IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Share Market Trend and Stock price prediction using Multivariate LSTM and supported by market Sentiments

Rajendra H Thakur, Dr. J. W. Bakal, Dr. Savita Sangam.

SCHOOL OF ENGINEERING AND APPLIED SCIENCES

University of Mumbai, Kalyan Sub Division, Kalyan,

1. Abstract

Obtaining accurate prediction of stock index and Stock prices significantly helps decision maker to take correct actions to develop a better economy. For many Trading houses and retail traders the inability to predict fluctuation of the stock market might cause serious profit loss. The main challenge is that we always deal with the dynamic market which is influenced by many factors in live market. They include political, financial and reserve occasions and unplanned events. Thus, stable, robust and adaptive approaches which can provide models have the capability to accurately predict stock index and trend of market are urgently needed. In this paper, we explore the use of Artificial Neural Networks (ANNs) and Support Vector Machines (SVM) along with Multivariate LSTM to build prediction models for the Nifty50 and Bank Nifty stock index as well as listed equity shares (stocks). Here we will do web scrapping for identifying market sentiments from different financial magazines and newspapers. The model will be a hybrid model where we will be using a combination of three different algorithms viz LSTM, Google Prophet and linear Regression. We will also show how traditional models such as multiple linear regressions (MLR) behave in this case. The developed models will be evaluated and compared based on a number of evaluation criteria for NSE as well as BSE. This study aims to significantly reduce the risk of trend prediction with machine learning and deep learning algorithms. Our motivation is based on the notion that financial planning guided by pattern discovery and prediction of stock index prices maybe more realistic and effective than traditional approaches, such as Autoregressive Integrated Moving Average (ARIMA) model

Index Terms— Tick Data, machine learning, market trend, prediction, stock market, technical indicators, Data mining, Time series Analysis, Typical Price.

2. Introduction:

Stock market prices or trends prediction making is a very challenging and difficult task. Prediction about stock market with high accuracy movement yield profit for investors and traders houses of the stocks as well as the retail traders like us can be benefited by the module. Because of the complexity of stock market ticker data (INR 0.05),

development of efficient models for prediction decision is very difficult, and the main thing is, it must be accurate and trustworthy. The trustworthiness can be achieved by testing the model 100s of time against multiple of stocks and the indices of Indian market such as NIFTY, BANK NIFTY, NIFTYIT, and NIFTYMETAL and so on.

In this project we will attempt to develop models for prediction of the Stocks and stock market and the trend whether it is Bullish or Bearish, to decide whether to buy/hold or sell the stocks using data mining and machine learning techniques. The machine learning technique like Neural Prophet, FLANN, LSTM, Naive Bayes, k-Nearest Neighbor (k-NN), Support Vector Machine (SVM), Artificial Neural Network (ANN) and Random Forest along with linear regression will be used for developing of prediction model. Technical indicators are calculated from the stock prices based on time-line data, Ticker price and it is used as inputs of the proposed prediction models. The main technical indicators used will be Moving Average (MA), Exponential Moving Average (EMA), Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD) and Williams %R. Ten years of stock market data will be used for signal prediction of stock and Market.

Based on the data set selected either for a specific stock or Sensex like Nifty50 or Bank Nifty, these models are capable to generate buy/hold or sell signal for stocks as an output.

The main goal of this project is to generate output signal to indicate Share or Stock market trend and stock prices as accurate as possible using data mining and machine learning techniques. Based on trend one can decide to buy, hold or sell the stocks. Accuracy of the derived signal using AI and algorithms will be improved by feeding market sentiments and current economic, news and social conditions. The market sentiments will be Scribed and derived by web scrapping the newspapers, blogs and news channels.

Algorithms Proposed	• Recurring Neural Network (RNN) • Long Short-Term Memory (LSTM) • Artificial Neural Network (ANN) • Fuzzy logics • Linear Regression • Random forest • Google Neural Prophet
Factors affecting Share market prices and trends	• Current Events • Earnings • Economy • Expectations • Emotion • Supply and demand • Company related factors • Investor sentiment / Market Sentiments • Interest rates • Politics • Current events • Natural calamities • Exchange rates • BSE website • Trading View site • Yahoo Finance and • Google
Interface used	• Yahoo API • Google Finance • NSE India

Historic Data from	• NSE India website • BSE website • Trading View site • Yahoo Finance and • Google
Project Language used	• Python
Market Sentiments news will be taken from	• https://www.tickertape.in/market-mood-index • https://youtu.be/rIq_X4gKXhI • https://ycharts.com/indicators/us_investor_sentiment • https://in.tradingview.com/chart/NmnCy9fz/ • https://www.moneycontrol.com/news/business/stocks/ • https://munafasutra.com/page/StockMarketNEWS • Angel one site
The Main Objectives of this study:	• Study the computational systems available for stock price prediction. • Study the computational systems available for stock price prediction. • Study the current Machine learning techniques used for stock prediction and describe their limitations. • Study the critical intrinsic and macro-economic features that directly affect the stock price. • Develop a model for prediction of trends in Stocks, Nifty 50 and Bank Nifty. • Develop stock prediction Model based on LSTM, ANNs and market Sentiments • Develop Stock portfolio model using Machine learning clustering techniques • Design a Computational Decision Support system to predict stock prices, market trend and future prices. • Describe the outcomes and implications of the study • Listing out any limitations of the designed model.
	•

3. Problem Statement

After doing extensive literature review, we find that there are many AI, ANN, computational and machine learning models used in stock price prediction, many of these models have focused on determining the daily movements of stock prices and mostly rely on technical indicators and ignore the fundamentals driving the stock price. We believe that there are no comprehensive machine models currently to predict long-term movements of stock prices (except Google Neural Prophet) which shall be very useful for individual stock investors. Also, the current Stock Portfolio construction, selection and optimization are mostly manual and the use of machine learning in this financial market area is yielding limited results. The portfolio building must be using FII and DI data which can be optioned by web scrapping from their portal.

Based on the available literature in this financial domain, LSTM, ANN and Google Neural Prophet are the best approach for stock price prediction supported by market sentiments after extensive comparative analysis between various computational models and find that choosing the correct input feature set to the ANN (RNN) consisting of fundamental financial parameters can improve the accuracy of stock price and Index prediction and we conclude that there is a strong need for developing a Machine learning model based on ANNs in support of market sentiment as an input with intrinsic features, as input set and also generate and select stock portfolios with Risk-Reward Ration analysis done by machine learning algorithms.

These algorithms work on provided training data taken from historic data and refined using testing with again

historic data. The historic data of market is static and does not have emotions and feeling of the market sentiments as it is only ticker data.

The news, emotions and market sentiments provide senses and dynamics to this ticker data to predict the accurate output like trend and stock prices.

Our model will use hybrid mode to predict accurate data using combination of different AI models and algorithms like LSTM, ANN and Neural Prophet and will use the market sentiments.

"It is said that Price is God" (Bhaav hi Bhagwan Hai") in the Share market and it respond to the market sentiments and live events.

1. Literature Review and Gaps

A detailed survey and study of literature from all over the world was done on current stock prediction computational models and their methods and usage of Decision Support systems in financial domain. Below some of the widely cited papers are listed in this domain.

There are many limitations in the current machine learning computational models including LSTM and ANNs used for stock price prediction because most of them use only technical historical data and ticker prices along with some technical indicator.

All these indicators are lagging indicators as they all are derived from price action and price action is always in present state and live in accordance of market. It is not predictable as price action depends on not only on historic data bur also current events and market sentiments.

Stock Market Prediction Using Artificial Neural Networks (September 2012) by Bing Yang uses ANN and Deep Neural Network to predict the Shanghai Stock market using historical data using Adam algorithm. The accuracy of their result has gone maximum to 72% to 73% of the realistic market.

Indian stock market prediction using artificial neural networks (ANN) on tick data by Dharmaraj, Abhishek and Vineet use almost same method ANN with Deep learning and they have found that the accuracy can go up to 80% to 87%; but still, it has limitation due to market sentiments.

A detail literature survey leads us to the conclusion that currently there is no any prediction model available which can predict share market trend and stock prices more accurately which could be nearer to real prices and the main reason is that all the models are working on static historical data.

4. Methods and Models

From the study of plenty of literature the main two approaches has been found for stock market or are there prediction namely fundamental analysis and technical analysis.

Fundamental Analysis and Technical Analysis, evaluation and selection are described below. The main four components of the model consist of:

1) LSTM and ANN based Stock prediction Model
2) Wilcoxon Norm
3) Google neural-Prophet
4) News Based Approach

I. Algorithms

- RNN - Recurring Neural Network (RNN) Algorithms
- LSTM - Long Short-Term Memory (LSTM) Algorithms
- ANN - Artificial Neural Network (ANN) Algorithms
- Fuzzy logics
- K-NN - k-nearest neighbors (KNN) Algorithms
- Linear Regression Algorithms
- Random forest Algorithms

II. Existing System

There are many prediction models available for stock price forecasting and trend identification; but most of them are used independently and isolated from real scenario.

Many of them are trying to be close to the real prices of stocks and trend; but lags in capturing market sentiments and investor moods. Also, sudden events happening during trading period are not being captured by such systems as they work on static historic data.

This is generally accomplished with the assistance of Genetic Algorithms (GA) or Artificial Neural Networks (ANN's), however this neglect to catch connection between stock costs of world Sensex conditions. Furthermore, in the Existing framework, sliding window calculation is being used. The Sliding window is the best approach to rebuild over a period arrangement data-set as a directed learning for the module. The utilization of earlier time ventures to anticipate whenever step is called sliding window technique.

Even the Google Neural Prophet extend the forecasting using historic Ticker data for many years is not able to capture the current trend which affects due to market sentiments.

III. Proposed System

Here we propose a new model for forecasting stock prices and stock trend using multiple algorithms, supporting each other by integrating their outputs and providing support from market sentiments and world news about world Sensex's.

This model forecast the prices of stocks (Equities) and also predicts the trend in such a way that the predicted price and trend is almost same as real prices and in same direction.

Market moves in the same direction as predicted and accuracy is high unless sudden event impacted which is not captured or updated in real time.

- The primary model used is LSTM

LSTM (Long Short-Term Memory) algorithm is used to provide efficient stock price prediction. LSTM algorithm is comparatively faster than other algorithms. It provides more perfect and efficient prediction and also suites best for non-linear data. These LSTM modules play a crucial role in machine learning.

The output results of indicators and oscillators are then introduced to the input of LSTM artificial neural network. Here the fundamental analysis module has been supported and aided by an algorithm that checks users' activity via Google Trends API.

The below diagram of LSTM represents an overview of my proposed system. The stock prices details from the ticker database are fed to the LSTM module, to train the module and predict the stocks.

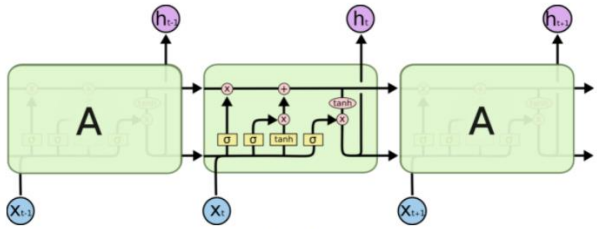

Fig. 4.1 – LSTM Primary module

These are the three parts of an LSTM cell are also known as gates. The first one is called Forget gate, and the second part is known as the Input gate and the last one is the Output gate.

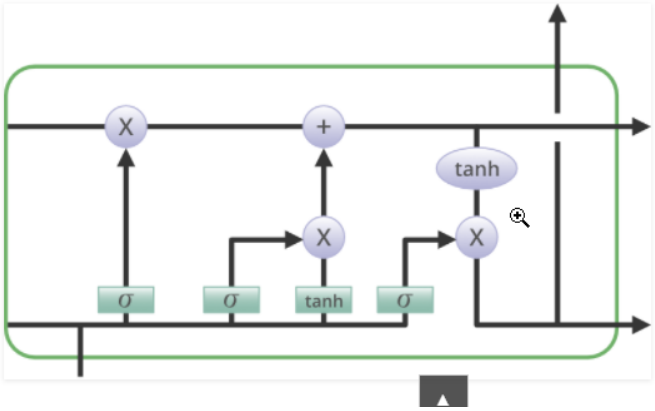

Fig. 4.2 – Process flow in LSTM system

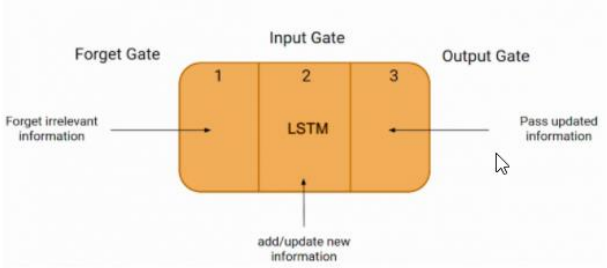

Fig. 4.3

Just like a RNN module, an LSTM module also has a hidden state where H (t-1) represents the hidden state of the previous timestamp (t-1) and Ht is the hidden state of the current timestamp of that level. In addition to that LSTM module also has a

cell state represented by C (t-1) and C (t) for previous and current timestamp respectively.

Now here, the hidden state is known as Short term memory and the cell state is known as Long term memory.

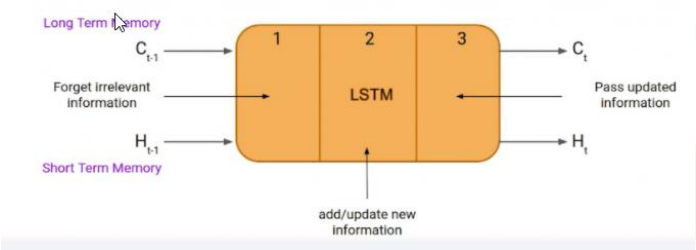

Fig. 4.4 – LSTM information flow.

Here it is interesting to note that the cell state carries the information along with all the timestamps it has.

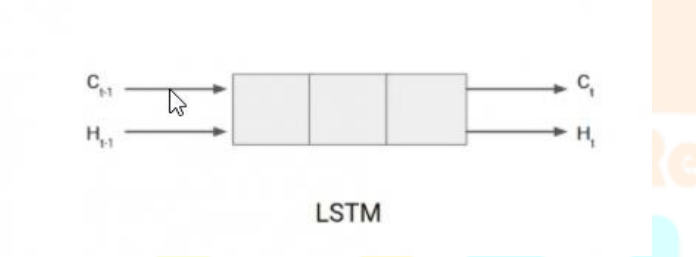

Fig. 4.5 – LSTM System

Objective

The LSTM is a special kind of Recurrent Neural Network which is capable of handling long-term dependencies.

IV. Architecture and working of an LSTM network module.

Introduction:

Long Short-Term Memory Network is an advanced RNN, a sequential network, which allows information to persist. LSTM is capable of handling the vanishing gradient problem faced by RNN. A recurrent neural network is also known as RNN and is used for persistent memory.

Let's say while watching a movie you remember the previous scene or while reading a novel you know what happened in the earlier chapter. Similarly, the RNNs work; they remember the previous information and use it for processing the current input. The shortcoming of RNN is, they cannot remember Long term dependencies (memories) due to vanishing gradient. LSTMs are explicitly designed to avoid the long-term dependency problems in the system.

LSTM Architecture

At a high-level LSTM works very much similar to an RNN cell. Below is the internal functioning of the LSTM module. The LSTM consists of mainly three parts, as shown in the image below and each part performs an individual function assigned to it.

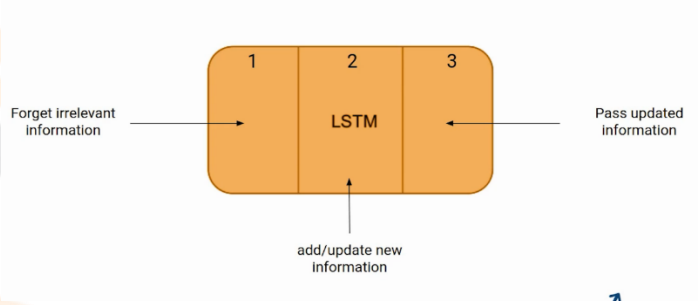

Fig. 4.6 – LSTM architecture Block Diagram

The first part of LSTM chooses whether the information coming from the previous timestamp is to be remembered or is irrelevant and can be forgotten. In the second part, the LSTM cell tries to learn any new information from the input fed to this cell. And at last, in the third part, the cell passes the newly updated information from the current timestamp to the next timestamp.

These are the three parts of an LSTM cell which are known as gates of LSTM. The first part of the LSTM is called Forget gate of LSTM, the second part is known as the Input gate of LSTM and the last one is the Output gate of the system LSTM.

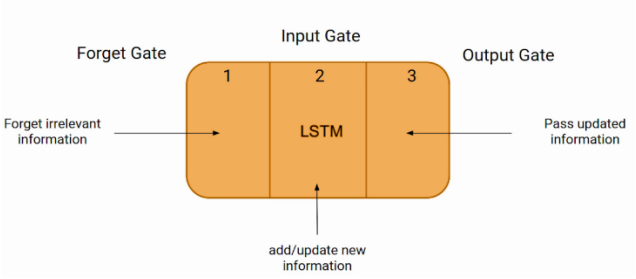

Fig. 4.7 – LSTM Components

Technically, just like a normal RNN, the LSTM also has a hidden state layer where H (t-1)

represents the hidden state of the previous timestamp and Ht is the hidden state of the current timestamp. In addition to hidden state layers LSTM also have a cell state represented by C (t-1) and C(t) for previous and current timestamp respectively.

In this LSTM module the hidden state is known as Short term memory and the Cell state is known as Long term memory of LSTM. Please see the following image.

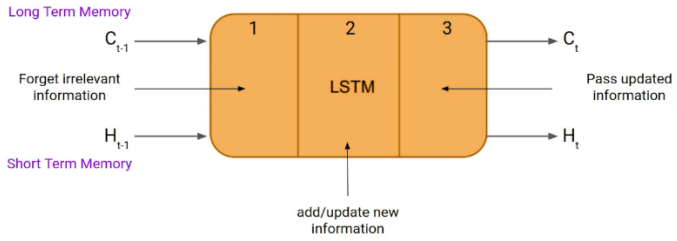

Fig. 4.8 – LSTM System module

Here it is interesting to note that the cell state of LSTM carries the information along with all the timestamps like previous, current and next.

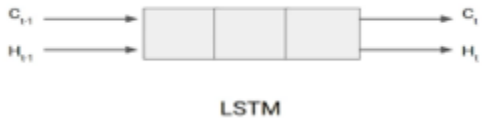

Fig. 4.9 – LSTM Symbolic Diagram

Kumar is a nice person.

Satish on the other hand is an evil person.

Now let's consider an example to understand how LSTM works. Let's take two sentences separated by a full stop. The first sentence is "**Kumar is a nice person**" and the second sentence is "**Satish, on the Other hand, is not nice person**". Now it is very clear that, in the first sentence we are talking about Kumar and as soon as we encounter the full stop (.) we started talking about Satish.

As we move from the very first sentence to the second sentence, our network has to realize that we are no more talking about Kumar. Now our subject is Satish. Here, the Forget gate of the network allows it to forget about the first subject Kumar. Now let's understand the roles played by these gates in our LSTM architecture.

1) Forget Gate:

Here, in a cell of the LSTM module, the very first step is to decide whether the module state should keep all the information from the previous timestamp C (t-1) or forget and leave it behind. Below is the formation of forget gate.

The previous information that is no longer useful in the cell state is removed with the forget gate. Two inputs x_t (input at the particular time t) and h_t-1 (previous cell output (t-1)) are fed to the gate and multiplied with weight matrices followed by the addition of bias. The resultant output is passed through an activation function which gives a binary output. If for a particular cell state the output is 0, the piece of information is forgotten and for the cell whose output is 1, the information is retained for future use.

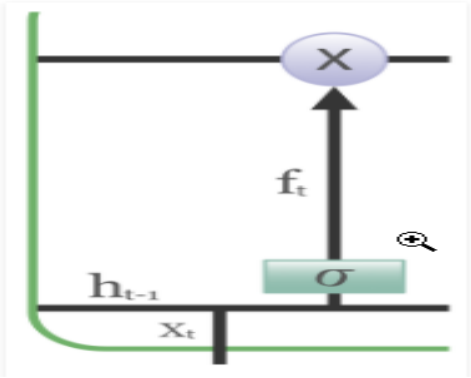

Fig. 4.10 – Forgot Gate of LSTM

Forget Gate:

- $f_t = \sigma\,(x_t * U_f + H_{t-1} * W_f)$

Now, let's try to understand the equation,

Here

Xt: input to the current timestamp.

Uf: This is weight associated with the input

Ht-1: This is the hidden state of the previous timestamp

Wf: This is the weight matrix associated with hidden state

A sigmoid function is applied over it. That will make f (t) a number between 0 and 1. This forgets gate ft of LSTM is later multiplied with the cell state of the previous timestamp (t-1) as shown below.

- **Forget Equation and Remember Equation**

$C_{t-1} * f_t = 0$ ...if $f_t = 0$ (forget everything)

$C_{t-1} * f_t = C_{t-1}$ ...if $f_t = 1$ (forget nothing)

2) Input gate:

An addition of useful information to the cell state is done by the input gate of LSTM. First, the information is regulated or channelized using the sigmoid function and filter the values to be remembered similar to the forget gate using inputs h_t-1 and x_t.

A vector is created using tanh function which gives an output ranging from -1 to +1, which contains all the possible values from h_t-1 and x_t. And at last, the values of the vector derived and the regulated values of cells are multiplied to obtain the useful information

Input Gate:

- $i_t = \sigma(x_t * U_i + H_{t-1} * W_i)$

Here,

Xt: An Input at the current timestamp t.

Ui: The weight matrix of input.

H(t-1): A hidden state at the previous timestamp.

Wi: The Weight matrix of input associated with hidden state.

Again, we have applied sigmoid function over it. As a result, the value of input gate, i at timestamp t will be between 0 and 1.

New information

- $N_t = \tanh(x_t * U_c + H_{t-1} * W_c)$ (new information)

The Nt won't be added directly to the cell state. Instead, it comes with the updated equation

$C_t = f_t * C_{t-1} + i_t * N_t$ (updating cell state)

Here, (Ct-1) is the cell state at the current timestamp (t) and others are the values we have calculated previously.

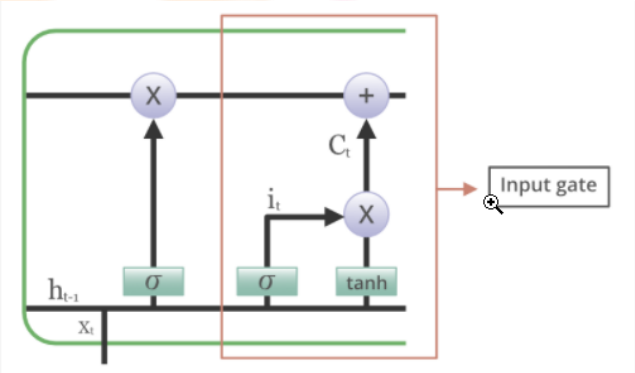

Fig. 4.11 – Input Gate of LSTM

3) Output gate:

The operation and action of extracting useful information from the current cell state to be presented as output is done by the output gate. Initially, a vector is generated by applying tanh function on the cell. Then, the information is streamlined and regulated using the sigmoid functions and filters it by the values to be remembered using inputs h_t-1 and x_t. And at the final stage, the values of the derived vector and the formatted and regulated values are multiplied to be sent as an output and input to the next cell of LSTM.

Output Gate:

- $o_t = \sigma(x_t * U_o + H_{t-1} * W_o)$

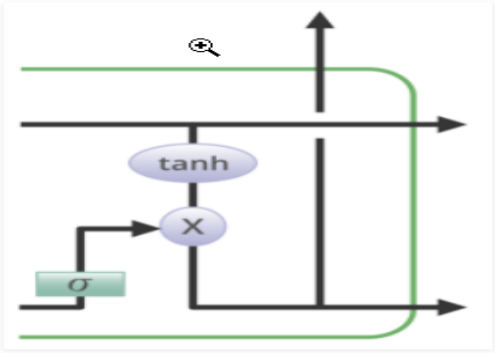

Fig. 4.12 – Out Put Gate of LSTM

How it works?

- LSTM calculation is comparatively quicker than Sliding window calculation.
- LSTM gives increasingly proficient and reliable forecast.
- LSTM suites best for non-direct information

A. Proposed System Design:

a) Download index/share value information

The NSE India and BSE India website hold and keep all historical ticker data prices for each and every stock. The offer price is the price of an individual organization. It comprises of date, most elevated price, least value, close value, number of bids and offers in the exchanges.

This information is also available in Yahoo finance, Google finance and many other websites who deals in economy market.

We use mostly Yahoo finance as data provider as it has facility of proving data in different time frames like 5 minutes, 15 minutes 30 minutes, 60 minutes and Daily closing price.

b) Data preprocessing

Information preprocessing gets ready crucial information for further preparing. It is utilized in database driven applications, for example, client relationship the board and principle-based applications (Neural Systems). The information preprocessing is an essential step in advance before preparing the information. The tick data is imported from Yahoo Finance and used for further processing. The information preprocessing has the accompanying advances:

c) Data exploration

The imported data was analyzed and checked for any anomalies or Null data.

d) Data cleaning

The imported data was cleaned and prepared in required format for dates and numbers.

Also checked for any missing values.

e) Data integration

This is the process of combining multiple sources into a single integrated dataset. The Data integration process is one of the main and most important components in data management.

- Schema integration: This involves Integration of metadata (a set of data that describes other data) from different sources.
- Entity identification problem: This process involves identifying entities from multiple databases. For example, the system or the use should know Script name (Equity name) of one database and Ticker price of another database belongs to the same entity (Stock).
- Detecting and resolving data value concepts: The data taken from different databases while merging may differ as per their formats. Like the attribute values and formatting formats from one database may differ from another database. For example, the date format may differ like “MM/DD/YYYY” or “DD/Mon/YYYY” or “MM/DD/YY”

f) Data transformation

- The changes made in the format or the basic structure of the dataset is called data transformation. This step can be simple or complex based on the requirements and the data type available. There are some ready methods available in data transformation where we convert the data format as per our requirement.
- Smoothing: With the help of algorithms, we can remove noise and some type of garbage from the dataset and helps in listing out and knowing the important features of the dataset. By smoothing we can find even a simple change that helps in prediction using the dataset.
- Aggregation: In this method, the data is stored and presented in the form of a summary in proper dataset format. The data set which is from multiple sources is

integrated into a single dataset with data analysis description. This is an important step since the accuracy of the data depends on the quantity and quality of the data prepared or available.

- Discretization: The continuous data and dataset is split into intervals. Discretization reduces the data size which becomes easy to do operations on it.
- Normalization: In this method, data is scaled so that it can be represented in a smaller range. Example ranging from -1.0 to 1.0. This help in better calculations.

g) Data reduction

The imported data was having plenty of information which was not required for predictions as well as calculation, so the unwanted data was removed and the crucial data only kept for the module.

h) Train the model

This is important part of any AI network or model. The designed model must be trained to learn the significant information of the dataset and the historical data which will be used to predict the future values. The way toward preparing includes furnishing a Machine learning calculation with preparing information. The learning calculation discovers designs in the preparation information that delineate info information credits to the objective. This progression of dataset is exceptionally significant for foreseeing the information. Also, the used size or number of records for training.
Here we have used different sets of data for training the module like for 1D time frame we have used 100 days for training and 100 days for testing. Whereas for 5 minutes time frame we have used 100 sets of 5 minutes values for training and 100 sets of 5 minutes modules for testing. Also same for 15 minutes time frame.
The module is designed in the view of Options and futures of stock index trading.

i) Run LSTM

LSTM module implies Long Short-Term Memory. This calculation has mainly three doors in particular Input entryway of module: Takes information and procedures the information. Yield door: Gives the outcome. Disregard entryway: Forgets the superfluous information. There are many associations into and out of the LSTM entryways, a couple of which are repetitive and recurring. The loads and burdens of these associations, which should be mastered amid preparing, decide how the entryways/ inputs work. The information door controls the degree or durability and strength to which another esteem streams into the cell, the overlook entryway controls the degree to which esteem stays in the cell and the yield entryway controls the degree to which the incentive in the cell is utilized. Here we use the built-in libraries of Python and we provide the required inputs.
The only drawback here is since I have taken much bigger size of data for training and testing, it takes little more time for prediction calculation.

j) Visualizing the predictions

When the LSTM module shows, it is fit to the prepared information, it very well may be utilized to anticipate the finish of-day stock closing cost of a discretionary stock or index. This expectation can be performed in two different ways:

Historic ticker data value graph:

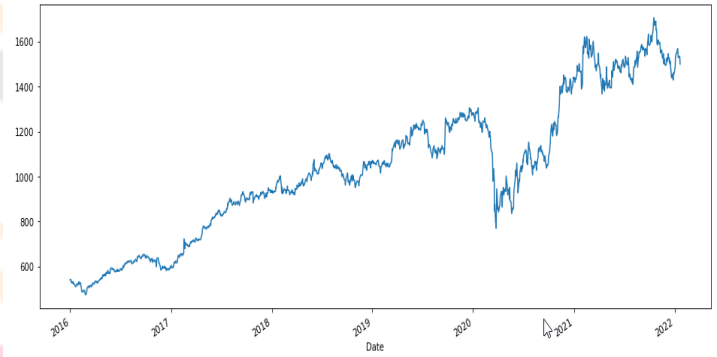

Fig. 4.14 - Actual closing prices over the time

Static – a straightforward and less exact strategy where the model is fit on all the preparation information. Each new time step is then anticipated each one in turn from test information.

Dynamic – an intricate, increasingly exact methodology where the model is refit for each time venture of the test information as new perceptions are made accessible.

Predicted ticker data value graph:

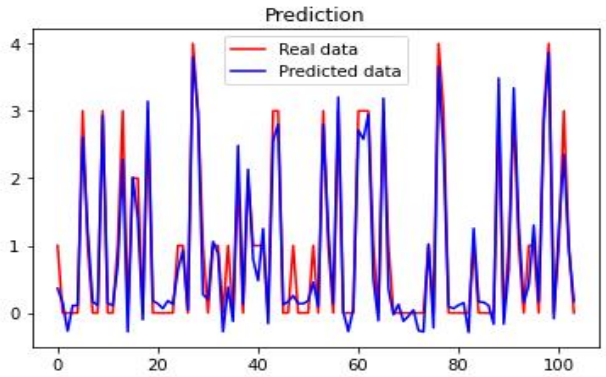

Fig. 4.13 - Actual and Predicted closing prices for limited data

The accurate and the exactness of the expectation model would then be able to be evaluated vigorously utilizing the RMSE (Root Mean Squared Error) metric. This is because of the way that neural systems when all is said is done (counting LSTM) will in general

give diverse outcomes with various beginning conditions on almost similar information.

Predicted ticker data value graph:

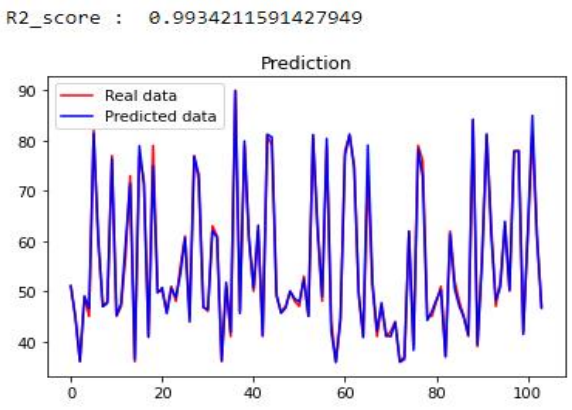

Fig. 4.15 - Actual and Predicted closing prices

At that point where rehashing of the model development and forecasting a few times (with various beginning conditions) and after that take the normal RMSE as a sign of how well the arrangement of it would be relied upon to perform on stock information That is, we will contrast our expectations and real patterns in stock value development that can be deduced from verifiable information. The historical datasets or ticker data will be used from either Google finance or Yahoo finance.

Fig. 4.16 – Overview of proposed system

5. Result

The results of this module are far better than any existing module as it has taken market sentiments as input to impact the forecasting of stock prices and indexes. This has brought the result of prediction as accurate as real.

As we go for small time frame, we get better results.

Accuracy: 98.306960464792

Fig. 5.1 – Actual and Predicted closing prices of BankNifty

If we are going for small time frame like 5MTF, 15 MTF or 1 HTF, we get very good results as it gets trained on smaller values and can predict in better way.

a) 1 Hour Time frame

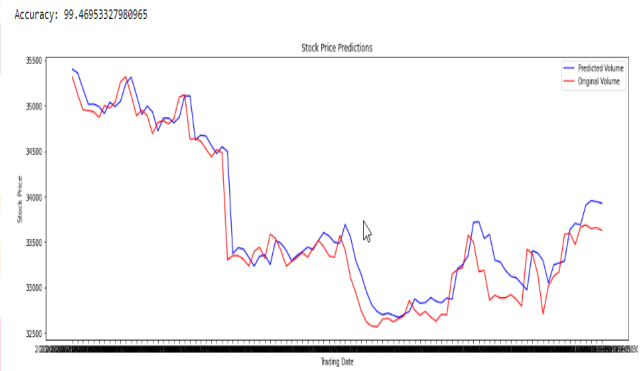

Fig. 5.2 - 1HTF Actual and Predicted Closing price

b) 15 minutes Time frame

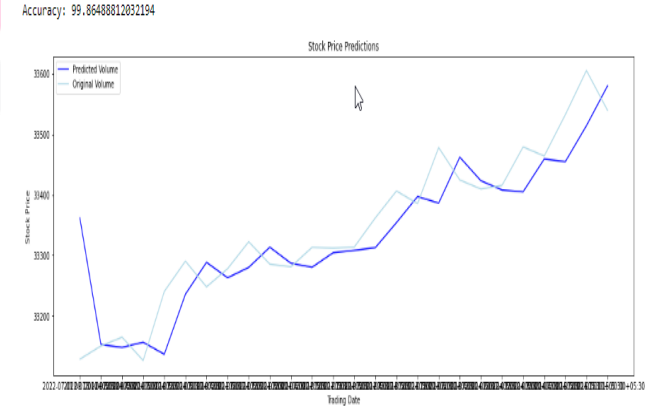

Fig. 5.3 – 15MTF Actual and Predicted Closing price

c) 5 minutes Time frame

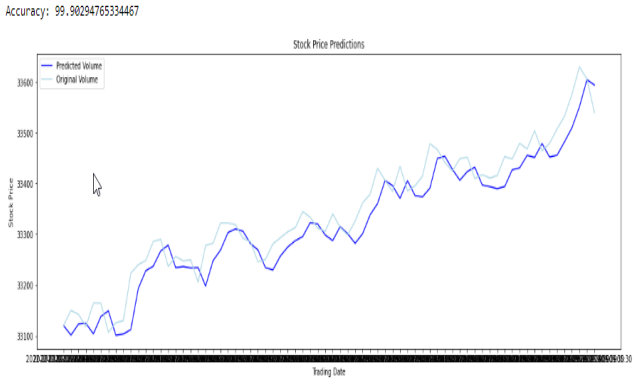

Fig. 5.4 – 5MTF Actual and Predicted Closing price

6. Conclusion and Future Works

Choice to purchase or move a stock is extremely muddled since numerous components can influence stock cost and market trend. This work shows a novel methodology Forward Forecast of Stock Price utilizing LSTM Machine Learning Algorithm. The sensitivity and uncertainty of securities exchange exchanging is developing quickly, which is urging specialists to discover new strategies for the expectation utilizing new systems. The results of comparison between Long Short-Term Memory (LSTM) and Sliding window, show that LSTM has a better prediction accuracy. The Guaging system isn't just helping the scientists however it likewise helps financial specialists and any individual managing the stock exchange. So as to help foresee the stock records, a Guaging model with great precision is required. In this work, we have utilized a standout among the most exact estimating innovation utilizing Recurrent Neural Network and Long Short-Term Memory unit which helps financial specialists, experts or any individual keen on putting resources into the share trading system by giving them a decent learning of things to come circumstance of money markets.

As it's well known, the stock market is a reflection of human emotions, though analysis has it's own limitations.

Future Work:

An extension of this stock prediction system would be to feed data from social media for sentimental analysis. It can be linked with the LSTM to train weights and further, improve accuracy. We will be also comparing the out puts using linear regression and random forest and combine result with be generated. Next part is to integrating this data into Google's Neural Prophet which is a type of hybrid model satisfying industry standard, to forecast stocks and trend for long duration for portfolio building. Thus, predicting the nearest possible values of actual data and exact trend of share market.

7. Data Availability

The dataset used in this study is open access and available at Yahoo finance, Google Finance, NSE Website as well as many financial sites.

8. Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

9. References:

1. J. T. Yao and C. L. Tan, ―Guidelines for financial prediction with artificial neural networks, ‖ in *Proc. 8th National Conference on Artificial Intelligence*, 2009.
2. Sentiment classification of social media reviews using an ensemble classifier

   By - Savita Sangam, Subhash Shinde

3. Zarandi, M. H. Fazel, B. Rezaee, I. B. Turksen, and E. Neshat, ―A type-2 fuzzy rule-based expert system model for stock price analysis, ‖ *Expert Systems with Applications*
4. Q. Bo, and K. Rasheed, ―Stock market prediction with multiple classifiers, ‖ *Applied Intelligence*
5. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling by Has¸im Sak, Andrew Senior, Franc¸oise Beaufays
6. NeuralProphet: Explainable Forecasting at Scale by Oskar Triebe a,1,_, Hansika Hewamalagec, Polina Pilyuginad, Nikolay Laptevb, Christoph Bergmeirc, Ram Rajagopala
7. Machine Learning for Stock Market Prediction with Step-by-Step Implementation by Analytics Vidhya
8. Forward Forecast of Stock Price Using LSTM Machine Learning Algorithm by Kavitha Esther Rajakumari, M. Srinivasa Kalyan, and M. Vijay Bhaskar
9. FLANN Based Model to Predict Stock Price Movements of Stock Indices by Chinmoy Mohapatra
10. Stock Market Prediction Using Artificial Neural Networks (September 2012)
11. A robust technique for exchange rate prediction using Wilcoxon Norm (January 2012)

12. On the development and performance evaluation of a multi-objective GA based RBF adaptive model for the prediction of stock indices (December 2013)
13. Stock Indices Prediction Using Radial Basis Function Neural Network (December 2012)
14. A Comparison between Regression, Artificial Neural Networks and Support
15. Vector Machines for Predicting Stock Market Indexes (November 2018)
16. Saudi Arabia stock prices forecasting using artificial neural networks
17. Indian stock market prediction using artificial neural networks (ANN) on tick data
18. Forecasting Stock Exchange Market using hybrid Neuro Fuzzy Model (Dec 2019)
19. Stock Price Prediction using Technical, Fundamental and News based Approach (March 2019)
20. Efficient Machine Learning Techniques for Stock Market Prediction (November 2013)
21. Design of a Decision Support System for Stock Price Prediction Using Artificial Neural Networks (January 2018)