A

A Major Project Report

On

##### ADVANCED ALGORITHM IN STOCK MARKET

PRICE FORECASTING

##### A PROJECT REPORT

###### ***Submitted by***

##### M.CHARAN

A.JAYASANKAR

A.PAVANKUMAR

B.CHAITANYA PRASAD

C.REDDYSEKHAR

***in partial fulfillment for the award of the degree***

***of***

##### DEPARTMENT OF CSE(AI&ML)

IN

BRANCH OF STUDY



**SRI VENKATESWARA COLLEGE OF ENGINEERING & TECHNOLOGY**

(Affiliated to JNTUA, Anantapur, Approved by AICTE, New Delhi, Accredited by NBA, New Delhi & NAAC with A+ Grade, Bangalore.

An ISO 9001-2000 Certified Institution), R.V.S NAGAR, CHITTOOR-517127

APRIL 2024

ADVANCED ALGORITHM IN STOCK MARKET PRICE FORECASTING

##### A PROJECT REPORT

###### ***Submitted by***

##### M.CHARAN(20781A3332)

A.JAYASANKAR(20781A3301)

A.PAVAN KUMAR(20781A3302)

B.CHAITANYA PRASAD(20781A3309)

C.REDDY SEKHAR(20781A3313)

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

in

# CSE(AI&ML)



**SRI VENKATESWARA COLLEGE OF ENGINEERING & TECHNOLOGY**

(Affiliated to JNTUA, Anantapur, Approved by AICTE, New Delhi, Accredited by NBA, New Delhi & NAAC with A+ Grade, Bangalore.

An ISO 9001-2000 Certified Institution), R.V.S NAGAR, CHITTOOR-517127

APRIL 2024



**SRI VENKATESWARA COLLEGE OF ENGINEERING & TECHNOLOGY**

**BONAFIDE CERTIFICATE**

Certified that this project report **“ADVANCED ALGORITHM IN STOCK MARKET PRICE FORECASTING ”**is the Bonafide work of “**M.Charan,A.Jayasankar,A.Pavankumar,B.Chaitanyaprasad,C.Reddysekhar”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

Dr.M.Lavanya Dr.R.Mohanraj

**HEAD OF THE DEPARTMENT SUPERVISOR**

Department of CSE(AI&ML) Department of CSE(AI&ML)

Sri Venkateswara College of Engineering Sri Venkateswara College of engineering

and Technology(SVCET)R.V.S Nagar, and Technology(SVCET)R.V.S Nagar,

Tirupati Road,Chittoor,A.P-517127 Tirupati Road,Chittoor,A.P-515127

TABLE OF CONTENTS

**CHAPTER NO. TITLE PAGE NO.**

ABSTRACT 6

**LIST OF TABLE 6**

LIST OF FIGURES 7

**LIST OF SYMBOLS 8**

**1. INTRODUCTION 7-9**

1.1 OVERVIEW

1.2 AIM AND OBJECTIVE

1.3 STOCK MARKET

1.4 MOTIVATION

**2. LITERATURE REVIEW 10**

2.1 GENERAL

**3. SYSTEM ANALYSIS 11-12**

3.1 OBJECTIVE

3.2 EXISTING SYSTEM

3.3 PROPOSED SYSTEM

**4. SYSTEM IMPLEMENTATION 13-21**

4.1 INTRODUCTION

4.2 SYSTEM REQUIREMENT

4.2.1 HARDWARE REQUIREMENT

4.2.2 SOFTWARE REQUIREMENT

4.3 ARCHITECTURE

**CHAPTER NO. TITLE PAGE NO.**

4.4 MODULE DECRYPTION

4.4.1 DATA PREPROCESSING

4.4.2 FEATURE SELECTION

4.4.3 BUILDING AND TRAINING MODEL

4.5 PYTHON TECHNOLOGY

4.5.1 PYTHON PLATFORM

4.5.2 PYTHON

4.5.3 PYTHON LIBRARY

**5. ALGORITHM 22-26**

**6. STEP OF IMPLEMENTATION 27-34**

**7. CONCLUSION AND FUTURE WORK 35**

7.1 CONCLUSION

7.2 FUTURE WORK

**8 REFERENCES 36-37**

# ADVANCED ALGORITHMS IN STOCK MARKET PRICE FORECASTING ABSTRACT

In recent years, there has been a growing trend in attempting to forecast market trends, which has garnered interest from economists, policymakers, academics, and market professionals. The goal of this proposed project is to explore and enhance supervised learning algorithms for predicting stock prices. Utilizing data mining for stock market analysis can be particularly beneficial for new investors, guiding their decisions in the stock market based on a variety of relevant factors. This encompasses daily stock market activities, such as the calculation of indexes like the Sensex and the trading of shares. The stock exchange offers a platform for the efficient and transparent trading of equity, debt instruments, and derivatives. Our project aims to develop software that can analyze historical stock data of specific companies using various influential parameters. By applying these parameters to different data mining algorithms, we can determine which algorithm yields the most accurate predictions. This approach will also assist in forecasting the future values of specific stocks

**CHAPTER 1 INTRODUCTION**

## OVERVIEW

In recent times stock market predictions is gaining more attention, maybe due to the fact that if the trend of the market is successfully predicted the investors may be better guided. The profits gained by investing and trading in the stock market greatly depends on the predictability. If there is a system that can consistently predict the direction of the dynamic stock market will enable the users of the system to make informed decisions. More over the predicted trends of the market will help the regulators of the market in taking corrective measures.

## AIM AND OBJECTIVE

The aim of the project is to examine a number of different forecasting techniques to predict future stock returns based on past returns and numerical news indicators to construct a portfolio of multiple stocks in order to diversify the risk. We do this by applying supervised learning methods for stock price forecasting by interpreting the seemingly chaotic market data.

## STOCK MARKET

A stock market, equity market or share market is the aggregation of buyers and sellers (a loose network of economic transactions, not a physical facility or discrete entity) of stocks (also called shares), which represent ownership claims on businesses; these may include securities listed on a public stock exchange as well as those only traded privately. Examples of the latter include shares of private companies which are sold to investors through equity crowd funding platforms. Stock exchanges list shares of common equity as well as other security types, e.g. corporate bonds and convertible bonds.

Stock price prediction is one of the most widely studied problem, attracting researchers from many fields. The volatile nature of the stock market makes it really difficult to apply simple time-series or regression techniques. Financial institutions and active traders have created various proprietary models to beat the market for themselves or their clients, but rarely did anyone achieve consistently higher than the average returns on investment. The challenge of stock market price forecasting is so appealing because an improvement of just a few points of percentage can increase the profit by millions of dollars. This paper discusses the application of Support Vector Machines and Linear Regression in detail along with the pros and cons of the given methods. The paper introduces the parameters and variables which can be used to recognize the patterns in stock prices which can be helpful in future stock prediction and how boosting can be integrated with various other machine learning algorithms to improve the accuracy of our prediction systems.

## MOTIVATION

Stock price prediction is a classic and important problem. With a successful model for stock prediction, we can gain insight about market behavior over time, spotting trends that would otherwise not have been noticed. With the increasingly computational power of the computer, machine learning will be an efficient method to solve this problem.

Thus, our motivation is to design a public service incorporating historical data and users predictions to make a stronger model that will benefit everyone.

# CHAPTER 2

# LITEATURE SURVEY

Predicting stock market prices has been a topic of interest among both analysts and researchers for a long time. Stock prices are hard to predict because of their high volatile nature which depends on diverse political and economic factors, change of leadership, investor sentiment, and many other factors. Predicting stock prices based on either historical data or textual information alone has proven to be insufficient. Existing studies in sentiment analysis have found that there is a strong correlation between the movement of stock prices and the publication of news articles. Several sentiment analysis studies have been attempted at various levels using algorithms such as support vector machines, naive Bayes regression, and deep learning. The accuracy of deep learning algorithms depends upon the amount of training data provided. However, the amount of textual data collected and analyzed during the past studies has been insufficient and thus has resulted in predictions with low accuracy. In our paper, we improve the accuracy of stock price predictions by gathering a large amount of time series data and analyzing it in relation to related news articles, using deep learning models. The dataset we have gathered includes daily stock prices for S&P500 companies for five years, along with more than 265,000 financial news articles related to these companies. Given the large size of the dataset, as an invaluable resource for training prediction models and performing inference for a given stock in real time. Index Terms-stock market prediction, cloud, big data, machine learning, regression.

# CHAPTER 3

# SYSTEM ANALYSIS

## OBJECTIVES

The aims of this project are as to identify factors affecting share market, To generate the pattern from large set of data of stock market for prediction of NEPSE and to predict an approximate value of share price to provide analysis for users through web application

The objective of the system is to give a approximate idea of where the stock market might be headed. It does not give a long term forecasting of a stock value. There are way too many reasons to acknowledge for the long term output of a current stock. Many things and parameters may affect it on the way due to which long term forecasting is just not feasible.

## EXISTING SYSTEM

Nowadays, as the connections between worldwide economies are tightened by globalization, external perturbations to the financial markets are no longer domestic. With evolving capital markets, more and more data is being created daily.

The intrinsic value of a company’s stock is the value determined by estimating the expected future cash flows of a stock and discounting them to the present, which is known as the book value. This is distinct from the market value of the stock, that is determined by the company’s stock price. This market value of a stock can deviates from the intrinsic value due to reasons unrelated to the company’s fundamental operations, such as market sentiment.

The fluctuation of stock market is violent and there are many complicated financial indicators. Only few people with extensive experience and knowledge can understand the meaning of the indicators and use them to make good prediction to get fortune. Most people have to rely solely on luck to earn money from stock trading. However, the advancement in technology, provides an opportunity to gain steady fortune from stock market and also can help experts to find out the most informative indicators to make better prediction. The prediction of the market value is of paramount importance to help in maximizing the profit of stock option purchase while keeping the risk low.

## PROPOSED SYSTEM

Linear Regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression.

### Advantages

* Space complexity is very low it just needs to save the weights at the end of training. Hence, it's a high latency algorithm.
* Its very simple to understand
* Good interpretability

# 

# 

# CHAPTER 4

**SYSTEM IMPLEMENTATION**

## INTRODUCTION

Design is a multi- step that focuses on data structure software architecture, procedural details, procedure etc… and interface among modules. The design procedure also decodes the requirements into presentation of software that can be accessed for excellence before coding begins. Computer software design change continuously as novel methods; improved analysis and border understanding evolved. Software proposal is at relatively primary stage in its revolution.

Therefore, software design methodology lacks the depth, flexibility and quantitative nature that are usually associated with more conventional engineering disciplines. However, methods for software designs do exit, criteria for design qualities are existing and design notation can be applied.

## SYSTEM REQUIREMENTS

### Hardware Requirements

* PROCESSOR: i3 or more
* RAM: 8 GB
* PROCESSOR: 2.4 GHZ
* MAIN MEMORY: 8GB RAM
* HARD DISK DRIVE: 1TB
* KEYBOARD :104 KEYS

### Software Requirements

* FRONT END: PYTHON
* IDE: ANACONDA, GOOGLE COLLAB
  + OPERATING SYSTEM: WINDOWS 10 OR MORE

## ARCHITECTURE

### Fig 4.1 Data Flow Diagram

***Fig 4.2 Architecture Design***

## MODULE DECRIPTION

The implementation of this project is divided into following steps

4.4.1 Data Preprocessing

* + 1. Feature selection

4.4.3 Building and Training Model

### Data Preprocessing:

The entries are present in the dataset. The null values are removed using df = df.dropna() where df is the data frame. The categorical attributes (Date,High,Low,Close,Adj value) are converted into numeric using Label Encoder. The date attribute is spitted into new attributes like total which can be used as feature for the model.

### Feature selection:

Features selection is done which can be used to build the model. The attributes used for feature selection are Date,Price,Adj close,Forecast X coordinate , Y coordinate, Latitude , Longitude, Hour and month,

### Building and Training Model:

After feature selection location and month attribute are used for training. The dataset is divided into pair of xtrain ,ytrain and xtest, y test. The algorithms model is imported form sklearn. Building model is done using model. Fit (xtrain, ytrain). This phase would involve supervised classification methods like linear regression, Ensemble classifiers (like Adaboost, Random Forest Classifiers), etc.

## PYTHON TECHNOLOGY

Python is an interpreted, object- oriented programming language similar to PERL, that has gained popularity because of its clear syntax and readability. Python is said to be relatively easy to learn and portable, meaning its statements can be interpreted in a number of operating systems, including UNIX- based systems, Mac OS, MS- DOS, OS/2, and various versions of Microsoft Windows 98. Python was created by Guido van Rossum, a former resident of the Netherlands, whose favourite comedy group at the time was Monty Python's Flying Circus. The source code is freely available and open for modification and reuse. Python has a significant number of users.

A notable feature of Python is its indenting of source statements to make the code easier to read. Python offers dynamic data type, ready- made class, and interfaces to many system calls and libraries. It can be extended, using the C or C++language.

Python can be used as the script in Microsoft's Active Server Page (ASP) technology. The scoreboard system for the Melbourne (Australia) Cricket Ground is written in Python. Z Object Publishing Environment, a popular Web application server, is also written in the Python language’s

### Python Platform

Apart from Windows, Linux and MacOS, C,Python implementation runs on 21 different **platforms**. Iron Python is a .NET framework based **Python** implementation and it is capable of running in both

A notable feature of Python is its indenting of source statements to make the code easier to read. Python offers dynamic data type, ready- made class, and interfaces to many system calls and libraries. It can be extended, using the C or C++language.

Python can be used as the script in Microsoft's Active Server Page (ASP) technology. The scoreboard system for the Melbourne (Australia) Cricket Ground is written in Python. Z Object Publishing Environment, a popular Web application server, is also written in the Python language’s

### Python Library

Machine Learning, as the name suggests, is the science of programming a computer by which they are able to learn from different kinds of data. A more general definition given by Arthur Samuel is –“Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed.” They are typically used to solve various types of life problems.

In the older days, people used to perform Machine Learning tasks by manually coding all the algorithms and mathematical and statistical formula. This made the process time consuming, tedious and inefficient. But in the modern days, it is become very much easy and efficient compared to the olden days by various python libraries, frameworks, and modules. Today, Python is one of the most popular programming languages for this task and it has replaced many languages in the industry, one of the reason is its vast collection of libraries. Python libraries that used in Machine Learning are:

* Numpy
* Scipy
* Scikit- learn
* Theano
* Tensor flow
* Keras
* Pytorch
* Pandas
* Matplotlib

**4.5.5.1*NumPy***

NumPy is a very popular python library for large multi- dimensional array and matrix processing, with the help of a large collection of high- level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High- end libraries like TensorFlow uses NumPy internally for manipulation of Tensors.

**4.5.5.2Scipy:**

SciPy is a very popular library among Machine optimization, linear algebra, integration and statistics. There is a difference between the SciPy library and the SciPy stack. The SciPy is one of the core packages that make up the SciPy stack. SciPy is also very useful for image manipulation.

**4.5.5.3 Scikit:**

Skikit- learn is one of the most popular ML libraries for classical ML algorithms. It is built on top of two basic Python libraries, viz., NumPy and SciPy. Scikit- learn supports most of the supervised and unsupervised learning algorithms. Scikit- learn can also be used for data- mining and data- analysis, which makes it a great tool who is starting out with ML.

**4.5.5.4 Theano:**

We all know that Machine Learning is basically mathematics and statistics. Theano is a popular python library that is used to define, evaluate and optimize

Mathematical expressions involving multi- dimensional arrays in an efficient manner. It is achieved by optimizing the utilization of CPU and GPU. It is extensively used for unit- testing and self- verification to detect and diagnose different types of errors. Theano is a very powerful library that has been used in large- scale computationally intensive scientific projects for a long time but is simple and approachable enough to be used by individuals for their own projects.

**4.5.5.5 Tensorflow:**

TensorFlow is a very popular open- source library for high performance numerical computation developed by the Google Brain team in Google. As the name suggests, Tensorflow is a framework that involves defining and running computations involving tensors. It can train and run deep neural networks that can be used to develop several AI applications. TensorFlow is widely used in the field of deep learning research and application.

**4.5.5.6 Keras:**

Keras is a very popular Machine Learning library for Python. It is a high- level neural networks API capable of running on top of TensorFlow, CNTK, or Theano. It can run seamlessly on both CPU and GPU. Keras makes it really for ML beginners to build and design a Neural Network. One of the best thing about Keras is that it allows for easy and fast prototyping.

**4.5.5.7 Pytorch:**

PyTorch is a popular open- source Machine Learning library for Python based on Torch, which is an open- source Machine Learning library which is implemented in C with a wrapper in Lua. It has an extensive choice of tools and libraries that supports on Computer Vision, Natural Language Processing(NLP) and many more ML programs. It allows developers to perform computations on Tensors with GPU acceleration and also helps in creating computational graphs.

**4.5.5.8 Pandas:**

Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training.

*In* this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high- level data structures and wide variety tools for data analysis. It provides many inbuilt methods for groping, combining and filtering data.

**4.5.5.9 Matplotlib:**

Matpoltlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named plot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, etc.

**CHAPTER 5**

**ALGORITHM**

Linear regression is one of the simplest algorithms used in stock market prediction. It predicts the future value of a stock based on historical data by fitting a straight line to the data points. However, the stock market is influenced by various factors, so linear regression may not always provide accurate predictions. It's often used in combination with other techniques for more robust forecasting.

**LSTM in Stock Market Price Forecasting**

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to capture long-term dependencies in sequential data, making it suitable for time-series forecasting tasks like stock market price prediction.

Here's a detailed explanation of LSTM in the context of stock market price forecasting:

**Sequential Data Handling:** Stock market data, such as prices over time, is inherently sequential. LSTM networks are well-suited for handling such data because they can remember information over long periods, which is crucial for capturing patterns in stock price movements.

**Memory Cells:** LSTMs have memory cells that maintain a memory state, allowing them to retain information over long sequences of data. These memory cells are equipped with gates (input gate, forget gate, and output gate) that control the flow of information into and out of the cell, regulating what information should be stored or discarded.

**Capturing Patterns:** LSTMs can capture complex patterns and relationships in the data. For stock market forecasting, these patterns may include seasonality, trends, and other cyclical behaviors. The network learns to extract features from the input data that are relevant for predicting future stock prices.

**Feature Extraction:** Before feeding the data into the LSTM network, feature extraction is often performed to enhance the model's ability to learn. Features could include historical prices, trading volumes, technical indicators (e.g., moving averages, RSI), and external factors (e.g., news sentiment, economic indicators).

**Training Process:** The LSTM network is trained on historical stock market data, where the input sequences are typically represented as sliding windows of past prices or features. The network learns to minimize the prediction error between its forecasts and the actual future prices. This is achieved through backpropagation and gradient descent.

**Model Evaluation:** Once trained, the LSTM model is evaluated on a separate validation dataset to assess its performance. Common evaluation metrics include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

**Prediction:** After training and validation, the LSTM model can be used to make predictions on unseen data, i.e., future stock prices. These predictions can provide insights for trading decisions, risk management, and portfolio optimization.

Overall, LSTM networks offer a powerful framework for stock market price forecasting by leveraging their ability to capture long-term dependencies and intricate patterns in sequential data. However, it's important to note that predicting stock prices accurately is challenging due to the inherent uncertainty and complexity of financial markets. Therefore, LSTM models are often used in conjunction with other techniques and tools to improve forecasting accuracy.

By leveraging these advanced algorithms and techniques, researchers and practitioners aim to enhance the accuracy, robustness, and interpretability of stock market price forecasting systems, ultimately empowering traders, investors, and financial analysts to make better informed decisions in dynamic and uncertain market environments.

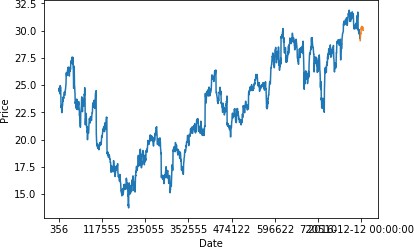
The model attained a Train Score of 0.00106 MSE (0.03 RMSE) and a Test Score of 0.00875 MSE (0.09 RMSE). It is observed that the system achieves greater accuracy with increased training and larger dataset sizes. The LSTM Model exhibits higher accuracy compared to the Regression-based Model.

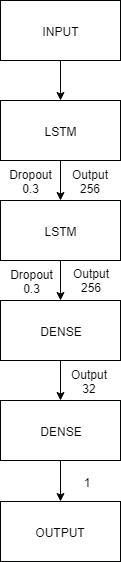
Fig. 2 LSTM Layers

LSTM, or Long Short-Term Memory, represents an advancement over Recurrent Neural Networks (RNN). Unlike RNNs, which primarily focus on recent information LSTM models excel in preserving information from previous states, enabling them to capture long-term dependencies. This feature allows LSTM models to process information over extended intervals, making them well-suited for tasks like stock market prediction.

# IMPLEMENTATION

The proposed system is trained and tested using a dataset obtained from Yahoo Finance. The dataset is partitioned into training and testing sets, and the system is assessed using different models.

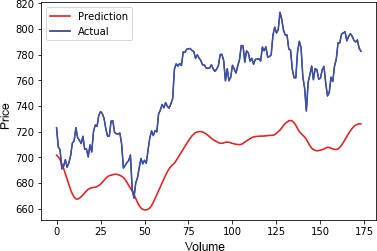
**A. Results of the Regression-Based Model**

Figure 3 depicts the results of applying the linear regression algorithm to the dataset, forecasting fluctuating prices over time.Figure 3: Plot of Price against Date Using

**Regression**

The graph shown in Figure 3 is generated using data with a batch size of 512 and trained over 90 epochs. The R-square confidence test resulted in a confidence score of 0.86625.

1. **Results of the LSTM-Based Model**



**Figure 4: Plot of Actual and Predicted Trends Using LSTM**

The efficiency of the LSTM-based model is notable, as the prediction closely approximates the real trend over extended periods of time. The model achieved a Train Score of 0.00106 MSE (0.03 RMSE) and a Test Score of 0.00875 MSE (0.09 RMSE). It is observed that increased training and larger dataset sizes contribute to higher accuracy. Overall, the LSTM Model demonstrated greater accuracy compared to the Regression-based Model.

**CHAPTER 6**

# STEP OF IMPLEMENTATION

### Step 1:Importing the Libraries

As we all know, the first step is to import the libraries required to preprocess Microsoft Corporation stock data and the other libraries required for constructing and visualizing the LSTM model outputs. We’ll be using the Keras library from the TensorFlow framework for this. All modules are imported from the Keras library.

### ****Step 2: Getting to Visualizing the Stock Market Prediction Data****

Using the Pandas Data Reader library, we will upload the stock data from the local system as a Comma Separated Value (.csv) file and save it to a pandas DataFrame. Finally, we will examine the data.

### ****Step 3: Checking for Null Values by Printing the DataFrame Shape****

In this step, firstly, we will print the structure of the dataset. We’ll then check for null values in the data frame to ensure that there are none. The existence of null values in the dataset causes issues during training since they function as outliers, creating a wide variance in the training process.

### ****Step 4: Plotting the True Adjusted Close Value****

The Adjusted Close Value is the final output value that will be forecasted using the Machine Learning model. This figure indicates the stock’s closing price on that particular day of stock market trading.

### ****Step 5: Setting the Target Variable and Selecting the Features****

The output column is then assigned to the target variable in the following step. It is the adjusted relative value of Microsoft Stock in this situation. Furthermore, we pick the features that serve as the independent variable to the target variable (dependent variable). We choose four characteristics to account for training purposes:

**Step 6: Scaling**

To decrease the computational cost of the [data](https://www.analyticsvidhya.com/blog/2022/06/python-stock-analysis-for-beginners/) in the table, we will scale the stock values to values between 0 and 1. As a result, all of the data in large numbers is reduced, and therefore memory consumption is decreased. Also, because the data is not spread out in huge values, we can achieve greater precision by scaling down. To perform this, we will be using the MinMaxScaler class of the sci-kit-learn library

### ****Step 7: Creating a Training Set and a Test Set for Stock Market prediction****

Before inputting the entire dataset into the training model, we need to partition it into training and test sets. The Machine Learning LSTM model will undergo training using the [data i](https://www.analyticsvidhya.com/blog/2022/06/python-stock-analysis-for-beginners/)n the training set, and its accuracy and backpropagation will be tested against the test set.To accomplish this, we will employ the TimeSeriesSplit class from the sci-kit-learn library. We will configure the number of splits to be 10, indicating that 10% of the data will serve as the test set, while the remaining 90% will train the LSTM model. The advantage of employing this Time Series split lies in its examination of data samples at regular time intervals.

### ****Step 8: Data Processing For LSTM****

Once the training and test sets are finalized, we will input the data into the LSTM model. Before we can do that, we must transform the training and test set data into a format that the LSTM model can interpret. As the LSTM needs that the data to be provided in the 3D form, we first transform the training and test data to NumPy arrays and then restructure them to match the format (Number of Samples, 1, Number of Features). Now, 6667 are the number of samples in the training set, which is 90% of 7334, and the number of features is 4. Therefore, the training set is reshaped to reflect this (6667, 1, 4). Likewise, the test set is reshaped.

### ****Step 9: Building the LSTM Model for Stock Market Prediction****

Finally, we arrive at the point when we construct the LSTM Model. In this step, we’ll build a Sequential Keras model with one LSTM layer. The LSTM layer has 32 units and is followed by one Dense Layer of one neuron.

We compile the model using Adam Optimizer and the Mean Squared Error as the loss function. For an LSTM model, this is the most preferred combination. The model is plotted and presented below.

### ****Step 10: Training the Stock Market Prediction Model****

Finally, we use the fit function to train the LSTM model created above on the training data for 100 epochs with a batch size of 8.

### ****Step 11: Making the LSTM Prediction****

Now that we have our model ready, we can use it to forecast the Adjacent Close Value of the Microsoft stock by using a model trained using the LSTM network on the test set. We can accomplish this by employing simple prediction model on the LSTM model

### Step 12: Comparing Predicted vs True Adjusted Close Value – LSTM

Finally, now that we’ve projected the values for the test set, we can display the graph to compare both Adj Close’s true values and Adj Close’s predicted value using the LSTM Machine Learning model.

model import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

import math

import csv

from sklearn.preprocessing import MinMaxScaler

from sklearn.metrics import mean\_squared\_error

from sklearn import linear\_model

from keras.models import Sequential

from keras.layers import Dense, Activation

from keras.layers import LSTM

dataset = pd.read\_csv('ICICI.csv', usecols=[4])

ClosePrice = dataset

#preprossecing

ClosePrice = np.reshape(ClosePrice.values, (len(ClosePrice),1))

scaler = MinMaxScaler(feature\_range=(0, 1))

ClosePrice = scaler.fit\_transform(ClosePrice)

# Splitting Training and Testing data

train\_Data = int(len(ClosePrice) \* 0.75) #75% data as train and 25% as test

test\_Data = len(ClosePrice) - train\_Data

train\_Data, test\_Data = ClosePrice[0:train\_Data,:],ClosePrice[train\_Data:len(ClosePrice),:]

def new\_dataset(dataset):

    data\_X, data\_Y = [], []

    for i in range(len(dataset)-2):

        a = dataset[i:(i+1), 0]

        data\_X.append(a)

        data\_Y.append(dataset[i+1, 0])

    return np.array(data\_X), np.array(data\_Y)

trainX, trainY = new\_dataset(train\_Data)

testX, testY = new\_dataset(test\_Data)

trainX = np.reshape(trainX, (trainX.shape[0], 1, trainX.shape[1]))

testX = np.reshape(testX, (testX.shape[0], 1, testX.shape[1]))

# ANN model

model = Sequential()

model.add(LSTM(32, input\_shape=(1, 1), return\_sequences = True))

.add(LSTM(16))

model.add(Dense(1))

model.add(Activation('linear'))

# Fitting model

model.compile(loss='mean\_squared\_error', optimizer='adagrad',metrics=['mse'])

history=model.fit(trainX, trainY, epochs=2, batch\_size=1, verbose=2)

# Predicting

trainPredict = model.predict(trainX) #Traning data

testPredict = model.predict(testX) #Testing data

# De-normalizing for ploating

trainPredict = scaler.inverse\_transform(trainPredict)

trainY = scaler.inverse\_transform([trainY])

testPredict = scaler.inverse\_transform(testPredict)

testY = scaler.inverse\_transform([testY])

# Traning dataset plot

trainPredictPlot = np.empty\_like(ClosePrice)

trainPredictPlot[:, :] = np.nan

trainPredictPlot[1:len(trainPredict)+1, :] = trainPredict

# Test dataset plot

testPredictPlot = np.empty\_like(ClosePrice)

testPredictPlot[:, :] = np.nan

testPredictPlot[len(trainPredict)+3:len(ClosePrice)-1, :] = testPredict

dates = []

prices = []

def getData(filename):

    with open (filename,'r') as csvfile:

        csvFileReader = csv.reader(csvfile)

        next(csvFileReader)

        for row in csvFileReader:

            dates.append(int(row[2].split('-')[0]))

            prices.append(float(row[3]))

    return prices,dates

def predict\_price\_LinearReg(dates, prices):

    dates = np.reshape(dates, (len(dates),1))

    prices = np.reshape(prices, (len(prices),1))

    linear\_mod = linear\_model.LinearRegression() # defining the linear regression model

    linear\_mod.fit(dates, prices) # fitting the data points in the model

    xyz = linear\_mod.predict(prices)

    next\_day\_index = np.array([len(prices)]).reshape(-1, 1)

    predicted\_price = linear\_mod.predict(next\_day\_index)[0][0]

    return predicted\_price, xyz

# Price for next day

last\_val = testPredict[-1]

next\_val = model.predict(np.reshape(last\_val, (1,1,1)))

print("\n-----Last Day Value:", last\_val.item())

print("-----Next Day Value Using ANN:", (last\_val+next\_val).item())

if last\_val.item() < (last\_val + next\_val).item():

   print("\n----Price will go up Buy share----")

else:

   print("\n----Alert Price will go down----")

prices,dates= (getData('ICICI.csv'))

predicted\_price,linearP = predict\_price\_LinearReg(dates, prices)

print ("\nNext day Using Linear Regression is: ", str(predicted\_price))

ClosePrice = scaler.inverse\_transform(ClosePrice)

plt.plot(ClosePrice, 'g', label = 'Orignal dataset')

plt.plot(trainPredictPlot, 'r', label = 'Traning set')

plt.plot(testPredictPlot, 'b', label = 'Predicted Price ANN')

plt.plot(linearP, color= 'black', label= 'Linear model')

plt.legend(loc = 'lower right')

plt.xlabel('Time in Days')

plt.ylabel('Stock prices')

plt.show()

# NMSE is used for evaluating the prediction accuracy of the model

trainScore = math.sqrt(mean\_squared\_error(trainY[0], trainPredict[:,0]))

print('Train NMSE: %.2f' % (trainScore))

# NMSE is used for evaluating the prediction accuracy of the model

testScore = math.sqrt(mean\_squared\_error(testY[0], testPredict[:,0]))

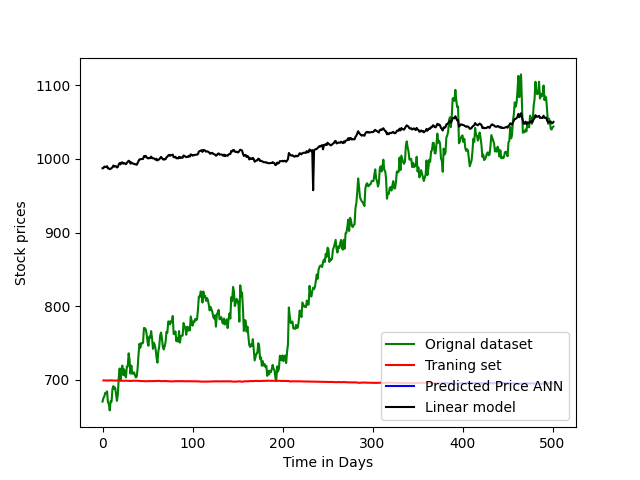
print('Test NMSE: %.2f' % (testScore))

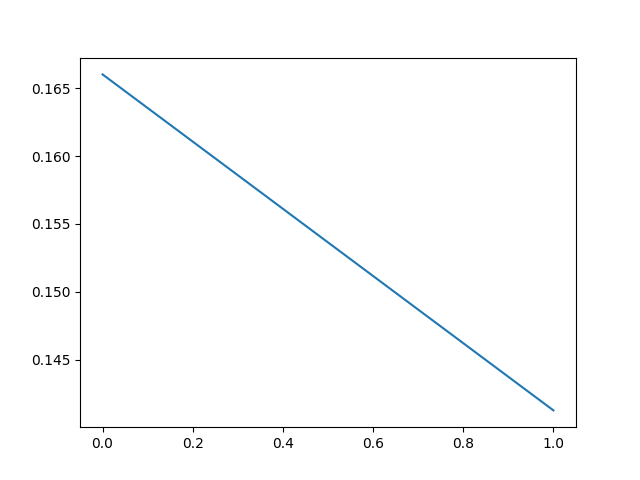
#Plot error graph

plt.plot(history.history['mse'])

plt.show()

**OUTPUT:**





# CHAPTER 7

# CONCLUSION AND FUTURE WORK

## CONCLUSION

By measuring the accuracy of the Linear Regression algorithms, we found that the most suitable algorithm for predicting the market price of a stock based on various data points from the historical data. The algorithm will be a great asset for brokers and investors for investing money in the stock market since it is trained on a huge collection of historical data and has been chosen after being tested on a sample data. The project demonstrates the machine learning model to predict the stock value with more accuracy as compared to previously implemented machine learning models.

## FUTURE WORK

Future scope of this project will involve adding more parameters and factors like the financial ratios, multiple instances, etc. The more the parameters are taken into account more will be the accuracy. The algorithms can also be applied for analyzing the contents of public comments and thus determine patterns/relationships between the customer and the corporate employ

# 

# 

# REFERECES

1. [Survey of stock market prediction using machine learning approach](https://ieeexplore.ieee.org/document/8212715/), [Ashish](https://ieeexplore.ieee.org/author/37085612540) [Sharma](https://ieeexplore.ieee.org/author/37085612540) ; [Dinesh Bhuriya](https://ieeexplore.ieee.org/author/37085626150) ; [Upendra Singh](https://ieeexplore.ieee.org/author/37085620610) [2017 International conference of](https://ieeexplore.ieee.org/xpl/conhome/8168799/proceeding) [Electronics, Communication and Aerospace Technology (ICECA)](https://ieeexplore.ieee.org/xpl/conhome/8168799/proceeding)
2. [Short-term prediction for opening price of stock market based on](https://ieeexplore.ieee.org/document/8342901/) [self-adapting](https://ieeexplore.ieee.org/document/8342901/) [variant PSO-Elman neural network](https://ieeexplore.ieee.org/document/8342901/),[Ze Zhang](https://ieeexplore.ieee.org/author/37086368722) ; [Yongjun](https://ieeexplore.ieee.org/author/37086369141) [Shen](https://ieeexplore.ieee.org/author/37086369141) ; [Guidong Zhang](https://ieeexplore.ieee.org/author/37086368421) ; [Yongqiang Song](https://ieeexplore.ieee.org/author/37086160390) ; [Yan Zhu](https://ieeexplore.ieee.org/author/37086368754), [2017 8th IEEE](https://ieeexplore.ieee.org/xpl/conhome/8336674/proceeding) [International Conference on Software](https://ieeexplore.ieee.org/xpl/conhome/8336674/proceeding) [Engineering and Service Science (ICSESS)](https://ieeexplore.ieee.org/xpl/conhome/8336674/proceeding)
3. [Combining of random forest estimates using LSboost for stock market](https://ieeexplore.ieee.org/document/8226316/) [index](https://ieeexplore.ieee.org/document/8226316/) [prediction](https://ieeexplore.ieee.org/document/8226316/),[Nonita Sharma](https://ieeexplore.ieee.org/author/37086311926) ; [Akanksha Juneja,2017 2nd](https://ieeexplore.ieee.org/author/37085755928) [International Conference for](https://ieeexplore.ieee.org/xpl/conhome/8168943/proceeding) [Convergence in Technology (I2CT)](https://ieeexplore.ieee.org/xpl/conhome/8168943/proceeding)
4. [Using social media mining technology to assist in price prediction of](https://ieeexplore.ieee.org/document/7509794/) [stock](https://ieeexplore.ieee.org/document/7509794/) [market,Yaojun Wang](https://ieeexplore.ieee.org/document/7509794/) ; [Yaoqing Wang](https://ieeexplore.ieee.org/author/37085805878),[2016 IEEE International Conference on Big](https://ieeexplore.ieee.org/xpl/conhome/7504794/proceeding) [Data Analysis (ICBDA)](https://ieeexplore.ieee.org/xpl/conhome/7504794/proceeding)
5. [Stock market prediction using an improved training algorithm of](https://ieeexplore.ieee.org/document/7879611/) [neural](https://ieeexplore.ieee.org/document/7879611/) [network,Mustain Billah](https://ieeexplore.ieee.org/document/7879611/) ; [Sajjad Waheed](https://ieeexplore.ieee.org/author/37085398742) ; [Abu Hanifa,2016 2nd](https://ieeexplore.ieee.org/author/37085998360) [International](https://ieeexplore.ieee.org/xpl/conhome/7871259/proceeding) [Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)](https://ieeexplore.ieee.org/xpl/conhome/7871259/proceeding)
6. [Efficacy of News Sentiment for Stock Market Prediction,Sneh](https://ieeexplore.ieee.org/document/8862265/) [Kalra](https://ieeexplore.ieee.org/author/37087039958) ; [Jay](https://ieeexplore.ieee.org/author/37085889190) [Shankar Prasad](https://ieeexplore.ieee.org/author/37085889190),[2019 International Conference on Machine](https://ieeexplore.ieee.org/xpl/conhome/8851231/proceeding) [Learning, Big Data,](https://ieeexplore.ieee.org/xpl/conhome/8851231/proceeding) [Cloud and Parallel Computing (COMITCon)](https://ieeexplore.ieee.org/xpl/conhome/8851231/proceeding)
7. [Literature review on Artificial Neural Networks Techniques](https://ieeexplore.ieee.org/document/8780437/) [Application for Stock](https://ieeexplore.ieee.org/document/8780437/) [Market Prediction and as Decision Support Tools](https://ieeexplore.ieee.org/document/8780437/),[Muhammad Firdaus](https://ieeexplore.ieee.org/author/37086291439) ; [Swelandiah](https://ieeexplore.ieee.org/author/37086928553) [Endah Pratiwi](https://ieeexplore.ieee.org/author/37086928553) ; [Dionysia Kowanda](https://ieeexplore.ieee.org/author/37086931330) ; [Anacostia Kowanda](https://ieeexplore.ieee.org/author/37086264111)

8.[Stock Market Movement Prediction using LDA-Online Learning](https://ieeexplore.ieee.org/document/8441038/) [Model](https://ieeexplore.ieee.org/document/8441038/),[Tanapon Tantisripreecha](https://ieeexplore.ieee.org/author/37086441707) ; [Nuanwan Soonthomphisaj,2018 19th](https://ieeexplore.ieee.org/author/37329185400) [IEEE/ACIS](https://ieeexplore.ieee.org/xpl/conhome/8422066/proceeding) [International Conference on Software Engineering, Artificial Intelligence, Networking](https://ieeexplore.ieee.org/xpl/conhome/8422066/proceeding) [and Parallel/Distributed Computing (SNPD)](https://ieeexplore.ieee.org/xpl/conhome/8422066/proceeding)

9.[Stock Market Prediction Analysis by](https://ieeexplore.ieee.org/document/8637365/) [Incorporating Social and News Opinion and Sentiment](https://ieeexplore.ieee.org/document/8637365/), [Zhaoxia Wang](https://ieeexplore.ieee.org/author/38184055800) ; [Seng-Beng](https://ieeexplore.ieee.org/author/37086021916) [Ho](https://ieeexplore.ieee.org/author/37086021916) ; [Zhiping Lin 2018 IEEE International Conference on Data Mining Workshops](https://ieeexplore.ieee.org/author/37277999300) [(ICDMW)](https://ieeexplore.ieee.org/xpl/conhome/8626049/proceeding)

10.[Stock Price Prediction Using News Sentiment Analysis](https://ieeexplore.ieee.org/document/8848203/) [Saloni](https://ieeexplore.ieee.org/author/37087015405) [Mohan](https://ieeexplore.ieee.org/author/37087015405) ; [Sahitya](https://ieeexplore.ieee.org/author/37087014008) [Mullapudi](https://ieeexplore.ieee.org/author/37087014008) ; [Sudheer Sammeta](https://ieeexplore.ieee.org/author/37087015796) ; [Parag, Vijayvergia](https://ieeexplore.ieee.org/author/37087014658) ; [David C. Anastasiu,2019 IEEE](https://ieeexplore.ieee.org/author/37085540985) [Fifth International Conference on Big Data Computing Service and Applications](https://ieeexplore.ieee.org/xpl/conhome/8843845/proceeding) [(BigDataService)](https://ieeexplore.ieee.org/xpl/conhome/8843845/proceeding)