Mini Project Report on

Machine learning model for stock value prediction

Submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted by:

Student Name :	University Roll No.:
Udit Aggarwal	2019191

Under the Mentorship of

Dr. Sharon Christa

Associate professor

Department of Computer Science and Engineering

Graphic Era (Deemed to be University)

Dehradun, Uttarakhand

January 2023

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the project report entitled **"Machine learning model for stock value prediction"** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering of the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Dr. Sharon Christa, Associate Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Name :	University Roll no. :
Udit Aggarwal	2019191

Table of Contents

Chapter 1

Introduction

The stock market is a financial marketplace where publicly traded companies' stocks are bought and sold. The stock market is often divided into two main sections: the primary market, where new issues are first sold to investors, and the secondary market, where existing securities are traded among investors. The prices of stocks on the stock market are determined by supply and demand, with the stock prices rising when demand is high and falling when demand is low. The stock market is often used as an indicator of the overall health of an economy.

1.1 Proposed work

Stock value prediction using machine learning refers to the use of machine learning algorithms to analyze historical stock market data and make predictions about future stock prices. Machine learning models can be trained on historical data, such as stock prices, trading volumes, and news articles, to identify patterns and trends that can be used to make predictions about future stock prices.

There are several types of machine learning algorithms that can be used for stock value prediction, including linear regression, decision trees, random forests, and neural networks. Each algorithm has its own strengths and weaknesses and can be used for different types of prediction tasks.

Linear regression, for example, is a simple algorithm that can be used to predict a continuous value, such as stock price. It works by fitting a line through a set of data points and using the line to make predictions about future values.

Decision trees and random forests are more complex algorithms that can be used for both classification and regression tasks. They work by building a tree-like model of decisions and outcomes, and can be used to make predictions based on multiple input variables.

Neural networks are a type of machine learning algorithm that are modeled after the structure and function of the human brain. They are particularly useful for tasks that involve pattern recognition, and can be used to make predictions based on large and complex sets of data.

It's important to note that stock value prediction using machine learning models is not an exact science and it's not guaranteed to be accurate all the time. Models may fail to predict certain scenarios, or they may not be able to take into account all the factors that can affect stock prices.

Moreover, stock market is a complex and dynamic system which is affected by various factors such as economic conditions, company's financial performance, political and social events, etc. So It's important to consider that predictions made by machine learning models should be used in conjunction with other forms of analysis and research.

1.2 Scope of the work

The scope of stock value prediction using machine learning is quite broad, as it can be applied to a wide range of financial markets and industries. Some of the main applications of stock value prediction using machine learning include:

1. Stock trading: Machine learning models can be used to predict future stock prices and help traders make more informed decisions about when to buy and sell stocks.

2. Portfolio management: Machine learning models can be used to analyze the performance of different stocks and make recommendations for creating and managing investment portfolios.

3. Risk management: Machine learning models can be used to identify and quantify risks associated with different stocks and to develop strategies for managing those risks.

4. Algorithmic trading: Machine learning models can be used to develop algorithms for automating the buying and selling of stocks.

5. Market analysis: Machine learning models can be used to analyze market trends and identify patterns that can be used to make predictions about future stock prices.

6. Sentiment Analysis: Machine learning models can be used to analyze news articles, social media posts, and other forms of text data to understand the overall sentiment about a particular stock and its potential impact on the stock price.

However, it's important to keep in mind that stock value prediction using machine learning is a complex and evolving field, and there are still many challenges that need to be overcome. For example, predicting stock prices is a difficult task due to the complexity and uncertainty of the stock market, and machine learning models may not always be able to take into account all the factors that can affect stock prices.

1.3 Problem Statement

Time series forecasting and modelling plays an important role in data analysis. Time series analysis is a specialized branch of statistics used extensively in fields such as Econometrics & Operation Research. Time series is being widely used in analytics & data science. Stock prices are volatile in nature and price depends on various factors. The main aim of this project is to predict stock prices using Long short term memory (LSTM).

Chapter 2

Literature Survey

2.1 Introduction

"Others' thought process" has forever been a significant piece of data for the greater part of us during the dynamic cycle. The Web and the Internet have now (in addition to other things) made it conceivable to learn about the conclusions and encounters of those in the tremendous pool of individuals that are neither our own colleagues nor notable expert pundits — that is, individuals we have never known about. What's more, on the other hand, an ever increasing number of individuals are making their perspectives accessible to outsiders through the Web. The interest that singular clients show in web-based conclusions about items and administrations, and the potential impact such sentiments employ, is something main impetus for this area of interest. Furthermore, there are many difficulties engaged with this cycle which needs to be trampled to accomplish legitimate results out of them. In this overview we broke down fundamental strategy that typically occurs in this cycle and measures that are to be taken to defeat the difficulties being confronted.

2.2 Existing methods

2.2.1 Stock Market Prediction Using Machine Learning

The examination work done by V Kranthi Sai Reddy Understudy, ECM, Sreenidhi Establishment of Science and Innovation, Hyderabad, India. In the money world stock exchanging is perhaps of the main movement. Financial exchange forecast is a demonstration of attempting to decide the future worth of a stock other monetary instrument exchanged on a monetary trade. This paper makes sense of the forecast of a stock utilizing AI. The specialized and essential or the time series investigation is utilized by the a large portion of the stockbrokers while making the stock forecasts. The programming language is utilized to anticipate the securities exchange utilizing AI is Python. In this paper we propose an AI (ML) move toward that will be prepared from the accessible stocks information and gain insight and afterward involves the obtained information for an exact expectation. In this setting this study utilizes an AI strategy called Help Vector Machine (SVM) to anticipate stock costs for the huge and little capitalizations and in the three unique business sectors, utilizing costs with both everyday and expert frequencies.

2.2.2 Forecasting the Stock Market Index Using Artificial Intelligence Techniques

The examination work done by Lufuno Ronald Marwala A paper submitted to the Staff of Designing and the Constructed Climate, College of the Witwatersrand, Johannesburg, in satisfaction of the prerequisites for the level of Expert of Science in Designing. The feeble type of Proficient Market speculation (EMH) states that it is difficult to conjecture the future cost of a resource in light of the data contained in the verifiable costs of a resource. This implies that the market acts as an irregular walk and subsequently makes determining unthinkable. Moreover, monetary determining is a troublesome undertaking because of the inborn intricacy of the monetary framework. The goal of this work was to utilize man-made consciousness (simulated intelligence) methods to display and foresee the future cost of a financial exchange file. Three man-made consciousness strategies, specifically, brain organizations (NN), support vector machines and neuro-fluffy frameworks are carried out in guaging the future cost of a financial exchange record in light of its verifiable cost data. Man-made brainpower procedures can think about monetary framework intricacies and they are utilized as monetary time series determining apparatuses.

Two procedures are utilized to benchmark the artificial intelligence methods, in particular, Autoregressive Moving Normal (ARMA) which is direct displaying strategy and irregular walk (RW) strategy. The trial and error was performed on information acquired from the Johannesburg Stock Trade. The information utilized was a progression of past shutting costs of the All Offer File. The outcomes showed that the three strategies can foresee the future cost of the Record with a satisfactory exactness. Each of the three computerized reasoning methods beat the straight model. Nonetheless, the irregular walk strategy out played out the wide range of various methods. These strategies show a capacity to foresee the future cost in any case, due to the exchange expenses of exchanging the market, it is preposterous to expect to demonstrate the way that the three methods can negate the frail type of market effectiveness. The outcomes show that the positioning of exhibitions support vector machines, neuro-fluffy frameworks, multi-facet perceptron brain networks is reliant upon the exactness measure utilized.

2.2.3 Automated Stock Price Prediction Using Machine Learning

The examination work done by Mariam Moukalled Wassim El-Hajj Mohamad Jaber Software engineering Division American College of Beirut. Generally and to anticipate market development, financial backers used to dissect the stock costs and stock pointers notwithstanding the news connected with these stocks. Thus, the significance of information on the stock cost development. The greater part of the past work in this industry zeroed in on either arranging the delivered market news as (positive, negative, unbiased) and showing their impact on the stock cost or zeroed in on the authentic cost development and anticipated their future development. In this work, we propose a mechanized exchanging framework that coordinates numerical capabilities, AI, and other outside elements like news' feelings to accomplish better stock expectation exactness and giving beneficial exchanges. Especially, we expect to decide the cost or the pattern of a specific stock for the approaching finish of-day thinking about the initial a few exchanging hours of the day. To accomplish this objective, we prepared conventional AI calculations and made/prepared numerous profound learning models thinking about the significance of the applicable news. Different examinations were led, the most elevated precision (82.91%) of which was accomplished involving SVM for Apple Inc. (AAPL) stock.

2.2.4 An Intelligent Technique for Stock Market Prediction

The exploration work done by M. Mekayel Anik · M. Shamsul Arefin (B) Division of Software engineering and Designing, Chittagong College of Designing and Innovation, Chittagong, Bangladesh. A financial exchange is a free organization of monetary exchanges among purchasers and dealers in view of stocks otherwise called shares. In financial exchanges, stocks address the possession claims on organizations. These may incorporate protections recorded on a stock trade as well as those just exchanged secretly. A stock trade is where dealers can purchase as well as sell stocks, bonds, and different protections. Financial exchange is an entirely weak spot for speculation because of its unstable nature. In the close to past, we dealt with colossal monetary issues because of gigantic drop in cost of offers in financial exchanges around the world. This peculiarity welcomed a weighty cost for the worldwide as well as on our public monetary design. Many individuals lost their keep going reserve funds of cash on the financial exchange. In 2010-2011 monetary year, Bangladeshi securities exchange confronted enormous breakdown [1]. This peculiarity can be managed particularly by severe checking and case financial exchange examination. In the event that we can break down securities exchange accurately in time, it can turn into a field of huge benefit and may turn out to be similarly less defenseless for the financial backers. Securities exchange is about forecast and quick dynamic about venture, which isn't possible without exhaustive examination of the market. In the event that we can anticipate the securities exchange by breaking down authentic information appropriately, we can stay away from the outcomes of serious market breakdown and to have the option to do whatever it may take to make market safe to such circumstances

2.2.5 Event Representation Learning Enhanced with External Common-sense Knowledge

The research work done by Xiao Ding, Kuo Liao, Ting Liu, Zhongyang Li, Junwen Duan Research Center for Social Computing and Information Retrieval Harbin Institute of Technology, China. Prior work has proposed effective methods to learn event representations that can capture syntactic and semantic information over text corpus, demonstrating their effectiveness for downstream tasks such as script event prediction. On the other hand, events extracted from raw texts lacks of common-sense knowledge, such as the intents and emotions of the event participants, which are useful for distinguishing event pairs when there are only subtle differences in their surface realizations. To address this issue, this paper proposes to leverage external common-sense knowledge about the intent and sentiment of the event. Experiments on three event-related tasks, i.e., event similarity, script event prediction and stock market prediction, show that our model obtains much better event embeddings for the tasks, achieving 78% improvements on hard similarity task, yielding more precise inferences on subsequent events under given contexts, and better accuracies in predicting the volatilities of the stock market.

Chapter 3

Methodology

3.1 Proposed System

The prediction methods can be roughly divided into two categories, statistical methods and artificial intelligence methods. Statistical methods include logistic regression model, ARCH model, etc. Artificial intelligence methods include multi-layer perceptron, convolutional neural network, naive Bayes network, back propagation network, single-layer LSTM, support vector machine, recurrent neural network, etc. I have used Long short-term memory network (LSTM).

3.1.1 Long short-term memory network:

LSTM (Long Short-Term Memory) is a type of Recurrent Neural Network (RNN) architecture that is designed to handle sequential data with long-term dependencies. The basic structure of an LSTM cell is composed of a series of gates, including an input gate, an output gate, and a forget gate, which are used to control the flow of information into and out of the cell.

The input gate controls the amount of information that is allowed to flow into the cell state, the output gate controls the amount of information that is allowed to flow out of the cell state, and the forget gate controls the amount of information that is discarded from the cell state.

Here is a simple diagram that illustrates the main components of an LSTM cell:

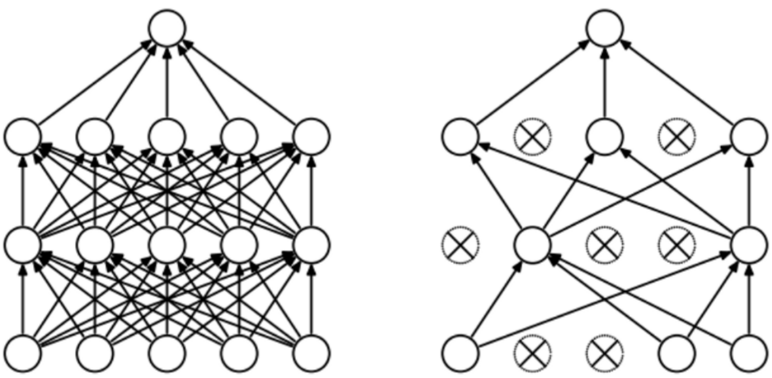

Fig. 3.1 LSTM cell diagram

In this diagram, the arrows represent the flow of information, and the circles represent the gates. The cell state is represented by the horizontal line in the center, and the input and output are represented by the arrows coming into and out of the top of the cell.

3.2 Methodology

We have imported various necessary libraries such as Numpy, Pandas, Matplotlib, and Keras to build our stock price prediction model using the LSTM. We have then read a CSV file containing the training data for Samsung's stock prices, and displayed the first few rows of the data.

```
In [5]: dataset=pd.read_csv('/Users/uditaggarwal/Downloads/Samsung_train_data.csv')
        dataset.head()
```

Out[5]:

	Date	Open	High	Low	Close	Adj Close	Volume
0	2012-01-03	21860.0	22100.0	21840.0	22100.0	17675.148438	16927750
1	2012-01-04	22100.0	22200.0	21500.0	21600.0	17275.257813	17103700
2	2012-01-05	21460.0	21580.0	21100.0	21100.0	16875.367188	17298400
3	2012-01-06	21120.0	21320.0	20600.0	20800.0	16635.439453	18816250
4	2012-01-09	20800.0	20820.0	20300.0	20320.0	16251.541016	19283000

Fig. 3.2 Read Dataset

Now, we have performed data analysis for stock price of Samsung. Fig. 3.3 represents the date, open, close, high, low, adjusted cose and volume of stock details.

```
In [6]: dataset.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1234 entries, 0 to 1233
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Date       1234 non-null   object
 1   Open       1234 non-null   float64
 2   High       1234 non-null   float64
 3   Low        1234 non-null   float64
 4   Close      1234 non-null   float64
 5   Adj Close  1234 non-null   float64
 6   Volume     1234 non-null   int64
dtypes: float64(5), int64(1), object(1)
memory usage: 67.6+ KB
```

Fig. 3.3 Stock dataset information

We will now clean the dataset by dropping any rows with missing values, and the closing process of the stocks are extracted and stored in the variable 'trainset'.

```
In [10]: dataset=dataset.dropna()
         trainset=dataset.iloc[:,4:5].values

In [11]: dataset.info()

         <class 'pandas.core.frame.DataFrame'>
         RangeIndex: 1234 entries, 0 to 1233
         Data columns (total 7 columns):
          #   Column     Non-Null Count  Dtype
         ---  ------     --------------  -----
          0   Date       1234 non-null   object
          1   Open       1234 non-null   float64
          2   High       1234 non-null   float64
          3   Low        1234 non-null   float64
          4   Close      1234 non-null   float64
          5   Adj Close  1234 non-null   float64
          6   Volume     1234 non-null   int64
         dtypes: float64(5), int64(1), object(1)
         memory usage: 67.6+ KB

In [12]: scale=MinMaxScaler(feature_range=(0,1))
         trainset=scale.fit_transform(trainset)
         trainset.shape

Out[12]: (1234, 1)
```

Fig. 3.4 Cleaned Data set

Now we will use a for loop to create a list of 60 day windows of the closing prices, and the corresponding 61st day closing price. These lists are then converted to numpy arrays, and the window array is reshaped to have a 3D structure, with the shape(number of windows, 60 time steps, 1 feature)

```
In [13]: W_train=[]
         Y_train=[]
         for i in range(60,1234):
             W_train.append(trainset[i-60:i,0])
             Y_train.append(trainset[i,0])

         W_train,Y_train=np.array(W_train),np.array(Y_train)

In [14]: W_train=np.reshape(W_train,(W_train.shape[0],W_train.shape[1],1))
         W_train.shape

Out[14]: (1174, 60, 1)
```

Fig. 3.5

A sequential model is then created using the Keras library, and several LSTM layers, dropout layers, and a dense layer are added to the model. The model is then compiled using the Adam optimizer and mean squared error as the loss function.

```
In [15]: technique=Sequential()

         technique.add(LSTM(units=100,return_sequences=True,input_shape=(W_train.shape[1],1)))
         technique.add(Dropout(0.2))

         technique.add(LSTM(units=100,return_sequences=True))
         technique.add(Dropout(0.2))

         technique.add(LSTM(units=100,return_sequences=True))
         technique.add(Dropout(0.2))

         technique.add(LSTM(units=100,return_sequences=False))
         technique.add(Dropout(0.2))

         technique.add(Dense(units=1))
         technique.compile(optimizer='adam',loss="mean_squared_error")

         2023-01-20 09:47:14.794224: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorF
         low binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following
         CPU instructions in performance-critical operations:  SSE4.1 SSE4.2
         To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.

In [16]: datatraining=technique.fit(W_train,Y_train,epochs=20,batch_size=32,verbose=2)
```

Fig. 3.6 Building the model

Now we plot the loss of the model during training. Afterwards we have read the test data, cleaned it and scaled the data, create the windows of 60 days, reshaped it and then make predictions using the previously trained model. Finally, we plot the actual stock prices and the predicted stock prices to visualize the results.

```
In [19]: testset=pd.read_csv('/Users/uditaggarwal/Documents/computer geu data/Samsung_test_data.csv')
         testset=testset.dropna()
         testset=testset.iloc[:,4:5]
         Y_test=testset.iloc[60:,0:].values
         inputclose=testset.iloc[:,0:].values
         inputclose_scaled=scale.transform(inputclose)
         inputclose_scaled.shape
         W_test=[]
         length=len(testset)
         time_step=60
         for i in range(time_step,length):
             W_test.append(inputclose_scaled[i-time_step:i,0])
         W_test=np.array(W_test)
         W_test=np.reshape(W_test,(W_test.shape[0],W_test.shape[1],1))
         W_test.shape

Out[19]: (925, 60, 1)

In [20]: Y_predict=technique.predict(W_test)
         Y_predict
```

Fig. 3.7 Reading the test data

```
ı [22]: plt.plot(Y_test,color='green',label='Actual stock prices')
        plt.plot(predicted_price,color='red',label='Predicted stock price')
        plt.title('Samsung stock price prediction')
        plt.xlabel('Time')
        plt.ylabel('Stock Price')
        plt.legend()
        plt.show()
```

Fig. 3.8 Plotting the predicted values

Chapter 4

Result and Discussion

4.1 Training model loss

We have compiled our model using the Adam optimizer and mean squared error as the loss function. Fig. 4.1 shows the loss of the model during training.

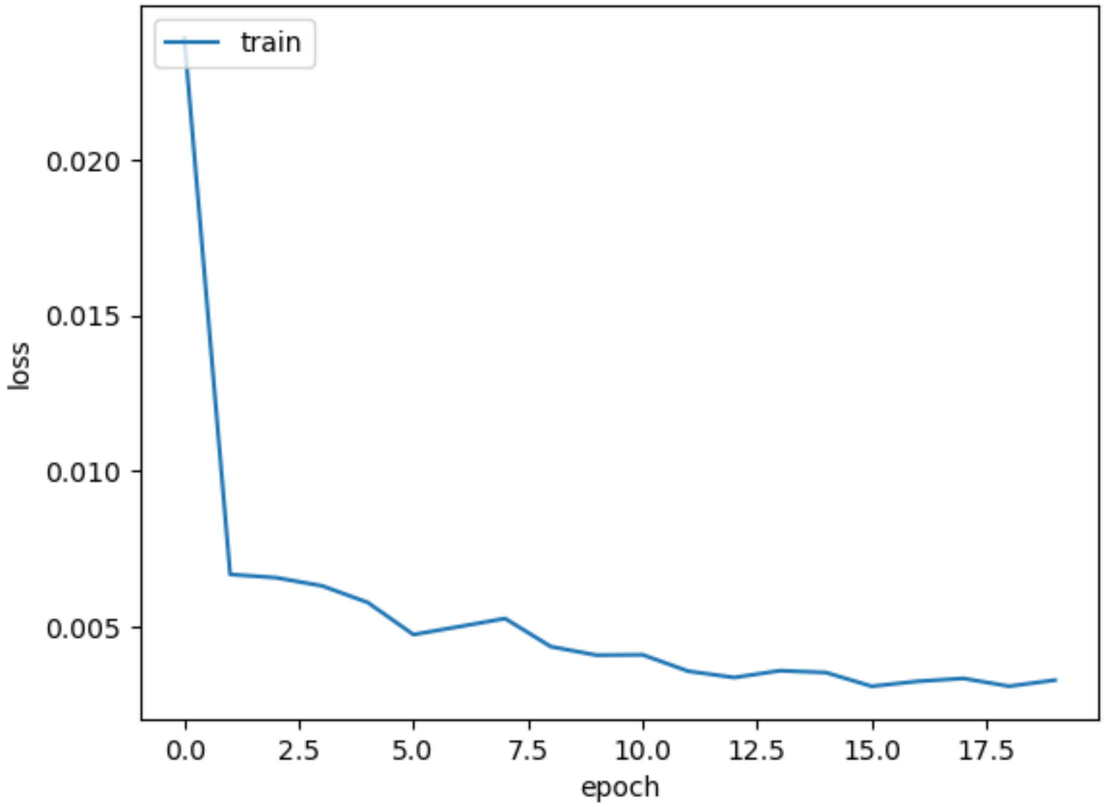

Fig. 4.1 Training model loss

In the above graph the x-axis represents the number of training epochs, and the y-axis represents the training loss. The training loss is a measure of how well the model is able to fit the training data. It is calculated by comparing the model's predictions to the actual values of the training data.

We have trained the model for 20 epochs and the loss is plotted at each epoch. The loss is decreasing as the number of epochs increases which means the model is learning from the training data and is getting better at making predictions.

4.2 Predicted values

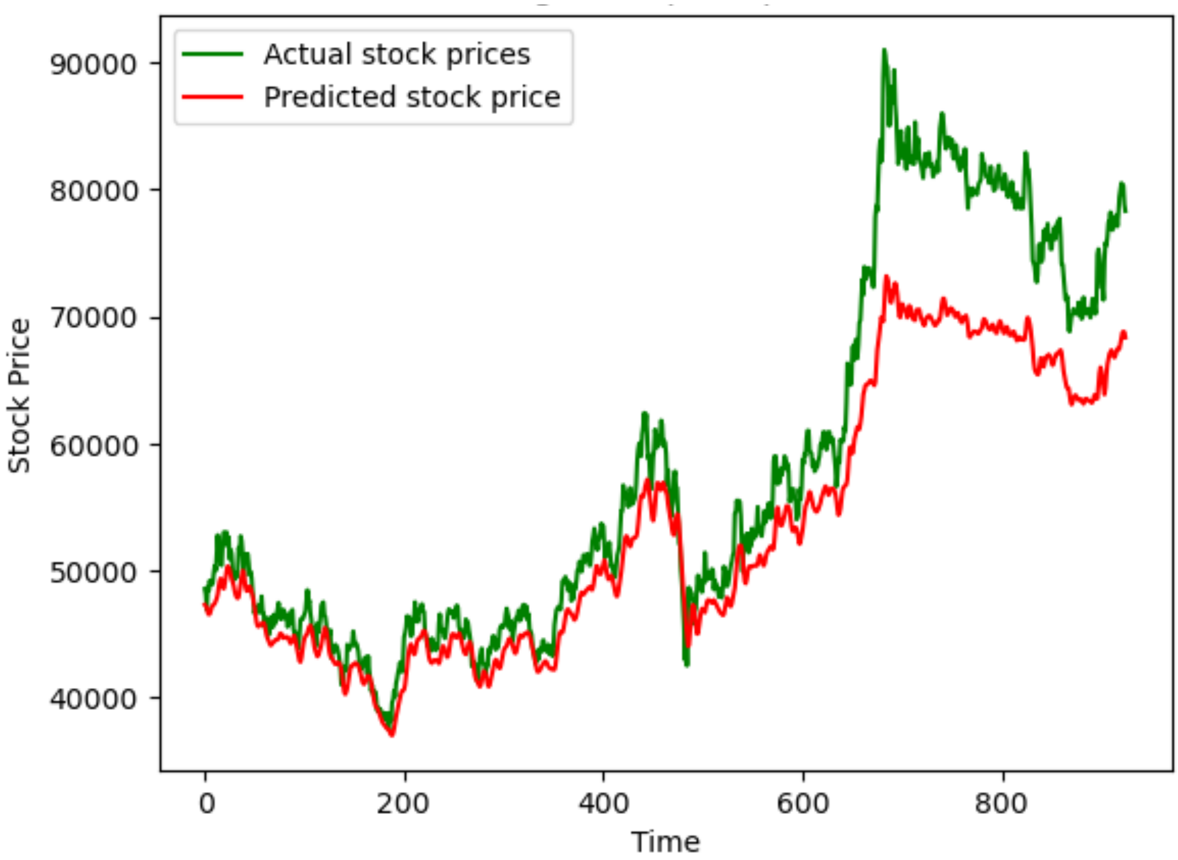

Fig. 4.2 Predicted values

The above graph is a plot of actual stock prices (green line) versus predicted stock prices (red line) for Samsung stock. The x-axis represents time, and the y-axis represents stock price. The plot shows how well the model's predictions align with the actual stock prices over time.

Chapter 5

Conclusion and Future Work

Stock value prediction using machine learning can be a useful tool for forecasting future stock prices. However, it is important to note that the stock market is inherently unpredictable and that no model can accurately predict future stock prices with 100% accuracy. Additionally, stock prices are influenced by a variety of factors, including economic indicators, company performance, and market sentiment, which can be difficult to fully capture in a machine learning model.

It is also important to keep in mind that the performance of the model can be affected by the quality and quantity of the training data. Additionally, the model's performance should be evaluated using appropriate metrics and be compared to a benchmark or other models.

In summary, stock value prediction using machine learning can be a useful tool, but it is important to keep in mind the limitations of the models and the inherent unpredictability of the stock market.

In the future, we can extend this application for predicting cryptocurrency trading and also, we can add sentiment analysis for better predictions.

References