LSTM vs. MLP in S&P500 Index Fund Forecasting

Eknoor Singh and Gurman Brar
Department of Systems Design Engineering
University of Waterloo
Waterloo, ON N2L 3G1
eknoor.singh@uwaterloo.ca
g9brar@uwaterloo.ca

May 28, 2024

Abstract

This report has the objective of testing the capabilities of two machine learning algorithms in attempts to forecast the S&P 500 stock on both the NYSE over a daily period. The necessary data was obtained through a custom API that interfaces with Yahoo Finance. Complementary stock indicators were created through the use of a custom Python library, meant for technical analysis of stocks. Both MLP and LSTM models were hyper-tuned and were comparatively assessed for their respective mean squared error. The designed MLP model proved to outperform the LSTM model, with lower MSE in the final test performance. However, both models showed resiliency in predicting the stock's trend and direction. Further recommendations and possible improvements to this report were discussed as the opportunities in this application of machine learning seem vast.

1 Introduction

1.1 Background & Motivation

In the realm of financial investing, companies offer a stake in their value through a concept referred to as stocks or securities [**1**]. These stocks are a percentage of ownership within the company, allowing shareholders to reap the benefits of the corporation they invested in through dividends and trading the stock at a fixed price [**1**]. Dividends are payouts from corporations that offer stakes within their companies as an incentive to get the public to invest [**1**]. Stock prices of corporations fluctuate with multiple causes, with major effects being caused by the economic laws of supply and demand, as well as the public sentiment of the company and their stock's value [**2**]. Investors aim to generate income from either of these two methods, with dividends being a fixed and passive form of income, while stock trading is more volatile in its nature.

In 2008, a global financial crisis shocked the world with the stock and house market crashing [**3**]. This event caused a tremendous decrease in stock prices in public companies, leaving shareholders facing great investment deficits. The community of investors likely wished to have the ability of foresight, as their losses could have been avoided if the decrease in stock prices could have been predicted. Meanwhile, the reported value of the stock market as of Q2 2023 is $108.6 Trillion [**4**]. The oppourtunity to create capital gains is tremendously higher in the current day, as compared to the global calamity the stock market faced in 2008. This leads to the motivation of this project, as we will test the capabilities of prediction models forecasting particular stock prices.

1.2 Relevant Literature

A literature review was performed to gauge how previous attempts at classification and prediction models were implemented to identify stock behaviour and forecast stock prices. Zhao Gao published a journal in 2019 that used classification and clustering algorithms to group stocks for their prospective capabilities [**5**]. K-nearest neighbors (KNN) and error back-propagation through an Artificial Neural Network (ANN) were used as classifying algorithms, and the k-means clustering algorithm was used as a comparative approach. Four distinct stock indicators were used as features in classifying/clustering the different stocks' nature. Various other articles experimented with the implementation of differing ANNs in attempts to forecast certain stock prices [**6**][**7**][**8**][**9**]. The varying methods of ANNs implemented in the aforementioned journals were Convolutional Neural Networks, Deep Neural Networks, Long Short-Term Memory Models (LSTM), Multi-Layer Perceptron Models (MLP), and other hybrid neural network implementations [**6**][**7**][**8**][**9**]. The outcomes of studying these relevant sources came two-fold, providing insights into model implementations and datasets. Out of the model implementations in these journals, the LSTM model performed remarkably in predicting stock prices with the given stock market datasets while it was often restricted due to the amount of data within these sets. A secondary model that stood out was the use of an MLP model to forecast stock price directions. Both of these models were found to be interesting models to implement and will be the applied models of this report. The stock market data that was used for all papers were the Istanbul Stock Exchange, the Brazilian Stock Index, and the Standard and Poor's 500 (S&P 500) [**6**][**7**][**8**][**9**]. The S&P 500 stock is widely known, with known daily fluctuations in opening and closing prices. It also has a great amount of historical data dating back to 1984 on Yahoo Finance [**10**], which combats the complication of LSTM performance with the limited datasets mentioned previously. Due to its familiarity and its substantial historical data, it was decided that the S&P 500 would be used as a dataset for input data for our models.

2 Methodology

2.1 Data Collection and Feature Generation

As discussed previously in Section 1.2, it was decided that the S&P 500's historical data was to be used as input for our MLP and LSTM models. An offered package in Python is the yfinance extension, which aids in importing financial data from Yahoo Finance's stock market [**11**]. Due to the objective of this project, daily data is needed to create a stock price forecast for future days. Upon pulling this data, a total of 9917 daily data entries were provided for the S&P 500, dating back just over 27 years. Of each of the daily entries for the S&P 500, there were multiple variables depicting the stock's performance on a given day. The variables provided when pulling from the S&P 500 were open price, close price, adjusted close price, stock volume, stock price daily high, and stock price daily low. The variables are the characteristics of the stock before, during, and after the closure of the stock market within one day.

Upon the collection of the data, the library pandas-ta was used to create meaningful indicators on the given stock variables [**12**]. The following indicators were generated from the stock variables that were pulled: Relative Strength Index (RSI) and Exponential Moving Average with differing day periods (20, 100, and 150 for EMAF, EMAM, and EMAS respectively). The RSI of a stock is a gauge of its momentum on the path it's traveling (trending upwards/downwards) and its resistance to change from its path [**13**]. With a general rule of thumb, when the RSI of a stock is >70, the stock is indicated to fall in price, and the opposite is true when the RSI is <30 [**13**]. The use of EMA, regardless of the time window, is to provide a view of the moving average of the stock's price, but with extra weight placed on more recent data points [**14**]. After creating our features, all features were normalized using the min-max scaler from the sci-kit learn package in Python.

2.2 MLP & LSTM Hyper-parameter tuning

To reiterate, we will be using both an MLP model architecture and an LSTM model architecture. For this report, the MLP architecture only has 2 hyperparameters; the number of hidden layers and the number of neurons in each of the hidden layers. In addition, the LSTM has 4 hyperparameters; the number of units in the LSTM layer, the number of neurons in the first and second hidden layers, and finally, the dropout rate between the hidden layers. For the hyperparameter tuning, we utilized an optimization tool called Optuna, which allows us to perform a hyperparameter tuning without analyzing the entire search space (as it is very computationally exhaustive given our large search space). For this report, we will only be running 20 trials for each of the models. The hyperparameter tuning scores each set of hyperparameters based on the MSE of the validation set after a new model is created using the respective hyperparameters and trained on the training data. The results of the MLP hyperparameter tuning can be found in Table 1 and the LSTM hyperparameter tuning results can be found in Table 2, in the upcoming sections.

3 Results

3.1 MLP Performance

Table 1: Optimal Hyper-parameters for MLP Model.

Optimal Hyper-parameters	
Number of Dense Layers	3
Number of Neurons in Layer #1	842
Number of Neurons in Layer #2	254
Number of Neurons in Layer #3	216
Validation Set MSE for hyper-parameters	0.00067

Using the optimization of hyper-parameters offered by Optuna, the number of dense layers and number of neurons within each layer were varied, as mentioned in Section 2.2. The optimal MSE that was produced by the MLP model was 0.00067, from the parameters specified in Table 1.

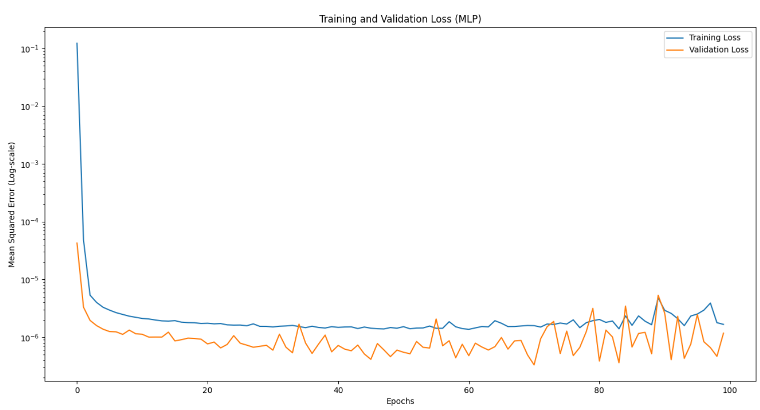

Figure 1: Training and Validation Loss Curves of MLP Model After Hyperparameter Tuning [**15**]

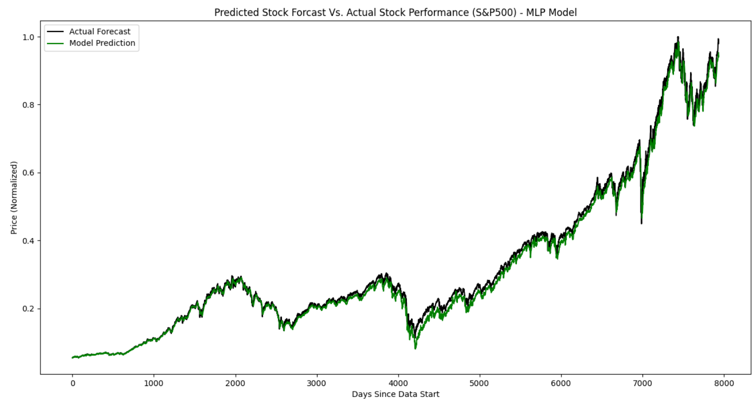

Figure 2: Actual S&P500 Price Movement Vs. MLP Model Prediction (Test Data) [**15**]

3.2 LSTM Performance

Table 2: Optimal Hyper-parameters for LSTM Model

Optimal Hyper-parameters	
Number of LSTM units	258
Number of Neurons in Layer #1	73
Number of Neurons in Layer #2	875
Dropout Rate	0.3245
Validation Set MSE for hyper-parameters	0.00057

Using the optimization of hyper-parameters offered by Optuna, the number of LSTM units, the number of neurons in the first and second layer, as well as the dropout rate were varied, as mentioned in Section 2.2. The optimal MSE that was produced by the LSTM model was 0.00057, from the parameters specified in Table 2.

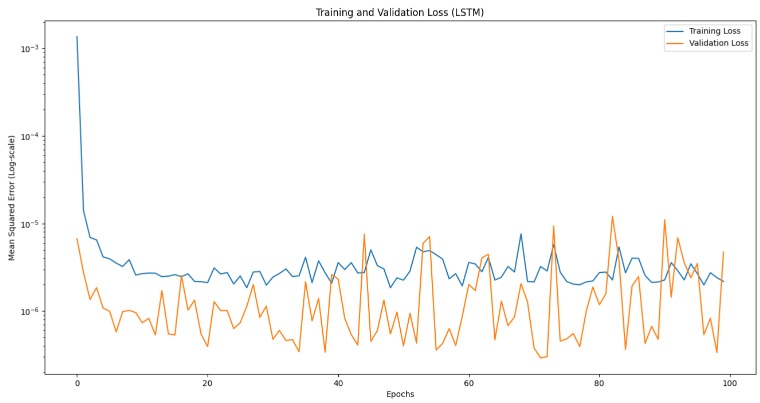

Figure 3: Training and Validation Loss Curves of LSTM Model After Hyperparameter Tuning [**15**]

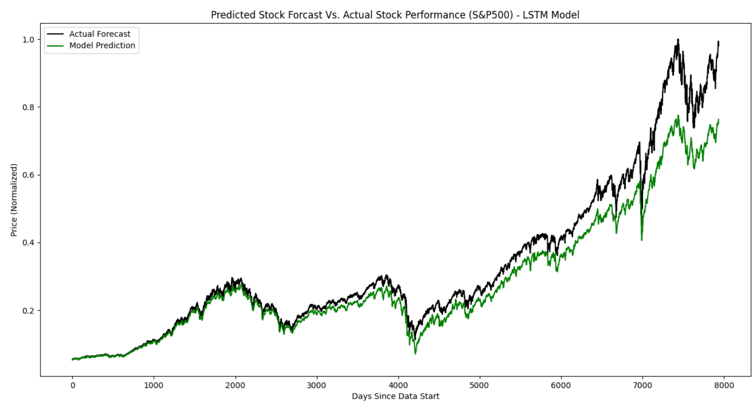

Figure 4: Actual S&P500 Price Movement Vs. LSTM Model Prediction (Test Data) [**15**]

3.3 Final Results

Table 3: Model performance on test data with optimized hyper-parameters

	MLP model	**LSTM model**
Final Test MSE	0.0169	0.0687

Based on the MSE results of both hypertuned parameters on the final test set (please refer to Table 3), we can conclude that the MLP model outperforms the LSTM model.

4 Discussion

4.1 Analysis of Results

When comparing the training curves of both the MLP and LSTM models, it can be noticed that the MLP model training had a more consistent decrease in loss when compared to the LSTM model which fluctuated frequently after about 20 epochs (Figures 1 and 3). This can potentially be attributed to the fact that the LSTM made use of a dropout layer whereas the MLP model did not. The dropout layer would have produced slight variations in the model for each of the forward and backward passes which would explain the fluctuations in the training curve (Figure 3). However, both the MLP and LSTM models had similar final losses as the number of epochs approached 100. In addition, it can be noted that both the LSTM and MLP had lower validation loss compared to training loss. This could be attributed to the fact that in general, there was less validation data when compared to training data which could result in a lower MSE. In addition, it could also be attributed to potential validation set overfitting caused by our hyperparameter tuning (where the validation loss was the value to minimize).

Taking a look at Figure 4, it can be noticed that as we get closer to the current day, the price predicted by our LSTM model deviates from the actual price but the general price trend still remains. In comparison, the MLP model (Figure 2) not only follows the actual price more accurately but also follows the trend accurately, producing an almost identical prediction. Since both models were only trained on S&P500 data and the MLP obtained a better final test set MSE than the LSTM, we can conclude that the MLP outperformed the LSTM.

4.2 Possible Improvements

The application of stock forecasting using machine learning algorithms has been explored with different methodologies, as discussed in Section 1.2. One possible improvement of results is to increase the number of different algorithms tested within this report. Various other algorithms have been tested in other pieces of literature, like CNNs, KNN, and other hybrid types of ANNs with back-propagation implementations. Other models, even an Encoder-Decoder based architecture, may see better results depending on how their architecture is varied, as compared to the MLP and LSTM results.

A second possible improvement to this report would the inclusion of more technical analysis indicators as offered features for the models. The library pandas-ta offers more than 130 indicators, the majority of which can be used as features in forecasting stocks [**12**]. With the addition of indicators, the models designed can forge better relationships with the features it has been given to predict the next stock price.

Lastly, a multimodal approach could be taken, with the MLP and/or LSTM, to be trained on more than a singular stock's data like the S&P 500. In this approach, the implemented models can better create relationships and decisions on how certain features directly affect the stock's velocity and inertia, regardless of what stock it is being tested on. This would aid in the creation of a more universal tool in the field of stock technical analysis.

5 Conclusion

In conclusion, we compared both an LSTM and MLP-based architecture for the task of predicting S&P500 index prices and price trends. Both the LSTM and MLP models were hyperparameter tuned using Optuna and the final hyperparameters were used to re-train both models and test against a final test set. The results of the final models MSE's on the test set can be found in Table 3. Based on these results, we can conclude that the MLP model out-performed the LSTM in predicting correct price and price movement, overall being the better model in predicting S&P500 trends and prices. Further modifications can be made to the MLP model architecture to reduce potential overfitting (such as the addition of dropout) when more stock data is used for training and the model is generalizing for multiple charts rather than just the S&P500.

6 References

[1] A. Hayes, "Stocks: What they are, main types, how they differ from bonds," Investopedia, https://www.investopedia.com/terms/s/stock.asp (accessed Dec. 21, 2023).

[2] D. R. Harper, "What drives the stock market?," Investopedia, https://www.investopedia.com/articles/basic (accessed Dec. 21, 2023).

[3] P. Kosakowski, "The fall of the market in the fall of 2008," Investopedia, https://www.investopedia.com/articles/economics/09/subprime-market-2008.asp (accessed Dec. 21, 2023).

[4] D. Neufeld, "The $109 Trillion global stock market in one chart," Visual Capitalist, https://www.visualcapitalist.com/the-109-trillion-global-stock-market-in-one-chart/ (accessed Dec. 21, 2023).

[5] Z. Gao, "The application of Artificial Intelligence in Stock Investment," Journal of Physics: Conference Series, vol. 1453, no. 1, p. 012069, 2020. doi:10.1088/1742-6596/1453/1/012069

[6] X. Zhong and D. Enke, "Predicting the daily return direction of the stock market using hybrid machine learning algorithms," Financial Innovation, vol. 5, no. 1, 2019. doi:10.1186/s40854-019-0138-0

[7] D. Senol and M. Ozturan, "Stock price direction prediction using artificial neural network approach: The case of Turkey," Journal of Artificial Intelligence,

vol. 1, no. 2, pp. 70–77, 2008. doi:10.3923/jai.2008.70.77

[8] Y. Hao and Q. Gao, "Predicting the trend of stock market index using the hybrid neural network based on multiple time scale feature learning," Applied Sciences, vol. 10, no. 11, p. 3961, 2020. doi:10.3390/app10113961

[9] C. M. Mesquita, R. Arantes de Oliveira, and A. C. Machado Pereira, "Combining an LSTM neural network with the variance ratio test for time series prediction and operation on the Brazilian Stock Market," 2020 International Joint Conference on Neural Networks (IJCNN), 2020. doi:10.1109/ijcnn48605.2020.9207408

[10] Y. Finance, "S&P 500 (ĜSPC) charts, Data & News," Yahoo! Finance, https://finance.yahoo.com/quote/%5EGSPC?p=%5EGSPC (accessed Dec. 21, 2023).

[11] R. Aroussi, "Yfinance," PyPI, https://pypi.org/project/yfinance/ (accessed Dec. 21, 2023).

[12] Twopirllc, "TWOPIRLLC/pandas-ta: Technical analysis indicators - pandas TA is an easy to use Python 3 pandas extension with 130+ indicators," GitHub, https://github.com/twopirllc/pandas-ta (accessed Dec. 21, 2023).

[13] J. Fernando, "Relative strength index (RSI) indicator explained with formula," Investopedia, https://www.investopedia.com/terms/r/rsi.asp (accessed Dec. 21, 2023).

[14] J. Chen, "What is Ema? how to use exponential moving average with formula," Investopedia, https://www.investopedia.com/terms/e/ema.asp (accessed Dec. 21, 2023).

[15] E. Singh, G. Brar, "LSTM vs. MLP in S&P500 Index Fund Forecasting", Dec 2023.