Indian Institute of Information Technology, Allahabad

Mini Project

Instructor: Dr. Ranjana Vyas

Analyzing use of different LSTM variants for Stock Price Prediction using Multivariate Time Series Dataset

<u>Group Members</u>

Shreyas Gupta	IIT2019102
Sarthak Maheshwari	IIT2019117
Garvit Chittora	IIT2019142
Mrinal Bhave	IIT2019152
Varun Tiwari	IIT2019154

Abstract

Prediction of stock prices has always been a matter of interest and open challenge for investors, financial institutions and researchers. However, given the volatile and unpredictable nature of these prices, this prediction has never been easy and tons of research has been done to build models for the same. This project attempts to explore the capabilities of the various variants of Long Short Term Memory(LSTM), a type of Recurrent Neural Network, in the prediction of future stock prices from NIFTY-50.

Introduction

The stock market is a public marketplace where shares of public companies are bought and sold everyday. It lets companies raise capital from the public and provides a way for the public to invest their money in equity.

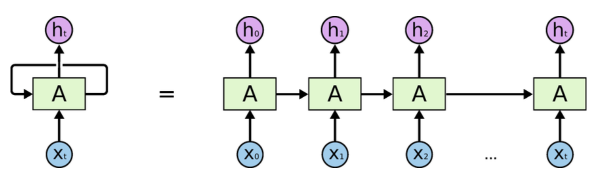
Prediction of the future movements of the stock market can be very lucrative for investors and institutions. The two distinct trading philosophies for stock market prediction are fundamental and technical analysis :

- Technical analysis focuses on the study of market movements through the use of charts and past price data.
- Fundamental analysis concentrates on the intrinsic value of a stock by analyzing financial statements of companies, current events and future plans.

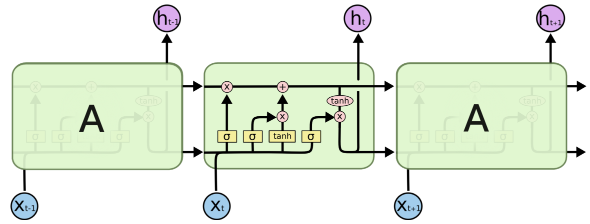
Our prediction model will focus entirely on technical analysis and won't have any information about the company that fundamental analysis utilizes.

Reliability of stock market prediction is important as it is very sensitive to the economy and can directly lead to huge financial loss/gains. Reliable predictions also help to keep the market in control and less volatile.

A specific way of analyzing a sequence of data points collected over an interval of time is termed as Time Series Analysis and many models have been developed for the same. The most basic being Recurrent Neural Network, which feeds on sequential data and produces output based on output from the previous cell.



Unfortunately, RNNs don't work for long term information because of its sequential nature where in the long term, memory can be easily corrupted due to constant large updates. To solve this issue, LSTM networks were explicitly designed to learn long term dependencies.



An LSTM chain not only keeps track of the output from the last module but also the long term cell memory. This is achieved by using 3 layers in each LSTM cell, they perform the following functions:

1. Decide whether we want to keep or forget previous cell memory
2. Decide what new information to update in cell memory
3. Decide what to output for this module

LSTM further has variants: Classic, Stacked and Bidirectional.
In Stacked LSTM, multiple LSTM cells are vertically stacked and each input in the sequence goes through a series of hidden layers to decide the output.
In Bidirectional LSTM, the LSTM chain is duplicated and the input sequence is processed in reverse in the second chain. This makes it so that each cell now has both past and future information to make a decision.

In this project we will explore the results of predicting stock prices using the different variants of LSTM.

Literature Survey

We review previous efforts to predict stock market prices using different machine learning models and how their results compare when applied on different types of data.

[1] evaluated the performance of bidirectional and stacked LSTM for stock market prediction. The performance of the models were also compared with a shallow and a unidirectional LSTM. It concluded that the bidirectional and stacked LSTMs had better performance for short term prices opposed to the long term prediction results.

[2] suggested that supervised learning classifiers be used to forecast stock price movement based on financial index data, and determine their ability. In the financial market computational analytical approaches have been portfolio modeling. The usage of SVM methodology has been shown in the paper and also shown that tactical methodologies can be applied to predict the stock prices.

[3] suggested a model which considers the historical equity share price of a company and applies LSTM. The proposed approach considers available historical data of a share and it applies prediction on a particular feature. The features of shares are opening price, day high, day low, previous day price, close price, date of trading. The proposed model uses the time series analysis in order to predict a share price for a required time span.

[4] compares the accuracy of autoregressive integrated moving average ARIMA and LSTM, as illustrative techniques when forecasting time series data. These techniques were executed on a set of financial data and the results showed that LSTM was far more superior to ARIMA.

[5] observe that for both LSTM and autoregressive integrated moving average with exogenous variables (ARIMAX) models a considerable improvement of the prediction of stock price movements can be achieved when including a sentiment analysis.

[6] employ LSTM networks as deep-learning methodology and obtain returns of 0.46% per day prior to transaction costs, therefore outperforming all the memory-free methods of earlier.

[7] observed that techniques like random forest, support vector machines were not exploited fully. They focused on preprocessing of the raw dataset and after that, reviewed the use of random forest, support vector machines on the dataset and the outcomes it generates. In addition, it examines the use of the prediction system in real-world settings and issues associated with the accuracy of the overall values given.

Problem Statement

Concluding from our introduction and literature survey, stock price prediction is very lucrative and plenty of models have been tried and tested for the same. Taking the past 20 years of statistics from Nifty-50 as our source of data, we must build and compare the stock price prediction models built from the three variants of LSTM: Classic, Stacked and Bidirectional.

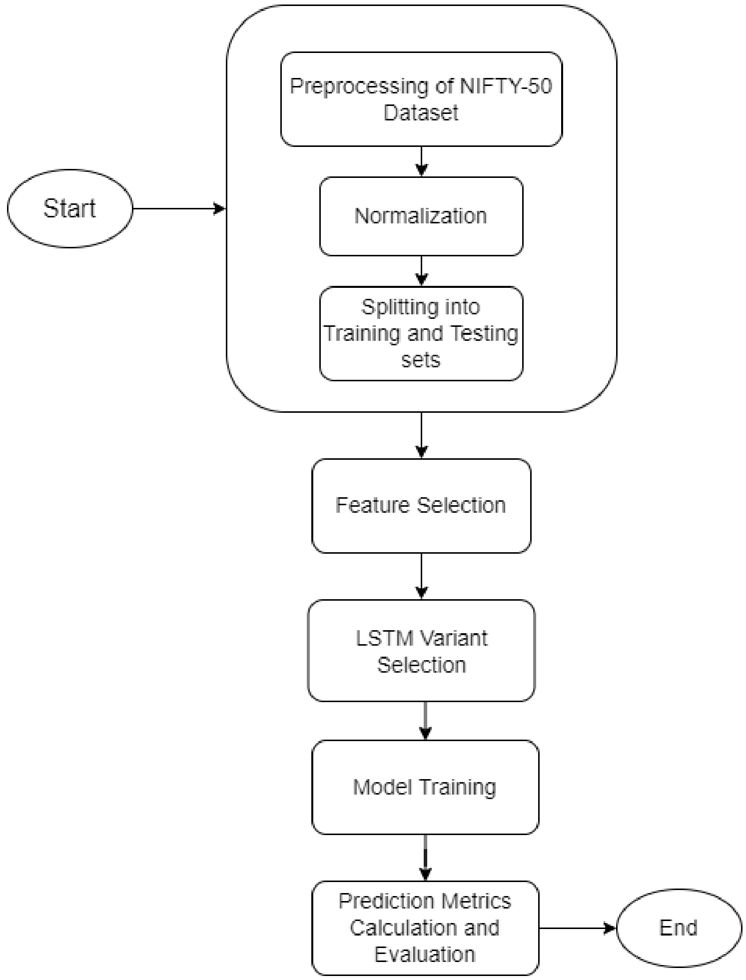
Base Paper

Gavriel, Stylianos (2021) “Stock Market Prediction using Long Short-Term Memory” [8]

Methodology

In this section included is an introduction with all the major steps which will be considered in this project:
(i) data acquisition,
(ii) data preprocessing,
(iii) details about the RNN-based models, and
(iv) the evaluation metrics.

Here is the flow chart for the methodology used for each model:



(i) **Dataset**

Dataset URL: https://www.kaggle.com/rohanrao/nifty50-stock-market-data

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

We chose this dataset because it contains 20+ years of price history and other metadata for 50 stocks, which is a large enough dataset to make a good model as data can be split between training and testing without compromises.

Data Set Description:
The data set consists of the following columns :

- Date
- Stock Symbol
- Type of Security (Equity)

- Previous day's close price
- Open price of the day
- Highest price of the day
- Lowest price of the day
- Last traded price of the day
- Close price of the day
- Volume

(ii) Data will be normalized using a python library sklearn. Feature scaling is a method to normalize the range of independent feature variables to a given range, which is 0 to 1 in our case. Data is then split into a training and a testing set at approximately 80:20 ratio.

(iii) Data will be used to train the models with all the combinations of feature sets possible. As many traditional predicting models, Closing Price will be used as a feature to train the control model. After evaluating it, multiple time scale feature models are created and trained. First a control model will be trained using the best possible features of the standard data-set, which will be selected by running tests for each combination of features and comparing their losses, and the label will be set to the Close price.

Multiple feature models will be then trained using the High, Open, Low, Last and Volume as features and further compared with the control model. Finally a comparison of the traditional models with the proposed multiple feature models will be made.

Coming to the specific parameters for the models, the neurons will use a rectified linear unit activation function and each model will have a dense output layer that makes a single value prediction.
The Classic LSTM model will have a single input layer with 16 neurons, the Stacked LSTM model will have two input layers with 16 neurons each and the Bidirectional LSTM model will have two classic lstm layers with 16 neurons, one forward and one backward whose results will be concatenated before passing on to the dense output layer.
Each model is trained over 200 epochs.

(iv) Evaluation of the methods will be made using root mean squared error and visuals will be created depicting predicted and real values, where N are the total number of values, Yi is the predicted price value and $\widehat{Yi}$ is the real price.

$$\textbf{RMSE} = \sqrt{\frac{1\sum_{i=1}^{N}(Yi - \widehat{Yi})^{2}}{N}}$$

Then final accuracy will be calculated using the sklearn metric *r2_score* which calculates the coefficient of determination or R2 score, which is the amount of the variation in the output dependent variable which is predictable from the input independent variables.

Types of LSTM

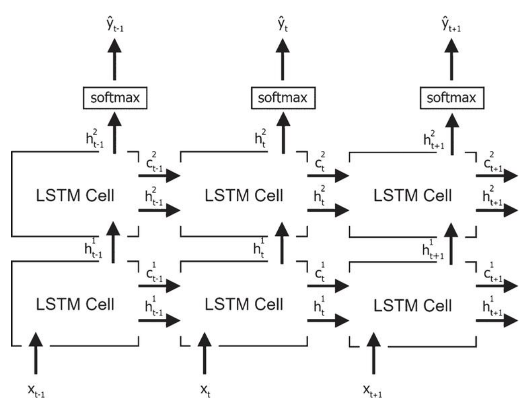
1. **Normal LSTM**
2. **Stacked LSTM**
3. **Bidirectional LSTM**

Normal LSTM:

Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can process not only single data points (such as images), but also entire sequences of data (such as speech or video). For example, LSTM is applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and anomaly detection in network traffic or IDSs (intrusion detection systems).

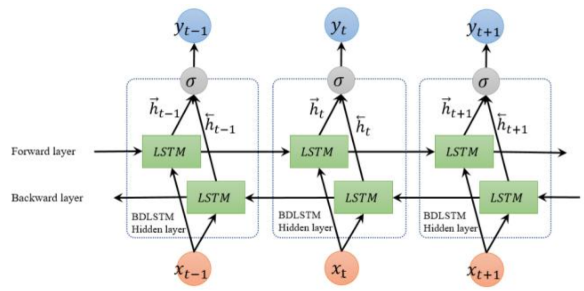
Stacked LSTM:

In Stacked LSTM, multiple LSTM cells are vertically stacked and each input in the sequence goes through a series of hidden layers to decide the output. This makes the model deeper and lets it account for more complex input patterns



Bidirectional LSTM:

In Bidirectional LSTM, the LSTM chain is duplicated and the input sequence is processed in reverse in the second chain. This makes it so that each cell now has both past and future information to make a decision instead of only past context.



Activity Schedule

Module Name	Estimated Completion Time	Date of Completion
Splitting the dataset into training, validation and test data and data normalization	1 week	26 September 2021
Building the model on Classic LSTM.	2 weeks	10 October 2021
Building the model on Stacked LSTM.	1-2 weeks	21 October 2021
Building the model on Bidirectional LSTM.	1-2 weeks	3 November 2021

Comparing results and making inferences.	1 week	10 November 2021

Results

Here are the score deviations for the top feature combinations for each LSTM variant. This is the RMS average deviation calculated across 10 stocks from the NIFTY-50 dataset.

Features	Classic LSTM	Stacked LSTM	Bidirectional LSTM
Open, Volume	0.03252	0.02609	0.03609
Low, Close	0.01682	0.01696	0.01418
Low, Volume	0.02798	0.02728	0.02555
Open, High, Low	0.0147	0.01465	0.01506
Open, Close, Volume	0.01601	0.01589	0.01615
High, Low, Close	0.01305	0.01337	0.01235
Low, Close, Volume	0.01635	0.01549	0.01518
Open, High, Close, Volume	0.0149	0.01435	0.01545
High, Low, Close, Volume	0.01377	0.0126	0.01319
Open, High, Low, Close, Volume	0.01371	0.01349	0.01338

As we can see from our table, the best possible accuracy was produced by Bidirectional LSTM for the feature set {High, Low, Close} with a deviation of 0.01235. For Stacked LSTM the best possible accuracy was produced by the feature set {High, Low, Close, Volume} with a deviation of 0.0126. For Classic LSTM the best possible accuracy was produced by the same feature set {High, Low, Close} with a deviation of 0.01305. Thus overall Bidirectional LSTM provided the best results and Classic LSTM provided the worst results. Overall it was observed that Volume as a feature was the worst for prediction and always degraded the accuracy of the model, however this degradation became insignificant as we started to select 4-5 features at once.

Conclusion

This study employed the use of various LSTM variants to predict the prices of stocks from the NIFTY-50 index. The three variants explored were: Classic, Stacked and Bidirectional. Overall, Bidirectional LSTM produced the best accuracy with the least deviation of 0.01235 for the feature set {High, Low, Close}. Stacked LSTM was at a close second with the least deviation of 0.0126, while Classic LSTM produced the worst result with least deviation of 0.01305. The outcomes from this research study hold practical importance because these models can be used for real time prediction of stock prices by HFT firms or hedge funds.

Future Work

1. **Expanding the models compared:** As of now, we are using 3 types of LSTM (Classical, Stacked and Bidirectional) to predict the stock market prices and are then comparing their results to find out which one of them is more accurate. We can further expand our observations by training other models like ARIMA, ARIMAX and Facebook's Prophet on our data and analysing their performance against each other in different scenarios.
2. **Adding Granularity:** In this study, we only looked at the daily stock price data. As a result, our parameters were calculated over weeks. This is different from actual hedge funds and quantitative trading institutions, who study on the order of minutes. We can further this study by looking at intraday data trends in addition to closing price data, by which, we can create more robust models that can predict sudden changes in momentum during intra-day trading more accurately.

References

[1] K. A. Althelaya, E. M. El-Alfy, and S. Mohammed. Evaluation of bidirectional lstm for short-and long-term stock market prediction. In 2018 9th International Conference on Information and Communication Systems (ICICS)

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

[3] Tao Xing, Yuan Sun, Qian Wang, Guo Yu. "The Analysis and Prediction of Stock Price", 2013 IEEE International Conference on Granular Computing (GrC).

[4] Siami-Namini, S., & Namin, A. S. (2018). Forecasting economics and financial time series: ARIMA vs. LSTM. preprint, arXiv:1803.06386

[5] Sharma, A., Tiwari, P., Gupta, A. & Garg, P. (2021). Use of LSTM and ARIMAX Algorithms to Analyze Impact of Sentiment Analysis in Stock Market Prediction. Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2020, 377–394.

[6] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270, 654–669.

[7] Hiba Sadia, Aditya Sharma, Adarrsh Paul, SarmisthaPadhi, Saurav Sanyal- "Stock Market Prediction Using Machine Learning Algorithms", IJEAT, 2019.

[8] Gavriel, Stylianos (2021) "Stock Market Prediction using Long Short-Term Memory"