MINI PROJECT REPORT

Stock Price Prediction

JULY 14, 2022

STOCK PRICE PREDICTION

A Project report submitted in partial fulfillment of the requirements for Bachelor Of Technology CSE 4th Semester Mini Project Work.

Submitted By

Kartikay
GE – 202016809

Under Mentorship and Guidance of

Mr. Kireet Joshi.

GraphicEra
Transforming dreams into reality

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GRAPHIC ERA DEEMED TO BE UNIVERSITY

CERITFICATE

This is to certify that project entitled "STOCK PRICE PREDICTION" submitted by Kartikay (2016809) in partial fulfillment of the requirements for 4th Semester Mini Project, of Graphic Era Deemed To Be University is a record of bonafide work carried out under my guidance and supervision.

Project Guide:

Mr. Kireet Joshi
Department of CSE
Graphic Era Deemed University
Dehradun

ACKNOWLEDGEMENT

I would like to express my profound and deep gratitude to my project guide Mr. Kireet Joshi for his guidance with unsurpassed knowledge and immense encouragement. I'm fortunate to get a chance to complete project under his exemplary guidance.

I would also like to express my heartfelt gratitude to Mrs. Garima Sharma and Mr. Guru Prasad who make my concepts of AI and ML crystal clear and imparted valuable knowledge regarding this field.

The success and final outcome of this project required lot of guidance and support from many people including my parents, teachers, classmates and many others whose names may not all be enumerated, and I am privileged to have got this all along the completion of this project.

ABSTRACT

In this project I attempt to implement machine learning model in Stock Price analysis and use it to predict stack prices. Machine learning (ML) is type of artificial intelligence that help software application to be more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

This project attempts to explore and deep dive into the predictive analysis field of machine learning. Prediction refers to the output of an algorithm after is has been trained on a historical dataset and applied to new data when forecasting the likelihood of a particular outcome. The algorithm will generate probable values for an unknown variable for each record in the new data, allowing the model builder to identify what that value will most likely be.

The objective is to predict the stock prices in order to make more informed and accurate investment decisions. I propose a stock price prediction system that integrates mathematical function, machine learning, and other external factors for purpose of achieving better stock prediction accuracy and issuing profitable trades.

There are two types of stocks. You may know of intraday trading by the commonly used term "day trading." Interday traders hold securities positions from at least one day to the next and often for several days to weeks or months. LSTMs are very powerful in sequence prediction problems because they're able to store past information. This is important in our case because the previous price of a stock is crucial in predicting its future price. While predicting the actual price of a stock is an uphill climb, we can build a model that will predict whether the price will go up or down.

INTRODUCTION

Motivation:

When you hear that 70% of trading volume in entire US stock market is generated by algorithms, you might think you are missing out something big. Are we the only fools who still trade the traditional way? If the machines dominate the market today, do us mere mortals even stand a chance against the might machines? Well, a big chunk of that automated 72% is a result of high-frequency trading algorithms trying to predict only milliseconds into the future. And those algorithms are usually very simple methods, nothing fancier than a chain of hardcoded rules or a simple linear regression model. So yes, if you race against milliseconds the you do need to be a machine.

But for casual investors like us this shouldn't really matter. Or should it? Machine learning models can learn much more complex patterns in data. Is it possible to predict longer-term price movements in the market using machine learning? Nobody knows for sure. Many large financial institutions are hiring data scientists, machine learning engineers, and deep learning experts with hefty salaries. So does that means it works for sure? Well, that may show us a sign about the trends in investing strategies, but institutional investing can be different from individual investing. For example, unless you are buying a very-low-volume stock, the shares you buy or sell barely have any impact on the price. But if you are buying and selling in large amounts, how you execute your trades can make a big difference. A machine learning model can help decide how you should split up your sales over time to avoid causing big price movements. From perspective of a individual trader a machine learning model can help a trader to predict the future values of a stock and help to analyze the stock market in better way.

Problem Statement:

To actually predict the price movements, you can try a lot of things, from very simple things such as training LSTMs or Temporal Convolutional Networks on historical data to overly complex models such as training Convolutional LSTMs on satellite imagery to predict macroeconomic movements. Time Series analysis plays an important role in data analysis. It is specialized branch of statistics used extensively in fields such as Econometrics and Operation Research. The main aim of this project is to predict stock prices using Long Short Term Memory(LSTM).

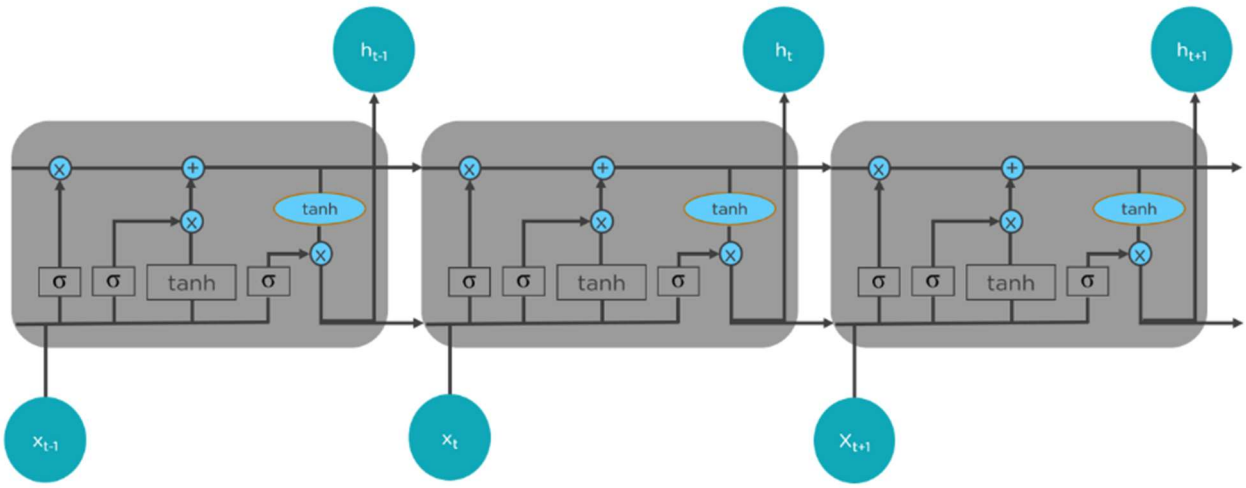
METHODOLOGY

Stock Price Prediction:

Stock Price Prediction using machine learning helps you discover the future value of company stock and other financial assets traded on an exchange. The entire idea of predicting stock prices is to gain significant profits. Predicting how the stock market will perform is a hard task to do. There are other factors involved in the prediction, such as physical and psychological factors, rational and irrational behavior, and so on. All these factors combine to make share prices dynamic and volatile. This makes it very difficult to predict stock prices with high accuracy.

Understanding Long Short Term Memory (LSTM)

LTSMs are a type of Recurrent Neural Network for learning long-term dependencies. It is commonly used for processing and predicting time-series data. In this project we will be using Long Short Term Memory to predict future values of stocks of a company.



From the image on the top, you can see LSTMs have a chain-like structure. General RNNs have a single neural network layer. LSTMs, on the other hand, have four interacting layers communicating extraordinarily.

LSTM works in three steps**:**

- The first step in LSTM is to decide which information to be omitted from the cell in that particular time step. It is decided with the help of a sigmoid function. It looks at the previous state (ht-1) and the current input xt and computes the function.
- There are two functions in the second layer. The first is the sigmoid function, and the second is the tanh function. The sigmoid function decides which values to let through (0 or 1). The tanh function gives the weightage to the values passed, deciding their level of importance from -1 to 1.
- The third step is to decide what will be the final output. First, you need to run a sigmoid layer which determines what parts of the cell state make it to the output. Then, you must put the cell state through the tanh function to push the values between -1 and 1 and multiply it by the output of the sigmoid gate.

With this basic understanding of LSTM, you can dive into the hands-on demonstration part of this tutorial regarding stock price prediction using machine learning.

Steps of Prediction:

1. Import the Libraries

For our objective we need to import various libraries:
We need more libraries which we can import afterwards.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pandas_datareader as data
!pip install --upgrade pandas-datareader
```

2. Import and load data in DataFrame

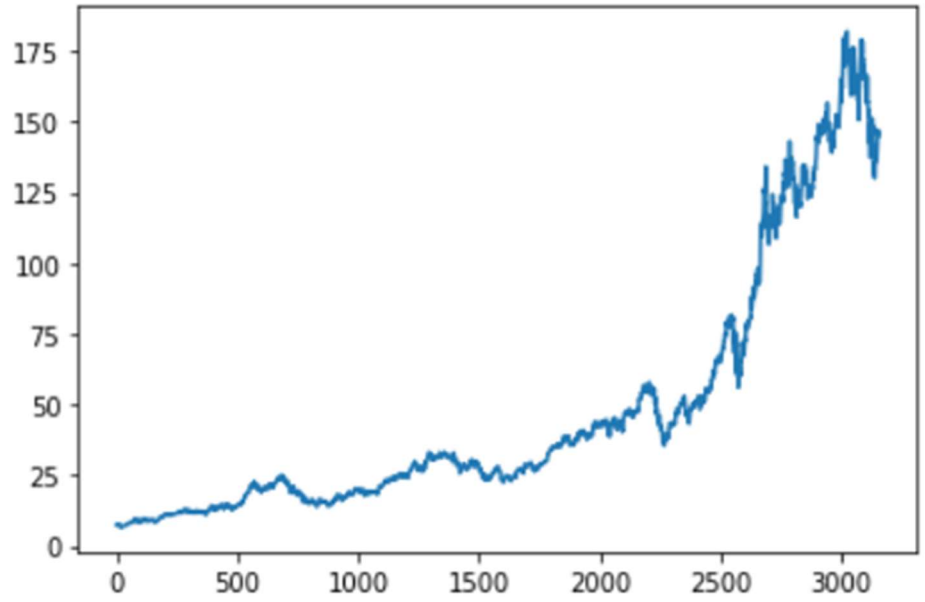
```
# We should take bigger timeframe for our model to be accurate
start = '2010-01-01'
end = '2022-07-12'

# User can take any stock picker
df = data.DataReader('AAPL','yahoo',start,end)
df.head()
```

Imported data look like this:

Date	High	Low	Open	Close	Volume	Adj Close
2009-12-31	7.619643	7.520000	7.611786	7.526071	352410800.0	6.434927
2010-01-04	7.660714	7.585000	7.622500	7.643214	493729600.0	6.535085
2010-01-05	7.699643	7.616071	7.664286	7.656429	601904800.0	6.546385
2010-01-06	7.686786	7.526786	7.656429	7.534643	552160000.0	6.442256
2010-01-07	7.571429	7.466071	7.562500	7.520714	477131200.0	6.430345

Plotting Data:



3. Calculating 100 and 200 days moving average and plotting both on graph.

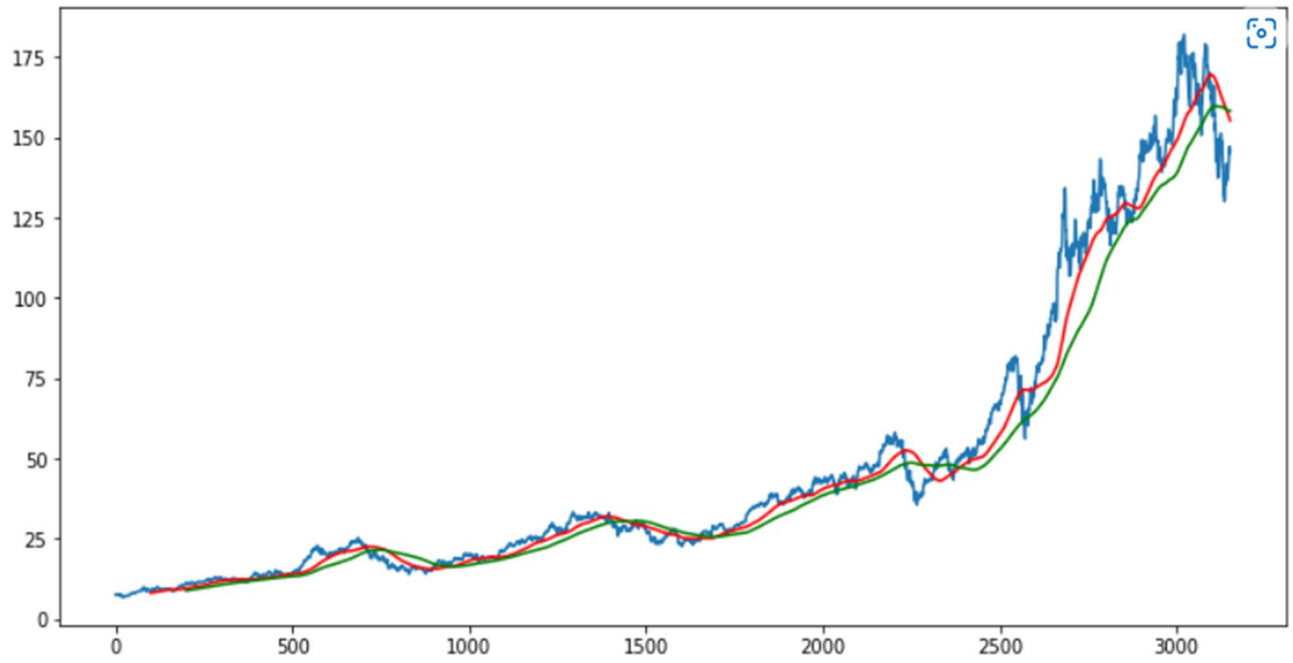
Brokers and analysts uses various charts and techniques to analyze market trends, but here we will be limiting ourselves to only 100 and 200 day moving average for analysis purpose, which is actually a pretty good tool for analysis purpose.

```
# If 100 days moving average if above 200 days moving average it means stock prices are inflating.
# 100 days moving average
# Moving average for first 100 days is not defined as we need at least 100
# days data to get 100 day moving average
ma100 = df.Close.rolling(100).mean()
ma100
```

```
ma200 = df.Close.rolling(200).mean()
ma200
```

```
# Plotting 200 days moving average graph
plt.figure(figsize = (12,6))
plt.plot(df.Close)
plt.plot(ma100,'r')
plt.plot(ma200,'g')
```

In graph below red line represents 100 day moving average and green line represents 200 day moving average.



4. Training Model

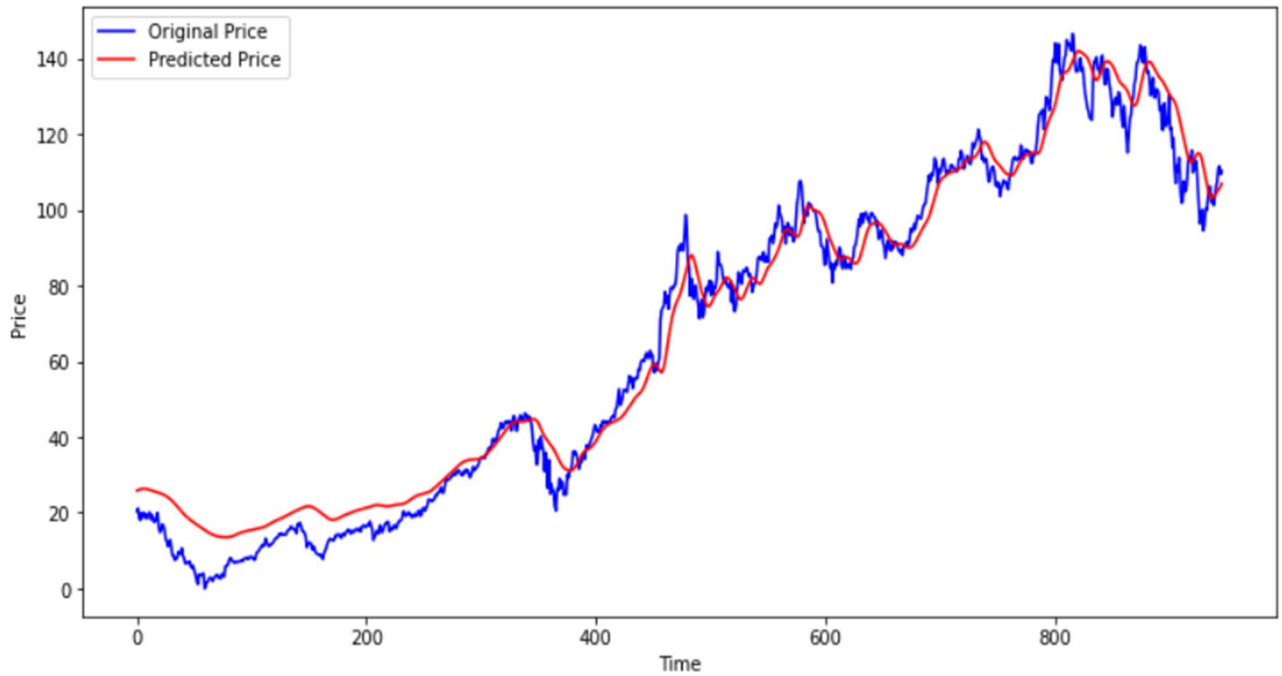
We will train our model on 70% of data scraped by us and will test the model on 30% of data.

```
# Training Model
model.compile(optimizer='adam', loss = 'mean_squared_error')
model.fit(x_train,y_train, epochs = 50)
```

```
Epoch 1/50
66/66 [==============================] - 11s 127ms/step - loss: 0.0264
Epoch 2/50
66/66 [==============================] - 8s 121ms/step - loss: 0.0066
Epoch 3/50
66/66 [==============================] - 8s 121ms/step - loss: 0.0061
Epoch 4/50
66/66 [==============================] - 8s 119ms/step - loss: 0.0047
Epoch 5/50
66/66 [==============================] - 8s 119ms/step - loss: 0.0043
Epoch 6/50
```

5. Making Prediction

After completion of step 4 our model is trained and we can use it to make predictions.
We plot a graph to show accuracy of a model by plotting the graph of predicted values and actual values of same graph.



From the graph we can see that our model is working fine and is almost accurately predicting the stock prices.

6. **Finally after we have made a complete model we can deploy it as Web App using Streamlit**.

Streamlit is an open source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib etc.

CONCLUSION AND FUTURE WORK

In this project I am predicting closing stock price of any given organization and developed a web app for the same using Streamlit and LSTM algorithm.

In future I want to integrate this model with sentiment analysis model which I have already develop, to make more accurate prediction about this stock prices by analyzing sentiments of public about a particular organization. The source code for sentiment analysis model can be accessed by this link.