Stock Price Prediction Based on Machine Learning Algorithms (LSTM, CNN, RNN)

Kavya B.G[1], Nishmitha.V[2], Sindhu.H.M[3], Tejashwini.B[4], Prof Kavyashree K.R
[1]Student, Department of CSE, BNM Institute of technology
[2]Student, Department of CSE, BNM Institute of technology
[3]Student, Department of CSE, BNM Institute of technology
[4]Student, Department of CSE, BNM Institute of technology
[5]Assistant professor, Department of CSE, BNM Institute of technology

Abstract: *Human intelligence is currently the dominant trend in stock market. This paper aims to build a model using Recurrent Neural Networks (RNN), Convolutional neural network (CNN) and especially Long-Short Term Memory model (LSTM) to predict future stock market values. The main objective of this paper is to analyze the performance of Machine learning algorithm in predicting stock price and analyze the accuracy obtained by various machine learning algorithms. Using this analysis in stock price prediction the result will be displayed in the form of table and graphical representation.*

Keywords: ***Accuracy, CNN, LSTM, Machine learning,NIFTY50, Prediction, RNN, Stock Market, Time series.***

1. INTRODUCTION

The Stock Market Prediction has been a critical area of research and is one of the trending applications of machine learning. Machine learning helps to discover the future value of company stocks, investment processes, and other assets. Several models provide a large number of methods that can be utilized with machine learning to predict an accurate value. This model uses the data efficiently to train and test the data, to give efficient results considering both weak and strong factors like open and close of the dataset. Machine learning models such as recurrent neural networks, long short-term memory, support vector machine, and convolutional neural network are seen to be used for effective statistics. Our paper aims to use an Machine Learning algorithm based on LSTM, RNN, and CNN to forecast the stock prices. The main objective here is to obtain the most accurate stock price,to predict future values of nifty-fifty data.

2. LITERATURE SURVEY

Divit Karmiani[1] did Several case studies on stock price prediction shows that the choice of algorithms depends upon parameters like time, variance and accuracy. If the requirement is high accuracy and low variance LSTM would be better choice compared to Backpropagation and SVM.

Ashwini Kanade [2] built a framework using Long Short Term Memory machine learning algorithm and adaptive stock technical indicators for efficient forecasting by using various parameters obtained from the historical data set considered for a particular company. This algorithm works on historical data retrieved from Yahoo Finance. For prediction of share price using Long Short-Term Memory, the results will attempt to predict whether a stock price in the future will be higher or lower than it is on a given day to increase transparency amonginvestors in the market.

Xianghui Yuan [3] construct a better-integrated stock selection model based on different feature selection and nonlinear stock price trend prediction methods. In this paper, the features are selected by various feature selection algorithms, and the parameters of the machine learning-based stock price trend prediction models are set through time-sliding window cross- validation based on 8-year data of Chinese A-share market. Through the analysis of different integrated models, the model performs best when the random forest algorithm is used for both feature selection and stock price trend prediction. Based on the random forest algorithm, a long-short portfolio is constructed to validate the effectiveness of the best model.

Naman Arora [4] have demonstrated a machine learning approach (deep learning) to predict stock market trend using different neural networks. Results show how history data has been used to predict stock movement with reasonable accuracy. Also, with T test result analysis we can conclude that LSTM performs better in comparison to Back propagation and SVM. For this implementation, we can conclude that if we incorporate all the factors that affect performance of the stock and feed them to neural network with proper data preprocessing and filtering, after training the network we will be able to have a model which can predict stock momentum more accurately and precisely for the better idea of stock value so that firms may have increased profit ratio as compared to what is might be going currently at that time. This will also lead to more transparency regarding stock as it will be easier for firms to analyze loses and achieve great success.

Isaac Kofi Nti [5] they introduced a novel "homogeneous" ensemble classifier (GASVM) based on Genetic Algorithm (GA) for feature-selection and optimization of SVM parameters for predicting 10-day-ahead price movement on the Ghana stock exchange (GSE). Accuracy metrics such as RMSE, MAE, AUC, Accuracy, Recall were compared, between proposed model (GASVM) and other state-of-the-art predictive models (DT, RF and NN). The GASVM showed a higher prediction accuracy of the GSE stock-price movement as compared with DT, RF and NN. The primary input of this study is the introduction of a GA as a feature selection mechanism to optimize the various design factors of the SVM simultaneously. This yielded evidence in the results obtained from the proposed model compared with the conventional SVM ensemble, random forest, decision trees, and neural network.

Anshul Mittal [6] proposed to apply sentiment analysis and machine learning principles to find the correlation between "public sentiment" and "market sentiment". they use twitter data to predict public mood and use the predicted mood and previous days' DJIA values to predict the stock market movements. In order to test their results, they propose a new cross validation method for financial data and obtain 75.56% accuracy using Self Organizing Fuzzy Neural Networks (SOFNN)on the Twitter feeds and DJIA values from the period June 2009 to December 2009. they also implement a naive portfolio management strategy based on their predicted values. their work is based on Bollen et al's famous paper which predicted the same with 87% accuracy.

3. PROPOSED MODEL

There are three main modules in the proposed model system

1. Registration
2. Stock Price Prediction
3. Chatbot

1. Registration:

   The Registration module will register users with User's name, email and password. After the Registration is successful the user can login to the system. Once login, the user will be directed to home page.

2. Stock Price Prediction:

   In the home page User will chose a company to invest in and upload the respective file in which it contains the stock price value of that company for last 10-15 years. This file will be taken as training data. Once system is trained user will get the predicted result of the stock price in the form of table and graphical representation for better understanding.

3. Chatbot:

In Chatbot the new users can know about the process of stock market, what is stock market, definitions of attributes, the initial investment amount and the queries related to stock market will be answered in the chatbot.

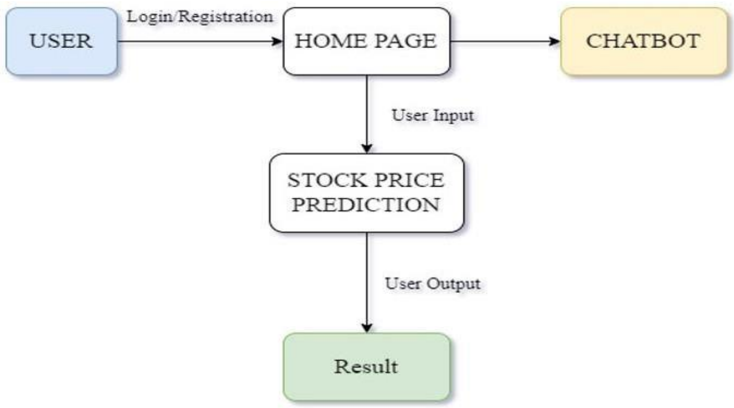

Fig.1 Front End System design

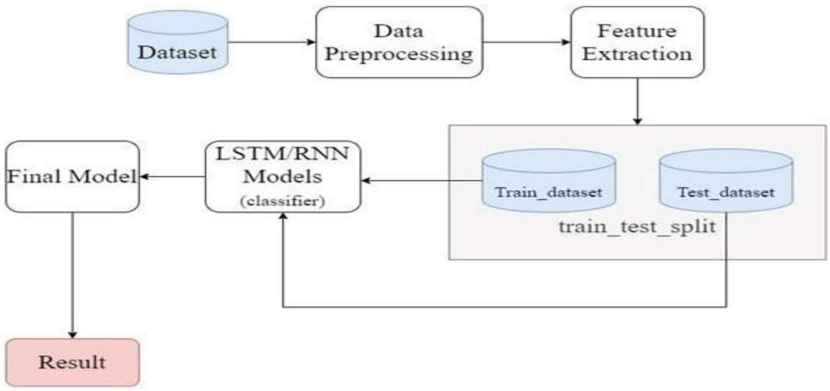

Fig.2 Back End System design

4. METHODOLOGY

METHODS

LSTM

Long short- term memory (LSTM) block or network is advancement to the simple recurrent neural network which can be used as a building component or block for an eventually better serial analysis using the recurrent neural network. LSTM block itself a recurrent network as it contains recurrent connections like connections as in a conventional recurrent neural network.

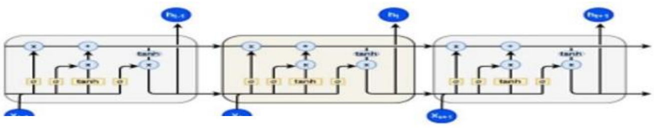

Fig.3 LSTM Model Structure

LSTM Algorithm:

```
def lstm_algo(self):

            yhat = model.predict(x_input, verbose=1)print("{}
            day output {}".format(i,yhat))
            temp_input.append(yhat[0][0])
            temp_input=temp_input[1:] #print(temp_input)
            lst_output.append(yhat[0][0])
            i=i+1else:
            x_input = x_input.reshape((1, n_steps, n_features))yhat =
            model.predict(x_input, verbose=0) print(yhat[0])
            temp_input.append(yhat[0][0])
            lst_output.append(yhat[0][0]) i=i+1


    length = len(lst_output)+len(training_set)+1a =
    np.arange(1,length)
    b = np.arange(len(training_set))d1 =
    np.array(training_set)
    d2= np.array(lst_output)
    d3 = np.concatenate([d1,d2])# plotting

    #plt.title("Line graph")
    fig = plt.figure(figsize=(7.2,4.8),dpi=65)
    plt.title("LSTM ")
    plt.xlabel("No of samples")
    plt.ylabel("Price")
    plt.plot(a, d3,label='Actual Price',color ="Red", linewidth=2) plt.plot(b,
    d1,label='Predicted Price',color ="Blue", linewidth=4)plt.legend(loc=4)
    plt.savefig('static/lstm_result.png') plt.close(fig)
    return lst_output
```

RNN

Recurrent Neural Networks are the class of Neural Networks where the units are recurrently connected. This allows them to use their internal memory for reclaiming the sequence of inputs, and to be used for handwritten recognition, manual generation, the stock request or speech recognition. Recurrent Neural Networks are used in this design since long- term dependences in the data need to be considered for the stock data. Due to the incapability of RNN to store the large amount of dataset using LSTM is a better option. And Long Short- Term Memory cells rather used traditional Neuron- similar cells. A truly significant interpretation of neural networks is honored as RNN, which is extensively employed in different problems. In a typical neural network, the input passes through some layers, and output is created.

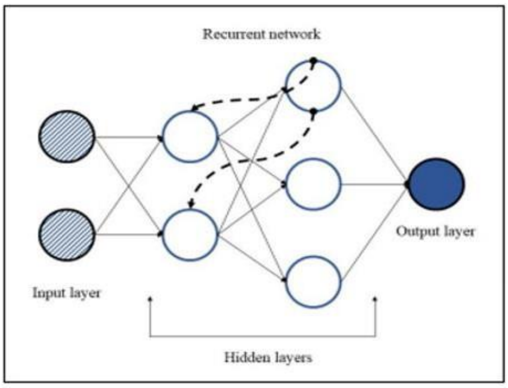

Fig.5 RNN Structure model

RNN Algorithm:

```
def rnn_algo(self):
      dat = self.name
      #df_train = pd.read_csv('Stock_Price_Train.csv')dataset =
      pd.read_csv('stocks data/'+self.name) test_loc =
      len(dataset)-int(len(dataset)*0.2)
      train = dataset.iloc[:test_loc]
      test = dataset.iloc[test_loc:]
      training_set = train.iloc[:, 2:3].values
      # Feature Scaling
      from sklearn.preprocessing
      import MinMaxScaler
      sc =MinMaxScaler(feature_range = (0, 1))
      training_set_scaled = sc.fit_transform(training_set)
      #Creating a data structure with xx timesteps and 1 output
      X_train = []
      y_train = []
      for i in range(10,len(training_set)):
          X_train.append(training_set_scaled[i-10:i, 0])
          y_train.append(training_set_scaled[i, 0])
      X_train, y_train = np.array(X_train), np.array(y_train)
```

For case, to predict stock request at a certain period, it's vital to observe the previous samples. RNN is named recurrent due to it does the same task for each item of a sequence when the output is related to the prior computed values. As another important point, RNN has a specific memory, which stores previous computed information for a long time.

CNN

In deep learning, a convolutional neural network (CNN or ConvNet) is a class of artificial neural network (ANN), utmost generally applied to analyze visual imagery. CNNs are also known as Shift Invariant or Space Invariant Artificial Neural Networks (SIANN), grounded on the partook- weight structure of the complication kernels or purifiers that slide along input features and give restatement- equivariant responses known as point charts. Counter-intuitively, utmost convolutional neural networks aren't steady to restatement, due to the down sampling operation they apply to the input. They've operations in image and videotape recognition, recommender systems, image category, image segmentation, medical image analysis, natural language processing, brain – computer interfaces, and monetary time series.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

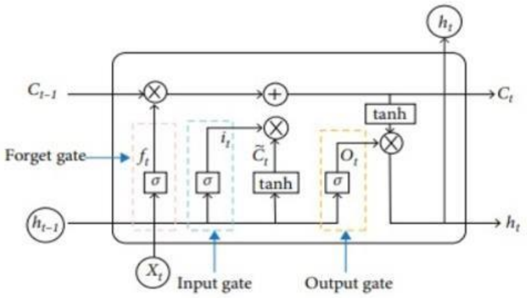

Fig.6 CNN Structure Model

CNN algorithm:

```
def cnn_algo(self):
    data = self.name
    dataset = pd.read_csv('dataset/'+self.name)
    training_set = dataset.iloc[:, 2].values
    # split a univariate sequence into samples
    def split_sequence(sequence, n_steps):
        X, y = list(), list()
        for i in range(len(sequence)):
            # find the end of this pattern
            end_ix = i + n_steps
            # check if we are beyond the sequence
            if end_ix > len(sequence)-1:
                break
            # gather input and output parts of the pattern
            seq_x, seq_y = sequence[i:end_ix], sequence[end_ix]
            X.append(seq_x)
            y.append(seq_y)
        return np.array(X), np.array(y)
```

CNNs use moderately little pre-processing compared to other image classification algorithms. This means that the network learns to optimize the purifiers (or kernels) through automated learning, whereas in traditional algorithms these purifiers are hand- manipulated.

DATASET

The data in this paper consist the data is the price history and trading volumes of the fifty stocks in the index NIFTY 50 from NSE (National Stock Exchange) India. All datasets are at a day-level with pricing and trading values split across .CVS files for each stock along with a metadata file with some macro-information about the stocks itself. The data spans from 1st January, 2000 to 30th April, 2021.

IMPLEMENTATION

LSTM, CNN, RNN algorithms are used to train and predict the Stock Price. The LSTM recurrent layer consists of memory units called as LSTM (). A fully connected layer that often goes around LSTM layers is used for outputting a prediction and is called as Dense (). By transforming a dataset using the function reshape () in NumPy.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

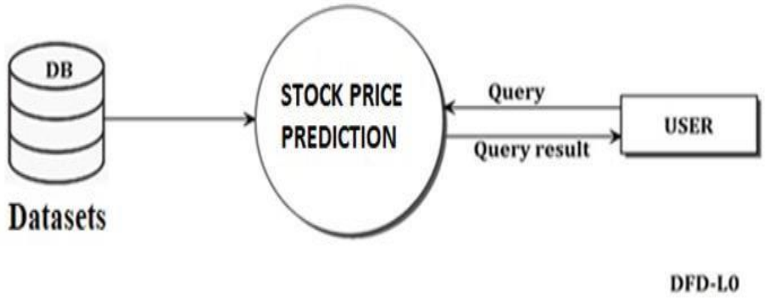

Fig.7 Data Flow Diagram (Level 0)

The most crucial step when starting with ML is to have dataset that we have collected from Kaggle. The collected data is a raw form which can't be directly fed into the machine. Here data pre-processing will takes place. Next the pre-processed data is ready for data preparation. The large amount of data will be allocated for training (80%) and rest of the data will be tested (20%). After the data is trained and tested the model is saved. For example if user gives a specific company name like Asian paints this query will be pre-processed in the dataset that is only on Asian paints. This pre- processed data will be given to the model, Then accuracy and prediction will be tested. Those query results will be given back to the user.

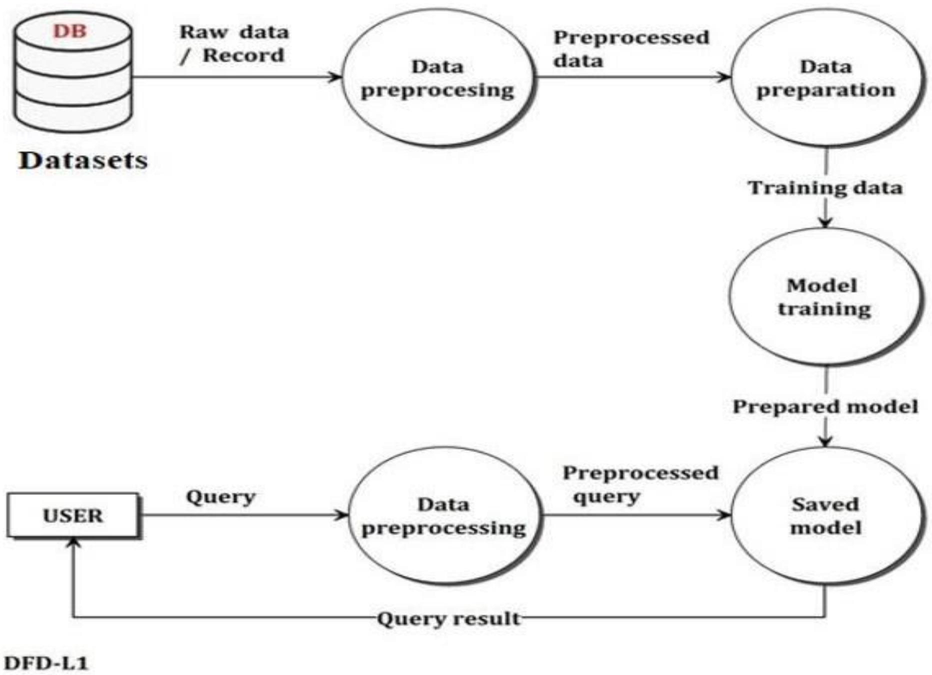

Fig.8 Data Flow Diagram (Level 1)

1	Symbol	Open	High	Low	Last Trade	Change	%Change	Traded Vo	Traded Val	52 Week H	52 Week L	365 Days %	30 Days % Change
2	NIFTY 50	11,879.20	11,932.65	11,853.95	11,913.45	5.3	0.04	5,489.76	16,548.31	12,103.05	10,333.85	13.65	5.05
3	ZEEL	289.25	307.9	285.45	305.2	17.7	6.16	209.73	628.83	506.9	199.15	-30.29	25.65
4	YESBANK	70	73.55	69.15	72.9	3.95	5.73	2,639.48	1,894.88	286	29	-67.32	81.8
5	BPCL	506.65	519.5	499	516.9	14.15	2.81	87.65	448.19	545	290.05	77.51	6.37
6	GAIL	126.9	130.8	125	130.2	3.15	2.48	128.01	164.52	189.6	119.5	-64.3	3.46
7	TATAMOT	169.2	172.95	169.05	171.95	2.9	1.72	221.93	378.94	239.35	106	-7.58	34.49
8	ICICIBANK	488	499	484.6	497.25	7.8	1.59	247.66	1,219.78	499	336	41.12	15.95
9	INDUSIND	1,416.00	1,454.90	1,407.40	1,445.00	22.2	1.56	35.21	504.68	1,834.40	1,188.05	-3.22	15.54
10	INFRATEL	218.3	222	213.5	221.3	3	1.37	71.8	156.59	335	176.35	-14.39	-14.7
11	BAJAJFINS	8,799.00	8,925.00	8,775.00	8,925.00	106.15	1.2	1.75	155.08	8,950.00	5,516.50	60.84	10.43
12	KOTAKBAN	1,594.90	1,619.40	1,592.00	1,619.00	18.75	1.17	29.24	471.27	1,683.95	1,115.75	40.33	2.11
13	AXISBANK	723.4	735.5	721.2	733.7	8.2	1.13	63.55	463.39	827.75	580.5	20.96	7.34
14	IOC	134.3	135.95	133.7	135.4	1.35	1.01	62.25	84.03	170.75	116.25	0.26	-6.46
15	SBIN	314.7	319.65	313.1	318.85	2.85	0.9	204.92	648.86	373.8	244.35	14.71	24.82
16	TATASTEEI	395.5	403.75	394	401	3.05	0.77	135.74	543.12	610.6	320.35	-31.2	16.28
17	HDFCBANI	1,252.55	1,268.50	1,250.50	1,265.30	9.7	0.77	71.43	900.46	1,282.70	954.1	-34.13	5.06
18	BHARTIAR	371.25	373.95	368.25	371.7	2.55	0.69	27.75	102.99	397	258.72	25.49	-5.55
19	JSWSTEEL	249.1	253.25	248.05	251.6	1.6	0.64	55.99	140.75	354.35	201.75	-26.98	13.61
20	ADANIPOF	386.15	389.15	382.5	388.2	2.05	0.53	23.01	89.19	430.6	292.1	18.12	-4.72
21	NTPC	118.95	121.75	117.75	118	0.6	0.51	229.43	273.81	145.85	106.67	-22.77	0.21
22	ONGC	138.4	139.75	136.65	138.85	0.55	0.4	70.32	97.4	178.9	115.55	-11.02	2.66
23	COALINDI	208.95	210.8	205.75	209.7	0.45	0.22	34.85	72.82	271.85	177.7	-20.54	10.31
24	BAJFINAN	4,160.00	4,188.00	4,142.90	4,167.00	7.55	0.18	7.69	320.59	4,280.00	2,233.10	82.96	7.31
25	HDFC	2,228.00	2,239.80	2,216.00	2,236.15	2.95	0.13	18.35	409.05	2,357.85	1,780.00	24.51	10.99
26	POWERGR	191.65	193.1	190.35	191.8	0.15	0.08	56.75	109.01	216.25	172.5	3.15	-3.33
27	WIPRO	257.2	257.5	253.4	256.5	0.1	0.04	19.02	48.61	301.6	226.99	-20.67	5.36
28	BAJAJ-AUT	3,238.00	3,254.00	3,221.00	3,239.80	-5.75	-0.18	2.78	90.1	3,289.00	2,442.20	22.15	9.52
29	TITAN	1,156.75	1,163.75	1,137.65	1,153.65	-3.1	-0.27	23.57	272.02	1,389.95	843.2	28.94	-7.63

Fig.9 Nifty50 dataset

5. RESULTS

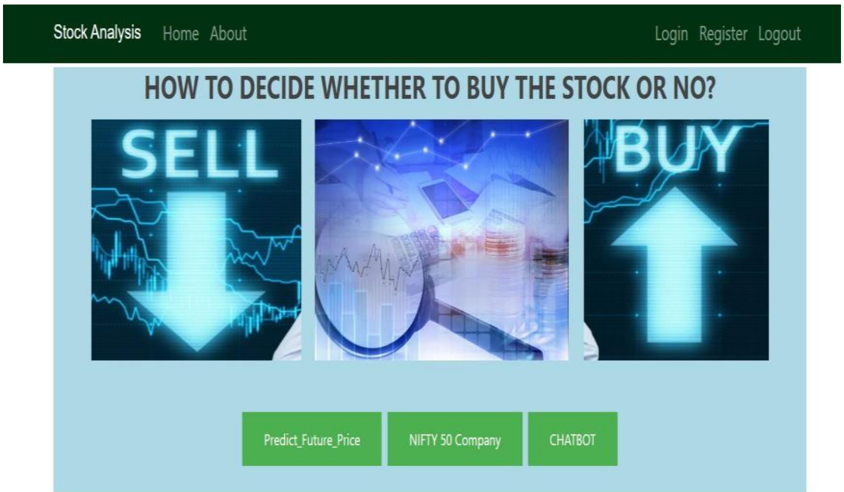

Fig.10 Stock market home page

The home page of the stock market price prediction has three options i.e predict_future_price NIFTY50 company, chatbot.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

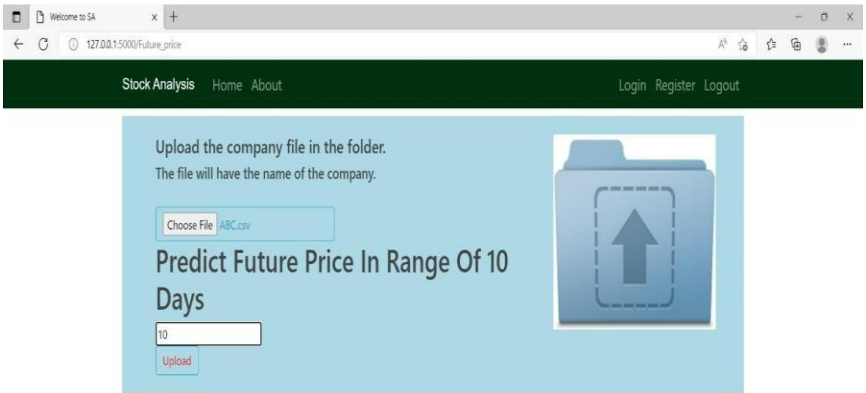

Fig.11 stock market prediction company selection

For predicting the future price of the stock, the user will have to select the file of the particular company and from the folder and upload the file for which company they want to check the future predicted price for the next 10 days. In order to predict future stock price the user will select the particular file from the database and upload the file.

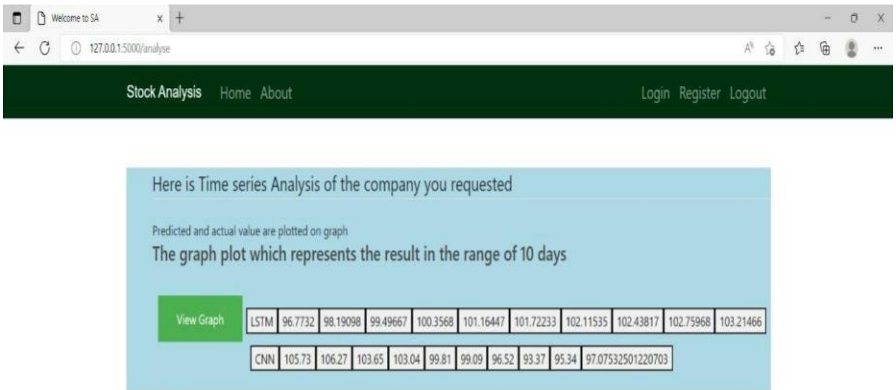

Fig.12 predicted output at the range of 10 days

After uploading the file with respect to chosen company the user will get the predicted analysis price in form of table that for a particular duration of time using LSTM and CNN Algorithm.

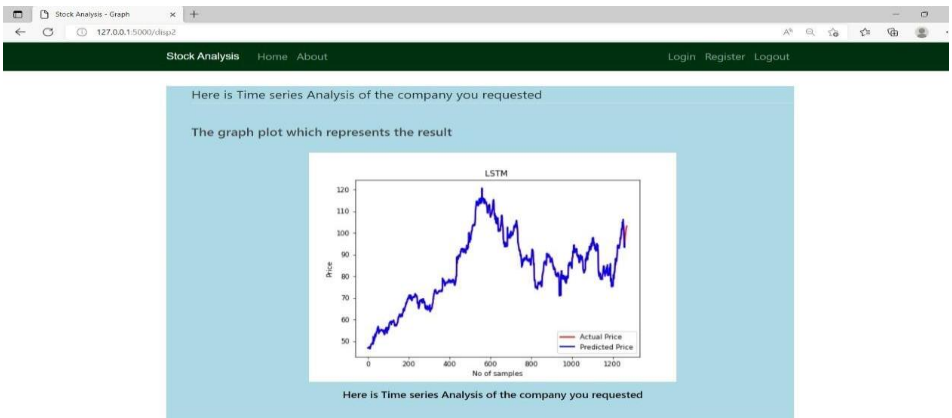

Fig.13 Graph printed According to the Attributes which selected from dataset

The graph is used to represents future price for easy understanding by the user.

In Automated chatbot the users query regarding any process about stock prediction and any questions regarding the attributes like open, high and other attributes will be provided.

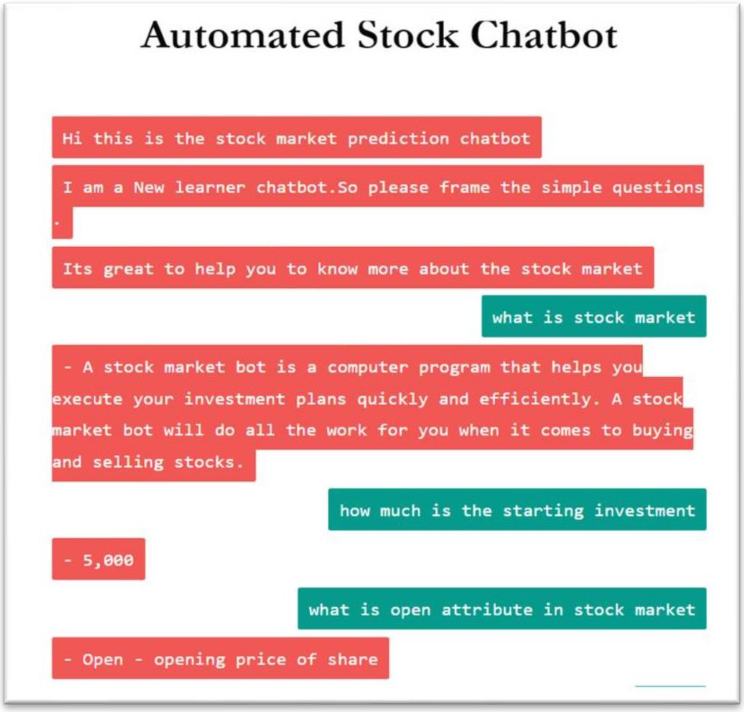

Fig.14 stock market chatbot

6. CONCLUSION

Various Machine Learning based models are used for predicting the daily trend of Market stocks. In this paper we have been utilized the algorithm like LSTM and RNN, CNN on the Nifty fifty companies' dataset. These techniques have shown an improvement in the accuracy of predictions, thereby yielding positive results. Using the above Machine Learning techniques in the prediction of stocks have yielded promising results. It has led to the conclusion that it is possible to predict stock price with more accuracy and efficiency using machine learning techniques. For the proposed system, the data is collected from various global financial markets and used along with Machine Learning algorithms to predict the variation in the stock prices. LSTM algorithm works on the large dataset value and provides better numerical results suggesting higher efficiency. Currently, this model can be successfully hosted on a web server and it can serve as a virtual stock market trading platform.

7. FUTURE ENHANCEMENTS

In the future, the stock market prediction system can be further improved by utilizing a much bigger dataset than the one being utilized currently. This would help to increase the accuracy of our prediction models. Furthermore, other Machine Learning model can also be analyzed for their accuracy rate. This application can be used to retrieve the current market scenario at any given point of time and allows a user to trade virtual money using real time data. By analyzing historical data as well as user's portfolio, the proposed system guides the user to buy stocks by predicting future trends in the stock market on a day-to-day basis. In future for better graphical representation of graph User Interfaces libraries can be used.

8. REFERENCES

[1]. Pooja Mehta, Sharnil Pandya and Ketan Kotecha, "**Harvesting social media sentiment analysis to enhance stock market prediction using deep learning",** 2020.

[2]. Anshul Mittal, Arpit Goel, "**Stock Prediction Using Twitter Sentiment Analysis",**2019.

[3]. Vaishnavi Gururaj, Shriya V R and Dr. Ashwini K "Stock **Market Prediction using Linear Regression and Support Vector Machines",** 2019.

[4]. Shahrokh Asadi a, Esmaeil Hadavandi A, Farhad Mehmanpazir b, Mohammad Masoud Nakhostin b, "**Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction",** 2019.

[5]. Isaac Kofi Nti*, Adebayo Felix Adekoya and Benjamin Asubam Weyori, **"Efficient Stock-Market Prediction Using Ensemble Support Vector Machine",** 2020.

[6]. Ashwini Saini , Anoop Sharma **"Predicting the Unpredictable: An Application of Machine Learning Algorithms in Indian Stock Market"**, 2020.