

**A Mini Project Report**

**on**

**“****Stock Price Prediction using LSTM and Technical indicators”**

**Submitted in partial fulfilment for the award of the degree of**

**BACHELOR OF TECHNOLOGY (HONOURS)**

**IN**

**COMPUTER SCIENCE (DATA SCIENCE)**

**Submitted by**

**M R Naveen Kumar**

**19BTRCR005**

**A Rishab Vanigotha**

**19BTRCR018**

**Sushil Bokade**

**19BTRCR017**

**Chethan S Pandit**

**19BTRCR002**

**Under the guidance of**

**Dr. John Basha**

**Designation**

**Faculty of Engineering & Technology**

**Jain (Deemed-To-Be University)**

**B. Tech (Honours) in Computer Science (Data Science)**

Jain Global Campus, Kanakapura Taluk - 562112

Ramanagara District, Karnataka, India

2021-2022.



**B. Tech (Honours) in Computer Science (Data Science)**

Jain Global Campus, Kanakapura Taluk - 562112

Ramanagara District, Karnataka, India

# **CERTIFICATE**

This is to certify that the project work titled **“Stock Price Prediction using LSTM and Technical indicators”** is carried out by **M.R.Naveen Kumar (19BTRCR005), A.Rishab Vanigotha (19BTRCR018), Sushil Bokade (19BTRCR017), Chethan S Pandit (19BTRCR002),** a bonafide students of Bachelor of Technology at the Faculty of Engineering & Technology, Jain (Deemed-to-be University), Bangalore in partial fulfilment for the award of degree, Bachelor of Technology (Honours) in Computer Science (Data Science), during the Academic year **2020-2021**.

|  |  |  |
| --- | --- | --- |
| **Guide Name****Designation** Faculty of Engineering & Technology,  Jain (Deemed-to-be University)  Date:  Signature: | **Prof. Mohammed Zabeeulla A N**  **Assistant Professor and Programme Coordinator,**  **Dept. of CSE**  Faculty of Engineering & Technology,  Jain (Deemed-to-be University)  Date:  Signature: | **Dr. Devaraj Verma,**  **Professor and Dy. HoD,**  **Dept. of CSE**  Faculty of Engineering & Technology,  Jain (Deemed-to-be University)  Date:  Signature: |

Name of the Examiner Signature of Examiner

1.

2.

# **DECLARATION**

We, **M.R.Naveen Kumar (19BTRCR005), A.Rishab Vanigotha (19BTRCR018), Sushil Bokade (19BTRCR017), Chethan S Pandit (19BTRCR002),** are students of sixth semester B. Tech (Honours) in **Computer Science (Data Science)**, at Faculty of Engineering & Technology, **Jain (Deemed-To-Be University)**, hereby declare that the project work titled **“Project Title”** has been carried out by us and submitted in partial fulfilment for the award of degree in **Bachelor of Technology (Honours) in** **Computer Science (Data Science)** during the academic year **2020-2021**. Further, the matter presented in the project has not been submitted previously by anybody for the award of any degree or any diploma to any other University, to the best of our knowledge and faith.

Student name : M R Naveen Kumar Signature

USN : 19BTRCR005

Student name : A Rishab Vanigotha Signature

USN : 19BTRCR018

Student name : Sushil Bokade Signature

USN : 19BTRCR017

Student name : Chethan S Pandit Signature

USN : 19BTRCR002

Place: Bengaluru

Date:

# **ACKNOWLEDGEMENT**

*It is a great pleasure for us to acknowledge the assistance and support of a large number of individuals who have been responsible for the successful completion of this project work.*

*First, we take this opportunity to express our sincere gratitude to* ***Faculty of Engineering & Technology, Jain (Deemed-to-be University),*** *for providing us with a great opportunity to pursue our Bachelor’s Degree (Honours) in this institution.*

*In particular we would like to thank* ***Dr. Hari prasad S A****,* ***Director****,* ***Faculty of Engineering & Technology****,* ***Jain (Deemed-to-be University),*** *for his constant encouragement and expert advice.*

*It is a matter of immense pleasure to express our sincere thanks to* ***Dr. Devaraj Verma****,* ***Professor and Deputy******Head****,* ***Department of Computer Science & Engineering****,* ***Jain (Deemed-to-be University),*** *for providing right academic guidance that made our task possible.*

*It is a matter of immense pleasure to express our sincere thanks to* ***Prof. Mohammed Zabeeulla, Program Coordinator of Data Science****,* ***Dept. of Computer Science & Engineering****,* ***Jain (Deemed-to-be University),*** *for providing right academic guidance that made our task possible.*

*We would like to thank our guide* ***Dr. John Basha, Designation****,* ***Dept. of Computer Science & Engineering****,* ***Jain (Deemed-to-be University),*** *for sparing his valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.*

*We would like to thank our Project Coordinator* ***Dr. S. Vijayakumar*** *and all the staff members of Computer Science & Engineering for their support.*

*We are also grateful to our family and friends who provided us with every requirement throughout the course.*

*We would like to thank one and all who directly or indirectly helped us in completing the Project work successfully.*

*Signature of Students*

# **TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | | Page No |
| **CERTIFICATE** | | ii |
| **DECLARATION** | | iii |
| **ACKNOWLEDGEMENT** | | iv |
| **TABLE OF CONTENT** | | v |
| **ABSTRACT** | | vi |
| **LIST OF FIGURES** | | vii |
| **LIST OF TABLES** | | viii |
| **NOMENCLATURE USED** | | viii |
| **Chapter 1** | | 1 |
| 1. **INTRODUCTION** | | 1 |
|  | * 1. Overview | 1 |
|  | * 1. Problem Definition | 1 |
|  | * 1. Objectives | 1 |
| **Chapter 2** | | 2 |
| 1. **LITERATURE SURVEY** | | 2 |
|  | * 1. Related Work | 2 |
|  | * 1. Existing System | 2 |
|  | * 1. Limitation of Existing System | 2 |
|  | * 1. Proposed System | 2 |
| **Chapter 3** | | 3 |
| 1. **METHODOLOGY** | | 3 |

|  |  |  |
| --- | --- | --- |
|  | * 1. Dataset | 3 |
|  | * 1. Architecture | 3 |
|  | * 1. Sequence Diagram | 3 |
| **Chapter 4** | | 4 |
| 1. **TOOL DESCRIPTION** | | 4 |
|  | * 1. Hardware Requirements | 4 |
|  | * 1. Software Requirements | 4 |
| **Chapter 5** | | 5 |
| 1. **IMPLEMENTATION** | | 5 |
| **Chapter 6** | | 6 |
| 1. **RESULTS AND ANALYSIS** | | 6 |
|  | * 1. Result Discussion | 6 |
|  | * 1. Comparison with Previous Studies | 6 |
|  | * 1. Analysis | 6 |
| **Chapter 7** | | 7 |
| 1. **CONCLUSIONS AND FUTURE SCOPE** | | 7 |
| **REFERENCES** | | 8 |
| **APPENDIX** | | 9 |

# **ABSTRACT**

Stock value prediction is a complex task that necessitates a solid algorithm foundation in order to compute longer-term share prices. Stock prices are correlated within the market, making it difficult to forecast costs. The proposed algorithm predicts share price using market data and machine learning techniques such as recurrent neural network named Long Short Term Memory, and weights are corrected for each data point using stochastic gradient descent. In comparison to currently available stock price predictor algorithms, this system will produce more accurate results. To influence the graphical outcomes, the network is trained and evaluated using various sizes of input data.

# **LIST OF FIGURES**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | **Fig. No.** |  |  | | **Description of the Figure** | **Page No.** | | 3.2.1 |  |  | | 3.3.1 |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  |  **NOMENCLATURE USED**  |  |  | | --- | --- | | Example |  | | ML | Machine Learning | | NN | Neural Network | | ANN | Artificial Neural Network | | RNN | Recurrent Neural Network | | LSTM | Long Short Term Memory | | NSE | National Stock Exchange | | MA | Moving Averages | | RSI | Relative Strength Index | | MACD | Moving Average Convergence Divergence | | TA | Technical Analysis | |  |  | |  |  |

# **Chapter 1**

**INTRODUCTION**

* 1. **Overview**

The share market is a marketplace where public company shares are traded. As previously discussed, the volatile nature of the stock market necessitates a great deal of analysis based on old data. Previous stock trend prediction algorithms make use of historical time series stock data. The statistical analysis of stock data is central to most scientific stock price forecasting procedures. The paper will develop a stock data predictor program that requires previous stock prices and data as training sets for the training program to predict the stock prices of a specific share. This model will develop a procedure.

This model takes into account a company's historical equity share price and employs an RNN (Recurrent) technique known as Long Short Term Memory (LSTM). The proposed approach takes into account a share's available historical data and predicts a specific feature. Shares have the following characteristics: opening price, day high, day low, previous day o price, close price, date of trading, total trade quantity, and turnover. The proposed model employs time series analysis to forecast a share price over a specified time period. The proposed will take into account an Indian stock exchange company called The National Stock Exchange of India Limited (NSE). The National Stock Exchange (NSE) is the Indian stock exchange entity. The NSE was the first exchange in India to provide a modern, up-to-date facility to investors spread across the country.

It is completely modern and equipped with all of the latest amenities, allowing investors to trade from anywhere in India. This is critical in reforming the Indian equity market in order to increase transparency, convergence, and efficiency in the capital market. The NSE's Common Index, The CNX NIFTY, is widely used by investors both in India and around the world. It facilitates the exchange, settlement, and clearing of equity and debt market transactions, as well as derivatives. This is one of India's most massive mazuma, currency, and index options trading exchanges in the world. A large number of domestic and ecumenical businesses are interested in the exchange. TATA, WIPRO, HDFC, and YES BANK LTD are among the regional companies. Among the pilgrim investors are only a few strategic holdings of the city party Mauritius Limited, five ecumenical holdings of Tigre. As suggested by , long-term memory networks (LSTM) are a type of recurrent neural network (RNN) capable of tackling linear problems. LSTM is a deep learning technique. Long term memory units (LSTM) are used to learn very long sequences. This is a more general version of the closed recursive system.

* 1. **Problem Definition**

Every day, the stock market is mentioned in the news. Every time it achieves a new high or low, you hear about it. If an effective algorithm could be established to anticipate the short term as well as Long term price of an individual stock, the rate of investment and business prospects in the stock market may grow. Many Supervised regression Algorithms like Linear, SVR, DecisiontreeRegressor etc., along with Artificial Neural Networks and Convolutional Neural Networks have been used in the past to predict stock prices, with an average error loss of 20-30%.

Also every day many students and traders emerges out for trading an amount in stock market, so as they can earn some profit out of it and meet their daily needs.

Throughout this study, we will explore if a LSTM, an Extension model of RNN may be used to forecast stock price with a lower proportion of inaccuracy.

* 1. **Objectives**
* Objective of this Project is to forecast the prices of the Stock, mainly NSE stocks, with high accuracy and less error using Stacked LSTM model.
* Also this Project provides the support of Technical Analysis to our Model to confirm if the forecasted of stock prices is highly accurate or not
* It gives a Long term and Short term overview of stock prices along with the trend of the stock i.e., Uptrend or Downtrend or Ranging Market
* It detects the Volatility, Price action, Momentum of the stock prices
  1. **Hardware and Software Tools Used**

**Software:**

* REQUIREMENTS

**Hardware:**

* Specifications of Hardware

# **Chapter 2**

**LITERATURE SURVEY**

The linear regression algorithm looks at two variables that define a single relationship. y = a1x+ a0 is the simplest form of the regression equation. It's simple to put into action. Overfitting can be avoided by reducing dimensionality and using cross validation. The SVM is a classification and regression training algorithm. With the use of regression techniques, Support Vector Machines (SVM) can address time-series domain problems. The kernel functions and mathematical programming approaches are two key components of SVM [1]. K-Nearest Neighbour is a supervised machine learning technique that can store and classify new case circumstances derived from previous ones. The algorithm finds the k closest observations for each datum and assigns the data point to the majority. The k-closest points are those that have the shortest Euclidean distance to the data point in question [2]. Linear Regression predicts the prices that are in a trend, in the sense either increasing linearly or decreasing linearly. But in reality stock values don’t just move up or down, they even move sideways which we call as ranging market. The linear prediction model is incapable of predicting a trend reversal. Furthermore, stock prices are influenced by a variety of factors that the algorithm ignores. SVM aims finding patterns in stock price volatility, but it fails to capture the long-term context-specific temporal connections that exist between stock prices [3]. As stock market being volatile most of the time, it creates noise (outliers). When there is more noise in the data, SVM underperforms because the target class overlap noise. In addition, determining hyper-parameters for the SVM is difficult. It causes the decision boundary to grow curvier and the variance to rise, increasing the probability of over-fitting [4]. Stock price movements may be forecasted using supervised learning classifiers based on financial index data, and their ability to anticipate stock price movement can be determined [5]. Long Short Term Memory (LSTM) networks are a form or an extension of recurrent neural network (RNN) that can solve involute or non-linear problems, and RNNs (Recurrent Neural Networks) are used to forecast stock values [6]. The most often used RNN architecture is the Long Short-Term Memory. In the hidden network layer, LSTM introduces a memory cell; a processing device that replaces conventional artificial neurons. With these memory cells, networks can effectively link memory and remote input in time, making it suitable for dynamically capturing data structure [7]. However these algorithms are effective, a model with a few percentage point higher accuracy, lower volatility, and faster prediction can significantly boost earnings.

# **Chapter 3**

**METHODOLOGY**

* 1. **Dataset**

The dataset will be obtained from the NSEpy python library. NSEpy is a library that allows you to extract historical and real-time data from the NSE's website. This Library strives to keep the API as simple as possible. The NSEpy Function get history returns a pandas data frame containing the price history of stocks/indices/derivatives.

* 1. **Architecture**

This model, which ran on GPU runtimes, was built using a variety of Python packages, as well as VScode and Google Colab. Our key libraries will be Pandas, NSEpy, NumPy, Matplotlib, TensorFlow, Keras, Sklearn, and datetime. Because it outperforms alternative designs like RNN, we choose the Stacked LSTM architecture as our foundation model.

LSTMs, or long-short-term memory networks, are a form of RNN that can learn long-term dependencies. They were first introduced by Hochreiter & Schmidhuber (1997), and they have since been refined and popularised by a number of people. They are now widely used and perform admirably in a wide range of settings. Long-term reliance is a problem that LSTMs were developed specifically to alleviate.

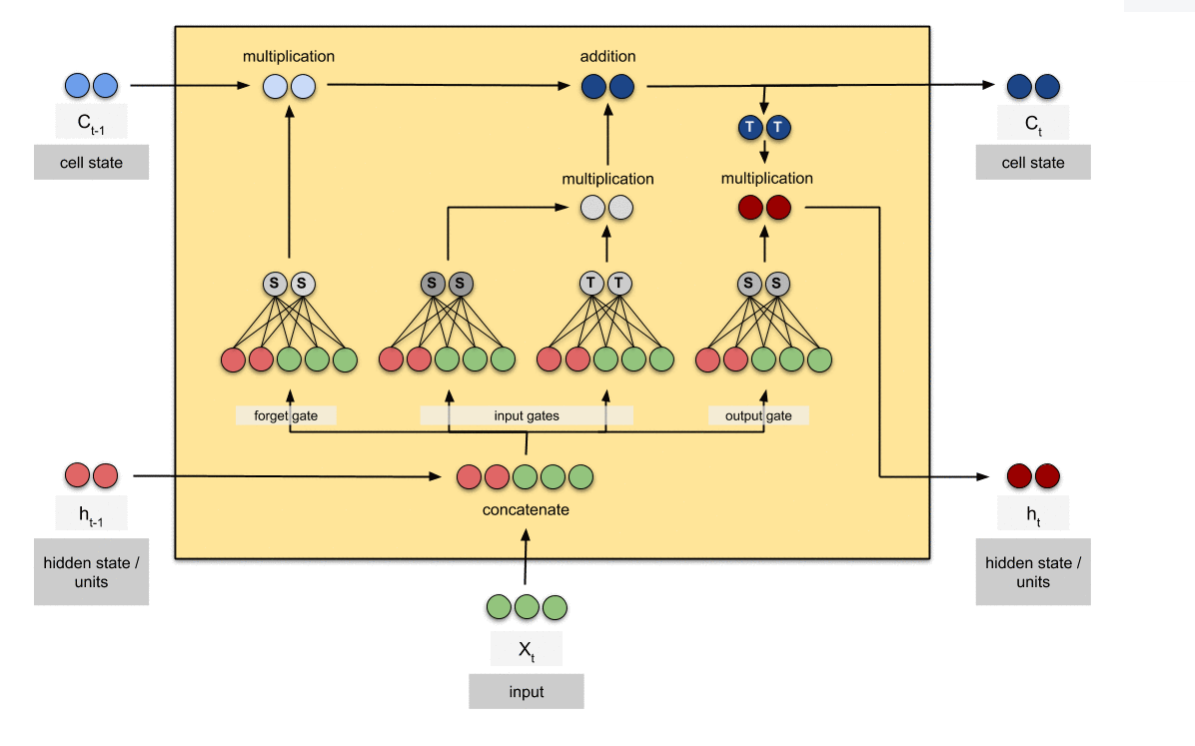
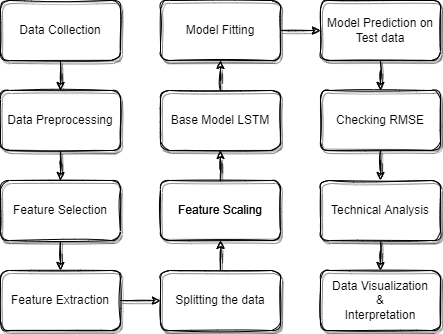


Fig 3.2.1 - LSTM Architecture

The first step of LSTM is to determine which cell state information should be discarded. The "forgot gate layer", a sigmoid class, makes this statement. For each cell state number, it checks the values ​​of ht-1 and xt and generates a number between 0 and 1. Number 1 means "completely preserved", while the number 0 means "completely garbage".

The next step is to determine in which cell state the new data should be stored. There are two factors to this. The "gate class", a sigmoid class, decides which variables to update first. Then, a vector t  of new candidate values ​​generated by a tanh layer will be used to update the state. In the next phase, we will combine these two elements to create a status update. Transition time from the previous cell state, Ct-1, to the current cell state, t  Because of the earlier stages, model already know what to do; all that's left is to do it. Model forget the things that model previously decided to forget, multiplying the previous state by the foot. Model then insert it into the formula because it \* t , scaled by the transformation of each state value, this is the most recent set of possible values. In the end, model have to decide what model is going to do. This output will be filtered and will depend on the state of our cells. To decide which aspect of the cell state should be generated, model first run a sigmoid layer. The state of the cell is then multiplied by the output of the sigmoid gate and sent through tanh (to force the values ​​to be between 1 and 1) to produce only the parts we want to produce.

* 1. **Sequence Diagram**

****

3.3.1 - Work flow

# **Chapter 4**

**TOOL DESCRIPTION**

This section gives a detailed description about the hardware tools and software tools involved in developing this system and how they are used.

* 1. **Hardware Requirements**
* Intel i3/i5/i7/i9 6th gen or above Processors
* Supported Discrete or Cloud GPU
* 6+ GB RAM
* Supports Windows 7 or later
* **<** 128 GB Storage
  1. **Software Requirements**
* Python version – 3.0 or later
* Jupyter Notebook / Any Python IDE / Any Code Editors with python extension
* TensorFlow version – 2.0 or later
* Keras version – 2.0 or later
* Python Libraries used – TensorFlow, Keras, NSEpy, Pandas, NumPy, Matplotlib, Scikit Learn, TA, Datetime

# **Chapter 5**

**IMPLEMENTATION**

ALGORITHM STEPS:

* **Importing all necessary python libraries:**

TensorFlow, NSEpy, and Matplotlib are the primary prerequisites. These are followed by other libraries such as Keras for implementing deep neural networks, Pandas and NumPy for pre-processing data, Scikit Learn for validating machine learning and deep learning models, and TA for technical analysis such as RSI, MACD, and BB, as well as Datetime for retrieving date and time.

* **Data Collection:**

Here, data is primarily acquired from NSEpy python library. NSEpy is a python library that allows you to extract historical and real-time data from the NSE's website. This Library strives to keep the API as simple as possible. The NSEpy Function get\_history returns a pandas data frame containing the price history of stocks/indices/derivatives.

* **Data Preprocessing:**

Here, data will be improved and cleaned up to prevent our model from fitting too closely or too loosely.

* **Feature selection and Extraction:**

This stage involves choosing the necessary or helpful variables from the dataset and processing those chosen variables such that a condensed version of the original features is created.

* **Splitting the data and Feature Scaling:**

Next, data will be divided into train and test datasets. The model is trained using a training dataset, and its performance against test data is tested using a testing dataset. This procedure aids in comparing the effectiveness of machine learning algorithms. The train split will then go through feature scaling, where data will be adjusted within a specific range to combat bias and outliers.

* **Model building and fitting:**

This step involves the creation of a Stacked LSTM model, which consists of multiple LSTM layers, Dropout layers, and one dense layer with hyper parameters. Because the model follows a sequence while training, it is a sequential model. Following that, the training dataset will be fed into the model along with various parameters such as epochs and batch size.

* **Model Evaluation:**

In this circumstance, the trained model will be tested with an unknown test split. This aids in determining the RMSE of our prediction; the lower the RMSE value, the better the model performs.

* **Technical Analysis, Data Visualization and Data Interpretation:**

***Moving averages Crossover strategy -***

We define a function where we find the crossover point of two moving averages, we can either use Simple Moving Averages or Exponential Moving Averages.

Two Moving averages are as follows:

* 100 MA – fast period
* 200 MA – slow period

When 100 days Moving Average crosses 200 days Moving Average upwards then it is interpreted as uptrend

When 100 days Moving Average crosses 200 days Moving Average downward then it is interpreted as downtrend.

***RSI strategy -***

RSI is a relative strength index, used to find the momentum of stock price. RSI is used to evaluate the rate and change of price movements of an investment, compare the magnitude of recent gains and losses over a specific time period. It's generally used to try to spot overbought or oversold levels in an asset's trading.

By default, the RSI is calculated for period of 14 days, that is, window size is 14

RSI is calculated using **RSIstep one** = 100 –

RSI reads the value from 1 to 100. When the RSI value is above 80% it is considered as Overbought and when the RSI value is less than 20% it is considered as oversold

Besides, using this strategy as an overbought or oversold detector, it is also used to provide bullish and bearish signals of price momentum.

***Bollinger Bands -***

Bollinger band is a technical analysis method used to measure the trend's strength.

Trend’s strength indicates the Volatility of the market.

Bollinger band consists of 2 bands and a Moving Average as follows:

* Moving Average (default): 20-days period
* Upper Band = 20+2\*S.D
* Lower Band = 20- 2\*S.D

Where S.D is standard deviation of 20days data (price)

95% of the stock prices move in between upper and lower bands. Hence, we define a function, when the price touches the upper band, we probably prefer selling as there is a high chance that the price start dropping down towards lower band and vice versa, just like over bought and over sold zones.

When the band width decrease, it is assumed to have a sudden volatility break out in short time towards up or down

To break through the upper zone, a significant price move is required. After a Bollinger Band verified W-Bottom, an upper band contact would suggest the commencement of an upswing. Prices rarely reach the lower band during an upswing.

To break through this lower zone, a significant price move is required. After a Bollinger Band verified M-Bottom, a lower band contact would suggest the commencement of an upswing. Prices rarely reach the upper band during a downswing.

***MACD strategy -***

MACD – Moving Average Convergence Divergence

MACD is a trend followed by momentum indicator that depicts the correlation between two price moving averages.

It basically that shows a macd line, signal line, macd histogram

* MACD line – (12 days EMA - 29 days EMA)
* Signal line: 9 days EMA of MACD line
* MACD histogram: MACD line – Signal line

Closing prices are used to calculate the Exponential Moving Averages (EMA)

Default values for setting MACD for long term and short-term analysis is 12, 26, 9. Depending on your trading style and objectives, values can be adjusted.

The MACD line fluctuates above and below the zero line, which is also known as the centreline.

We define a function which shows a BUY signal when MACD line crosses above the Signal line and shows a SELL signal when Signal line crosses above the MACD line.

***EMA Strategy -***

Exponential Moving Average is a trend indication is mostly used to find the uptrend or downtrend of a market.

We define a simple function which compares the High and Low prices of a stock with EMA

For long term it recommended to use 200 days of EMA and for short term its recommended to use 100 or 50 days of EMA.

When any number (recommended is 6 days) of days, High and Low prices are compared with the corresponding EMA value to detect whether Stock is in uptrend or downtrend

When 6 consecutive High prices > EMA and Low prices > EMA then it is depicted as Uptrend and similarly Low price < EMA and High price < EMA then it is considered as Down trend, neither of the above 2 cases it is considered as Ranging market

Based on Uptrend and Downtrend we decide whether to buy the stock or sell the stock

* Uptrend: Prefer Buying
* Downtrend: Prefer Selling
* Ranging: Prefer Holding

We integrate all mentioned Technical Analysis tools with the Stacked LSTM model to evaluate how well the model is predicting.

# **Chapter 6**

**RESULTS AND ANALYSIS**

This section is containing a description about the main findings of this proposed system with the figures, detailed explanation by comparing with previous studies and analysis.

* 1. **Result Discussion**

When a model with many LSTM layers is run, say one with 128 units and a dropout of 0.2, another with 100 units and a dropout of 0.3, a third with 60 units and a dropout of 0.4, and a fourth with 60 units and a dropout of 0.5, To create an ideal stacked LSTM model, the model ran 50 epochs with 69 batches.

|  |  |  |
| --- | --- | --- |
| **Training Error Metrics** | **SBI** | **JSW Steel** |
| RMSE | 18.285 | 11.239 |
| MAE | 13.586 | 8.0152 |

|  |  |  |
| --- | --- | --- |
| **Testing Error Metrics** | **SBI** | **JSW Steel** |
| RMSE | 17.252 | 40.021 |
| MAE | 13.168 | 32.041 |

**Training data prediction -**

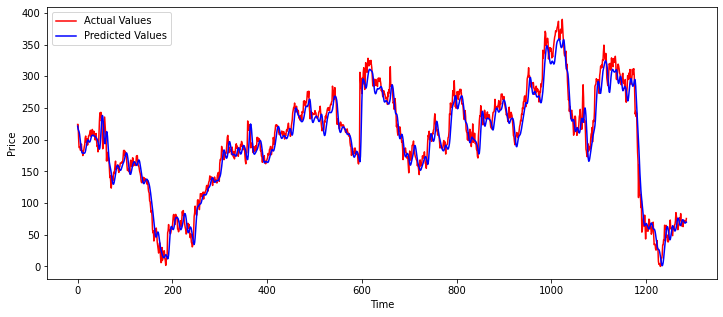
****

Fig 6.1.1 – Training Prediction line plot

**Testing data prediction**

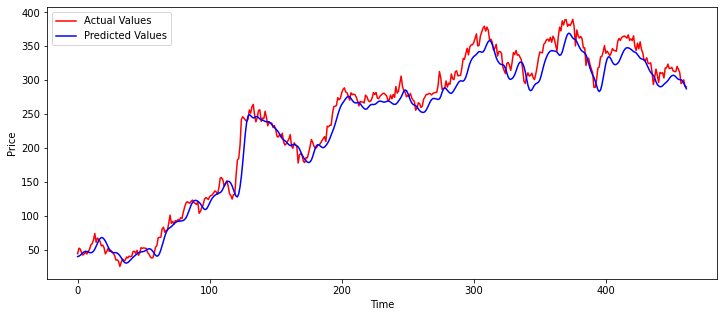
****

Fig 6.1.2 – Testing Prediction line plot

* 1. **Comparison with Previous Studies**

# **Chapter 7**

**CONCLUSIONS AND FUTURE SCOPE**

Explain the project work with respect to future work.

# **REFERENCES**

[1] Y. Xia, Y. Liu and Z. Chen, ”Support Vector Regression for prediction of stock trend,” 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering, Xi’an, China, 2013, pp. 123-126, doi: 10.1109/ICIII.2013.6703098.

[2] Teja Kodali, ”Using kNN Classifier to Predict Whether the Price of Stock Will Increase”. [online]. Available: <https://datascienceplus.com/knnclassifier-to-predict-price-of-stock/>

[3] Sai Krishna Lakshminarayanan and John McCrae, ”A Comparative Study of SVM and LSTM Deep Learning Algorithms for Stock Market Prediction”. [online]. Available: <http://ceur-ws.org/Vol-2563/aics41.pdf>

[4] D. Karmiani, R. Kazi, A. Nambisan, A. Shah and V. Kamble, ”Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market,” 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 2019, pp. 228-234, doi: 10.1109/AICAI.2019.8701258.

[5] Raut Sushrut Deepak, Shinde Isha Uday, Dr. D. Malathi, “Machine Learning Approach In Stock Market Prediction”, IJPAM 2017

[6] M. S. Hegde, G. Krishna and R. Srinath, "An Ensemble Stock Predictor and Recommender System," 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, 2018, pp. 1981-1985

[7] M. Roondiwala, H. Patel and S. Varma, "Predicting stock prices using LSTM," International Journal of Science and Research (IJSR), vol. 6, no. 4, pp. 1754-1756, 2017

**Sources –**

<https://towardsdatascience.com/making-a-trade-call-using-simple-moving-average-sma-crossover-strategy-python-implementation-29963326da7a>

<https://sentimentrader.com/blog/ultimate-guide-to-macd#:~:text=The%20optimal%20MACD%20setting%20was,ema%20of%20the%20MACD%20Line>

<https://medium.datadriveninvestor.com/3-simple-technical-indicators-for-long-term-investing-a100f02b9bed>

<https://blog.finology.in/investing/trend-analysis>

APPENDIX - I

**Example**

**SOURCE CODE**

**GitHub**: (link will be uploaded shortly, after implementing all the censorship and privacy measures.)

**Note**: This web application is still in development phase and shall be proprietary and may be

commercialized of the creators wish to do it.