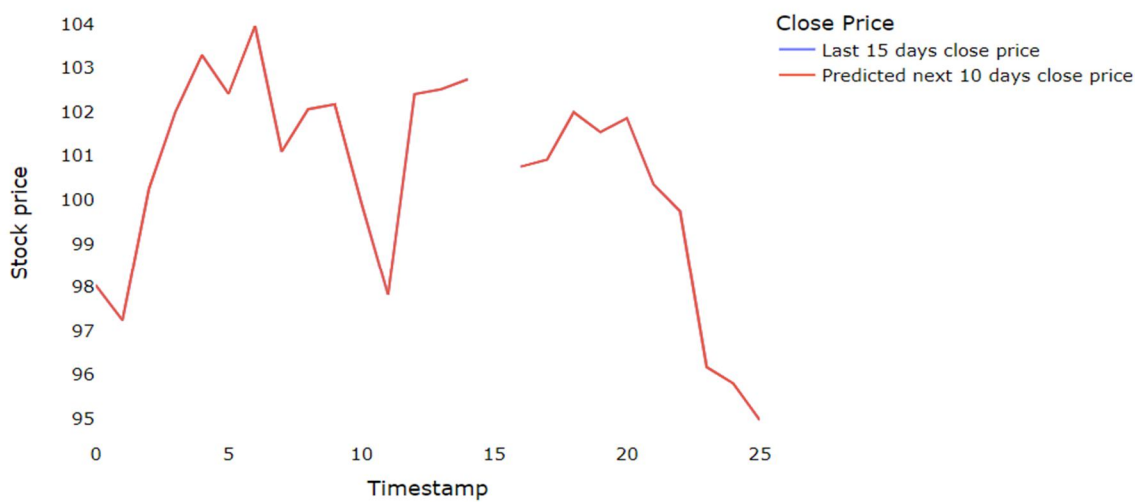


Fig14: Compare last 15 days vs next 10 days

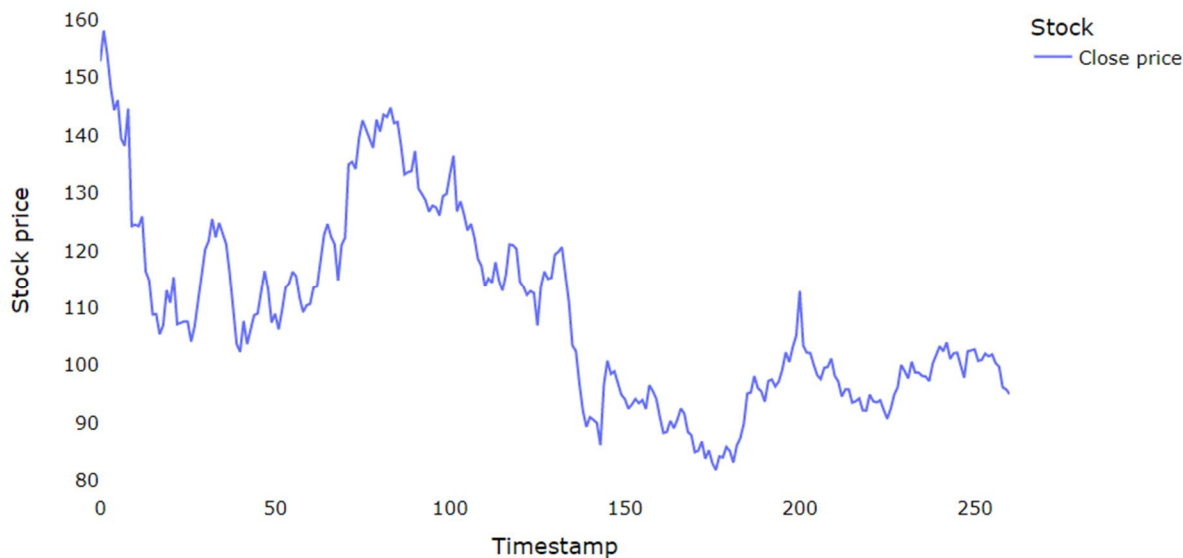


Fig15: Plotting whole closing stock price with prediction

VIII. COMPARATIVE ANALYSIS

A. Training Data

	LSTM	KNN
RMSE	3.335606591	5.661812451
MSE	11.12627133	32.05612023
MAE	2.609865596	4.270053207
Variance Regression	0.9477137884	0.8418939114
R2 Score	0.9447498871	0.8408178078

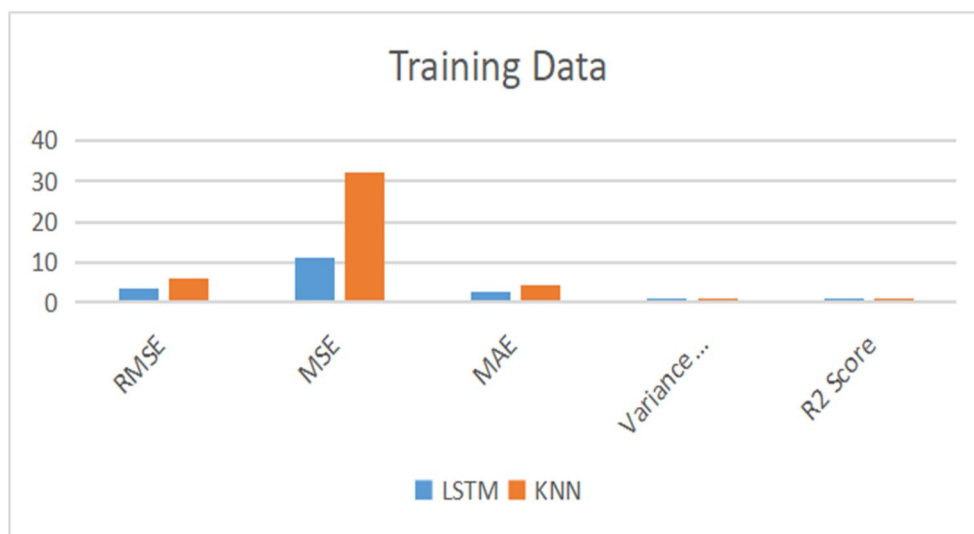


Fig.16: Training Data

B. Testing Data

	LSTM	KNN
RMSE	3.385129958	5.572973119
MSE	11.45910484	31.05802938
MAE	2.544681706	4.419565204
Variance Regression	0.6777050402	0.01736691839
R2 Score	0.5850649297	0.1246136405

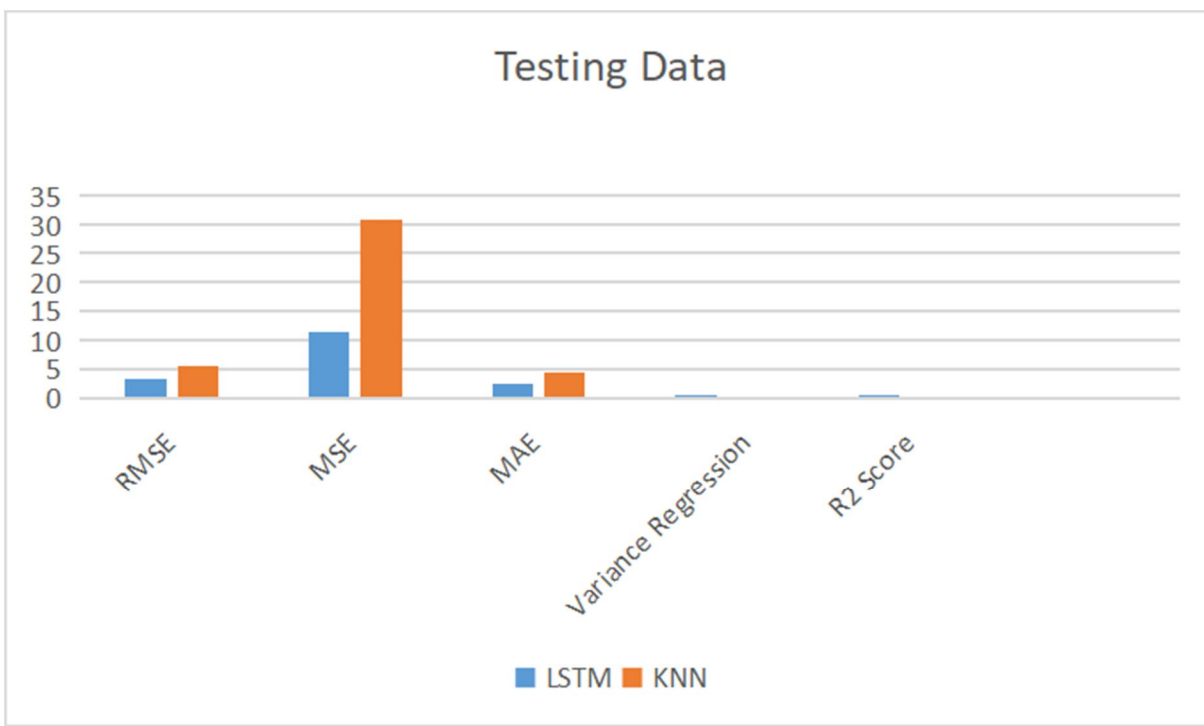


Fig17: Testing Data

From these bar graphs and the tables, we can observe that there is not much difference between the values of evaluation metrics in both the LSTM and KNN models during the training and testing times. In both testing and training data, LSTM outperforms KNN.

C. Combined Analysis

	LSTM		KNN	
	TRAIN	TEST	TRAIN	TEST
RMSE	3.335606591	3.385129958	5.661812451	5.572973119
MSE	11.12627133	11.45910484	32.05612023	31.05802938
MAE	2.609865596	2.544681706	4.270053207	4.419565204
Variance Regression	0.9477137884	0.6777050402	0.8418939114	0.01736691839
R2 Score	0.9447498871	0.5850649297	0.8408178078	0.1246136405

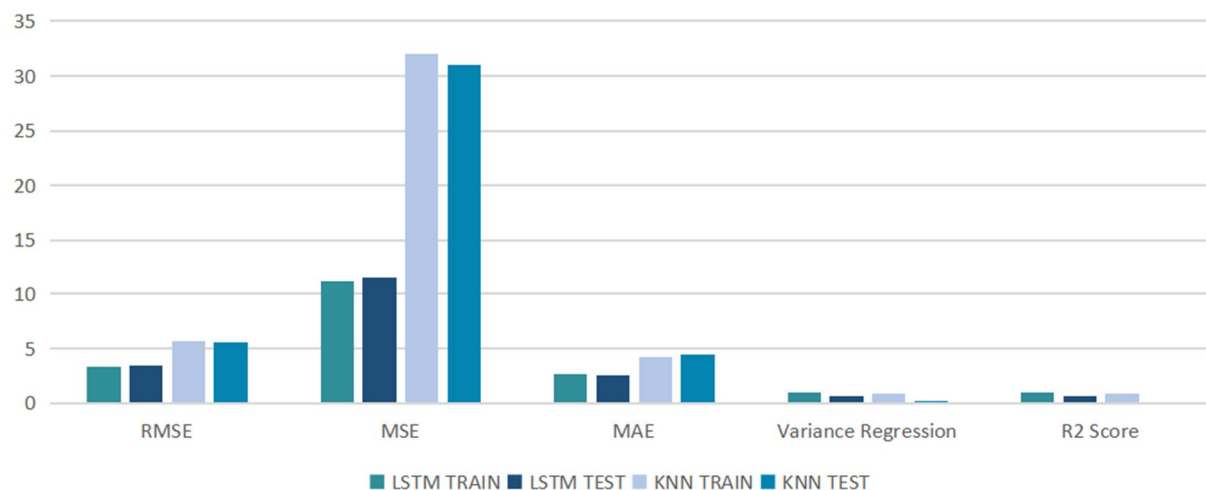


Fig18: Comparative Analysis

The table and the graphs show that, in terms of accuracy for stock market prediction, the LSTM model outperforms the KNN model. The performance measures of the two models can be compared to draw this conclusion.

In comparison to the KNN model, the LSTM model has lower values for RMSE, MSE, and MAE, thus indicating lower prediction errors. The LSTM model also exhibits a higher variance regression value and R2 score, indicating a better overall fit and an explanation for greater variance in the data.

Thus, it can be concluded that the LSTM model is preferable over the KNN model for predicting stock market prices. This result, however, may not necessarily apply to other datasets or assessment techniques because it is based on the specific data and evaluation measures employed in this investigation.

D. Quality Assurance

Quality assurance is an essential part of any project, especially machine learning models like forecasting that use LSTMs. Quality assurance processes ensure that standards are reliable, accurate, and valid and meet end-user expectations. The main points of good security for market forecasting using LSTMs are:

- 1) **Good Data:** Good data used to train and evaluate the model is important. Be mindful of the accuracy and reliability of the forecast. Data must be clean, consistent, and error-free. Data security processes may include data validation, data maintenance, and data normalization processes.
- 2) **Model Validation:** The accuracy and reliability of the model must be checked through rigorous testing and validation. This may include testing the model against a database or using cross-validation techniques to test its performance on different datasets. A valid model should include performance evaluation using metrics such as accuracy, precision, recall, and F1 score.
- 3) **Robustness:** The model must be robust and capable of meeting new needs or inputs, such as changes in the market or changes in stock prices. Strengthening can be achieved through methods such as regular work, learning together, and good work.
- 4) **Maintenance:** The model should be maintained and updated to ensure it remains accurate and viable over time. This will include monitoring performance standards, updating weak standards, and reintroducing new data standards.

IX. CONCLUSION

In brief, the objective of this study is to enhance and evaluate the efficacy of LSTM and KNN models in forecasting stock prices. The findings indicate that the LSTM model demonstrates superior performance compared to the KNN model in terms of accuracy and predictive capability. This project exemplifies the significance of basic data and the process of refining the data utilized to inform the model. Furthermore, the project serves to illustrate the significance of refining the negative model in order to achieve optimal performance. A comparative analysis of the LSTM and KNN models offers valuable insights into the respective strengths and limits of both models when employed for the prediction of stock prices. The experiment additionally showcases the potential of deep learning algorithms in the field of finance, namely in the realm of stock price prediction.



Subsequent endeavors will encompass the incorporation of other pertinent functionalities inside the database, as well as the evaluation of the efficacy of alternative machine learning methods. In its entirety, this study contributes to the extant body of literature pertaining to the prediction of stock prices, thereby offering valuable insights into the application of deep learning techniques in the realm of financial forecasting.

X. FUTURE SCOPE

There are many future directions for this project. A promising development is to explore different types of machine learning or deep learning for stock price prediction, such as convolutional neural networks (CNN) or recurrent neural networks (RNN).

Another potential area of improvement is to add additional features to the model that can improve forecasting, such as newspapers, social opinion analyses, or economic assessments.

Additionally, this project can be scaled in real time, providing real-time cost estimates using data flow technologies such as Apache Kafka or AWS Kinesis. This is useful for traders and investors who need new information to make informed decisions. As a result, this project completed two machine learning models for stock price prediction and followed the coding and testing processes. There are many potential development areas and future directions for this work, including exploring different algorithms or architectures and generating real-time price predictions, including other tasks.

REFERENCES

- [1] [Hinawi/journals/jat/2019/4145353](#)
- [2] [Predicting-stock-price-machine-learningnd-deep-learning-techniques-python/Analyticsvidya](#)
- [3] [Stock prediction machine learning/Simplilearn](#)
- [4] [Predicting stock prices using machine learning/neptune.ai](#)



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)