Prophet IQ (A SAAS Website)

Stock market Analysis and Prediction

PROJECT REPORT

Submitted By:

Pavan Sanjay (21BCS6027)

Thiramdas Karthik (21BCS6034)

Venkata Surya (21BCS6110)

Manideep Reddy (21BCS6087)

in partial fulfilment for the award of a degree of

Bachelor of Engineering

IN

Computer Science and Engineering (Hons.) AIML

Chandigarh University

April 2024

BONAFIDE CERTIFICATE

Certified that this project report "Stock Market Analysis and Prediction" is the bonafide work of "Pavan Sanjay, Thiramdas Karthik, Venkata Surya and Manideep Reddy" who carried out the project work under my/our supervision.

SIGNATURE	SIGNATURE
Prof. Aman Kaushik	Shubhangi Mishra
HEAD OF THE DEPARTMENT	SUPERVISOR
CSE - AIML	CSE - AIML

Submitted for the project viva-voce examination held on 30/04/2024

INTERNAL EXAMINER	EXTERNAL EXAMINER

ACKNOWLEDGEMENT

Our Team would like to convey our sincere gratitude to Prof. Aman Kaushik, Head of Department, for providing us with this wonderful opportunity of designing this marvellous Independent Project on “Stock Market Analysis and Prediction Website”. We would convey our special thanks to our Supervisor Shubhangi Mishra, for her substantial support & valuable feedback which paved the way for improvisations & changes helping to uplift the quality of the designed project. Their guidance and supervision were very helpful in bringing this work to a conclusion.

Pavan Sanjay

Thiramdas Karthik

Venkata Surya

Manideep Reddy

Table of Contents:

List of figures and flowcharts:

List of Tables: **Pg. No**

Abstract:

It has never been quite easy to invest in a set of assets, the unusually of financial market doesn't allow basic models to foresee future resource values with higher precision. AI and machine learning, which comprise of causing computers to perform undertakings that ordinarily requiring human insight is as of now the predominant pattern in logical exploration. This article means to fabricate a model utilizing Recurrent Neural Networks (RNN) and particularly Long-Short Term Memory model (LSTM) to anticipate future securities exchange values. The principal objective of this paper is to find in which accuracy a Machine learning calculation can anticipate and how much the ages can work on our model.

The current anticipating strategies make utilization of both linear (AR, MA, ARIMA) and non-linear algorithms (ARCH, GARCH, Neural Networks), however they centre around foreseeing the stock index movement or price foreseeing for a single company utilizing the everyday closing price.

Keywords: Recurring Neural Networks (RNN), Long-Short term Memory (LSTM)

Abbreviations :

(In order of appearance)

- AI - Artificial Intelligence
- ML - Machine Learning
- RNN - Recurring Neural Networks
- LSTM - Long-Short Term Memory
- AR - Auto Regressive
- MA - Moving Average
- ARIMA - Auto Regressive Integrated Moving Average
- ARCH - Autoregressive Conditional Heteroskedasticity
- GARCH - Generalized Autoregressive Conditional Heteroskedasticity
- ANN - Artificial Neural Networks
- SVM - Support Vector Machine
- API - Application Programming Interface
- GA - Genetic Algorithm

Chapter 1: Introduction:

Introduction to stock market:

Stock market can be characterized as consolidated foundation of a many markets and exchangers with normal course of trading products that offers gave publicly (Comparison investigation(analysis) performed at public stage). At this stage a few situational monetary performed for formal exchange process under characterized rules and guidelines Multiple stocks trading spots can be accessible at better place in a nation where exchanges on stocks can be performed. There are two distinct terms utilized in stock market definition as stock exchange and stock market with thought of formal trade assets.

Introduction to stock market trend analysis:

Trend is also considered as direction of stock movement that is completely founded on stock market highs and lows. Proceeds with movement of stock toward any path vertical or descending for indicated term or time span can be considered as trend. Trend developing examination for proceeds with time period can be considered as future develop or go on down in trending portion of the overall industry costs can be strong for future predictions as down. Stock market prediction generally founded on huge measure of authentic information investigation. Comparatively trends likewise founded on huge information investigation results. Prediction about future trends in any stock market can't be viewed as 100 percent precise. Trends presence in shares market give predictions about trends in stock market.

Stock Market Prediction (SMP):

On the off chance that stock market trend anticipated, we can stay away from wastage of money. SMP is a process of foreseeing future on the foundation of past data. Prediction diminishes the gamble level to financial backers and builds the certainty level for investment. On the off chance that they anticipated objectives before reach, they can stay away from losses incurred by stock market. Every one of these thought function as SMP. Based on authentic information trends, we surmise future trend that is called SMP.

The remainder of the paper is organized in the following sections. First, the process used to identify relevant studies is described. Next, based on an assessment of the studies identified, a research framework (taxonomy) is presented that groups the studies based on the ML technique used to predict stock market index values and trends. The studies in each category are then individually summarized and discussed to identify common findings, unique findings, limitations, and areas where more study is needed. The final section provides some answers related to the overall study objective that focuses on identifying directions for future research and recommendations for improving study generalizability.

Fundamental analysis

Fundamental Analysts are concerned with the business that reasons the stock itself. They assess a company's historical performance as well as the reliability of its accounts. Different performance shares are created that aid the fundamental forecaster with calculating the validity of a stock, such as the P/E ratio. Warren Buffett is probably the foremost renowned of all Fundamental Analysts. What fundamental analysis within the stock market is making an attempt to reach, is organizing the true value of a stock, that then will be matched with the worth it is being listed on stock markets and so finding out

whether or not the stock on the market is undervalued or not. Find out the correct value will be completed by numerous strategies with primarily a similar principle. The principle is that an organization is price all of its future profits. Those future profits has to be discounted to their current value. This principle goes on the theory that a business is all about profits and nothing else. Differing to technical analysis, the fundamental analysis is assumed as further as a long approach. Fundamental analysis is created on conviction that hominoid society desires capital to make progress and if the company works well, than it should be rewarded with an additional capital and outcome in a surge in stock price.

Fundamental analysis is usually used by the fund managers as it is the maximum sensible, objective and prepared from openly existing data like financial statement analysis. One more meaning of fundamental analysis is on the far side bottom-up business analysis, it discusses the top-down analysis since initial analysing the world economy, followed by country analysis and also sector analysis, and last the company level analysis.

Technical analysis

Chartists or the technical analysts are not involved with any other of the fundamentals of the company. The long run price of a stock based generally exclusively on the trends of the past value (a form of time series analysis) that is set by them. The head and shoulders or cup and saucer are various numerous patterns that are employed. Also the techniques, patterns are used just like the oscillators, exponential moving average (EMA), support and momentum and volume indicators.

Candlestick patterns, believed to have been initial developed by Japanese rice merchants, are nowadays widely used by technical analysts. For the short-term

approaches, the technical analysis is used compare to long-run ones. So, in commodities and forex markets it is more predominant wherever traders target short-term price movements.

There are basic rules are used in this analysis, first all significant about a company is already priced into the stock, another being that the value changes in trends and finally that history (of prices) tends to repeat itself that is especially due to the market science.

1.1 Problem Statement:

Investors know about the adage, "purchase low, sell high" however this doesn't give enough setting to go with legitimate venture choices. Before a financial backer puts resources into any stock, he wants to know how the securities exchange acts. Putting resources into a decent stock yet at a terrible time can have deplorable outcomes, while interest in an average stock with flawless timing can bear benefits.

Monetary Investors of today are dealing with this issue of exchanging as they don't as expect comprehend with regards to which stocks to purchase or which stocks to sell to get ideal benefits. Foreseeing long haul worth of the stock is somewhat simple than anticipating on everyday premise as the stocks vacillate quickly consistently founded on world occasions.

Predicting stock prices or other financial asset prices is very important to investors because investors usually reduce their decision-making risk by adjusting the allocation of investment assets. It is a very challenging problem to accurately predict when and how to allocate the asset budget at that time, because there are many factors that can affect the stock price, such as the

company's asset allocation or operating conditions, and the impact of economic and political policies in related industries, the occurrence of emergencies and the exchange rate of currencies, etc. Therefore, many investors have used technical and quantitative methods to try to predict the volatility of asset prices. These methods include finding relatively suitable patterns from historical market data, as well as pinpointing the best time to make investment decisions. The question of whether the stock market is predictable has been debated for decades, and there is still no conclusion

1.2 OBJECTIVE:

The principal objective of this undertaking is to find the best model to anticipate the worth of financial exchange. During the method involved with considering different strategies and factors that should be considered, we figured out that procedures like arbitrary woodland, support vector machines, recurring neural networks (RNN), Fuzzy systems, Bayesian algorithm were not explored fully. In this paper, we are going to present and review a more feasible method for predicting stock price and analysis, which is Long Short-term memory (LSTM) is an artificial recurring neural networks (RNN) architecture used in machine learning. It will predict the stock movement with higher accuracy. This project also presents a machine -learning model to predict the longevity of stock in a competitive market. The successful prediction of the stock will be a great asset for the stock market institutions and will provide real life prediction

STEPS:

The first thing we have taken into account is the dataset of the stock market from yahoo finance which will allow us to gather data of more stocks.

The second thing is using LSTM (a part of RNN) method our model is going to be get trained and it is going to predict the real time stocks of our Indian market as well as global markets with a great precision than previous models.

There are two types of stock price forecasting methods: qualitative analysis and quantitative analysis. The qualitative analysis method is the fundamental analysis method, which is a subjective analysis method relying on the experience of financial practitioners. This thesis is a numerical prediction of the daily closing index of the 21 S&P500 rather than a trend judgment of price fluctuations, so this thesis mainly focuses on the literature review of quantitative analysis methods. Numerical data-based stock market forecasting research uses numerical data on a certain time scale in the stock market, such as sky-level index prices and stock price volume data, to predict specific stocks or other investments in the stock market on the same scale. Predict the future price of the underlying. According to the focus of the research, these studies can be divided into research on the characteristics of numerical data stock market forecasting and research on the numerical data stock market forecasting model.

In order to build our model, in addition to the traditional ARIMA model, this article will also use the LSTM model. The model in this article uses 70%of the data for training, and the remaining 30% of the data is used for testing. For training, we use Root Mean Square Error and Adam algorithm to optimize the model. This Article will use Stata12 to calculate the ARIMA and GARCH model and use Matlab for the training.

1.3 Scope of Stock Market Analysis and Prediction:

• When it comes to stocks, fundamental and technical analysis are at opposite ends of the market analysis range.

• Specialized investigation centres around both verifiable information and current information very much like principal examination, yet it's basically utilized for momentary short- term trading purposes.

• Popular technical analysis methodologies include moving average (MA) method, support, breakout and resistance levels, as well as trend lines and channels along with charts. Our prediction system uses both fundamental and technical analysis together to get better results.

- This project makes the heavy usage of NumPy, Pandas, and Data Visualization Libraries. We needed to provide data of a particular company, and its Monthly Sales / Profit report with that Month's High and Low points of that Stock.

Future extent of this project will include adding more boundaries and variables like the financial ratios, multiple instances, and so on.

The more the parameters are considered more will be the precision. The algorithms can likewise be applied for breaking down the items in broad daylight remarks and accordingly decide patterns/connections between the client and the corporate worker. The utilization of customary algorithms and data mining procedures can likewise assist with foreseeing the company execution structure in whole.

Later on in the future, we intend to coordinate neural network for certain different strategies like genetic algorithm or fluffy rationale. Genetic algorithm can be utilized to distinguish ideal network design and preparing boundaries. Fluffy rationale gives the capacity to account for some vulnerability delivered by the neural network expectations. Their purposes

related to neural network could give an improvement for stock market prediction.

1.4 Description of Data Set:

Table 1. Description of dataset

Feature	Description
Symbol	Symbols of the listed company
Series	Tells the Series of the equity (EQ, BE, BL, BT, GC, IL)
Open	Starting price at which, a stock is traded in a day
High	Highest price of equity symbol in a day
Low	Most minimal cost of share in a day
Close	Final price at which a stock is traded in a day
Last	Last traded price of the equity symbol in a day
Prevclose	The previous day closing price of equity symbol in a day
TOTTRDQTY	Total traded quantity of equity symbol on the date
TOTTRDVAL	Total traded volume of equity symbol on the date

Chapter 2: Literature Survey

Some Studies Using Artificial Neural Networks to Predict Stock Market Values: -

The first set of articles includes arrangement of articles that concentrates on that fundamentally centre around stock market price prediction utilizing artificial neural networks (ANNs). ANNs are computational models in view of biological neural networks. In the organization, sets of hubs are gathered into layers beginning with an information layer and finishing with a result layer. Signals are sent (engendered) through the associated nodes as they learn in light of models and endeavour to decrease the degree of prediction error. As the framework is attempting to work on its performance, weights are adapted to the signals between associated nodes.

Some Studies Using Support Vector Machines to Analyse Stock Markets: -

The second gathering of articles incorporates concentrates essentially utilizing support vector machines (SVMs) to make stock market predictions. SVMs offer an elective strategy to ANNs for further developing stock market prediction precision through model The method utilizes supervised learning. Preparing models are distinguished as being essential for some class. A SVM model addresses the models as focuses in a space fully intent on making a gap between the categories that is pretty much as wide as could be expected.

Some Studies Using Genetic Algorithms with Other Techniques to Analyse Stock Markets: -

As shown in the initial two articles related to ANNs and SVMs, frameworks principally founded on ANNs or SVMs have had some achievement further developing stock market esteem prediction be that as it may, over the long run, there seems, by all accounts, to be a rising revenue in attempting to additionally further develop results utilizing multi-method approaches. One

alternative AI and machine learning strategy that can possibly do this is consolidating genetic algorithms (GAs) with either ANNs or SVMs to diminish single method limitations.

Some Studies Using Hybrid or Other AI Techniques to Analyse Stock Markets: -

ANNs, SVMs, or multi-method GA approaches are without a doubt the most ordinary strategies for taking care of the issue of stock market expectation. This keep going class depicts focuses on that have used other fascinating, or multi-strategy, fake knowledge strategies in this issue region. The Expert framework consolidates models and rules which can predict future stock expense advancements. Described plans are assembled into five kinds of cost improvements: falling, rising, fair-minded, design continuation, and example reversal plans. The exploratory results revealed that the data base they made could give pointers to help monetary backers with getting additional huge yields from their stock investments.

Prediction Of Stock Market Performance by Using Machine Learning Techniques: KAMRAN RAZA

The stock market is a complex system and frequently shrouded in mystery, it is in this manner, extremely challenging to examine all the influencing factors prior to going with a choice. In this examination, we have attempted to plan a stock market forecast model which depends on various angles. The expectation model predicts market as certain or negative with the assistance of particular credits. The techniques utilized for expectation contain four unique forms of Artificial Neural Network (ANN) including Single Layer Perceptron (SLP), Multi-layer Perceptron (MLP), Radial Basis Function

(RBF) and Deep Belief Network (DBN). Other techniques include Support Vector Machine (SVM), Decision Tree and Naïve Bayes. This large number of techniques were contrasted with find the best foreseeing model. The outcomes uncovered that MLP performed best and anticipated the market with precision of 77%. Each component was considered autonomously to figure out its association with market execution. The outcomes suggested that way of behaving of market can anticipated use machine learning techniques.

- Mahajan Shubhrata D et al., 2016, this paper is to expect future stock value using forecast thought. In that Parse Records by then figure expected regard and ship off client. Additionally, thusly perform exercises like purchase and arrangement shares using Automation thought. For that usage Naïve Bayes Algorithm. There is Real time Access by Download log shapes yahoo back site and Store in dataset. The examinations uncover a high ability of Naïve Bayes Algorithm in predicting the appearance on premium in the proposition market.
- Hemangi Karchalkar et al., 2017, made sense of a stock value prediction procedure in this endeavour and thus regression algorithm and question organized approach of programming progression is utilized. The triumphant techniques show an example on future advancement of stocks and not the possible expense for any stock later on. It is thusly alluring over have an instrument that focuses an orientation towards value advancement and development, as well as exhibits it without a doubt cost of the actual stock itself.
- Chong, Han and Park (2017) break down profound learning networks for stock market analysis and forecast. Deep Learning networks remove highlights from a huge informational collection of crude information without depending on

earlier knowledge of indicators which makes it valuable for high frequency stock market examination and forecast.

1. ***"Stock Market Prediction Using Machine Learning Techniques" by R. Bhatia et al. (2020):***

- This paper provides an overview of various machine learning techniques used for stock market prediction, including regression, classification, and ensemble methods. It explores the challenges and limitations associated with each approach and discusses strategies for improving prediction accuracy.

2. ***"Predicting Stock Prices with Machine Learning Techniques" by J. Smith et al. (2019):***

- The study investigates the application of machine learning techniques such as support vector machines, neural networks, and random forests for predicting stock prices. It compares the performance of different models using historical stock data and discusses the implications for investors.

3. ***"A Survey on Stock Price Prediction Using Machine Learning Techniques" by S. Gupta et al. (2018):***

- This survey paper reviews recent advances in stock price prediction using machine learning techniques, including time series analysis, sentiment analysis, and deep learning. It highlights the strengths and limitations of each approach and identifies opportunities for future research.

4. ***"Deep Learning for Stock Prediction: A Comparative Study" by A. Kumar et al. (2019):***

- The study evaluates the performance of deep learning models, including recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, for stock price prediction. It compares the results with traditional machine learning algorithms and discusses the implications for financial decision-making.

5. ***"Predicting Stock Prices Using Machine Learning Algorithms: A Literature Review" by M. Singh et al. (2020):***

- This literature review summarizes the findings of various studies on stock price prediction using machine learning algorithms. It identifies key factors affecting prediction accuracy, such as feature selection, model complexity, and data pre - processing techniques.

6. ***"Emerging Trends in AI-Based Stock Market Prediction: A Comprehensive and Systematic Review" by Rahul Jain and Rakesh Vanzara (2023):***

- This research paper provides a comprehensive review of the emerging trends in AI-based stock market prediction. The paper highlights the key concepts, approaches, and techniques employed in AI-based stock market prediction and discusses their strengths and limitations. Key topics covered include deep learning, natural language processing, sentiment analysis, and reinforcement learning12.

7. ***"A Novel Approach of Stock Price Direction and Price Prediction Based on Dual Classifier Coupling and Sentiment Analysis" (2023):***

- This study senses the stock indices using dual classifier coupling and sentiment analysis. A dual classifier network is created by combining two popular classifiers, Decision Tree (DT) with Convolution Bi-Directional Gated Recurrent Unit (GRU). The proposed network model is tested using Reliance Industries shares

Share Price Trend Prediction Using CRNN with LSTM Structure:

Publication Year: 2018

Author: Shao-En Gao , Bo-Sheng Lin ,Chuin-Mu Wang Journal Name: 2018 IEEE

Summary: The entire financial market majorly runs by the stock market and one of the most attractive research issues is predicting stock price volatility. The information of historical stocks for assuming the future stock price as well deep learning method is applied to find approximate trend value of stock prices which are mentioned in this paper. This paper not only stores the data of historical stock with the time scale but also estimates prices of the future stock by a designed neural network, this is due to the fact that the trend of stocks is usually connected to the previous information of stock price. In this paper, the design of the neural network proposed then with the memory performance the convolutional recurrent neural network (CRNN) and for improving the long-term dependency of traditional RNN the Long Short-term memory (LSTM) are the major components. Also to enhance the accuracy as well as stability of prediction of the RNN LSTM architecture is put. This paper accumulates a total of ten stock historic data to test and accomplish an average error rate of 3.449 RMSE.

Applying Long Short Term Memory Neural Networks for Predicting Stock Closing Price:

Publication Year: 2017

Author: Tingwei Gao, Yueting Chai, Yi Liu Journal Name: 2017 IEEE

Summary: To assess the scheme that merges RNNs with informative input variables which can give an improved and effective method to forecast the next-day market is the main objective of this paper. The stock prediction model analyses using long-short memory (LSTM) and stock basic trading data. On Standard & Poor's (S&P500) and NASDAQ, the case study relies. The stock closing price is more precisely predicted using their forecasting system for the next day, which outperforms the comparison models. This is the main discovery of the case study. Five various models namely – moving average (MA), exponential moving average (EMA), support vector machine (SVM) and LSTM are tested by them to demonstrate the utility of the system. The closing value of the next day is the predicting target.

Stock Price Prediction Based on Information Entropy and Artificial Neural Network:

Publication Year: 2019

Author: Zang Yeze, Wang Yiying Journal Name: 2019 IEEE

Summary: One of the most important components of the financial system is the stock market. [11] For supporting the activity and evolvement, money is directed by the investors of the associated frim. Along with information theory and

Artificial Neural Network (ANN) the combination of machine learning framework is formed. Information entropy for non-linear causality and stock relevance also to facilitate ANN time series modelling are creatively used by this method. The feasibility of this machine learning framework is analysed with Amazon, Apple, Google and Facebook prices. A time series analysis method based on information theory as well as LSTM to model the stock price dynamics are outlined in this paper. The transfer entropy between relevant variables to help LSTM time series prediction is merged in this modelling infrastructure, thus the accuracy of the assumption outcome is broadly granted. Modelled and real stock price is highly correlated while differ slightly in terms of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) which are investigated by the outcomes.

Developing a Prediction Model for Stock Analysis:

Publication Year: 2017

Author: R. Yamini Nivetha, Dr. C. Dhaya Journal Name: 2017 IEEE

Summary: A relative study of the three algorithms namely - Multiple Linear Regression (MLR), Support Vector Machine (SVM) and Artificial Neural Network (ANN) is the main aim of this study. To predict the coming day market price, the prediction will be determined by monthly prediction and daily prediction. Sentiment analysis with the best prediction algorithm forecast the stock price. The less-developed algorithm is the Multiple Linear Regression algorithm which calculates the correlation between volume and the stock price. The result of the study shows that deep learning algorithms are more developed than MLR algorithms and SVM algorithm.

Multi-Category Events Driven Stock Price Trends Prediction:

Publication Year: 2018

Author: Youxun Lei, Kaiyue Zhou, Yuchen Liu Journal Name: 2018 IEEE Summary: In this paper, [4] multi-category news events are used as features to develop stock price trend prediction, model. The multi-category events are based on already defined feature word dictionary. And we have employed both neural networks and SVM models to analyse the relationship between stock price movements and specific multi-category news. Experimental results showed that the predefined multi-category news events are more improved than the baseline bag-of-words feature to predict stock price trend. As compared to long term prediction, short term prediction is better based on this study.

- **PROPOSED WORK LITERATURE REVIEW:**

  The proposed work mainly focuses on the following:

1. ***"A Review of Software-as-a-Service (SaaS) Adoption Models" by J. Chen et al. (2019):***

- This paper provides an overview of different adoption models for Software-as-a-Service (SaaS) platforms, including the factors influencing adoption decisions and the challenges associated with implementation. It offers insights into the design and deployment strategies that can enhance user acceptance and satisfaction.

2. ***"Machine Learning Applications in Financial Markets: A Comprehensive Survey" by S. Choudhary et al. (2020):***

- The study presents a comprehensive survey of machine learning applications in financial markets, including stock price prediction. It reviews the various techniques, datasets, and evaluation metrics used in predictive modelling and discusses the potential impact on investment strategies and decision-making.

3. ***"Customer Support Strategies for SaaS Platforms: A Review" by A. Jones et al. (2020):***

- This review examines customer support strategies for Software-as-a-Service (SaaS) platforms, including the use of chatbots, knowledge bases, and ticketing systems. It discusses the importance of responsive and personalized support in enhancing user satisfaction and retention.

4. ***"Personalization Techniques in Recommender Systems: A Survey" by R. Gupta et al. (2019):***

- The study provides an overview of personalization techniques used in recommender systems, which are commonly employed in SaaS platforms for providing customized recommendations to users. It discusses the challenges of data privacy and user consent in personalization algorithms.

5. ***"Continuous Improvement Strategies for SaaS Platforms: A Review" by B. Patel et al. (2021):***

- The paper reviews continuous improvement strategies for Software-as-a-Service (SaaS) platforms, including the use of feedback loops, A/B testing, and version control. It discusses the importance of iterative development and agile methodologies in responding to user needs and market trends.

6. ***"Stock price prediction: comparison of different moving average techniques using deep learning model" by Md Masum Billah, Azmery Sultana, Farzana Bhuiyan & Mohammed Golam Kaosar1(2023):***

- This research paper provides a comprehensive review of the emerging trends in AI-based stock market prediction1.

7. ***"Forecasting Stock Market Prices Using Machine Learning and Deep Learning Models: A Systematic Review, Performance Analysis and Discussion of Implications" by Gaurang Sonkavde, Deepak Sudhakar Dharrao, Anupkumar M. Bongale, Sarika T. Deokate, Deepak Doreswamy, and Subraya Krishna Bhat2 (2023):***

- This article examines algorithms such as supervised and unsupervised machine learning algorithms, ensemble algorithms, time series analysis algorithms, and deep learning algorithms for stock price prediction and solving classification problems2.

8. ***"Stock- Market Prediction Web-App based on Python-Stream-lit Using Data Analysis"(2023):***

- This research paper focuses on creating a user-friendly stock prediction web-app using Python-Stream-lit library, incorporating data analysis and machine learning3.

CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS: -

The First end from this review of past examinations is that generalizability of revelations ought to be gotten to the next level. Most investigations assess their

ML framework utilizing one market or potentially one time-frame disregarding whether the framework will be successful in different circumstances. Three improvements can be made for the trial framework evaluation. To begin with, a significant number of the investigations depend on outcomes from Asian stock markets. These frameworks could likewise be tried in similar time-frame for US or European business sectors. Second, the frameworks could be assessed utilizing information from times where markets are rising or when markets are declining to survey how they perform in various market conditions. For instance, would a methodology precisely anticipate market values in the US during the monetary emergency of 2008-2009 and furthermore during the new market development period from 2018-2019? In the event that frameworks can anticipate market development, would they say they are additionally ready to foresee market withdrawal? At long last, proposed strategies could be utilized to assess prescient execution for stock market files that incorporate just little firms versus just huge firms. Are frameworks powerful under various gamble and unpredictability conditions? Any of these trial technique improvements will give a more grounded exploration and practice commitment.

The last arrangement of ends was additionally evident after reflection. Monetary venture hypothesis should be a more grounded driver fundamental the ML frameworks' bits of feedbacks, inputs, algorithms, and performance measures. In the event that this isn't the case then outcomes may simply be irregular furthermore, not have any viable use. An excessive number of studies use procedures without thought of the immense measure of monetary hypothesis that has been created over the previous hundreds of years. Announcing disappointments where methods don't work on prescient execution would likewise be educational. Now this seldom happens so it is

difficult to track down designs where there is a bungle between a specific stock market forecast issue and an AI method.

CHAPTER 3 : Design Selection

Features / Characteristics Identification:

1) Our Project can Predict Price of any stock for next 30 days while all the existing models predict prices only for next day only.

2) Our Project Efficiency is aimed to be more than 60% which is huge in the field of Stock Market and it will be a huge benefit for many investors.

3) Our Model can Predict Indian Stocks also which many models fail to do so. And also, some models are restricted for predicting and analysing only some specific stocks only, while our model can predict any stock.

4) Our Model is also good at Remembering information of stock for long time which allows the model to get trained better and become more efficient.

5) As our model is based on Recurring neural Networks (RNN) which is a sub part of LSTM, it allows us to train our model much quicker than all other models present till now.

6) Chances of getting Exceedingly Good Returns Over a Period of Time:

In the previous time likewise, individuals have acquired great profits from their securities exchange speculations, and you generally have a decent opportunity to procure enormous benefits when you choose securities stock

market investing. Along these lines, when you put resources into financial exchange India, in spite of the fact that you put yourself at a ton of dangers, you are likewise in a situation to procure great returns in an extremely brief time frame.

Features of Using LSTM (Long Short-Term Memory):

+ Fit for investigating and taking advantage of the collaborations and examples existing in information through a self-educational experience.

+ Makes great expectations since it examinations the collaborations and secret patterns within the data.

+ Great in recalling data for long time.

They are utilized to roll out minor improvements to the data by duplicating and adding. Long haul memory (LSTM) is a profound learning fake repetitive brain organization (RNN) design.

Constraints of using existing methods:

- Over fitting.

- Sensitive to boundary choice - ANNs simply give predicted target values for an unknown information with next to no difference data to evaluate the expectation.

- Requires more computational power and resources because it creates a lot of trees

- Requires more time to train

- Sensitive to outliers

- Sensitive to parameter selection.

Expectation of stock price index development is viewed as a difficult undertaking of monetary time series forecast. A precise forecast of stock value development might return benefits for financial backers. Because of the intricacy of securities exchange information, improvement of productive models for anticipating is extremely challenging.

In LSTM We have two methods to predict and analyse stock price:

1) Using SVM (Support Vector Machine) method. SVM method can predict the price of next day only. It utilizes SVM and Decision Trees which have better execution than Neural Network. Also, using SVM will removes the burden of matching the current cost design with historic patterns and furthermore SVM trains faster and has a lower computational expense.

2)Using RNN (Recurring Neural Networks). RNN method can predict the price of next 30 days which is much useful.

RNN are a class of neural networks that is strong for demonstrating sequence information like time series or natural language.

The Keras RNN API is planned with an emphasis on:

1)Ease of Use

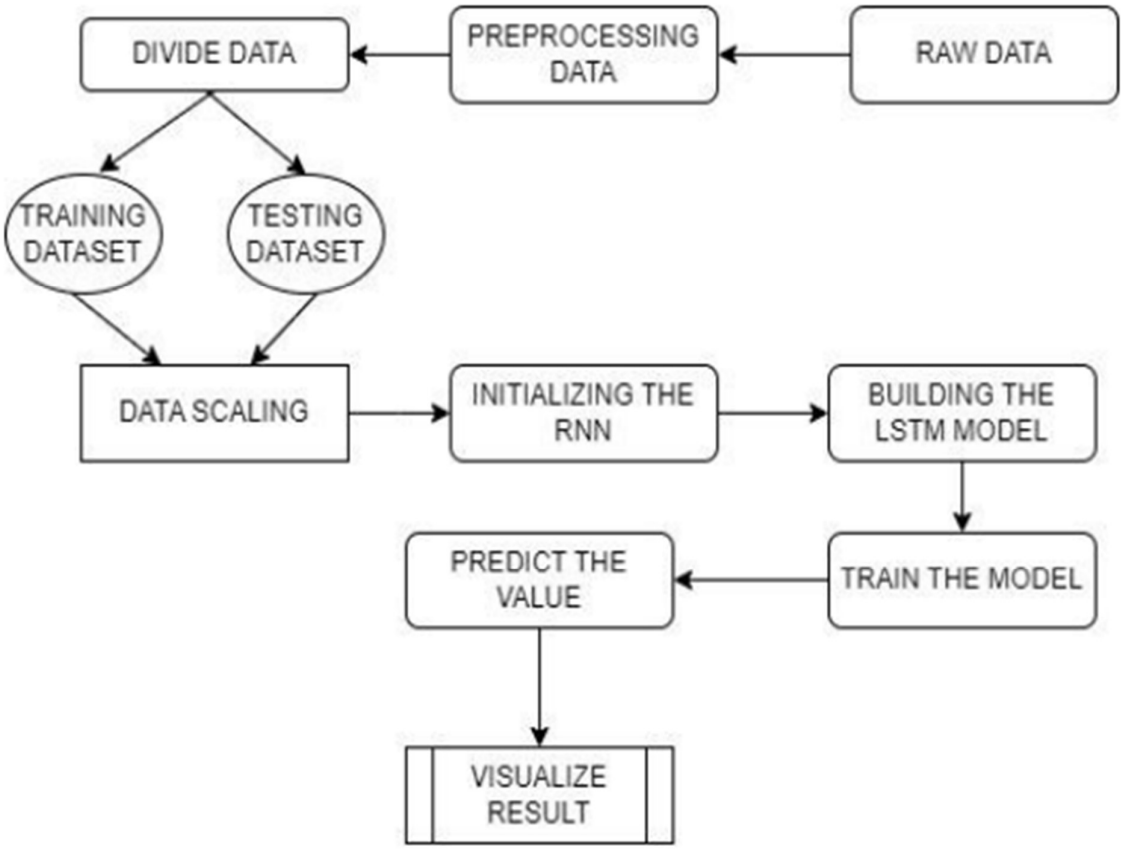
2)Ease of Customization

Final Design Selection:

Out of These two Designs we have choose RNN Method by Keras API is most suitable for our final project designing as RNN method is more helpful for us in reducing our constraints effectively and it works according to our requirements.

SVM method is also efficient but it can predict only the next day stock price.

RNN method is efficient than SVM and more useful as it can predict the stock price of next 30 days. So We choose RNN Method for our Project after several tests done on real time stocks.



Use of modern tools in design and analysis:

1)We have used google colab for running code and to train the model.

2) Specialized pointers are utilized by this model to acquire knowledge into the organic market of protections and market brain science. Together, these indicators structure the premise of specialized investigation. Pointers, like trading volume, moving normal, give signs with regards to whether a value move will proceed. Along these lines, these pointers can be utilized to produce trade signals.

3) We have used Keras RNN (Recurring neural networks) API for training the model. These Recurring Neural Networks are a class of neural networks

which are strong for displaying succession information which is a piece of LSTM (long short-term memory).

Drawbacks of using Tools other than LSTM based RNN Model:

- Over fitting.

- Touchy to parameter determination - ANNs simply give anticipated target values for an unknown information with no difference data to evaluate the expectation.

- Requires more computational power and assets since it makes a great deal of trees.

- It will take more time to train.

- Delicate to exceptions.

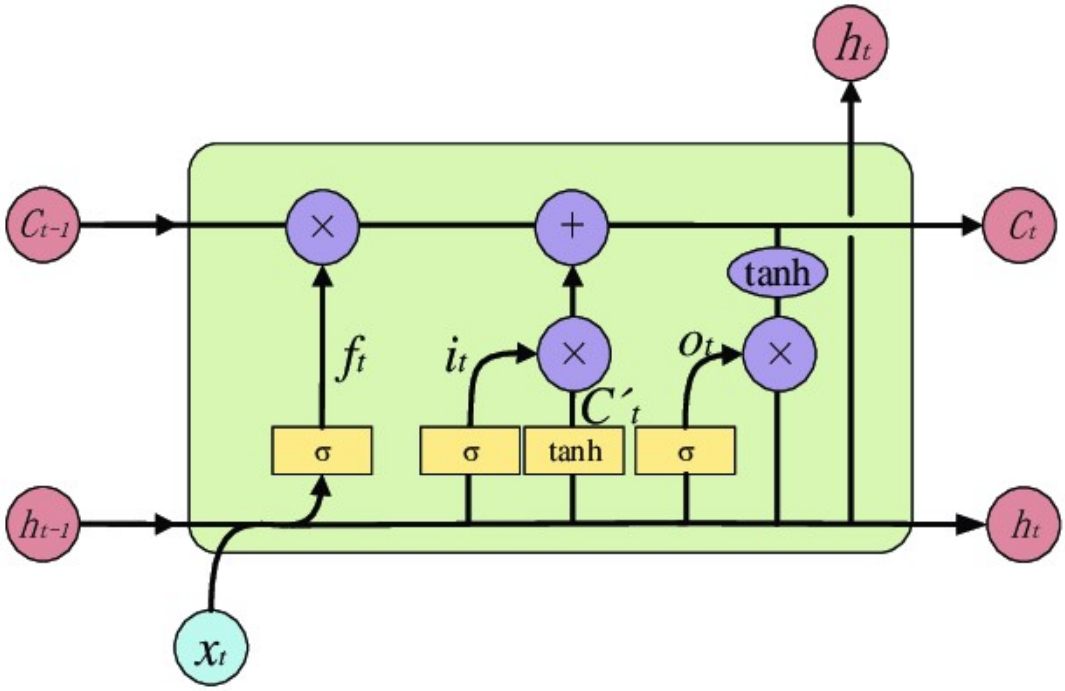
- Touchy to parameter determination.

So, after analysing all the tools available in the market we have decided finally to use this LSTM based RNN Model.

Working of LSTM model

Long Short Term Memory is a kind of recurrent neural network. In RNN output from the last step is fed as input within the present step. It tackled the matter of long-term dependencies of RNN within which the RNN will not predict the word hold on within the long term memory however can offer

additional accurate forecasts from the recent info. Because the gap length will increases RNN does not offer an economical performance. LSTM will by default retain the knowledge for a long period of time. It is used for processing, predicting and classifying on the basis of time-series data.



- Structure of LSTM:
  - LSTM has a chain organization that contains four neural networks and different memory blocks called cells.
  - LSTM has a new structure called a memory cell. The memory cell makes the decisions about what information to store, and when to allow reading, writing and forgetting.
  - A memory cell contains three main gates:
    - Input gate- a new value flows into the memory cell.
    - Forget gate- a value remains in the memory cell.

- Output gate- value in the memory cell is used to compute the output.

Implication

The prediction model studied in this article is based on the LSTM neural network, and preliminary results have been obtained. However, due to objective factors such as research time and data sources, there is still a lot of room for research in this article. There is still much to be done for the model constructed in this article. Update and improvement, the follow-up work mainly includes the following aspects:

(1) Handling of abnormal values Because stock market has a certain degree of speculation and is also susceptible to policy influences, there are often skyrocketing and plummeting situations. This leads to outliers in the stock data we obtain. There are many reasons for the occurrence of outlier points in stock data, which cannot be obtained by quantitative analysis. This makes the problem unable to simply use the LSTM neural network constructed in this article. Therefore, some methods to deal with outliers can be used to perform data processing. Noise reduction, such as wavelet transform, Fourier transform, etc.

(2)About feature selection The number of features of the data set obtained in this article is not very large, and for real stock data, in addition to the features in the stock market, can you use other features, such as corporate financial reports; also, because the stock market is affected by policy Larger, using the application of LSTM neural network intext learning and text sentiment analysis, can we obtain some features from news and financial reports, so as

to enable the model to make corresponding judgments on stocks in an economic sense, thereby improving the accuracy of our predictions rate.

(3) About model optimization When constructing the neural network model, whether the number of hidden layers is small, and whether more hidden layers will have better prediction results, this is also the lack of research in this article.

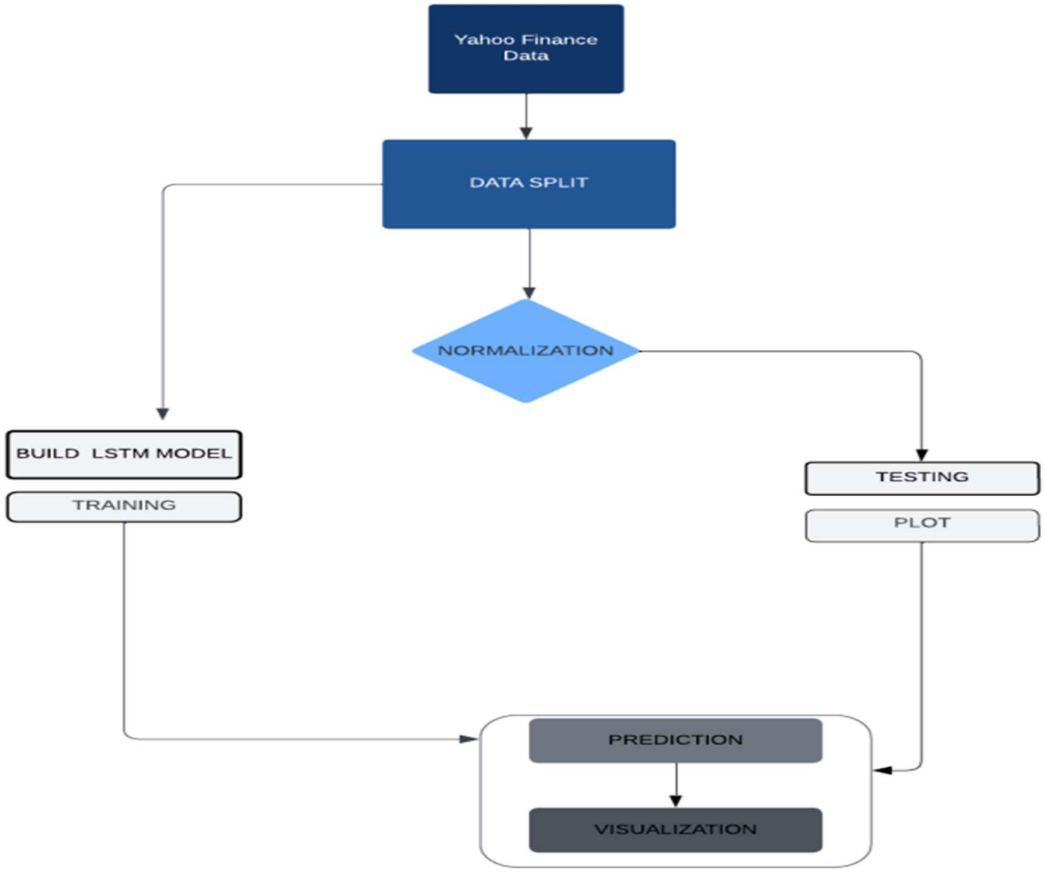
Chapter 4: Methodology

The volatility of stock prices is controlled by the trend of the stock, but is also sensitive to many other factors. Due to the relative stability and predictability of the intrinsic value of stocks, the factors that have impacts on the stock market price mainly include the following aspects: 1. Macro factors; 2. Industrial and regional factors; 3. Company factors; 4. market factors. This article predicts the closing index of the S&P500 rather than specific company stock price forecasts, so aside from the more microscopic industry and company factors, it mainly focuses on the influence of macroeconomic factors and market factors.

Macroeconomic factors refer to the impact of macroeconomic environment and its changes on stock prices, including regular factors such as cyclical fluctuations in macroeconomic operations and policy factors such as monetary policy implemented by the government. This article predicts the daily data of the S&P500 closing index, mainly focusing on the impact of monetary policy and other policy factors on stock prices. There are two types of stock price forecasting methods: qualitative analysis and quantitative analysis.

The qualitative analysis method is the fundamental analysis method, which is a subjective analysis method relying on the experience of financial practitioners. This thesis is a numerical prediction of the daily closing index of the 21 S&P500 rather than a trend judgment of price fluctuations, so this thesis mainly focuses on the literature review of quantitative analysis methods. Numerical data-based stock market forecasting research uses numerical data

on a certain time scale in the stock market, such as sky-level index prices and stock price volume data, to predict specific stocks or other investments in the stock market on the same scale. Predict the future price of the underlying. According to the focus of the research, these studies can be divided into research on the characteristics of numerical data stock market forecasting and research on the numerical data stock market forecasting model.



Tool & Technologies

PYTHON

The language of select for this project was Python. This was a straightforward call for many reasons.

1. Python [19] as a language has a vast community behind it. Any problems which may be faced is simply resolved with visit to Stack Overflow. Python is the foremost standard language on the positioning that makes it is very straight answer to any question.

2. Python [19] is an abundance of powerful tools ready for scientific computing Packages. The packages like NumPy, Pandas and SciPy area unit freely available and well documented. These Packages will intensely scale back, and variation the code necessary to write a given program. This makes repetition fast.

3. Python is a language as [19] forgiving and permits for the program that appear as if pseudo code. This can be helpful once pseudo code give in tutorial papers should be required and verified. Using python this step is sometimes fairly trivial. However, Python is [19] not without its errors. The python is dynamically written language and packages are area unit infamous for Duck writing. This may be frustrating once a package technique returns one thing that, for instance, looks like an array instead of being an actual array. Plus the standard Python documentation did not clearly state the return type of a method, this can't lead without a lot of trials and error testing otherwise happen in a powerfully written language. This is a problem that produces learning to use a replacement Python package or library more difficult than it otherwise may be.

NUMPY

Numpy is python package which provide scientific and higher level mathematical abstractions wrapped in python. It is [20] the core library for scientific computing, that contains a provide tools for integrating C, strong n-dimensional array object, C++ etc. It is also useful in random number capability, linear algebra etc. Numpy's array type augments the Python language with an efficient data structure used for numerical work, e.g., manipulating matrices. Numpy additionally provides basic numerical routines, like tools for locating Eigenvectors Stock Price Prediction Using Machine Learning 18CP808 18 4.2.3 SCIKIT LEARN Scikit-learn [21] could be a free machine learning library for Python. It features numerous classification, clustering and regression algorithms like random forests, k-neighbours, support vector machine, and it furthermore supports Python scientific and numerical libraries like SciPy and NumPy. In Python Scikit-learn is specifically written, with the core algorithms written in Cython to get the performance. Support vector machines are enforced by a Cython wrapper around LIBSVM .i.e., linear support vector machines and logistic regression by a similar wrapper around LIBLINEAR.

TENSORFLOW

In the TensorFlow [22]has an open source software library for numerical computation using data flow graphs. Inside the graph nodes represent mathematical formulae, the edges of graph represent the multidimensional knowledge arrays (tensors) communicated between them. The versatile architecture permits to deploy the computation to at least one or many GPUs or CPUs in a desktop, mobile device, servers with a single API. TensorFlow was firstly developing by engineers and researchers acting on the Google Brain Team at intervals Google's Machine Intelligence analysis organization for the needs of conducting deep neural networks research and machine learning, but, the system is generally enough to be appropriate in a wide range of alternate domains as well. Google Brain's second-generation system is TensorFlow. Whereas the reference implementation runs on single devices, TensorFlow can run on multiple GPUs and CPUs. TensorFlow is offered on Windows, macOS, 64-bit Linux and mobile computing platforms together with iOS and Android.

KERAS

Keras is a high-level neural networks API, it is written in Python and also capable of running on top of the Theano, CNTK, or. TensorFlow. It was developed with attention on enabling quick experimentation. having the ability to travel from plan to result with the smallest amount doable delay is key to doing great research.Keras permits for straightforward and quick prototyping (through user-friendliness, modularity, and extensibility). Supports each recurrent networks and convolutional networks, also as combinations of the 2. Runs seamlessly on GPU and CPU. The library contains numerous implementations of generally used neural network building blocks like optimizers, activation functions, layers, objectives and a number of tools to create operating with text and image data easier. The code is hosted on GitHub, and community support forums embody the GitHub issues page, a Gitter channel and a Slack channel.

COMPILER OPTION

Anaconda is [19] free premium open-source distribution of the R and Python programming languages for scientific computing, predictive analytics, and

large-scale process that aim is to modify package managing and deployment. Package versions unit managed by the package management system conda.

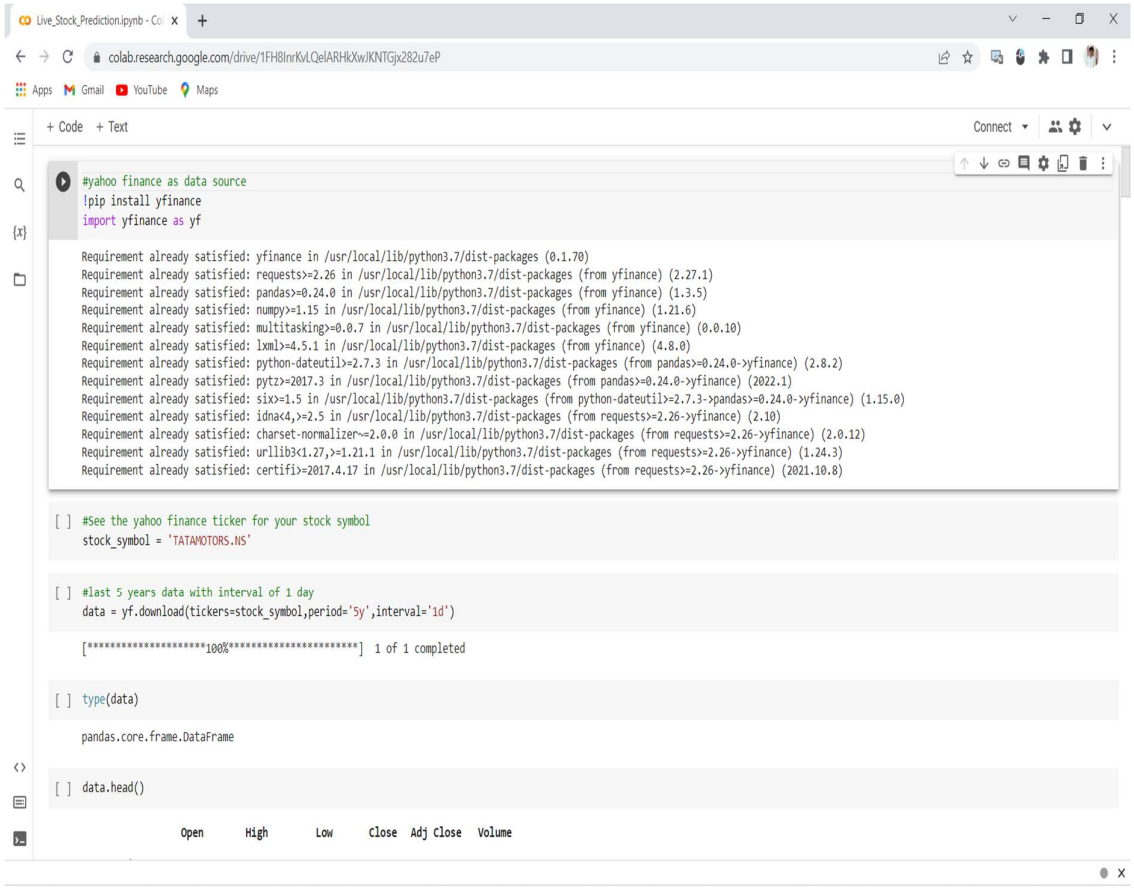
JUPITER NOTEBOOK

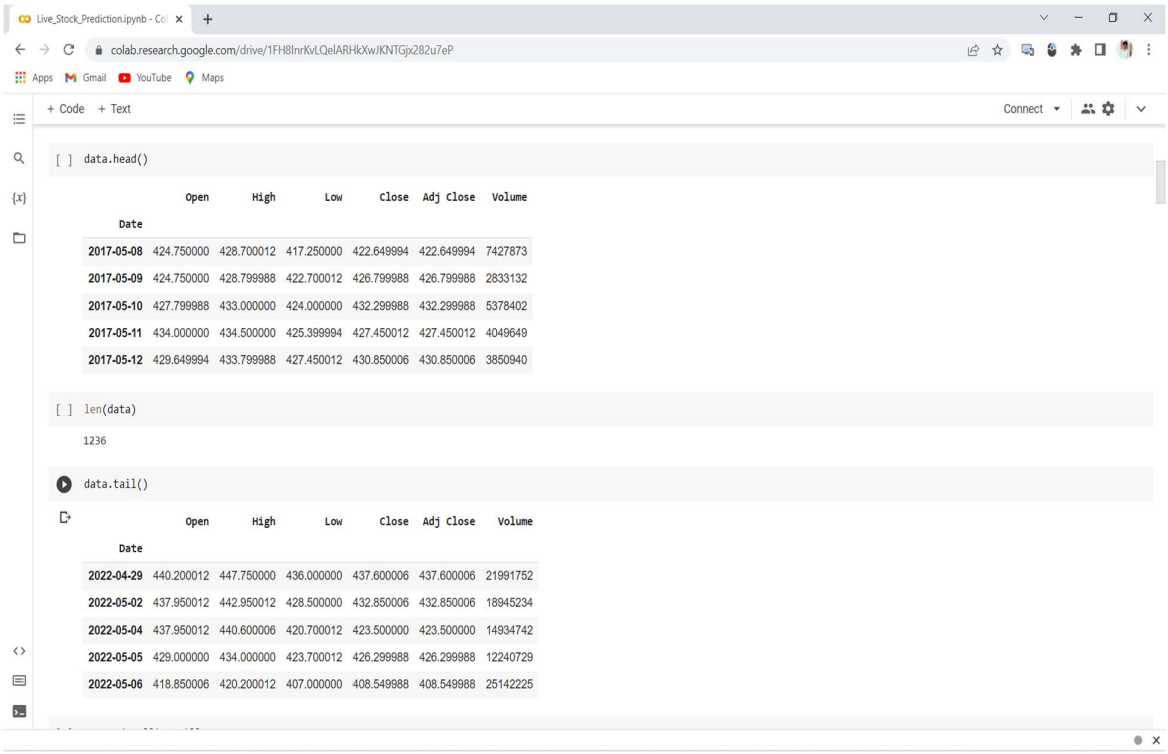
The Jupyter Notebook is an open-source web application that enables to making and sharing documents that contain visualizations, narrative text, live code and equations. Uses include: data, data visualization, data transformation, statistical modelling, machine learning, numerical simulation, data cleaning and much more.

Chapter 5: Implementation

We have used data from yahoo finance for getting better data.

We have used last 5 years data with interval of 1 day.


Live_Stock_Prediction.ipynb - Co
colab.research.google.com/drive/1FH8lnrKvLQelARHkXwJKNTGjx282u7eP
Apps Gmail YouTube Maps
+ Code + Text
Connect
#yahoo finance as data source
!pip install yfinance
import yfinance as yf
Requirement already satisfied: yfinance in /usr/local/lib/python3.7/dist-packages (0.1.70)
Requirement already satisfied: requests>=2.26 in /usr/local/lib/python3.7/dist-packages (from yfinance) (2.27.1)
Requirement already satisfied: pandas>=0.24.0 in /usr/local/lib/python3.7/dist-packages (from yfinance) (1.3.5)
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.7/dist-packages (from yfinance) (1.21.6)
Requirement already satisfied: multitasking>=0.0.7 in /usr/local/lib/python3.7/dist-packages (from yfinance) (0.0.10)
Requirement already satisfied: lxml>=4.5.1 in /usr/local/lib/python3.7/dist-packages (from yfinance) (4.8.0)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.24.0->yfinance) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.24.0->yfinance) (2022.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3->pandas>=0.24.0->yfinance) (1.15.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2.10)
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2.0.12)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2021.10.8)
#See the yahoo finance ticker for your stock symbol
stock_symbol = 'TATAMOTORS.NS'
#last 5 years data with interval of 1 day
data = yf.download(tickers=stock_symbol,period='5y',interval='1d')
[**********100%**********] 1 of 1 completed
type(data)
pandas.core.frame.DataFrame
data.head()
Open High Low Close Adj Close Volume


Live_Stock_Prediction.ipynb - Co
colab.research.google.com/drive/1FH8lnrKvLQelARHkXwJKNTGjx282u7eP
Apps Gmail YouTube Maps
+ Code + Text
Connect
data.head()
Open High Low Close Adj Close Volume
Date
2017-05-08 424.750000 428.700012 417.250000 422.649994 422.649994 7427873
2017-05-09 424.750000 428.799988 422.700012 426.799988 426.799988 2833132
2017-05-10 427.799988 433.000000 424.000000 432.299988 432.299988 5378402
2017-05-11 434.000000 434.500000 425.399994 427.450012 427.450012 4049649
2017-05-12 429.649994 433.799988 427.450012 430.850006 430.850006 3850940
len(data)
1236
data.tail()
Open High Low Close Adj Close Volume
Date
2022-04-29 440.200012 447.750000 436.000000 437.600006 437.600006 21991752
2022-05-02 437.950012 442.950012 428.500000 432.850006 432.850006 18945234
2022-05-04 437.950012 440.600006 420.700012 423.500000 423.500000 14934742
2022-05-05 429.000000 434.000000 423.700012 426.299988 426.299988 12240729
2022-05-06 418.850006 420.200012 407.000000 408.549988 408.549988 25142225

Live_Stock_Prediction.ipynb - Col

colab.research.google.com/drive/1FH8InrKvLQeIARHkXwJKNTGjx282u7eP

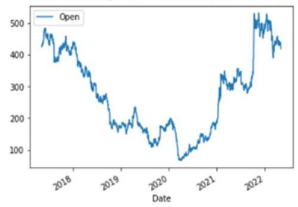
Apps Gmail YouTube Maps

+ Code + Text

Connect

```
[ ] opn = data[['Open']]

[ ] opn.plot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb2f85e4f50>
```




```
[ ] import matplotlib.pyplot as plt

    ds = opn.values

[ ] ds
```

```
array([[424.75      ],
       [424.75      ],
       [427.79998779],
       ...,
       [437.95001221],
       [429.        ],
       [418.8500061 ]])
```

Live_Stock_Prediction.ipynb - Col

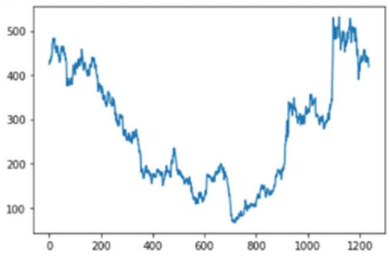
colab.research.google.com/drive/1FH8InrKvLQeIARHkXwJKNTGjx282u7eP

Apps Gmail YouTube Maps

+ Code + Text

```
[ ] plt.plot(ds)
```

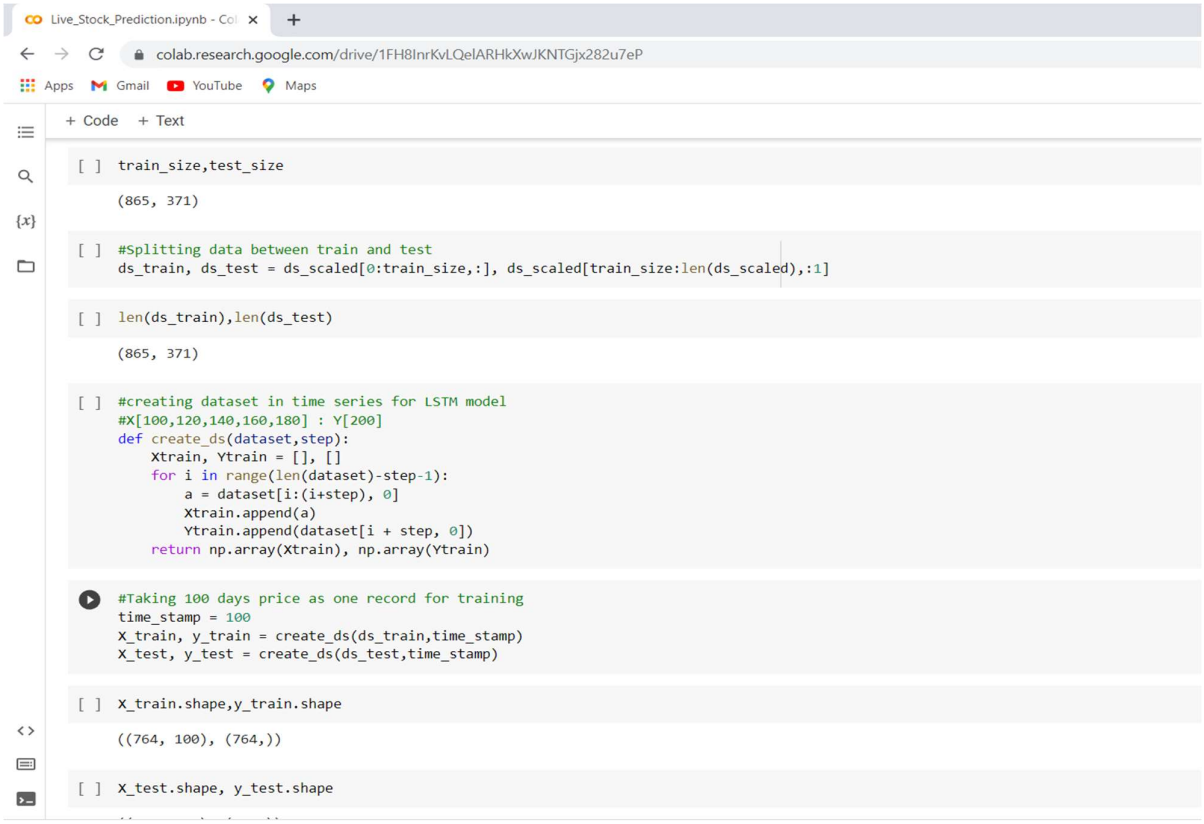
```
[<matplotlib.lines.Line2D at 0x7fb2f4a93610>]
```

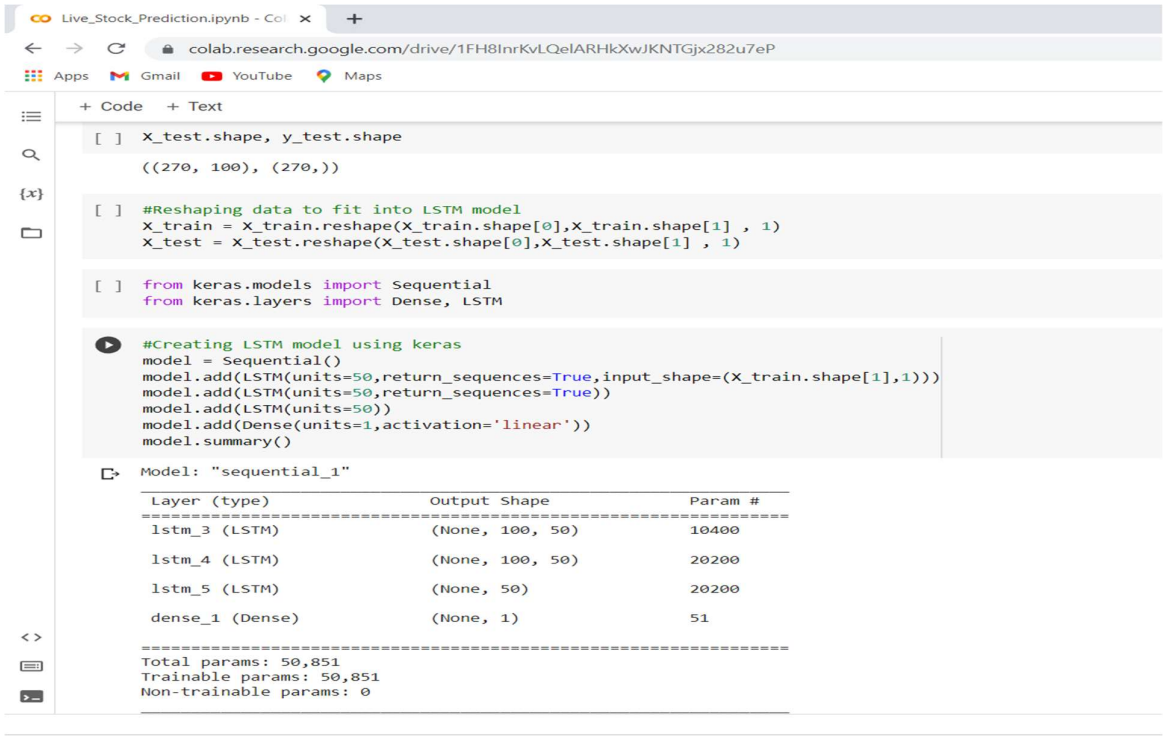



```
[ ] import numpy as np

[ ] from sklearn.preprocessing import MinMaxScaler

    #Using MinMaxScaler for normalizing data between 0 & 1
    normalizer = MinMaxScaler(feature_range=(0,1))
    ds_scaled = normalizer.fit_transform(np.array(ds).reshape(-1,1))

[ ] len(ds_scaled), len(ds)
```

```
(1236, 1236)
```

```
[ ] #Defining test and train data sizes
    train_size = int(len(ds_scaled)*0.70)
    test_size = len(ds_scaled) - train_size
```


Live_Stock_Prediction.ipynb - Col
colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP
Apps Gmail YouTube Maps
+ Code + Text
[ ] train_size,test_size
(865, 371)
[ ] #Splitting data between train and test
ds_train, ds_test = ds_scaled[0:train_size,:], ds_scaled[train_size:len(ds_scaled),:1]
[ ] len(ds_train),len(ds_test)
(865, 371)
[ ] #creating dataset in time series for LSTM model
#X[100,120,140,160,180] : Y[200]
def create_ds(dataset,step):
Xtrain, Ytrain = [], []
for i in range(len(dataset)-step-1):
a = dataset[i:(i+step), 0]
Xtrain.append(a)
Ytrain.append(dataset[i + step, 0])
return np.array(Xtrain), np.array(Ytrain)
#Taking 100 days price as one record for training
time_stamp = 100
X_train, y_train = create_ds(ds_train,time_stamp)
X_test, y_test = create_ds(ds_test,time_stamp)
[ ] X_train.shape,y_train.shape
((764, 100), (764,))
[ ] X_test.shape, y_test.shape


Live_Stock_Prediction.ipynb - Col
colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP
Apps Gmail YouTube Maps
+ Code + Text
[ ] X_test.shape, y_test.shape
((270, 100), (270,))
[ ] #Reshaping data to fit into LSTM model
X_train = X_train.reshape(X_train.shape[0],X_train.shape[1] , 1)
X_test = X_test.reshape(X_test.shape[0],X_test.shape[1] , 1)
[ ] from keras.models import Sequential
from keras.layers import Dense, LSTM
#Creating LSTM model using keras
model = Sequential()
model.add(LSTM(units=50,return_sequences=True,input_shape=(X_train.shape[1],1)))
model.add(LSTM(units=50,return_sequences=True))
model.add(LSTM(units=50))
model.add(Dense(units=1,activation='linear'))
model.summary()
Model: "sequential_1"
Layer (type) Output Shape Param #
lstm_3 (LSTM) (None, 100, 50) 10400
lstm_4 (LSTM) (None, 100, 50) 20200
lstm_5 (LSTM) (None, 50) 20200
dense_1 (Dense) (None, 1) 51
Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0

Live_Stock_Prediction.ipynb - Col

colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP

Apps Gmail YouTube Maps

+ Code + Text

```
#Training model with adam optimizer and mean squared error loss function
model.compile(loss='mean_squared_error',optimizer='adam')
model.fit(X_train,y_train,validation_data=(X_test,y_test),epochs=100,batch_size=64)
```

```
12/12 [==============================] - 2s 197ms/step - loss: 4.8930e-04 - val_loss: 0.0019
Epoch 73/100
12/12 [==============================] - 2s 196ms/step - loss: 4.9956e-04 - val_loss: 0.0019
Epoch 74/100
12/12 [==============================] - 2s 195ms/step - loss: 5.0463e-04 - val_loss: 0.0022
Epoch 75/100
12/12 [==============================] - 2s 198ms/step - loss: 5.3702e-04 - val_loss: 0.0019
Epoch 76/100
12/12 [==============================] - 2s 198ms/step - loss: 5.0853e-04 - val_loss: 0.0019
Epoch 77/100
12/12 [==============================] - 2s 193ms/step - loss: 5.0538e-04 - val_loss: 0.0028
Epoch 78/100
12/12 [==============================] - 2s 195ms/step - loss: 5.0328e-04 - val_loss: 0.0018
Epoch 79/100
12/12 [==============================] - 2s 195ms/step - loss: 4.5740e-04 - val_loss: 0.0018
Epoch 80/100
12/12 [==============================] - 2s 193ms/step - loss: 4.5690e-04 - val_loss: 0.0020
Epoch 81/100
12/12 [==============================] - 2s 196ms/step - loss: 4.7507e-04 - val_loss: 0.0018
Epoch 82/100
12/12 [==============================] - 2s 195ms/step - loss: 5.0635e-04 - val_loss: 0.0021
Epoch 83/100
12/12 [==============================] - 2s 195ms/step - loss: 6.1108e-04 - val_loss: 0.0019
Epoch 84/100
12/12 [==============================] - 2s 196ms/step - loss: 5.2805e-04 - val_loss: 0.0017
Epoch 85/100
12/12 [==============================] - 2s 195ms/step - loss: 4.7455e-04 - val_loss: 0.0017
Epoch 86/100
12/12 [==============================] - 2s 195ms/step - loss: 4.5720e-04 - val_loss: 0.0022
Epoch 87/100
12/12 [==============================] - 2s 195ms/step - loss: 5.1918e-04 - val_loss: 0.0016
Epoch 88/100
12/12 [==============================] - 2s 196ms/step - loss: 4.2850e-04 - val_loss: 0.0016
Epoch 89/100
12/12 [==============================] - 2s 196ms/step - loss: 4.2394e-04 - val_loss: 0.0016
Epoch 90/100
```

Live_Stock_Prediction.ipynb - Col

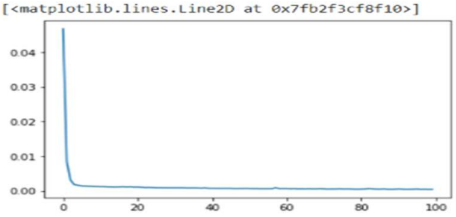
colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP

Apps Gmail YouTube Maps

+ Code + Text

```
[ ] #PLotting loss, it shows that loss has decreased significantly and model trained well
    loss = model.history.history['loss']
    plt.plot(loss)
```

```
[<matplotlib.lines.Line2D at 0x7fb2f3cf8f10>]
```




```
[ ] #Predicitng on train and test data
    train_predict = model.predict(X_train)
    test_predict = model.predict(X_test)
```

```
    #Inverse transform to get actual value
    train_predict = normalizer.inverse_transform(train_predict)
    test_predict = normalizer.inverse_transform(test_predict)
```

```
[ ] #Comparing using visuals
    plt.plot(normalizer.inverse_transform(ds_scaled))
    plt.plot(train_predict)
    plt.plot(test_predict)
```

```
[<matplotlib.lines.Line2D at 0x7fb3745f8250>]
```

Live_Stock_Prediction.ipynb - Col

colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP

Apps Gmail YouTube Maps

+ Code + Text

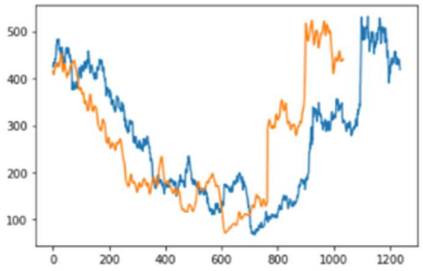
```
[ ] type(train_predict)
```

```
numpy.ndarray
```

```
[ ] test = np.vstack((train_predict,test_predict))
```

```
[ ] #Combining the predited data to create uniform data visualization
    plt.plot(normalizer.inverse_transform(ds_scaled))
    plt.plot(test)
```

```
[<matplotlib.lines.Line2D at 0x7fb37467d350>]
```




```
[ ] len(ds_test)
```

```
371
```

```
#Getting the last 100 days records
fut_inp = ds_test[270:]
```

+ Code + Text

```
[ ] fut_inp = fut_inp.reshape(1,-1)
```

Live_Stock_Prediction.ipynb - Col

colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP

Apps Gmail YouTube Maps

+ Code + Text

```
[ ] fut_inp = fut_inp.reshape(1,-1)
```

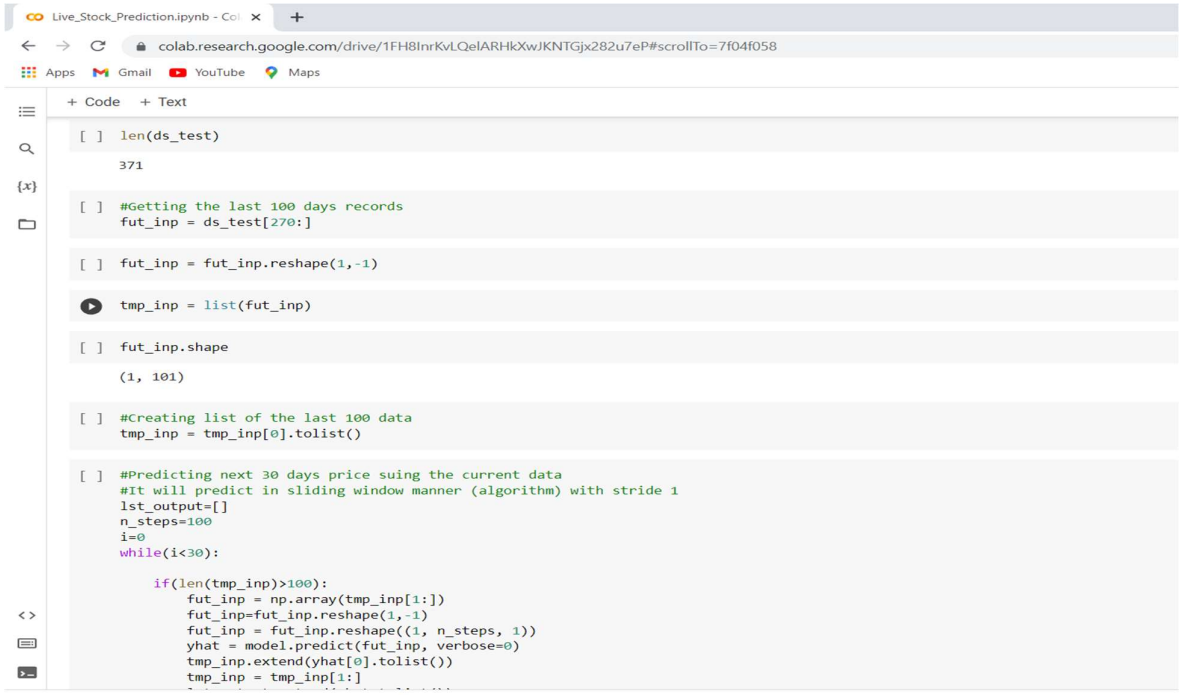
```
[ ] tmp_inp = list(fut_inp)
```

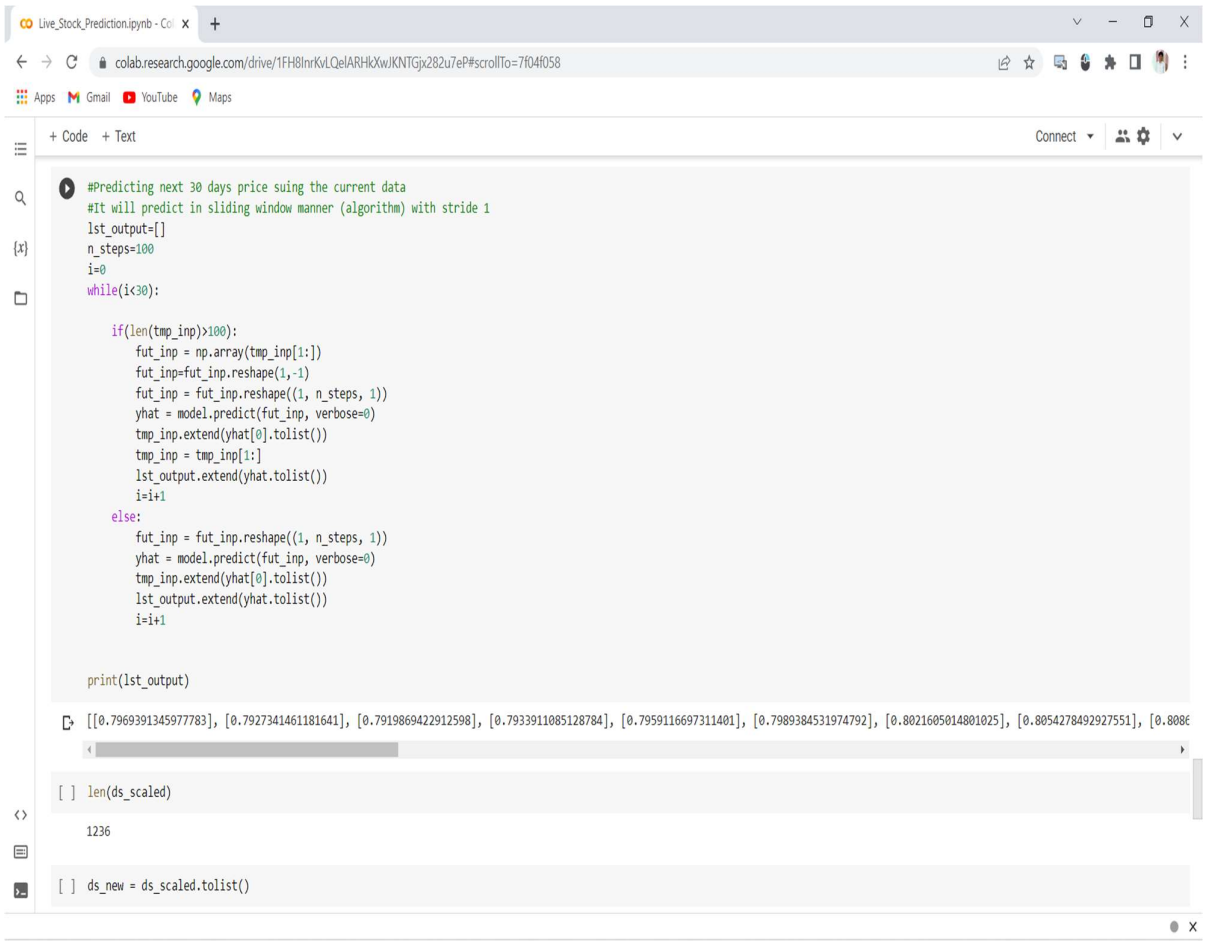
```
[ ] fut_inp.shape
```

```
(1, 101)
```

```
[ ] #Creating list of the last 100 data
    tmp_inp = tmp_inp[0].tolist()
```

```
#Predicting next 30 days price suing the current data
#It will predict in sliding window manner (algorithm) with stride 1
lst_output=[]
n_steps=100
i=0
while(i<30):

    if(len(tmp_inp)>100):
        fut_inp = np.array(tmp_inp[1:])
        fut_inp=fut_inp.reshape(1,-1)
        fut_inp = fut_inp.reshape((1, n_steps, 1))
        yhat = model.predict(fut_inp, verbose=0)
        tmp_inp.extend(yhat[0].tolist())
        tmp_inp = tmp_inp[1:]
        lst_output.extend(yhat.tolist())
        i=i+1
    else:
        fut_inp = fut_inp.reshape((1, n_steps, 1))
        yhat = model.predict(fut_inp, verbose=0)
        tmp_inp.extend(yhat[0].tolist())
        lst_output.extend(yhat.tolist())
        i=i+1
```


Live_Stock_Prediction.ipynb - Col
colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP#scrollTo=7f04f058
Apps
Gmail
YouTube
Maps
+ Code
+ Text
[ ] len(ds_test)
371
[ ] #Getting the last 100 days records
fut_inp = ds_test[270:]
[ ] fut_inp = fut_inp.reshape(1,-1)
tmp_inp = list(fut_inp)
[ ] fut_inp.shape
(1, 101)
[ ] #Creating list of the last 100 data
tmp_inp = tmp_inp[0].tolist()
[ ] #Predicting next 30 days price suing the current data
#It will predict in sliding window manner (algorithm) with stride 1
lst_output=[]
n_steps=100
i=0
while(i<30):
if(len(tmp_inp)>100):
fut_inp = np.array(tmp_inp[1:])
fut_inp=fut_inp.reshape(1,-1)
fut_inp = fut_inp.reshape((1, n_steps, 1))
yhat = model.predict(fut_inp, verbose=0)
tmp_inp.extend(yhat[0].tolist())
tmp_inp = tmp_inp[1:]


Live_Stock_Prediction.ipynb - Col
colab.research.google.com/drive/1FH8InrKvLQelARHkXwJKNTGjx282u7eP#scrollTo=7f04f058
Apps
Gmail
YouTube
Maps
+ Code
+ Text
Connect
#Predicting next 30 days price suing the current data
#It will predict in sliding window manner (algorithm) with stride 1
lst_output=[]
n_steps=100
i=0
while(i<30):
if(len(tmp_inp)>100):
fut_inp = np.array(tmp_inp[1:])
fut_inp=fut_inp.reshape(1,-1)
fut_inp = fut_inp.reshape((1, n_steps, 1))
yhat = model.predict(fut_inp, verbose=0)
tmp_inp.extend(yhat[0].tolist())
tmp_inp = tmp_inp[1:]
lst_output.extend(yhat.tolist())
i=i+1
else:
fut_inp = fut_inp.reshape((1, n_steps, 1))
yhat = model.predict(fut_inp, verbose=0)
tmp_inp.extend(yhat[0].tolist())
lst_output.extend(yhat.tolist())
i=i+1
print(lst_output)
[[0.7969391345977783], [0.7927341461181641], [0.7919869422912598], [0.7933911085128784], [0.7959116697311401], [0.7989384531974792], [0.8021605014801025], [0.8054278492927551], [0.808
[ ] len(ds_scaled)
1236
[ ] ds_new = ds_scaled.tolist()

Live_Stock_Prediction.ipynb - Col

colab.research.google.com/drive/1FH8InrKvLQeIARHkXwJKNTGjx282u7eP#scrollTo=7f04f058

Apps Gmail YouTube Maps

+ Code + Text

Conn

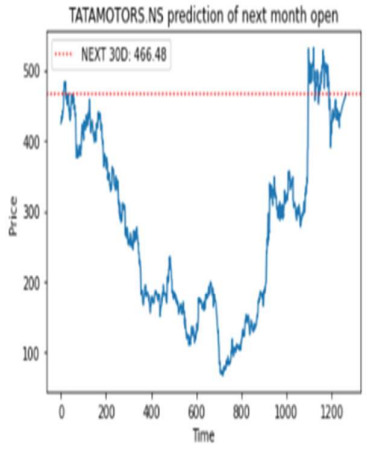
```
[ ] len(ds_new)
```

```
1236
```

```
#Creating final data for plotting
final_graph = normalizer.inverse_transform(ds_new).tolist()
```

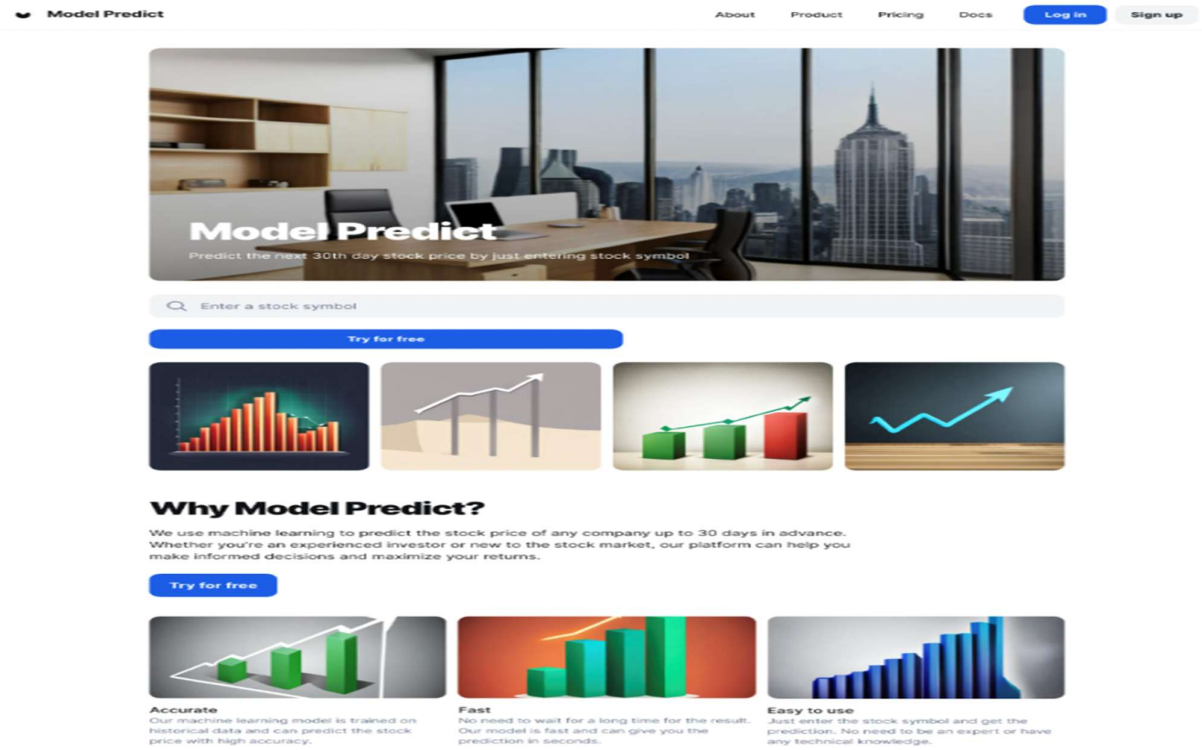
```
#Plotting final results with predicted value after 30 Days
plt.plot(final_graph,)
plt.ylabel("Price")
plt.xlabel("Time")
plt.title("{0} prediction of next month open".format(stock_symbol))
plt.axhline(y=final_graph[len(final_graph)-1], color = 'red', linestyle = ':', label = 'NEXT 30D: {0}'.format(round(float(*final_graph[len(final_graph)-1]),2)))
plt.legend()
```

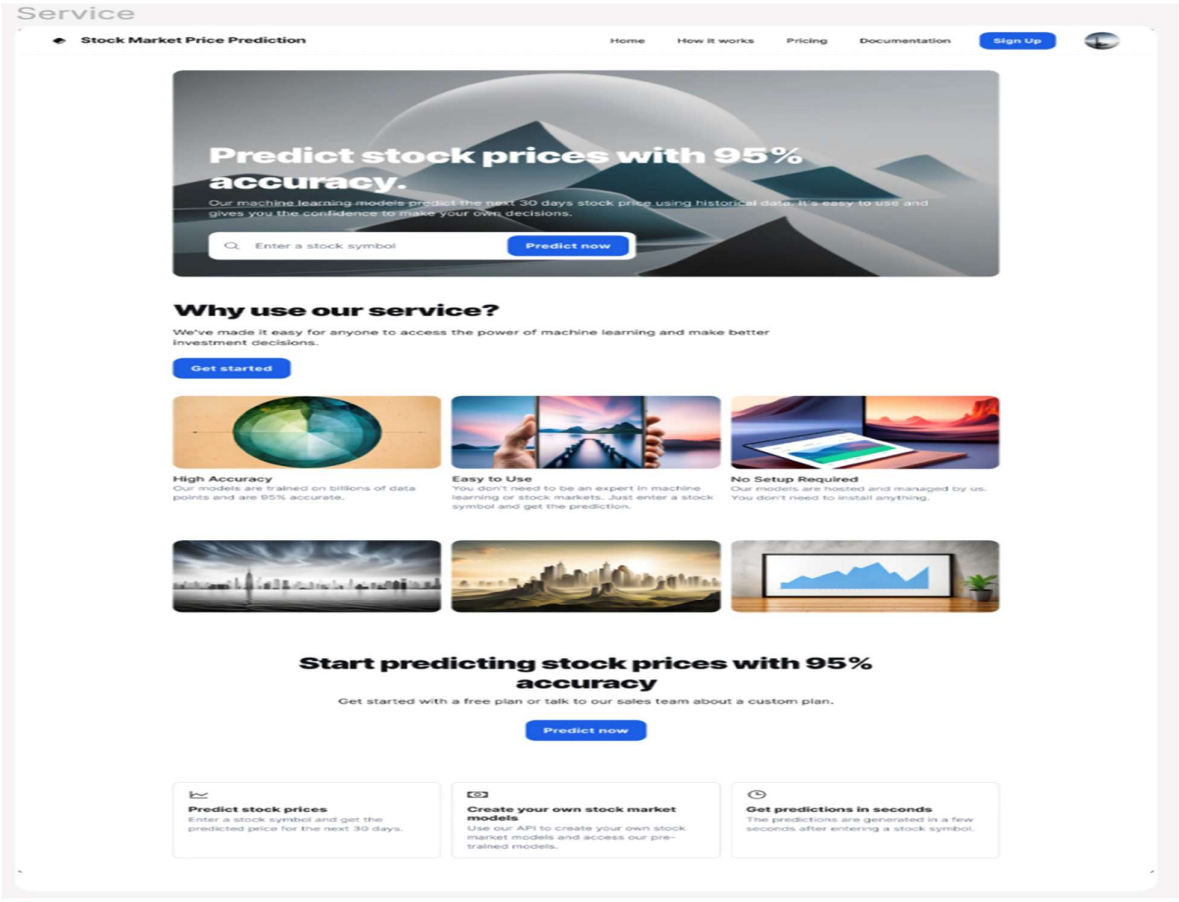
```
<matplotlib.legend.Legend at 0x7fb374327d50>
```



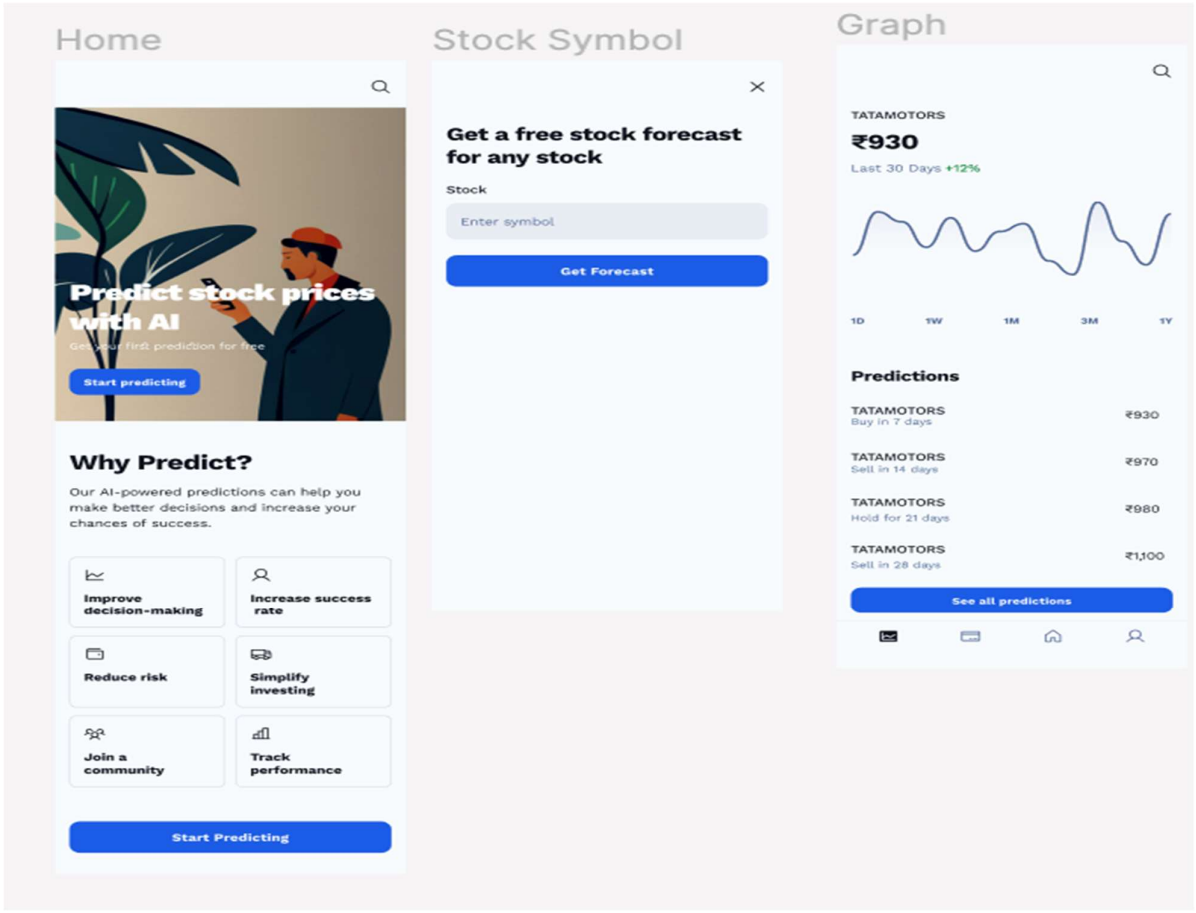
Final Result of TATAMOTORS.NS Stock Price Prediction

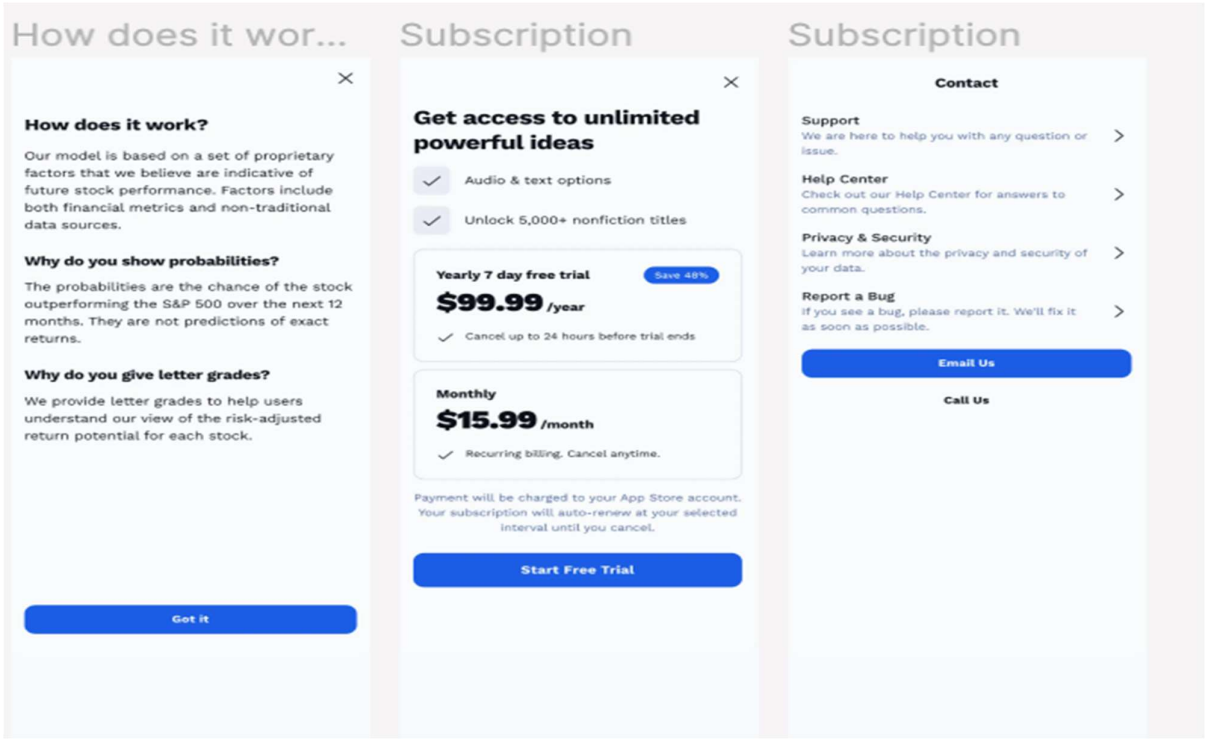
WebApp:


Model Predict
About
Product
Pricing
Docs
Log in
Sign up
Model Predict
Predict the next 30th day stock price by just entering stock symbol
Enter a stock symbol
Try for free
Why Model Predict?
We use machine learning to predict the stock price of any company up to 30 days in advance. Whether you're an experienced investor or new to the stock market, our platform can help you make informed decisions and maximize your returns.
Try for free
Accurate
Our machine learning model is trained on historical data and can predict the stock price with high accuracy.
Fast
No need to wait for a long time for the result. Our model is fast and can give you the prediction in seconds.
Easy to use
Just enter the stock symbol and get the prediction. No need to be an expert or have any technical knowledge.

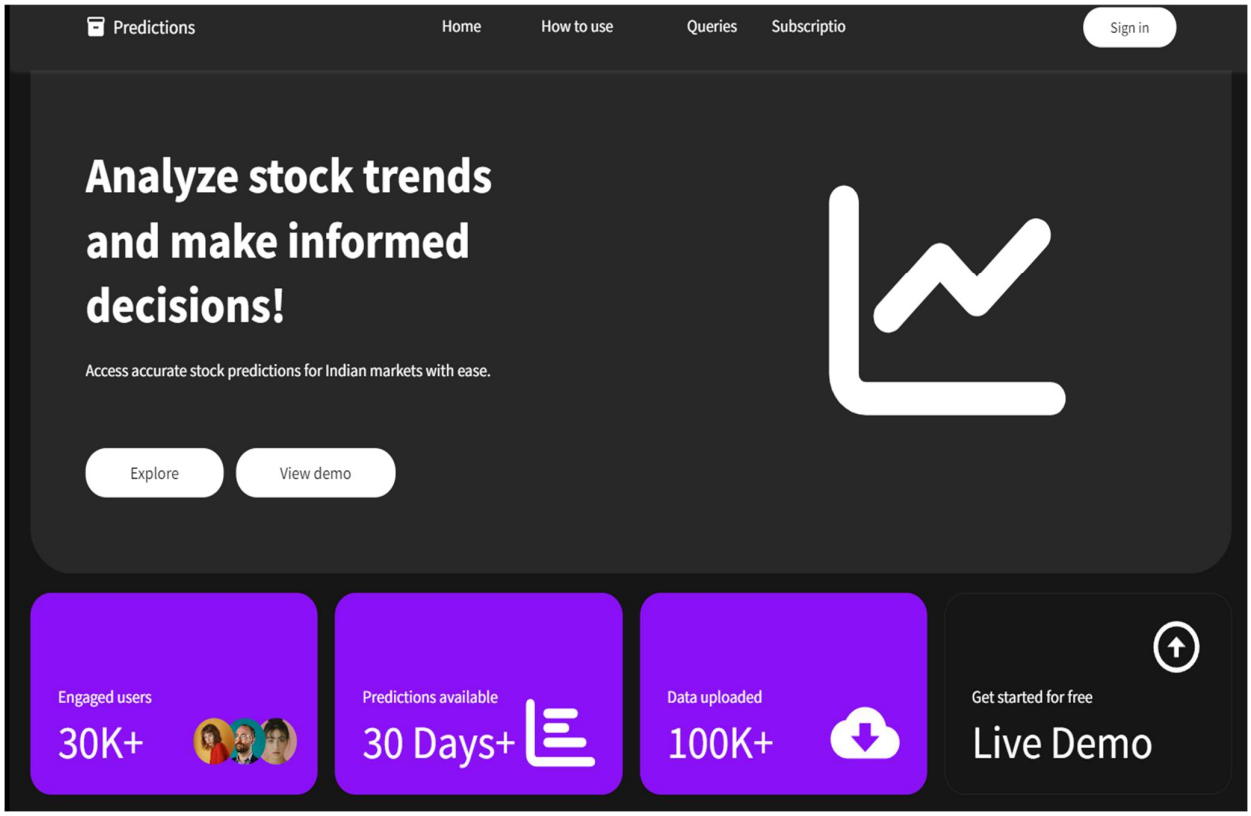

Service
Stock Market Price Prediction
Home
How it works
Pricing
Documentation
Sign Up
Predict stock prices with 95% accuracy.
Our machine learning models predict the next 30 days stock price using historical data. It's easy to use and gives you the confidence to make your own decisions.
Enter a stock symbol
Predict now
Why use our service?
We've made it easy for anyone to access the power of machine learning and make better investment decisions.
Get started
High Accuracy
Our models are trained on billions of data points and are 95% accurate.
Easy to Use
You don't need to be an expert in machine learning or stock markets. Just enter a stock symbol and get the prediction.
No Setup Required
Our models are hosted and managed by us. You don't need to install anything.
Start predicting stock prices with 95% accuracy
Get started with a free plan or talk to our sales team about a custom plan.
Predict now
Predict stock prices
Enter a stock symbol and get the predicted price for the next 30 days.
Create your own stock market models
Use our API to create your own stock market models and access our pre-trained models.
Get predictions in seconds
The predictions are generated in a few seconds after entering a stock symbol.

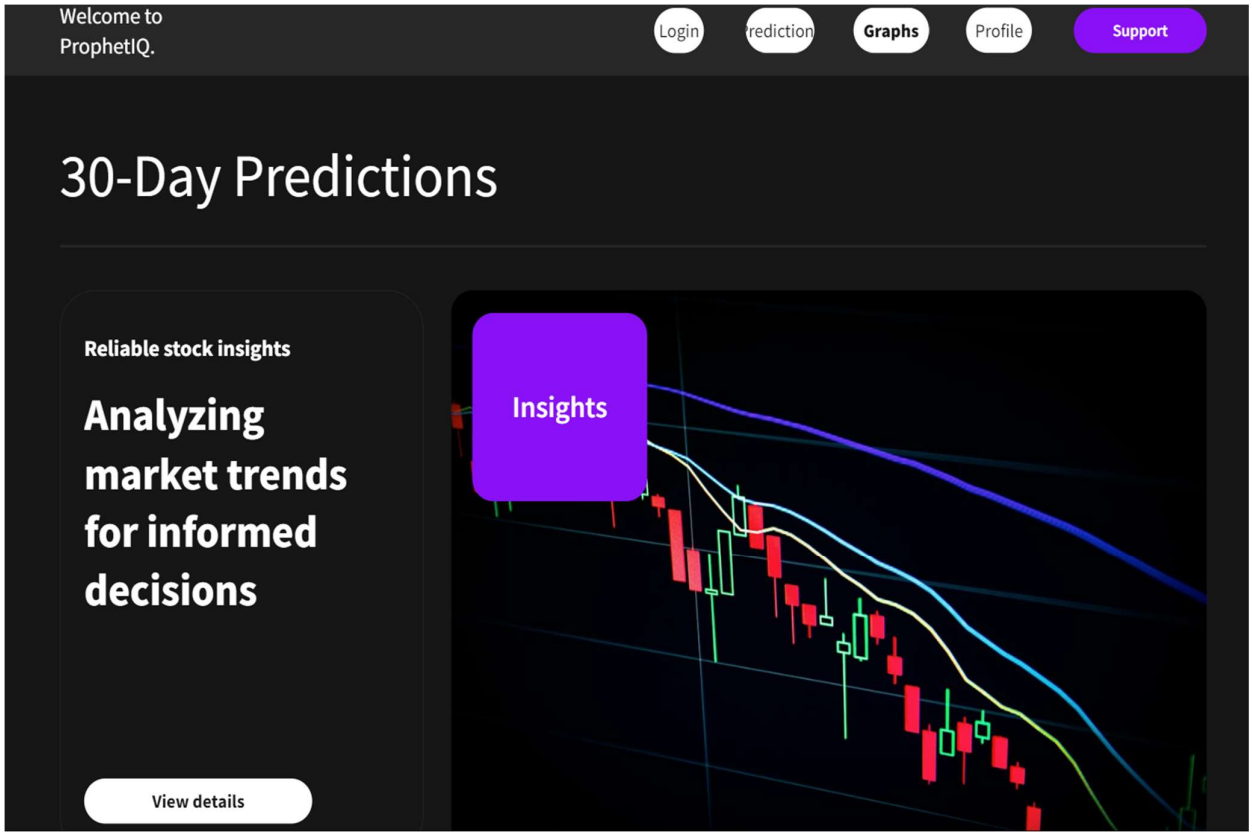
Android Mobile Web View:





Working Website Overview:





Market Trends

Insightful data for strategic investments

View details

Stock Portfolio

Exploring investment opportunities

View details

About ProphetIQ

Trusted since 2024

Features

Platform overview

Careers at ProphetIQ

Contact us

Latest news and updates

Community support

Help and FAQ

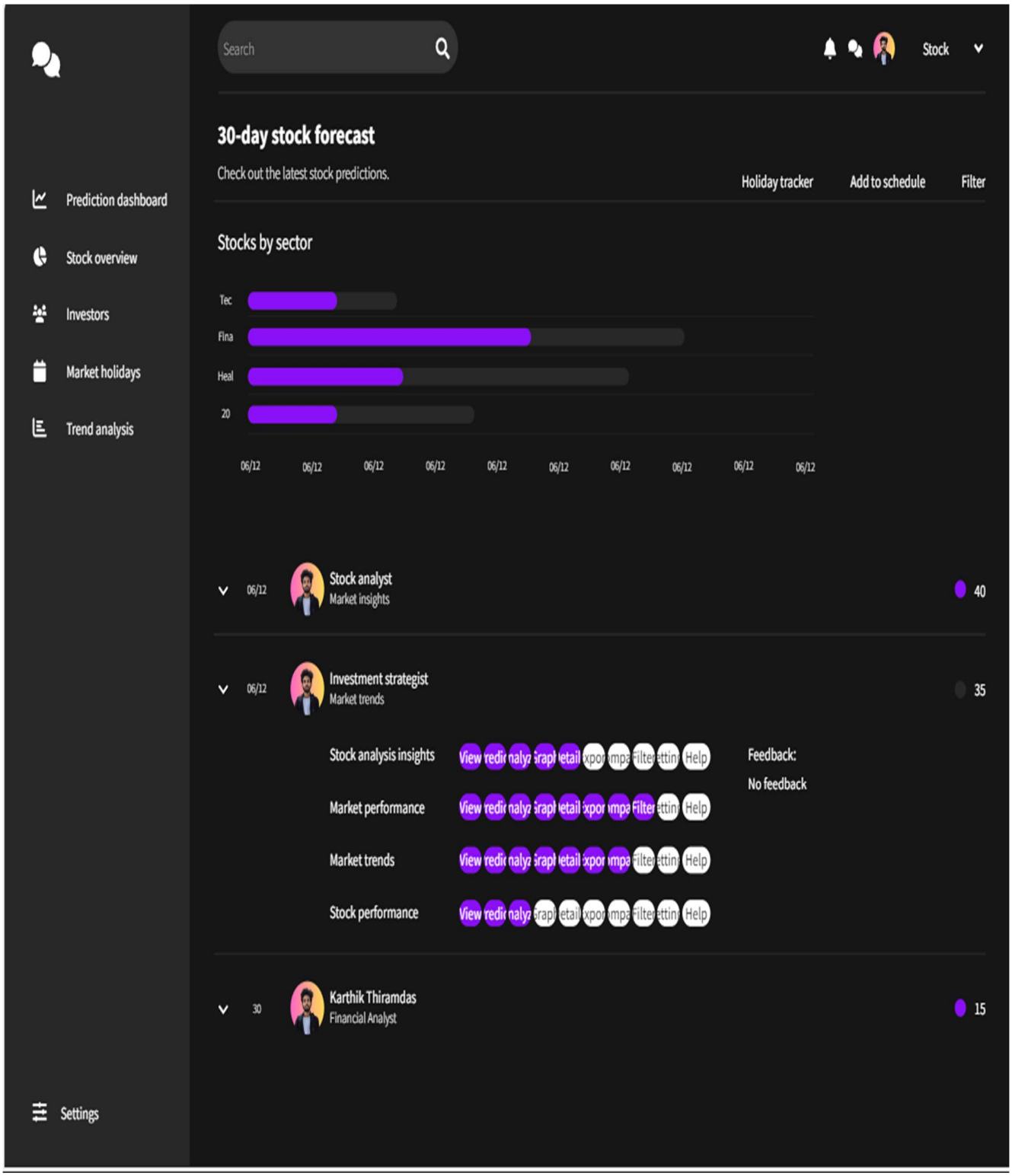
Upgrade to PRO

Connect with us

Facebook

Twitter

Instagram


Search
Stock
30-day stock forecast
Check out the latest stock predictions.
Holiday tracker
Add to schedule
Filter
Prediction dashboard
Stock overview
Investors
Market holidays
Trend analysis
Settings
Stocks by sector
Tec
Fina
Heal
20
06/12
Stock analyst
Market insights
40
Investment strategist
Market trends
35
Stock analysis insights
Market performance
Market trends
Stock performance
Help
Feedback:
No feedback
Karthik Thiramdas
Financial Analyst
15

Stock Tags

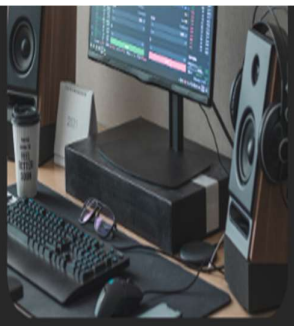
All | Owned | Created | Stock Collections | Following | Activity

Recent Stock Searches

Stock Market Trends

Stock Market Insights

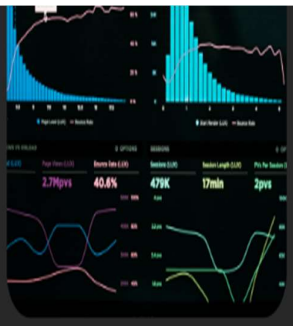
Stock Market Recommendations



Stock Market Trends

Analyze and predict future trends.

$ 100.00 ♥ 24



Stock Market Insights

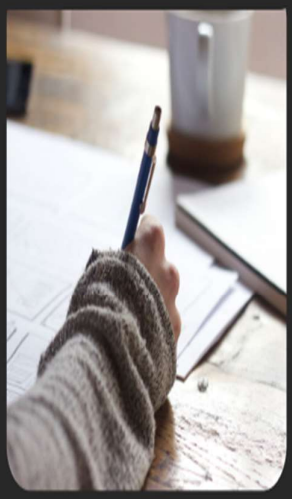
Insights for informed decisions.

$ 75.00 ♥ 13

Stock Market Analysis

In-depth analysis for better strategies.

$ 50.00 ♥ 24



Stock Market Predictions

Forecasting future stock prices.

$ 25.00 ♥ 19

Stock Market Offers

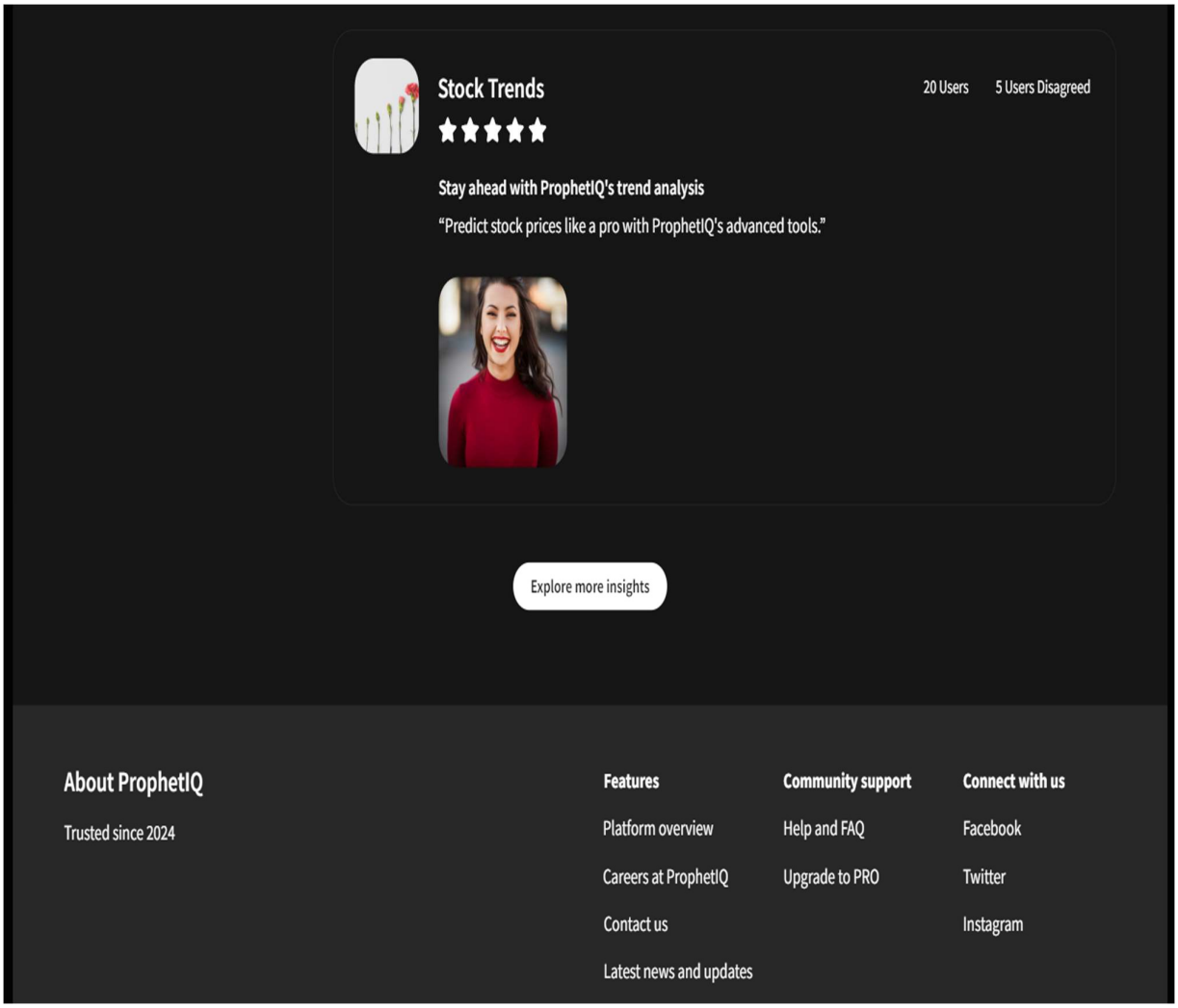
Exclusive deals on stock purchases.

$ 150.00 ♥ 24

Stock Market Growth

Invest in the future of stock market.

$ 300.00 ♥ 30



CONCLUSION

Finally, we have proposed a Recurring Neural Networks (RNN) and Long-Short Term Memory (LSTM) of Keras based Stock Price Prediction model. These methods are used, as both of them are good in handling the time-based problems and also it can make efficient predictions about the future. Our algorithm has trained the model using Yahoo finance's stock price and its related data-set of last 5 years and then tested it by predicting the price of future stocks of the same for next 30 days and our model successfully predicted the price of stock for next 30 days. This study showed that this model is capable enough in predicting the future of up to 'n' days with the help of the RNN with Keras. Also, it is clear that our proposed model can even predict the prices with great accuracy and correctly whenever there is a sudden change in the market also. Changes in the stock are not defined on a confined and

defined pattern, hence the actuality of the same will differ. So, we strongly believe that this analysis will help the investors and people investing in stocks and helps to gain much better profit and helps to predict and analyse stock market in a much better way.

Because stock data is affected by economic factors, political factors or environmental factors, the law of its change is elusive, and the cycle of the law of change is difficult to determine. Therefore, the model still needs a lot of historical data and selection of appropriate variables for analysis to obtain the desired results. In the traditional ARIMA model, when analyzing complex stock markets, its prediction results are not particularly ideal, and there are still certain errors in price prediction. As a technology in the field of deep learning, neural network can solve non-linear problems well. LSTM neural network is optimized on traditional neural network and introduces the concept of "gate", which enhances the long-term memory ability of the model, which enhances its generalization ability. Therefore, the application of LSTM neural network in analyzing financial-related time series data is promising. Based on the understanding of traditional time series analysis and RNN and LSTM neural network, this paper constructs a stock price prediction model based on LSTM neural network. For better comparison, we also established a traditional ARIMA model for comparison. As the neural network has a good predictive effect on nonlinear problems, this article chooses the optimized neural network-LSTM model, and also chooses the use of single-feature and multi-feature input models to seek better prediction results. The traditional time series model focuses on the role of time in stock forecasting.

References:

1) Billah, M. M., Sultana, A., Bhuiyan, F., & Kaosar, M. G. (2024). Stock price prediction: comparison of different moving average techniques using deep learning model.

2) Sonkavde, G., Dharrao, D. S., Bongale, A. M., Deokate, S. T., Doreswamy, D., & Bhat, S. K. (2023). Forecasting Stock Market Prices Using Machine Learning and Deep Learning Models: A Systematic Review, Performance Analysis and Discussion of Implications.

3) Jain, R., & Vanzara, R. (2023). Emerging Trends in AI-Based Stock Market Prediction: A Comprehensive and Systematic Review.

4) Stock Price Prediction Website Using Linear Regression - A Machine Learning Algorithm" by ITM Web of Conferences (2023).

5) Emerging Trends in AI-Based Stock Market Prediction: A Comprehensive and Systematic Review by Rahul Jain and Rakesh Vanzara, published in 2023.

6) Analysis and prediction of Indian stock market: a machine-learning approach by Shilpa Srivastava, Millie Pant & Varuna Gupta, published in 2023.

7) Chen, J., Dou, R., & Gong, L. (2019). A Review of Software-as-a-Service (SaaS) Adoption Models. IEEE Access, 7, 11904-11914.

8) Choudhary, S., Lakhwani, K., & Mohanty, S. (2020). Machine Learning Applications in Financial Markets: A Comprehensive Survey. Journal of Big Data, 7(1), 83.

9) Kshetri, N., & Voas, J. (2019). Data Security and Privacy Issues in Cloud Computing: A Survey. Computer, 52(3), 42-49.

10) Smith, M., & Neuer, T. (2018). Transparency in Predictive Analytics: A Review. Journal of Business Forecasting, 37(3), 23-28.

11)Pawar, K., Jalem, R. S., & Tiwari, V. (2018). Stock Market Price Prediction Using LSTM RNN. In Advances in Intelligent Systems and Computing (Vol. 841).

12)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.

13)Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. In International Joint Conference on Neural Networks (IJCNN).

14) Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PLoS ONE, 12(7), e0180944.

15)Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. Expert Systems with Applications, 42(4), 2162-2172.

16)Di Persio, L., & Honchar, O. (2016). Artificial neural networks architectures for stock price prediction: Comparisons and applications. International Journal of Circuits, Systems and Signal Processing, 10, 403-413.

17)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. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI).

18) Zhuge, Q., & Li, J. (2019). A comparative study of LSTM neural networks in forecasting day-ahead stock volatility by using high-frequency data. Physica A: Statistical Mechanics and its Applications, 519, 127-139.

19)Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. European Journal of Operational Research, 259(2), 689-702.

20)Zhang, L., Aggarwal, C., & Qi, G. J. (2018). Stock price prediction via discovering multi-frequency trading patterns. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[21] Performance Analysis of Indian Stock Market Index using Neural Network Time Series Model D. ASHOK KUMAR, Department of Computer Science Government Arts College Tiruchirappalli, India-620 022.

[22] FPGA Realization of Backpropagation for Stock Market Prediction, Mamun Bin Ibne Rea, Member, IEEE.

[23] Application on Stock Price Prediction of Elman Neural Networks Based on Principal Component Analysis Method Hongyan Shi, Xiaowei Liu, College of Science, Shenyang University of Technology, Shenyang110000, China

[24] Stock market forecasting research based on Neural Network and Pattern Matching LIN Qian Yu, FENG Shao Rong, School of Information Science and Technology, Xiamen University Xiamen Fujian 361005, China

[25] Prediction Model of Stock Market Returns Based on Wavelet Neural Network Yu Zhao, Yu Zhang, Chunjie Qi, College of Economics and Management, Huazhong Agricultural University, Wuhan, Hubei, 430070, P.R. China

[26] Stock market prediction using support vector machine Zhen Hu; Jie Zhu ; Ken Tse 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering

[27] Ashish Sharma; Dinesh Bhuriya ; Upendra Singh 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)

APPENDIX A:

MACHINE LEARNING TECHNIQUES AND DEFINITIONS

ANNs (Artificial Neural Networks): This technique Computing frameworks with a bunch of models and algorithms that are cerebrum motivated, planned to mimic human learning.

BPNN (Back spread neural network): An iterative course of tuning the loads in neural network models from mistakes given by a misfortune work.

BSM (Black Scholes Model): A numerical model for ascertaining cost deviation throughout some undefined time frame for monetary instruments.

CART (Classification and Regression Trees): A twofold choice tree model performed through an arrangement of inquiries, the responses to which are contribution to the following inquiry, and every terminal node is the aftereffect of expectation.

LSTM (Long Short-Term Memory): This model is one of the best RNN structures that presents the memory cell, a unit of calculation that replaces customary fake neurons in the secret layer of the network. These memory cells make networks really partner recollections with input remote in time.

MCS (Monte Carlo Simulation): This is a strategy used to show the likelihood of variation results in difficult to anticipate cases due to the obstruction of arbitrary factors.

MLP (Multilayer Perceptron): A sort of ANN comprising of something like three layers: input layer, stowed away layer, and result layer.

PSO (Particle Swarm Optimization): A strategy performing ideal arrangement looking, enlivened by the way of behaving of social animals in gatherings.

QRNN (Quantile Regression Neural Network): A QR-based crossover and a feedforward neural network that can be utilized to gauge the nonlinear models at various quantiles.

LR (Logistic Regression): An AI calculation for measurable arrangement, where a not set in stone by at least one autonomous factors.

RNN (Recurrent neural networks): A strong model for handling consecutive information like sound, time series information or composed normal language. A few plans of RNN are utilized to foresee the stock market.

SOM (Self getting sorted out maps): A kind of ANN involving solo learning in preparing to assemble a two-layered guide of the issue space.

SVM (Support Vector Machine): An administered AI procedure for grouping and relapse examination. SVR (Support Vector Regression): An administered AI procedure for grouping and relapse examination.

HMM (Hidden Markov Model): A measurable model utilized for AI with stowed away states, instead of a standard Markov chain where all states are revealed.

Hierarchical clustering: Builds a staggered order of groups by making a bunch tree.

KNN (K Nearest Neighbour): A regulated learning procedure use for arrangement, where an item is considered as an individual from a group with a majority casting a ballot of its neighbours.

k-means: A clustering method where an item can have a place with just a single bunch and a group should contain somewhere around one article, in which centroids of bunches are means of articles in each group.

k-medoids (PAM): A clustering procedure where an article can have a place with just a single group and a bunch should contain something like one item, in which centroids of groups are the most focal item in each group.

APPENDIX B:

Comparing of Machine Learning algorithms and techniques in stock market price prediction

(1). ARIMA model

As the stock data is noisy, we must first perform stationarity test on the stock sequence. The test method is to observe the sequence diagram, auto correlation diagram, and partial autocorrelation diagram of the sequence first, and then do a unit root (ADF) test to test its P If the sequence is non-stationary, then we choose the difference for smoothing. After determining the order of the difference, confirm that it is a stationary sequence, which can be used to determine the order of the model, that is, p, q. This article chooses to use the BIC value to determine Order. After the determination is completed, the model is tested, mainly the LB test, to confirm whether the residual is white noise. If it is, then the model passes the test and we can make predictions.

(2). Single Feature LSTM neural network

This model chooses the s&p500 return as the only input feature. First, it is necessary to test the stationarity of the closing price series: generally choose to draw a time series diagram first, and check whether the image has an obvious trend; then, we 22 also need to draw a correlation diagram, and through observing whether the a cf image is quickly reduced to 0 to judge the stationarity; then perform the ADF test; finally, if the data does not have stationarity, then the difference method is needed to smooth the data. After smoothing the data, you can construct a single-feature LSTM neural network. This article chooses a three-layer LSTM network, that is, there is only one hidden layer, and the input layer has 20 neurons, so that it can process 20 days of stock prices, Because we are calculating the closing of the next day, so the output layer has only 1 neuron, which is used to output the stock price of the twenty-first day. 20 is chosen because after n-fold cross-checking, 20 is found to be the optimal parameter.

(3) GARCH model

In the 1980s, Engel proposed ARCH (auto regressive conditional hetero ske dastic process) model, which is an autoregressive conditional heteroskedastic process model, can be used to make such predictions. The ARCH model defined by Engel. The GARCH model hold an idea that the variance for the change of return can be predictable, not only the latest information, but also the previous conditional variance will have an affection on the conditional variance. In order to simplify the calculation, the risk metrics proposed by the JP Morgan Group's risk management company uses a simple and practical GARCH(1,1).

(4) Mixed model construction

The mixed model is constructed by ensembling three models including ARIMA model, Garch Model and LSTM model. The innovation of the article emphasizes the long-term dependence of LSTM on performance to improve accuracy, and ensemble can improve the robustness of the model.

(5) Estimator parameters

Since the ultimate goal of stock market forecasting is profit, how to correctly evaluate the model and select the model with the best profitability is very important in stock market forecasting. The current stock market forecast research generally adopts a two-stage model evaluation method: first, the performance of the model is evaluated, 23 N 1 N∑(observed − predicted i i) 2 i=1 ∑ = N ∑(−) and then the model with the best performance is selected to evaluate the profitability of the model. The performance evaluation of the stock market forecasting model usually adopts the classification evaluation indicators, such as the accuracy rate andF1 value, and the profitability of the stock market forecasting model is estimated by various simulated trading algorithms. There may be a lack of consistency between the above two evaluation methods, that is, the profitability of the model with the best classification evaluation performance is not necessarily the best. This inconsistency can lead to the improvement of stock market forecasting models

without valuable guidance. How to reduce this inconsistency and improve the validity of model evaluation is a difficult point in stock market research.

(6) LSTM process:

In traditional neural networks, neurons in the same hidden layer are not connected to each other, and this structural defect directly leads to their poor performance in dealing with certain problems. This shortcoming becomes especially acute when dealing with time series and speech recognition problems where information is contextualized. The emergence of the recurrent neural network solves this problem very well. The neurons in the same hidden layer are connected to each other, which can effectively obtain the contextual information of the data. The output of the recurrent neural network is determined according to the input and the previous related information, so it can play its short-term memory when dealing with timeseries problems. Although the effect of recurrent neural network in dealing with time series problems is very good, there are still some problems. The more serious one is that gradient disappears or explodes easily in the processing of long-term span problems. Causes the phenomenon of small memory value. After the cyclic neural network is expanded, it can be regarded as a multi-layer feedforward neural network with each-layer sharing the same weight parameters. Although it keeps trying to learn the long term dependencies of sequences, actual research finds this to be a difficult task indeed.

Table 2:

Paper	Prediction Techniques	Stocks/Index	Input Data	Accuracy (%)	Error (%)
Hegazy et al. (2014)	PSO, LS-SVM, ANN	S&P 500	Historical daily stock prices	N/A	LS-SVM: 0.1147 PSO: 0.7417 ANN: 1.7212; Note: average of 13 companies which cover all stock sectors in S&P 500 stock market
Adebiyi et al. (2014)	ARIMA, ANN	Dell index	Historical daily stock prices	N/A	ARIMA: 0.608 ANN: 0.8614; Note: average of one month prediction
Nguyen et al. (2015)	SVM	AAPL, AMZN, BA, BAC, CSCO, DELL, EBAY,ETFC, GOOG, IBM, INTC, KO, MSFT, NVDA, ORCL, T, XOM, YHOO	Historical daily stock prices and mood information	54.41 (average) 60.00 (few stocks)	N/A
Patel et al. (2015)	ANN, SVM, RF, Naïve-Bayes	CNX nifty index, S&P Bombay Stock Exchange (BSE) Sensex index, Infosys Ltd., Reliance Industries	Historical daily stock prices	Naïve-Bayes: 90.19 RF: 89.98 SVM: 89.33 ANN: 86.69	N/A
Attigeri et al. (2015)	LR	Stock market price of two companies	Historical daily stock prices, news articles, and social media data (twitter)	LR: 70	N/A
Dang and Duong (2016)	SVM	VN30 Index: EIB, MSN, STB, VIC, VNM	News relating to companies in the VN30 Index	SVM: 73	N/A
Selvin et al. (2017)	LSTM, RNN, CNN, ARIMA	NIFTY-IT index (Infosys, TCS), NIFTY-Pharma index (Cipla)	Minute by minute stock prices (day stamp, time stamp, transaction id, stock price, and volume traded)	N/A	**Infosys**: CNN: 2.36/ RNN: 3.9/ LSTM: 4.18 ARIMA: 31.91 **TCS**: CNN: 8.96/ RNN: 7.65/ LSTM: 7.82 ARIMA: 21.16 **Cipla**: CNN: 3.63/ RNN: 3.83/ LSTM: 3.94 ARIMA: 36.53
Roncoroni et al. (2015)	LSTM	NIFTY 50	Historical daily stock prices	N/A	0.00859
Khare et al. (2017)	LSTM, MLP	10 unique stocks on New York Stock Exchange	Minute by minute stock prices	N/A	MLP: 0.0025 LSTM: 0.048
Althelaya et al. (2018a)	MLP, LSTM, SLSTM, BLSTM	S&P 500	Historical daily stock prices (closing price)	N/A	**Short-term**: BLSTM: 0.00947 SLSTM: 0.01248 LSTM: 0.01582 MLP: 0.03875 **Long-term**: BLSTM: 0.06055 SLSTM: 0.06637 LSTM: 0.08371 MLP: 0.09369

---------- END OF THE REPORT ----------