International Journal of Engineering Sciences & Emerging Technologies, May 2024.
ISSN: 22316604 Volume 11, Issue 3, pp: 546-557 ©IJESET

STOCK PROGNOSTICATION: PROTOCOL BASED ON INDIVIDUAL STOCK PRICES PREDICTION USING LSTM

Richa Saxena[1], Anuj Sharma[2], Hadiya Khaleeq[3], Sushant Singh[4], Tanveer Alam[5]

[1]Assistant Professor, CS&E Department, MIT, Moradabad
richasaxena2006@gmail.com
[2]B. Tech 4th Year, CS&E Department, MIT, Moradabad
as.anuj963852@gmail.com
[3]B. Tech 4th Year, CS&E Department, MIT, Moradabad
hadiachoudhry8@gmail.com
[4]B. Tech 4th Year, CS&E Department, MIT, Moradabad
sushantsingh4506@gmail.com
[5]B. Tech 4th Year, CS&E Department, MIT, Moradabad
mbdtanveeralam@gmail.com

ABSTRACT

In today's time, investing in the Stock market is a prominent way to create capital. Investing in the stock market requires financial knowledge, and years of experience, which causes difficulty for the newcomers to enter the market or lose capital. We propose a solution by deep learning-based model. This research aims to build a Model for, the Stock Market to predict the price of stocks and trends. This model is based on a variant of the recurrent neural network called Long Short-term memory. The proposed solution includes Data Extraction, Data Pre-processing, Training Testing on the stock data. Model functions by analyzing the series of data or sequential data and predicting the next possible output. After evaluating different Models used for Stock prediction, this model uses fewer features which reduces the data pre-processing to the minimum and achieve higher accuracy. This system will be beneficial to the stock investing firms as an analytical tool and also for newcomers.

KEYWORDS

Deep Learning, Long Short-Term Memory, Sigmoid Function, Recurrent Neural Network, Tanh Function, Random Forest, Support Vector Machine.

1. INTRODUCTION

As stock market is the vast platform in which investors are interested to invest their money in the companies with trending price to get some profit in the future. Sometimes stock market prices are very unpredictable, and the stock market goes down and investors lose his money that they have invested in the stock market.

Our model will help investor to invest in the companies in which there are good chances of getting profit. It will help the user by predicting the future price of the stock by using the available data of the previous price point that are stored in the dataset which we will use as a raw data for our modeland this prediction will be done with the help of Deep Learning Algorithms.

2. Jupyter

Jupyter is an application consists of 3 parts:

2.1 Notebook document:

It is a document produced by the Jupyter Notebookwhich contains element like rich text, programming code, equations, results and the executable codewhich is required to do data analysis.

2.2 Jupyter Notebook App:

It is based on Server – Client Application that hasfeature like editing and running documents written in the notebook document via a web browser. It can be executed on a local desktop which requires no internet access or can be accessed through the internet on a remote server.

2.3 Kernel:

Kernel in the Jupyter is a "computational engine".It is the main component of the Jupyter notebook, which executes the code contained in the Notebook document. When the Notebook document is launched in Jupyter, the kernel performs the computation on thenotebook document and displays the result of the executable code.

3. Long Short-Term Memory(LSTM) and Sentimental Analysis

This is a variant of RNN which is used in Deep Learning. In LSTM previous data are stored in the cells and then that stored data were used when needed, it will be comparing old, stored data with thecurrent data and gives us the best predicted accuracyfor the stock.

LSTM consist of three parts that is Forget Gate, Output Gate, Input Gate.

3.1 Recurrent Neural Network(RNN):

It is an artificial neural network that works on solving sequential data or time-series data problems.These Networks are commonly used for Problems related to Language translation, natural language processing, speed recognition, etc.

As the name suggests, these networks are recurrent and every single input data is compute by same function and the output depends on the previous oneand then the output is replicate and sent back to the neural net as a current input to find the next output as shown in figure 1.

Working of RNN:

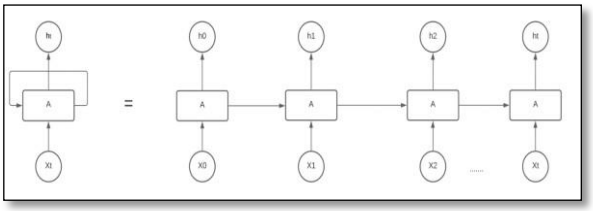


Figure 1: Basic Architecture of RNN

In this, X (0) will be sent as an input from the Sequential Data, and H (0) will be the output. Then the H (0) output with the X (1) input will be feedback to the network for the next step, which will give H (1). And in the next Step H (1) and X (2) will be fedtogether in the network for the next H(t) and so, and that's how it remembers the information points throughout training.

Current State Formula is:

$$V_t = f(V_{t-1}, x_t) \quad (3.1)$$

Activation Function used:

$$V_t = \tanh (W_{vp} = V_{t-1} + W_{xv}X_t) \quad (3.2)$$

W is *weight*, **V** is the *single hidden vector*, **Wvp** is *the weight at* the *previous hidden state*, **Wxv** is the *weight at* the *current input state*, **tanh** is the *Activation function*, that implements a non-linearity that

squashes the activations to the range [-1.1].

Output:

$Z_t = W_{vy} h_t$ (3.3)

Zt is output state.

Wvy is weight at the output stateLimitations of RNN:

- Training of Recurrent Neural Nets isDifficult
- The Vanishing Gradient Problem
- RNNs cannot be stacked
- Difficult to on longer sequences

3.2 Why LSTM over StandardRNN

RNN works on the sequence of data and is useful when the difference between the appropriate information is small and, sometimes, we only need recent information to do the current task. For those tasks, RNN is very prominent but as the data increases and the gap between information increases, RNN is not able to process the connectedinformation. LSTM is used as it can retain the relevent information for a longer period. LSTM cansolve all the problems that RNN can solve, but RNNcannot solve all the problems that LSTM can solve.

3.3 Long Short-Term MemoryNetworks

Long Short-Term Memory Network are designed toremember information for longer period. RNN will be considered as a base for Long Short-Term Memory Network which commonly known as LSTM. RNN is considered as a best practice to be used in order to analyze long-term dependencies..

These are a type of RNNs which are effective at analysing and learning any order dependence in sequence data issues. It is difficult to understand what LSTMs are and how they work, there are different terms used in this like bidirectional and sequence-to-sequence.

As with all RNNs, LSTM also has a chain like structures of repeating neural net modules. Modulesare linked and has a different structure, unlike the RNNs simple structure as shown in figure 2 & 3.

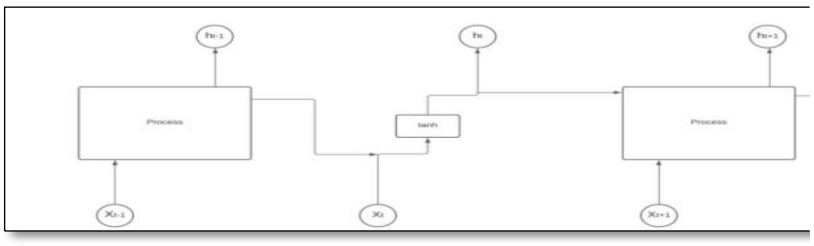


Figure 2: The Structure of RNN containsSingle layer

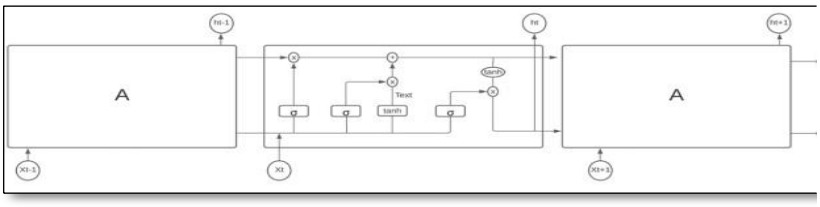


Figure 3: Structure of LSTM contains fourInteracting layers

3.3.1 Working of LSTM:

LSTM has a cell state layer that runs straight to end of the chain structure, having only minor linear interactions. LSTM can remove or add information in the cell state. This addition and removal of information are done by the structures called gates as shown in figure 4, 5 & 6.

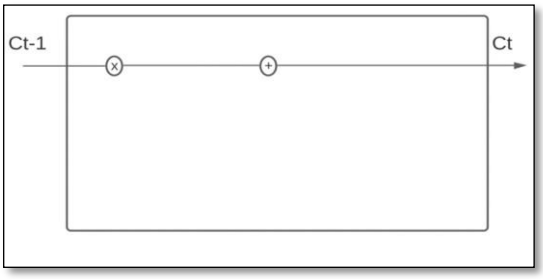


Figure 4: LSTM Working Model

The very first Step is to conclude which information will be discarded from the cell state as shown in figure 5. This decision is made by a sigmoid layer called Forget gate layer. It takes the data at x_t and ht-1, andas the sigmoid function is used, it gives a number asoutput in range 0 and 1 for the each cell state. Output 1 represents don't remove the information and keeping it, and output 0 represents removing the information completely from the cell state layer.

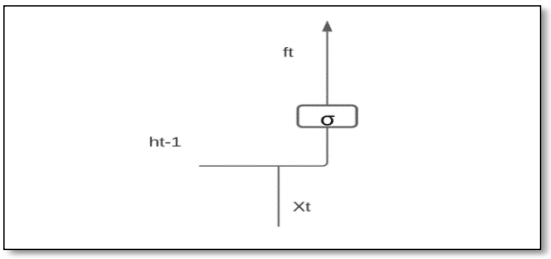


Figure 5: Forget Gate

Second Step is to choose what new informationis needed to be placed. There are two parts of adding relevant information into the cell the state. First, the input layer decide what informationis need to be updated. Second, a new vector of new selected value ~Ct that could be append to the state generated by tanh layer as shown in figure 6. And then, these two are combined to create an updated cell state.

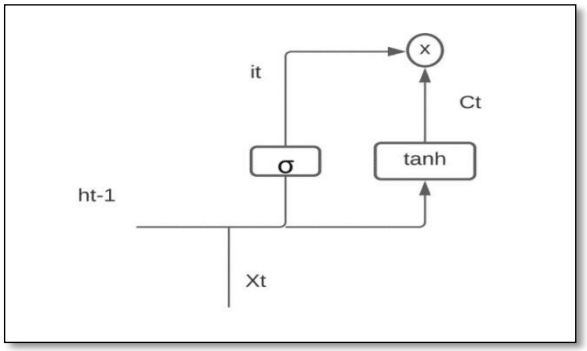


Figure 6: Input gate

Now, this step further divided into three parts at thevery first level the sigmoid layer will choose

which part of the cell state is going to be the output at second level. Tanh will be used on the previously selected cell state layer so that the output values lieson a scale of -1 to 1, at the third level output of the second level will be multiplied to the output of the sigmoid gate so that the final output will be achieved as shown in figure 7.

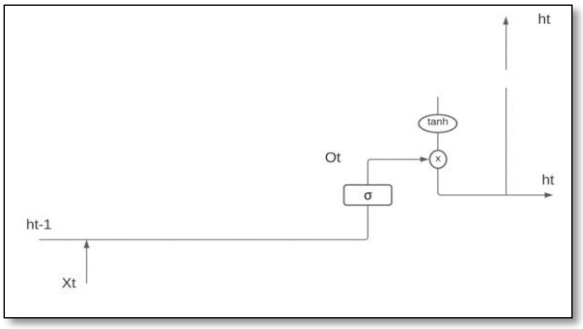


Figure 7: Output

4. Project Survey

As concluded by Nath and Hasan in **Hidden Markov Model (HMM),** for the stocks of fourdifferent Airlines [1]. They reduced their model into four states of price that are: Closing, Opening, the highest, the lowest. A drawback for their model wasthat it was limited to small datasets, and this cannotlead to the generality for the market.

Lee MC in **SVM with A Hybrid Feature Selectionto predict the Stock Trend** model used **SVM,** this model includes hybrid feature selection method [2]. Limitation for this model was that it only compares with only **Back Propagation Neural Network.**

Kara applied **Artificial Neural Network** (**ANN)** and **Support Vector Machine (SVM),** for theirmodel [3]. It works on the stock price movement index. Their model works on stock market, featuresbut it doesn't predict anything. Model never workedon the stock market price structure, neither it has the nobility and also it is not described how model works.

Thakur and Kumar applied **Multi categoryclassifier** and **Random Forest (RAF),** for their model [4]. They develop a System for Hybrid Financial trading (it is an exchange through which trader can trade in both the system's automated one and transaction floor brokers to complete their transaction). Developed model generates three typesof signals that are: **- Buy Signal, Hold Signal, Sell Signal**. The Problem with this model was that is lackin financial domain, and it works on a single stock ata time don't give any compared result between multiple stocks.

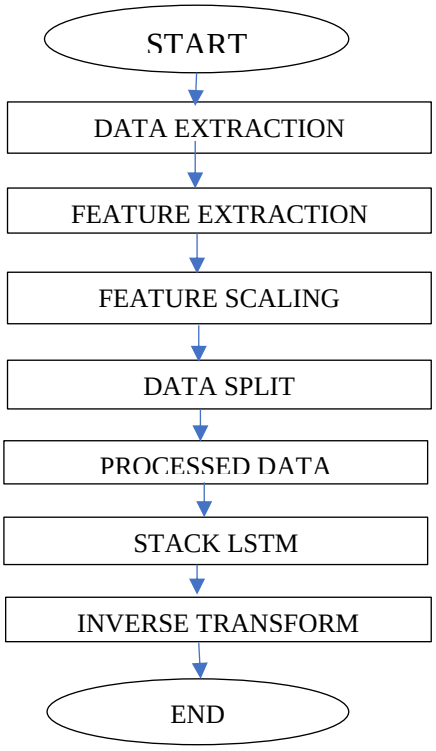
5. Methodology and data

In this model we are extracting data directly from **Yahoo Finance.**

Yahoo Finance is a site which tracks the market and stocks and data of each and every company and provide up to date data and portfolio management resources and financial news, etc.

For the building of this model, we are using LSTM, this is the best algorithms that will be there which gives the maximum accuracy for the stock that we are searching for In our model the data which will be considered further divided into two parts the first part which is 70% of the data is used for training purpose and the rest part of data which is 30% is used for testing purpose.

5.1 Long Short-Term Memory(LSTM) Model



5.1.1 Data extraction:

Data Extraction is done using Yahoo Finance dynamically, in this we must enter the company's ticket name to extract data.

E.g.:- Out of which only four Columns used from the dataset in this model are four categories of price: Low, High, Open, Close as shown in figure 8.

Date	High	Low	Open	Close	Volume	Adj Close
2011-03-16	168.139999	162.869995	164.699997	164.699997	5208100	164.699997
2011-03-17	166.300003	160.779999	165.910004	160.970001	6471400	160.970001
2011-03-18	163.539993	160.589996	161.190002	161.820007	7442700	161.820007
2011-03-21	165.789993	161.720001	163.369995	164.520004	4055100	164.520004
2011-03-22	164.440002	162.250000	164.070007	162.600006	3611400	162.600006

Figure 8: Extracted Data

5.1.2 Feature Extraction:

Important columns were extracted in this processfrom the entire data as shown in figure 9.

For e.g.: - Columns used from the dataset in thismodel are four categories of price: Low, High, Open, Close.

```
df1=df.reset_index()['Close']

df1

0        164.699997
1        160.970001
2        161.820007
3        164.520004
4        162.600006
           ...
2705    3391.350098
2706    3381.830078
2707    3466.300049
2708    3377.419922
2709    3400.350098
Name: Close, Length: 2710, dtype: float64
```

Figure 9: Feature Extraction

5.1.3 Feature Scaling:

In this model used dataset is scaled between 0 to 1 shown in figure 10. So that the accuracy gained will be maximum. Usingthis feature model will not be overfitted.

```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
df1=scaler.fit_transform(np.array(df1).reshape(-1,1))

print(df1)

[[1.04468800e-03]
 [0.00000000e+00]
 [2.38067612e-04]
 ...
 [9.25748684e-01]
 [9.00855357e-01]
 [9.07277584e-01]]
```

Figure 10: Scaling Data

5.1.4 Data Split (Training and Testing):

In this model the data which will be considered divided into two parts as shown in figure 11. The first part which is 70% ofthe data is used for training purpose and the rest partof data which is 30% is used for testing purpose.

Figure 11: Splitting Training and TestingData

5.1.5 Processed Data:

After applying all the above steps, the data is processed and then algorithm will be applied to the processed data.

5.1.6 Stack LSTM:

It's an architecture in which multiple LSTM layersare used so that data will be processed multiple times and in the first layer it takes in order and in sequential next layer it takes one input as shown in figure 12. This process makes it more accurate.

```
In [108]:
training_size=int(len(df1)*0.65)
test_size=len(df1)-training_size
train_data,test_data=df1[0:training_size,:],df1[training_size:len(df1),:1]

In [109]: training_size,test_size

Out[109]: (1761, 949)
```

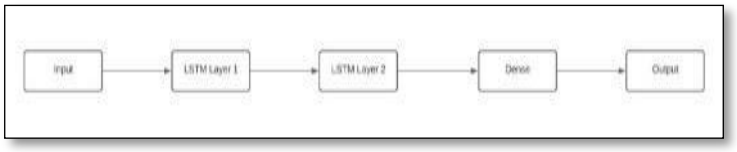


Figure 12: Stacked LSTM

5.1.7 Inverse Transform:

It convertsscaled data into original values as shown in figure 13.

Figure 13: Inverse data

6. Proposed solution

Algorithm: Long Short-Term Memory

function ModelCompile() Stack_method = Sequential() Layer_1 = LSTM(50, input_size =

(X_train.shape[1], X_train.shape[2]))Layer_2 = Dense (1)

Loss Function = mseOptimizer = adam

Metrics = f1, metrics. binary_accuracy,metrics.mse, metrics.mae

return LSTMmodelend function

function MAIN()

ModelCompile(i)

FitModel(i_training, Y_training, epochs=100,batchsize=64)

EvaluateModel(i_testing, Y_testing)model using test data

end function

7. Result

Working of the model is we firstly extract datasetdynamically using yahoo finance and then on the **Extracted Data** we have applied **FeatureExtraction** as we are using only closing data for theprediction so this will help in removing other unnecessary columns from the dataset, after applying all the above steps we have rescaled variable in the range of (0 and 1) by using **Feature Scaling**. **Data Split (Training and Testing)** now we have trained and tested our data in the ratio of 70% and 30% as this will give good accuracy, **Processed Data** after we have applied all thesesteps, we got our processed data and then we are applying the main algorithm and in this we have applied multiple layers of **LSTM** to get more accuracy which is Stacked LSTM and then the output we got from the LSTM model is

used in **Inverse Transform** to scale it back to its original range.

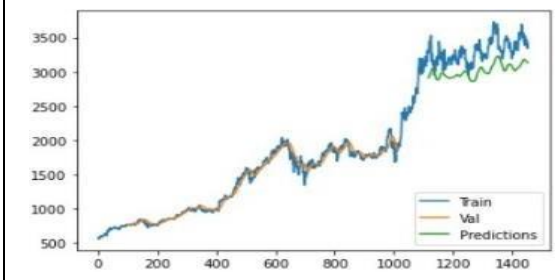
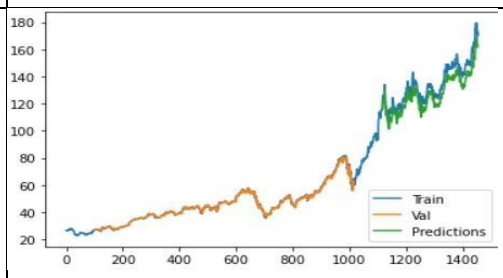
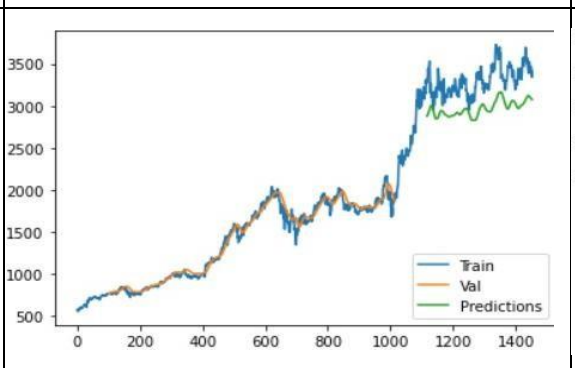
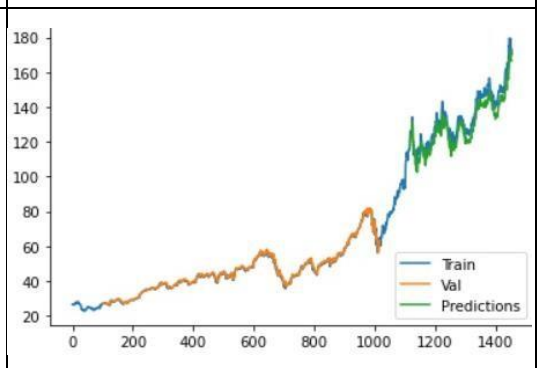
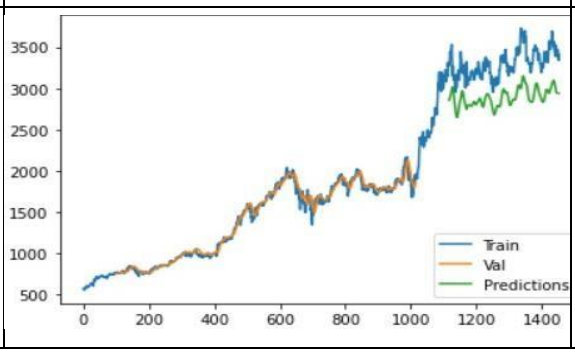
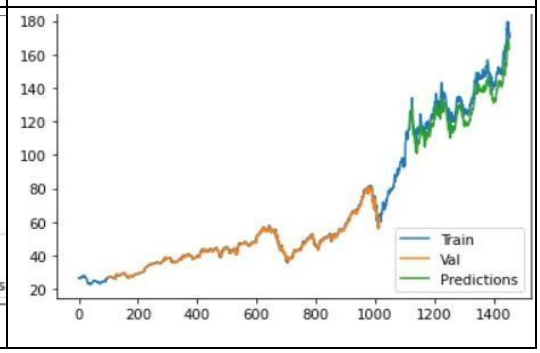
The model is built to predict future price of the stockand the predicted output for the dataset of company“Amazon” is:

1) For 22 Dec 2021 is for Amazon closingprice is: -

Figure 14: Output Predicted by Our ModelBased on Closing Time

Date	Open	High	Low	Close*	Adj Close**	Volume
Dec 22, 2021	3,385.40	3,441.00	3,370.01	3,420.74	3,420.74	2,751,800

Figure 15: Original Stock Price in The Market Compares with The Closing Price

Epochs	AMAZON	APPLE
10		
15		
25		

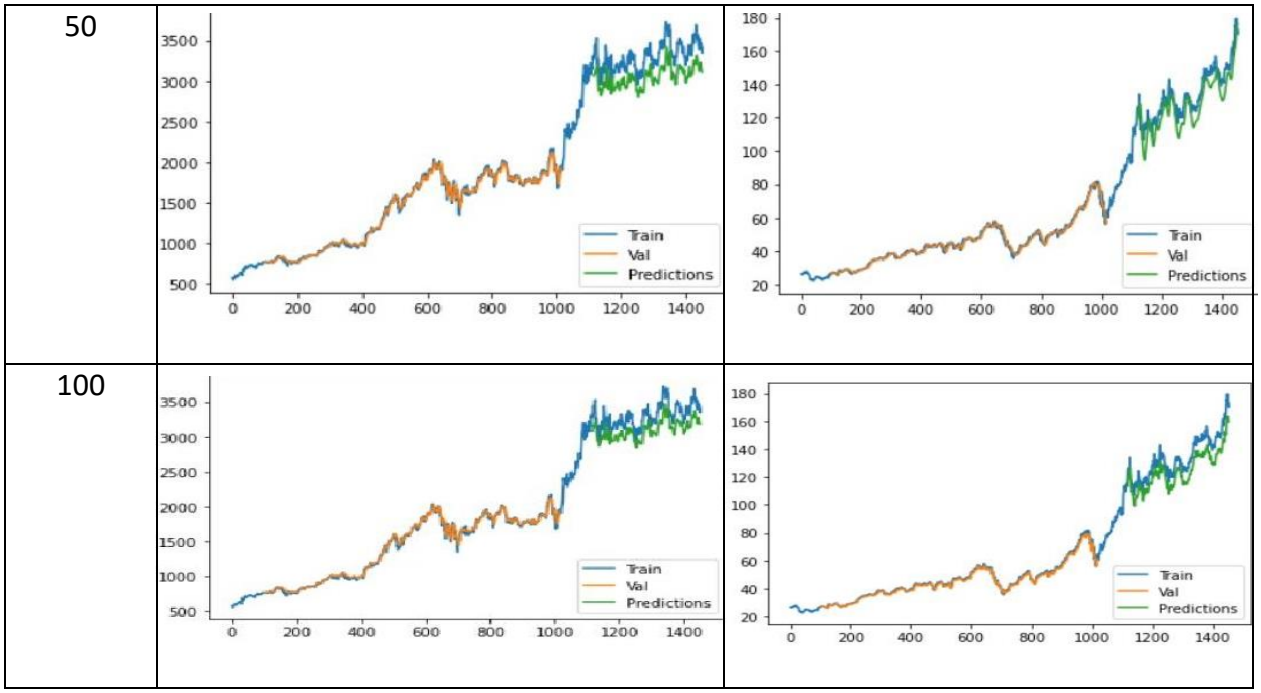


Figure 16: Visualization of Training and Prediction Data

8. Conclusion

Models like SVM, Linear Regression, etc. need many features for the prediction which need more preprocessing to extract more features from dataset,these features are dependent on each other. The novelty features of this paper are that we use **Long Short-Term Memory (LSTM)** model, the LSTM isa variant of RNN having different gates and architecture which analysis the time series and sequential data.

In this model only one feature is enough "close price" from the dataset that helps in reducing the preprocessing stage, and variables are independent give more accurate prediction than any other models[12].

In this model only one feature is enough "close price" from the dataset that helps in reducing the preprocessing stage, and variables are independent give more accurate prediction than any other models[12].

This System consists of Seven states: First extracting dataset dynamically form yahoo finance using pandas library, and on the data **Feature Extraction** is applied to extract the closing price from the dataset. **Feature Scaling** is done on the closing price to scale the value in the range of [0,1] to fit in the model. **Data Split (Training and Testing)** now the data is cleaned data after which it will be split into testing and training dataset in the ratio of 30% and 70% as this will give good accuracy, **Processed Data** after these preprocessingsteps, we got our processed data and then this processed data will be fed to the stacked LSTM model which consist of the smultiple layer of LSTMto get more accuracy than Output will be provided by the LSTM model will be in **Inverse Transformation** to change it back to its original format to display to the user.

This paper proposes LSTM to display the future value of the Cipla and Adani port assets as this model provides 70-80 percent on an average of 75%accuracy in tracing the future opening price for boththe assets.

REFERENCES

[1] Hassan MR, Nath B. Stock market forecasting using Hidden Markov Model: a new approach. In: Proceedings—5th international conference onintelligent systems design and applications 2005, ISDA'05. 2005. pp. 192–6. https://doi. org/10.1109/ISDA.2005.85

[2] Lee MC. Using support vector machine with a hybrid feature selection method to the stock trend prediction. Expert Syst Appl. 2009;36(8):10896–904. https://doi.org/10.1016/j.eswa.2009.02.038.

[3] Kara Y, Acar Boyacioglu M, Baykan ÖK. Predicting direction of stock price index movement using artifcial neural networks and support vector machines: the sample of the Istanbul Stock Exchange. Expert Syst Appl. 2011;38(5):5311– 9. https://doi.org/10.1016/j.eswa.2010.10.027.

[4] Thakur M, Kumar D. A hybrid fnancial trading support system using multi-category classifers and random forest. Appl Soft Comput J. 2018;67:337– 49. https://doi.org/10.1016/j.asoc.2018.03.006.

[5] Atsalakis GS, Valavanis KP. Forecasting stock market short-term trends using a neuro-fuzzy based methodology. Expert Syst Appl. 2009;36(7):10696–707.

[6] Tsai CF, Hsiao YC. Combining multiple feature selection methods for stock prediction: union, intersection, and multiintersection approaches. Decis Support Syst. 2010;50(1):258–69. https://doi.org/10.1016/j.dss.2010.08.028.

[7] Tushare API. 2018. https://github.com/waditu/tushare. Accessed 1 July 2019.

[8] Wang X, Lin W. Stock market prediction using neural networks: does trading volume help in short-term prediction?. n.d.

[9] Weng B, Lu L, Wang X, Megahed FM, Martinez W. Predicting short-term stock prices using ensemble methods and online data sources. Expert Syst Appl. 2018;112:258–73. https://doi.org/10.1016/j.eswa.2018.06.016.

[10] Zubair M, Fazal A, Fazal R, Kundi M. Development of stock market trend prediction system using multiple regression. Computational and mathematical organization theory. Berlin: Springer US; 2019. https://doi.org/10.1007/s1058 8-019-09292-7.

[11] Shen J, Shafq MO. Deep learning convolutionalneural networks with dropout—a parallel approach. ICMLA. 2018;2018:572-7

[12] Vaishnavi Gururaj, Shriya V R and Dr. Ashwini K. Stock Market Prediction using Linear Regressionand Support Vector Machines. International Journalof Applied Engineering Research ISSN 0973-4562 Volume 14, Number 8 (2019) pp. 1931-1934. Research India Publications.

AUTHERS

Richa Saxena received her M.Tech degree in Computer Science & Engineering in 2013 from TMU. She is currently an Assistant Professor in CSE Department at MIT with 14 years of professional experience. She has wide expertise in Cloud Computing with a strong background in Networking, Ethical Hacking, Python, IOT, and has completed several FDP's along with STC's from NITTTR, Chandigarh.

Anuj Sharma is an undergraduate B.Tech student in Computer Science & Engineering from MIT and will graduate in 2022. He has an interesting area in Front End Development and UX&UI Designing. He is certified in Artificial Intelligence + Machine Learning from IIT Kanpur. He has knowledge regarding C and Java.

Hadiya Khaleeq is an undergraduate B.Tech student in Computer Science & Engineering from MIT and she will be graduate in 2022. She has a great interest in Machine Learning including Python & MySQL. She has done training in Machine Learning and Artificial intelligence from IIT Kanpur. She has knowledge regarding Java, MySQL and C.

Sushant Singh is an undergraduate B.Tech Student in Computer science and engineering from MIT and will graduate in 2022. He has great interest in Game development, Front End development and Deep learning. He has done training in Machine Learning and AI from Prutor. He also has knowledge regarding C, C++ and Core Java.

Tanveer Alam is an undergraduate B.Tech in computer Science & Engineering from MIT and will graduate in 2022. He has strong interest in the field of Data Science and Machine learning. He is certified in Artificial Intelligence & Machine learning from IIT Kanpur. He also has knowledge regarding C, HTML and SQL.