STOCK MARKET ANALYSIS USING DEEP LEARNING AND EFFICIENT FRONTIER ALGORITHM

Krish Shah[1], Neha Joshi[2], Abhishek Kumar Saxena[3], Benoi Alex[4]

[1-4]Student, Dept. of Computer Science and Engineering, MIT School of Engineering, MIT ADT University, Pune Maharashtra, India

---***---

Abstract - *The stock market or share market is one of the foremost difficult and complex tasks to do business. Tiny ownership companies, brokerage companies, banking sectors, etc., all rely on it to form revenue and divide risks; a really difficult model. As we all know machine learning and artificial intelligence have always helped the world in finding solutions to almost every problem. So, this paper proposes to use statistics like efficient frontier and machine learning algorithms like long short-term memory(LSTM), to predict the longer-term stock worth for exchange by open supply libraries and antecedent algorithms to assist build this unpredictable format of business a little predictable. We have a tendency to see that this implementation can bring acceptable results. The end result is totally supported numbers and assumes a lot of axioms that will or might not follow within the planet thus because of the time of prediction, and will help the investor to make better decisions which would directly contribute to his profile.*

***Key Words*: Stock market, machine learning, artificial intelligence, statistics, algorithms, efficient frontier, LSTM.**

1. INTRODUCTION

STOCK MARKET is one in all the oldest ways where a traditional person would trade stocks, create investments, and earn some cash out of firms that sell a region of itself. This technique proves to be a possible investment scheme if done safely. However, the costs and therefore the liquidity of this platform is extremely unpredictable and this is often wherever we have a tendency to bring technology to assist us out. Machine learning is one such tool that helps us win what we would like. The subsequent three paragraphs can, in short, make a case for the key elements of this paper:

The stock market as we all know could be an important commerce platform that has an effect on everybody at a private and national level. The essential principle is kind of easy, corporations can list their shares within the corporations as tiny commodities known as Stocks. They are doing therefore so as to boost cash for the firm. An organization lists its stock at a value known as the giving or initial public offering. This is often the selling price at which the corporation sells the stock and raises cash. Once those stocks are the property of the owner associated, he could sell them at any value to an Emptor at an Exchange like Bombay Stock Exchange or metropolis securities market. Traders and patrons continue to commerce these shares at their own value however the corporation solely gets to stay the cash created throughout the commerce. The continuing hopes of shares from one party to a different so as to create a lot of profits end up in a rise of the value of the actual share once each profitable deal. However, if the corporate problems a lot of stocks at lower commerce, then the value for exchange goes down and traders suffer a loss. This actual development is the reason for the concern folks have in investment available markets and therefore the reason for the autumn and rise of stock costs in a very shell.

Now if we have a tendency to try and graph the stock market worth over the fundamental measure (say six months), is it extremely arduous to predict the future outcome on the graph? Somebody's brain is extremely capable of extending the graph many coordinates by simply straightforward watching it for many minutes. And if we have a tendency to crowd cipher i.e. create a bunch of random individuals trying to extend the graph by a hard and fast quantity of your time (say a week), we'll get an awfully cheap and approximate answer to a true-life graph. As a result, several brains can try and interpret the pattern and create a guess and this such activity has verified to be plenty more triple-crown in observation than it looks in theory. Having aforementioned that, predicting the true worth of the stock is best calculable by the tactic of crowd computing. However, because it is greatly evitable that crowd computing may be a terribly slow activity, we have a tendency to try and use a laptop here to simulate such examples with a lot of scientific and mathematical approaches.

In machine learning, we need a layer to understand the sequence of the time series data, so that sequence prediction problem doesn’t arise, the technique we are using is known as Long short-term memory and is normally used because of its very straightforward and effective approach. With the recent breakthroughs that have been happening in data science, it is found that for almost all of these sequence prediction problems, Long Short-Term Memory networks (LSTM) have been observed as the most effective solution.

So, with the fundamental information of exchange, graphs, and knowledge analysis coupled with machine learning; we are currently ready to devise the program.

2. RELATED WORK

[1] 'Network approach for stock market data mining and portfolio analysis'

In this paper, data mining based on networks is done to identify the important players. A portfolio analysis of the most influential player is used to reveal the crucial sectors of the market.

[2] 'Stock Market Behaviour Prediction using Stacked LSTM Networks'

In this thesis, the stacked Long Short Term Memory network model for predicting stock market behaviour is used. The data is composed of historic stock market data from the American Stock Exchange. Results obtained show that by making use of a stacked Long Short Term Memory network model, future stock market behaviour can be predicted.

[3] 'Stock Market Prediction Using Machine Learning.'

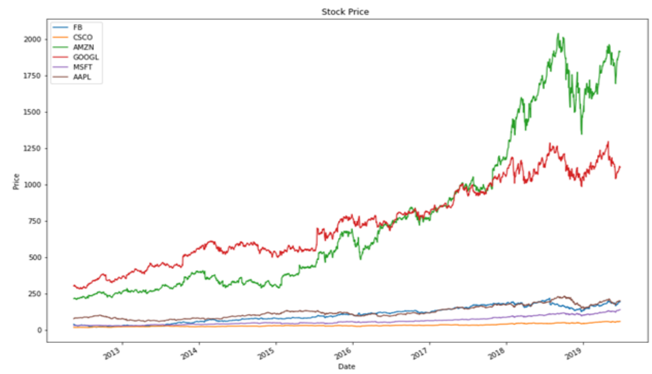
Considering the factors like open, close, low, high, and volume, this paper aims to use machine learning algorithms like regression and LSTM to predict the future price of the stocks.

[4] 'A Deep Learning Model for Predicting Buy and Sell Recommendations in Stock Exchange of Thailand using Long Short-Term Memory'

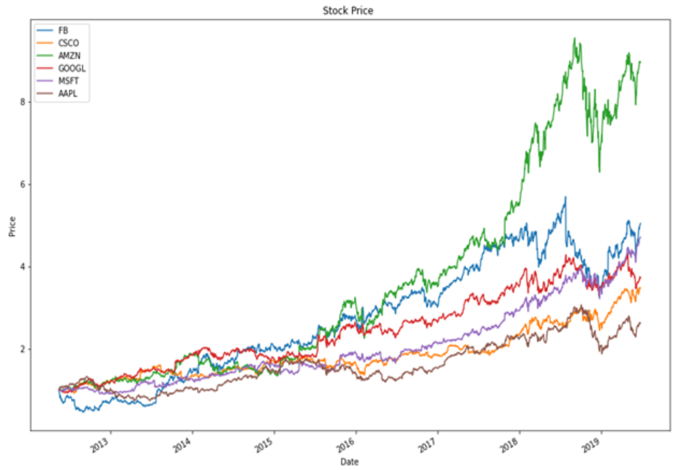
This research adopts deep learning techniques for predicting buy and sell recommendations in the Stock Exchange of Thailand using Long Short-Term Memory. The proposed model can capture long term dependencies in stock price data in order to enhance prediction accuracy.

3. DATA VISUALISATION

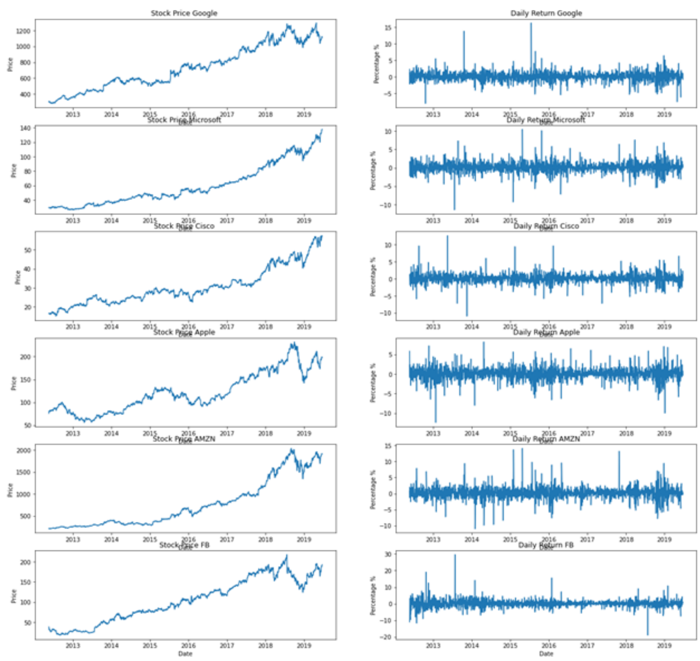
The below figure shows the Closed price of the stock for various companies which is used to calculate the returns. We see the above two companies are way too far from other companies because of their values.



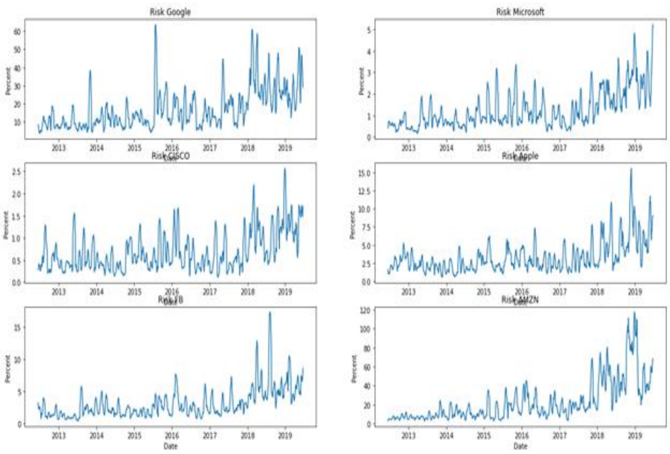
To get rid of this problem of flattening other values we have normalized the price. So every stock starts from the same starting point.



The below figure shows the daily return calculated which is the percent change in the value of prices for a particular time period. This is calculated for the whole period of 11 years.



The below graph shows the risk calculated across 11 years with simple standard deviations from returns calculated.



The below figure shows the covariance among various stocks which is used to calculate efficient frontier. Less the covariance among the companies better returns on low risk.

4. DATA PREPROCESSING

In machine learning, we cannot use the raw data directly in our program as the data could not be the way our program requires so devising our data in a format required is called preprocessing of the data. The goal of data preprocessing is to find out what information we require and to modify our data accordingly. It could be done following the given:

[1] **Efficient Frontier:**

1. Find the percent change between all stock Open prices to get daily returns for the daily stock price.
2. Calculate the risk from the returns calculated using standard deviation.
3. Calculate the portfolio returns and risk from the historical data return and risk according to the weights giving by the user

[2] **Long Short Term Memory:**

1. Gather the historical 11 years data and scale it between 0 and 1 using min-max scaler in python
2. To input into LSTM, we need to convert into the sequence of 60 historical entries and 61st entry as a target value for all the 11years data making a list of array of 60 entries with 1 target value

5. ALGORITHMS USED

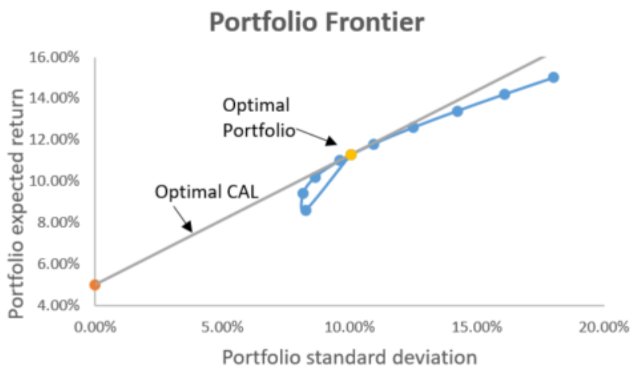
5.1 Efficient Frontier

Different combinations of securities produce different levels of return. The efficient frontier represents the simplest of those securities combinations -- folks that produce the utmost expected return for a given level of risk. The efficient frontier is that the basis for contemporary portfolio theory.

It is a graph that maps out all possible portfolios with different asset weight combinations, with levels of portfolio variance graphed on the x-axis and portfolio expected return on the y-axis.

For every point on the efficient frontier, there's a minimum of 1 portfolio that's ready to be constructed from all available investments that have the expected risk and return like that point.

The optimal portfolio consists of a risk-free asset and an optimal risky asset portfolio. The optimal risky asset portfolio is at the aim where the Capital Allocation Line (CAL) is tangent to the efficient frontier. This portfolio is right because the slope of CAL is that the best possible, which suggests we achieve the right returns per additional unit of risk. The graph below illustrates this:



The tangent portfolio weights are calculated as follows:

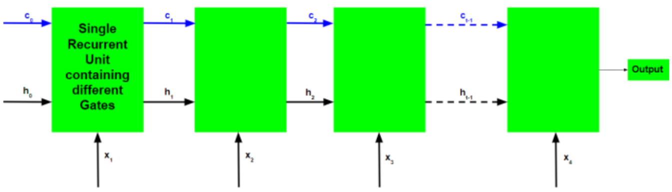
$$w_s^{TP} = 1 - w_b^{TP}$$

$$w_b^{TP} = \frac{(E(R_b)-R_f)\sigma_s^2-(E(R_s)-R_f)\rho_{sb}\sigma_s\sigma_b}{(E(R_b)-R_f)\sigma_s^2+(E(R_s)-R_f)\sigma_b^2-(E(R_b)-R_f+E(R_s)-R_f)\rho_{sb}\sigma_s\sigma_b}$$

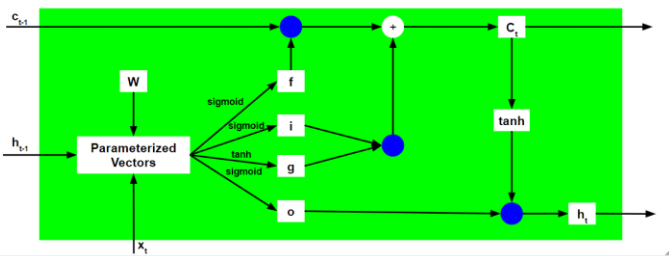
5.2 LSTM(Long Short -Term Memory)

An LSTM recurrent unit tries to "remember" all the past knowledge that the network has seen so far and to "forget" irrelevant data. This is done by introducing different activation function layers called "gates" for various purposes. Each LSTM recurrent unit also maintains a vector called the Internal Cell State which conceptually describes the information that was chosen to be retained by the previous LSTM recurrent unit.

The basic work-flow of a Long Short Term Memory Network is similar to the work-flow of a Recurrent Neural Network with the only difference being that the Internal Cell State is also passed forward along with the Hidden State.

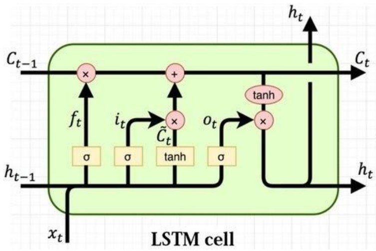


The working of LSTM is illustrated as below:-



Note that the blue circles denote element-wise multiplication. The weight matrix W contains different weights for the current input vector and the previous hidden state for each gate.

Just like Recurrent Neural Networks, an LSTM network also generates an output at each time step and this output is used to train the network using gradient descent.

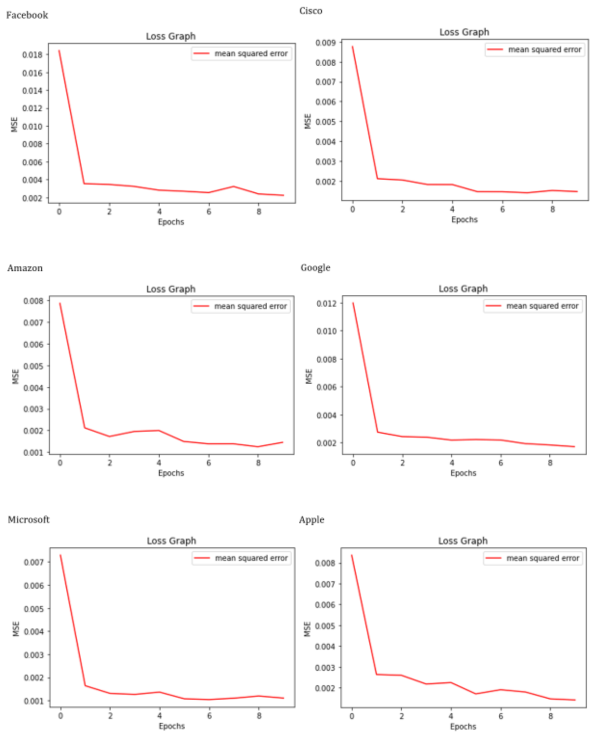


$$i_t = \sigma(x_t U^i + h_{t-1} W^i)$$
$$f_t = \sigma(x_t U^f + h_{t-1} W^f)$$
$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$
$$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g)$$
$$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$$
$$h_t = \tanh(C_t) * o_t$$

6. RESULTS

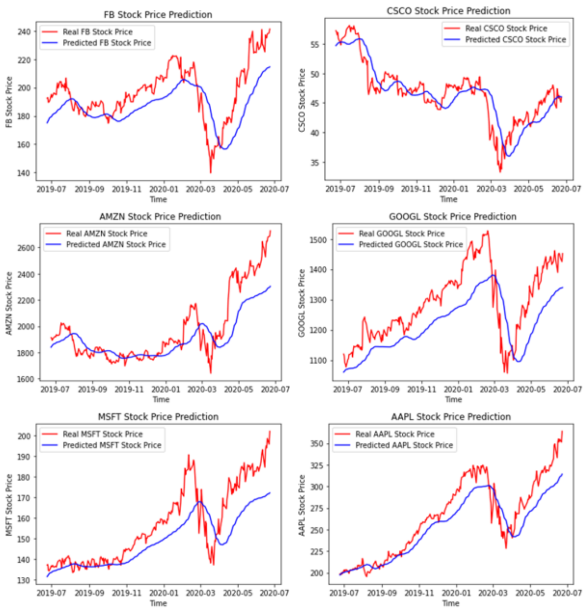
Loss graphs

The following are the loss graphs for model training of each company in the prototype. The loss used for this model is the root mean square error.

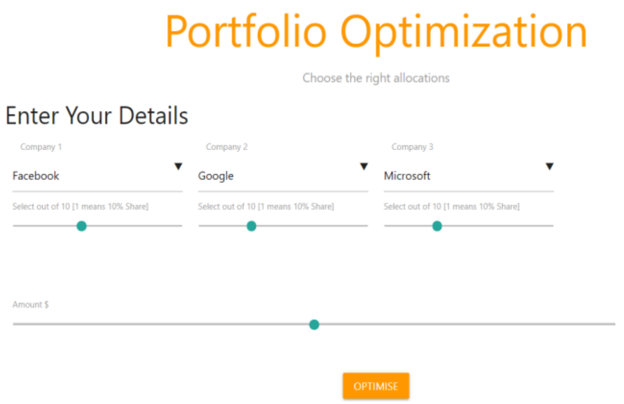


Evaluation on 2019 Data

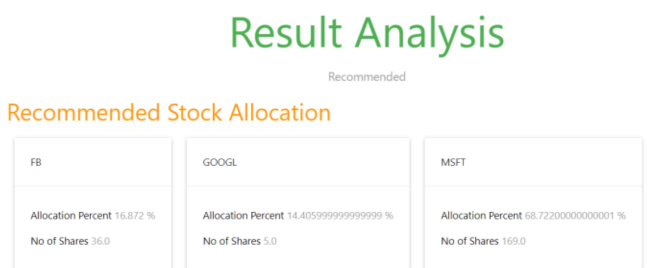
The following are the figures showing the test result on 2019 stock price data which includes predicted and the actual price for each company.



Output Screen



• This is the initial page where we ask users to select companies and distribute the shares amongst them.
• There are small sliders that move from 0-10, where 10 represents 100% and 1 represents 10%.
• There is a bigger slider ranging from 0-1 Lakh that shows the initially allocated amount.
• There is an optimise button at the bottom which when clicked triggers the model and starts training for the past 11 years and we test the model on last 1 year data, then it generates the result.



• Once the results are generated by trained model on the test set, the result is generated in above shown format.
• It shows the recommended stock allocation percentage by our model.
• It also shows the number of shares needs to be allocated.

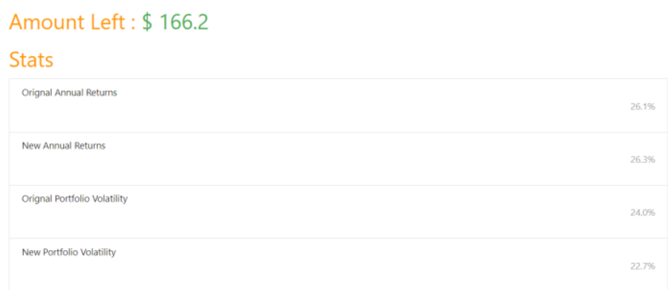

Predicted Next Day Price

Date	Stock Price
2020-06-25	$ 365.27493
2020-06-26	$ 365.2863
2020-06-27	$ 365.0903
2020-06-28	$ 364.6098
2020-06-29	$ 363.77048
2020-06-30	$ 362.53204

Date	Stock Price
2020-06-25	$ 1394.2506
2020-06-26	$ 1394.7504
2020-06-27	$ 1394.9144
2020-06-28	$ 1394.6339
2020-06-29	$ 1393.8732
2020-06-30	$ 1392.6548

Date	Stock Price
2020-06-25	$ 215.15019
2020-06-26	$ 215.09521
2020-06-27	$ 214.98503
2020-06-28	$ 214.78886
2020-06-29	$ 214.51898
2020-06-30	$ 214.17027

Then five different models are trained for five different companies with the data of the last 60days and then it predicts the open price of stock for the next 6 days.



• It shows the amount left after the allocation.
• It shows the annual return before and after our allocation
• It also shows the risk before and after our allocation.

7. CONCLUSION

Determining the stock market forecasts has always been challenging work for business analysts. Thus, as we can see above in our implementation, we train the data using an existing stock dataset that is available. We use this data to predict and forecast the stock price of 6 days into the future. Thus, the project applies the Machine Learning technology for stock price forecast which will provide the research of the stock market development a new thought & we aim to make use of huge datasets to predict the stock market prices.

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude towards my teacher Prof. Dipti Pawar and Prof. Sonali Deshpande for providing us with the golden opportunity to make this wonderful project on the topic 'Stock Market Analysis using Deep Learning and Efficient frontier algorithm', which also helped us in doing a lot of research and learning about new things.

We are highly indebted to MIT School of Engineering for their guidance and support as well as providing the necessary information and resources regarding the project.

REFERENCES

[1] Susan George, Manoj Chagat, "Network approach for stock market data mining and portfolio analysis" IEEE, 2017.

[2] Samuel Olusegun Ojo, Pius Adewale Owolawi, Mardi Mphahlele, and Juliana Adeola Adisa, "Stock Market Behaviour Prediction using Stacked LSTM Networks'" IEEE, 2019.

[3]Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan, "Stock Market prediction using Machine Learning", IEEE 2018

[4]Thaloengpattarakoon Sanboon, Komal Keatruangkamala, Saichon Jaiyen, "A Deep Learning Model for Predicting Buy and Sell Recommendations in stock Exchange of Thailand using Long Short-Term Memory."

[5]Agarwal (July 14, 2017). "Introduction to the Stock Market". Intelligent Economist. Retrieved December 18, 2017.

[6]Hurwitz, E., and Marwala, T., 2012. Common mistakes when applying computational intelligence and machine learning to stock market modelling. arXiv preprint arXiv:1208.4429.

[7]Ashish Sharma, Dinesh Bhuriya, Upendra Singh. "Survey of Stock Market Prediction Using Machine Learning Approach", ICECA 2017.

[8]Loke.K.S. "Impact Of Financial Ratios And Technical Analysis On Stock Price Prediction Using Random Forests", IEEE, 2017.