Stock Price Prediction

using

Neural Network

By Naomi Vaid,

Junior at University of South Florida,

Tampa FL

Table of Contents

ACKNOWLEDGMENT

I would like to thank Odysseas Drosis from Cornell University for providing me with guidance and knowledge needed for carrying out this report. I would like to thank Daniela Ganelin from Massachusetts Institute of Technology for her contributions as well. I would like to thank my professors and fellow student in University of South Florida for their support. I would also like to thank staff of InspiritAI for their extended guidance.

Naomi Vaid
October 2023

ABSTRACT

The economy is an ever changing area with new and different trends each year. Understanding and being able to predict new changes and directions for the economy is necessary for businesses to make informed decisions, assists in business planning and investment and helps individuals have control over their finances. The stock market is what allows individuals and groups to clearly see how certain markets and industries are doing. People can analyze patterns and developments to exercise good economic judgment. A few examples of applications of stock price prediction are: aids in risk management, enables algorithmic trading strategies, contributes to market analysis, portfolio management, which all lead to efficient financial decision processes.

INTRODUCTION

Traditional stock price prediction methods include technical analysis, fundamental analysis, time series analysis, etc. When using traditional stock price prediction methods, there are quite a few challenges faced. First is market volatility. Stock markets tend to be highly volatile as they are influenced by many factors. Traditional methods fall short of taking into account these unpredictable and sudden changes which can lead to inaccurate results. Second is the data quality/quantity. Since traditional methods rely heavily on historical data, there might be issues if this data isn't readily available or accurate. For instance, some data might have missing values, inconsistencies, etc. Third is overfitting and model complexity. Traditional methods might have been heavily trained on past data which means the prediction model is overfitting. This will cause it to only be accurate on past data and it won't be able to accurately predict and work with future data. AI is considered to be a promising approach for stock price prediction for various reasons. It can easily handle complex relationships, i.e. capturing and analyzing large amounts of data. Since the stock market in its essence is all about capturing data and understanding it, AI is the perfect tool to use for it. With the rapid advancement and growth of AI it has developed the ability to process and handle vast amounts of data. It can handle much larger amounts of data that people can, and at a much faster rate. This is another factor that makes it extremely useful for a stock price prediction model. Lastly, the use of AI leads to automation which eliminates the need for manual analysis by traders and investors which can save a great deal of time and money. Our model utilizes neural networks and the 5 company stock prices we focused on were: Tesla, Starbucks, Amazon, Microsoft, Apple.

DATASET

For the stock price prediction program, we used data imported from Yahoo Finance. This is an efficient way to import data using libraries as our code was in python. The data imported from

Yahoo Finances contained the stock prices for hundreds of thousands of companies, including Tesla, Starbucks, Amazon, etc from the last 5 years. It included various categories of prices such as the opening and closing ones. We had access to quality data because of this which prevented any discrepancies in values, any missing data, etc. It was also a quick way to gather and analyze large amounts of data as we didn't have to manually search for the stock prices of a company ourselves and input it. Yahoo Finance is also frequently updated and revised which provides us access with the best real time data- and the stronger our data- the better our program performs. First, data is imported via Yahoo Finance, then the necessary libraries (yfinance, pandas, numpy) are installed. Then, an object called *Ticker* is created which is the stock symbol for the company whose stock prices we are analyzing. Next we retrieve historical stock data from the last 5 years and this data is stored in a data frame. This includes the open, close, high, low, volume, dividends and stock splits as some of its columns. To preprocess the data, depending on what column of data we are analyzing, the flow of the program slightly varies. For instance, if we are only interested in analyzing the open prices, the unneeded columns from the dataframe are removed, making it contain only the required columns (i.e. open prices). The libraries used were Yfinance, Numpy and Sickit Learning.

MODELS & METHODS USED

We used a neural network for our program. The neural network is derived from the way the human brain works. It consists of 10 billion neurons and six thousand times as many synapses between them, which connects them. All the inputs received by the body are processed in this network located in the brain. See Figure I below.

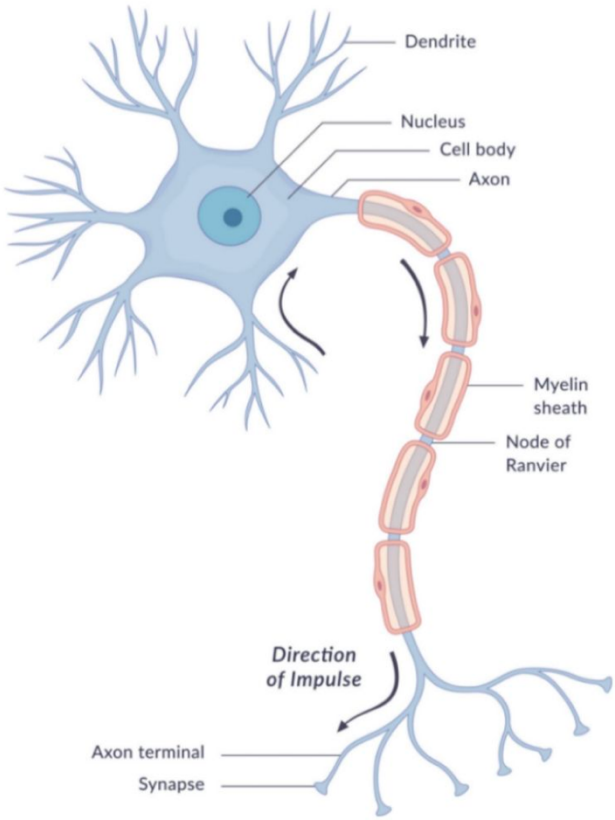


Figure I: Biological Neuron
Reference : www.thoughtco.com

The flow of the neural network we used was similar to the biological neuron. See Figure II below. We used 2 inputs and had 2 hidden layers, the first layer of size 100 and the second layer of size 50.

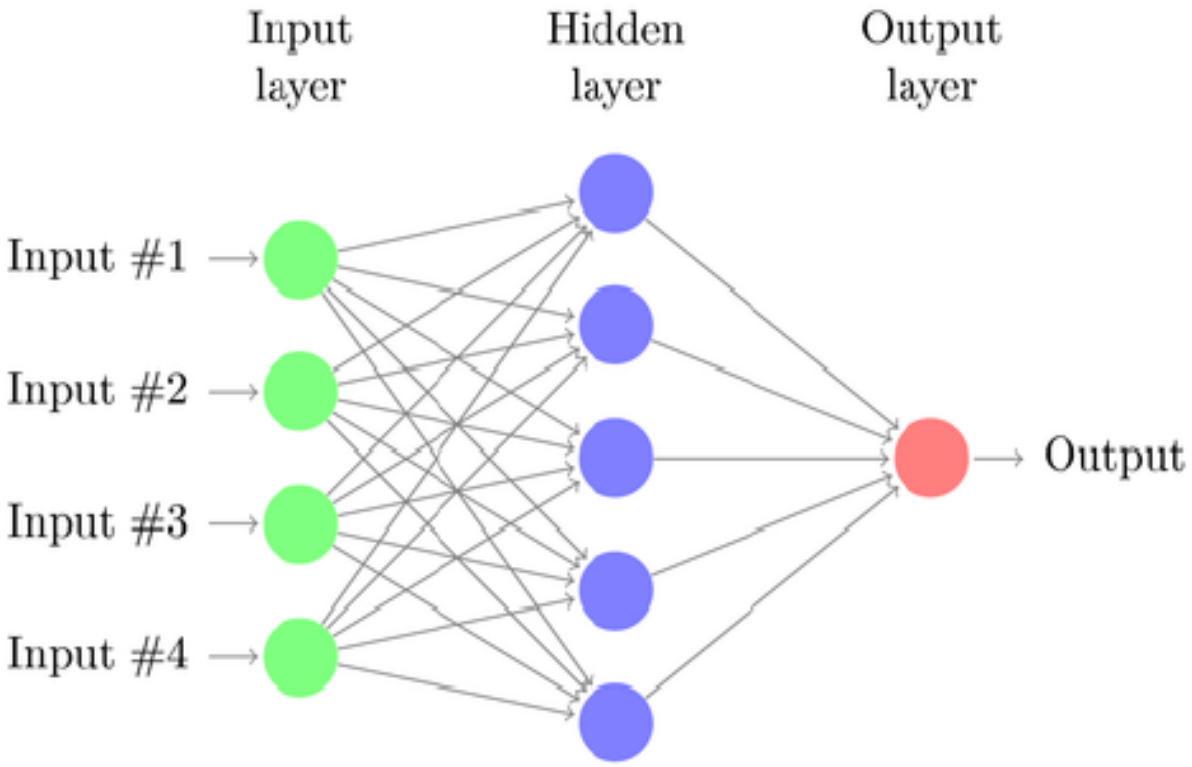


Figure II: Neural network layers
Reference: www.quora.com

The advantages of neural networks are as follows:

A. Generalization capability
B. Robustness
C. Ease in mapping of input/output
D. Model does not require assumptions
E. Flexibility in usage

The drawbacks are:

A. Blackbox characteristics
B. Overfitting possibility
C. Input requires knowledge and expertise
D. Training can be time consuming

Neural networks are great for nonlinear relationships which take into account market changes, and while our program doesn't currently take market changes into factor, this is still an important feature to have for the future. The neural network we used was a Multilayer Perceptron (MLP). The program had a maximum iteration of 2000. It also had a random state of 42. MLPs are a feed forward neural network that provide flexibility. Feed forward neural networks are commonly used in predictions as they are excellent at capturing patterns and trends. Additionally, it can

handle large data sets and process vast amounts of data quickly and efficiently. We used two sets of training data in the model: 'X_train' (input feature: ⅔ of the data) and 'Y_train' (target label: ⅓ of the data). The neural network was iterated 2000 times.

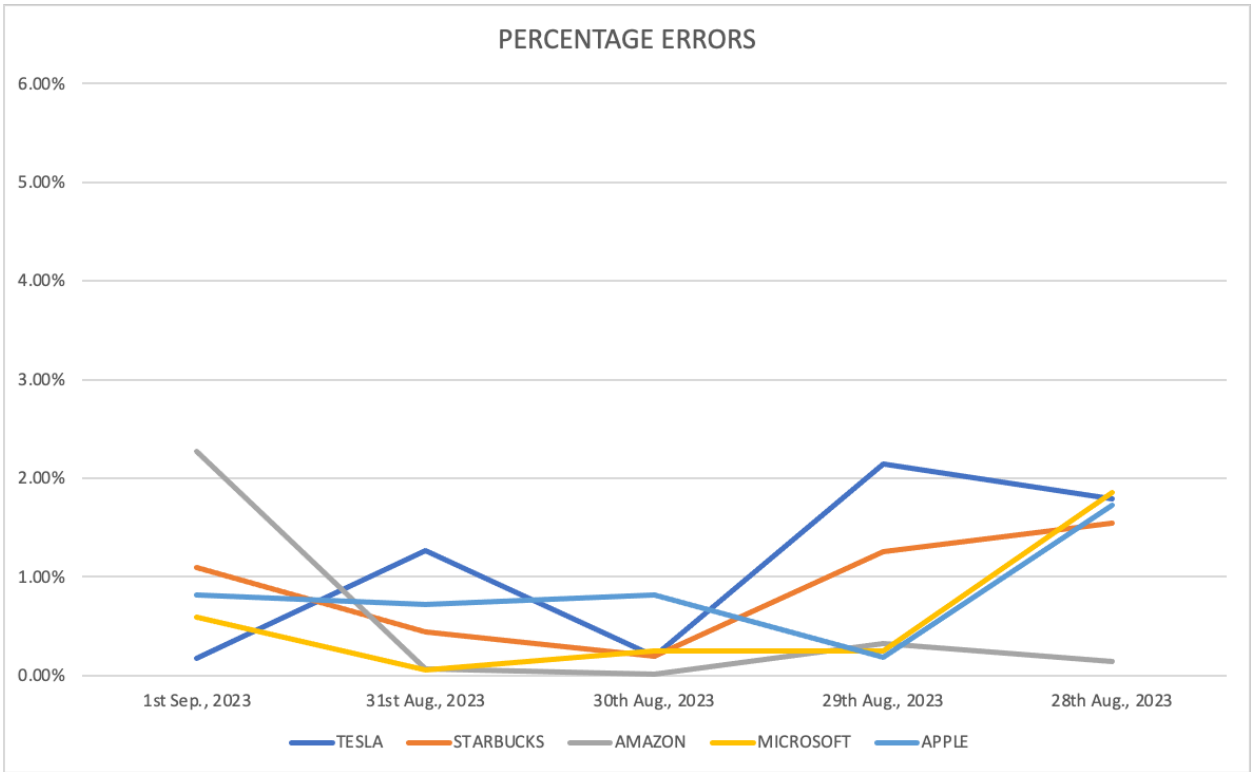
```
# Train-test split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.33, random_state=42)

# Neural Network Model (MLP Regressor)
neural_network = MLPRegressor(hidden_layer_sizes=(100, 50), max_iter=2000, random_state=42)
neural_network.fit(X_train, Y_train)
Y_pred_neural = neural_network.predict(X_test)
neural_mse = mean_squared_error(Y_test, Y_pred_neural)
print("Neural Network Mean squared error: %.2f" % neural_mse)
```

This number was chosen as it provides a sufficient balance to allow the model to train from the data but also avoids excessive training time. The model was trained via the fit method. It adjusts its internal parameters to minimize the mean squared error while training. After training in both models, the model is used to make predictions on the test data ('X_test'). Then, the mean squared error value and percentage difference between the predicted values versus actual test labels are calculated and used as an evaluation metric.

RESULTS

PERCENTAGE DIFFERENCE	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
TESLA	0.17%	1.27%	0.20%	2.14%	1.79%
STARBUCKS	1.10%	0.44%	0.20%	1.26%	1.55%
AMAZON	2.27%	0.07%	0.01%	0.33%	0.14%
MICROSOFT	0.59%	0.06%	0.25%	0.25%	1.86%
APPLE	0.82%	0.72%	0.82%	0.19%	1.73%

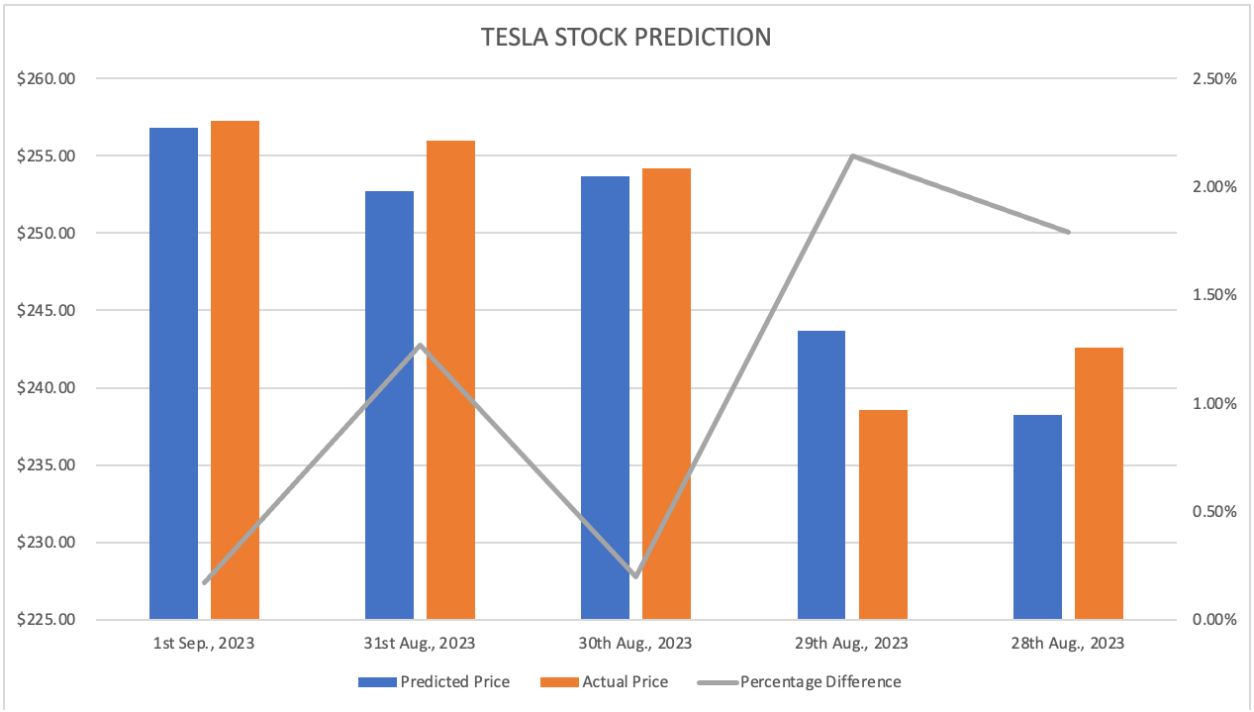


As shown in the table and graph above, we looked at data from 28th August 2023-1st September 2023. A critical part of the study was the evaluation of the result predicted. We compared the actual opening stock prices on those days to our neural network's predicted price. To evaluate the performance of the models, we used mean squared error as well as percentage error. Our percentage errors were low in value, with the highest being 2.27% and the lowest being 0.01%.

COMPANY SPECIFIC OUTCOMES

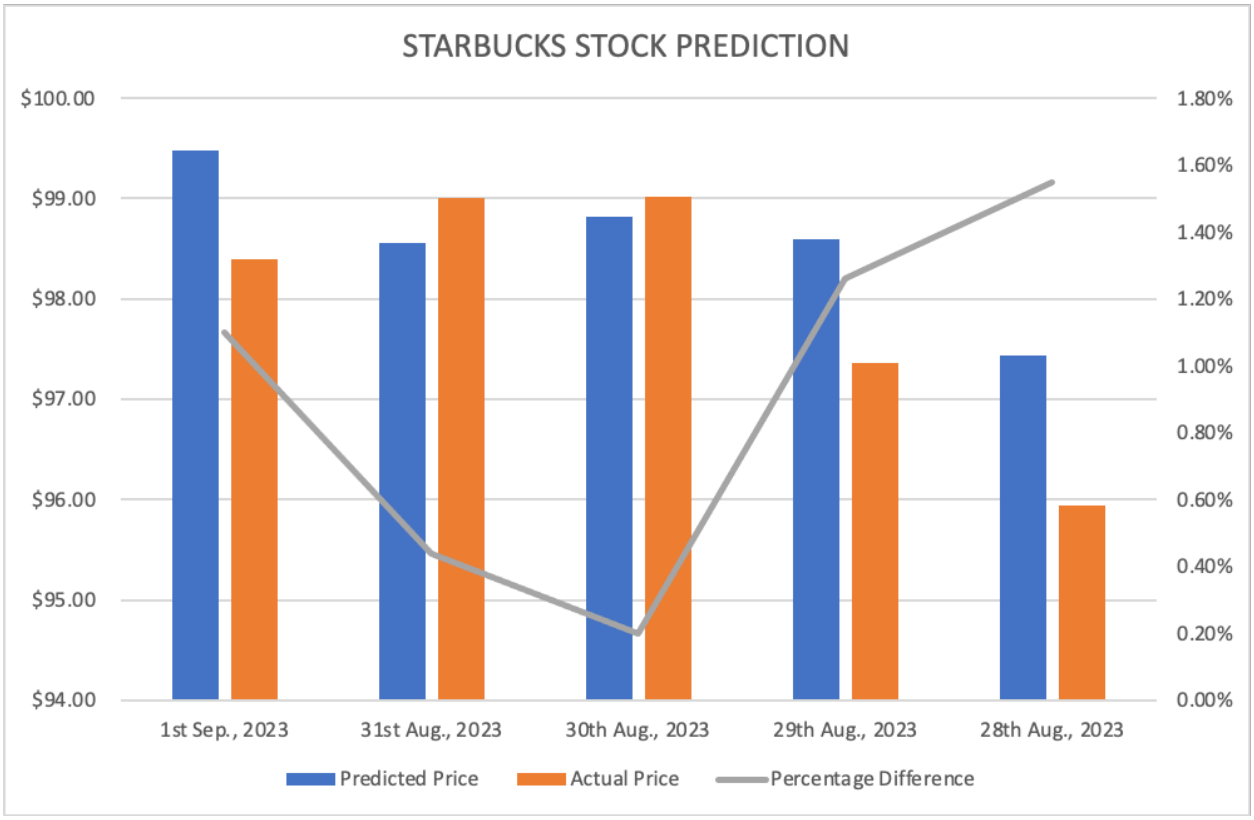
TESLA

Date	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
Predicted Price	$256.83	$252.74	$253.69	$243.69	$238.24
Actual Price	$257.26	$255.98	$254.20	$238.58	$242.58
Percentage Difference	0.17%	1.27%	0.20%	2.14%	1.79%
MSE	116.62	108.83	64.27	642.95	731.7



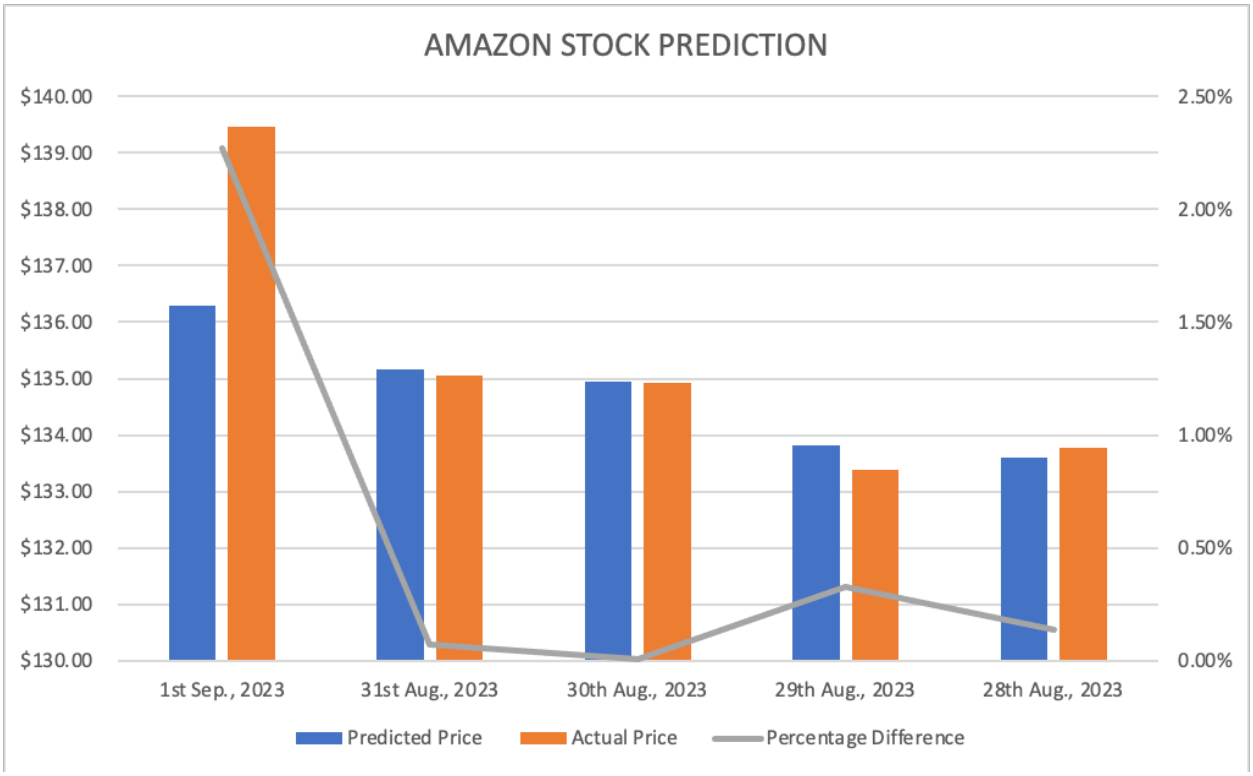
STARBUCKS

Date	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
Predicted Price	$99.48	$98.56	$98.82	$98.59	$97.43
Actual Price	$98.40	$99.00	$99.02	$97.36	$95.94
Percentage Difference	1.10%	0.44%	0.20%	1.26%	1.55%
MSE	4.25	4.2	2.26	19.59	23.64



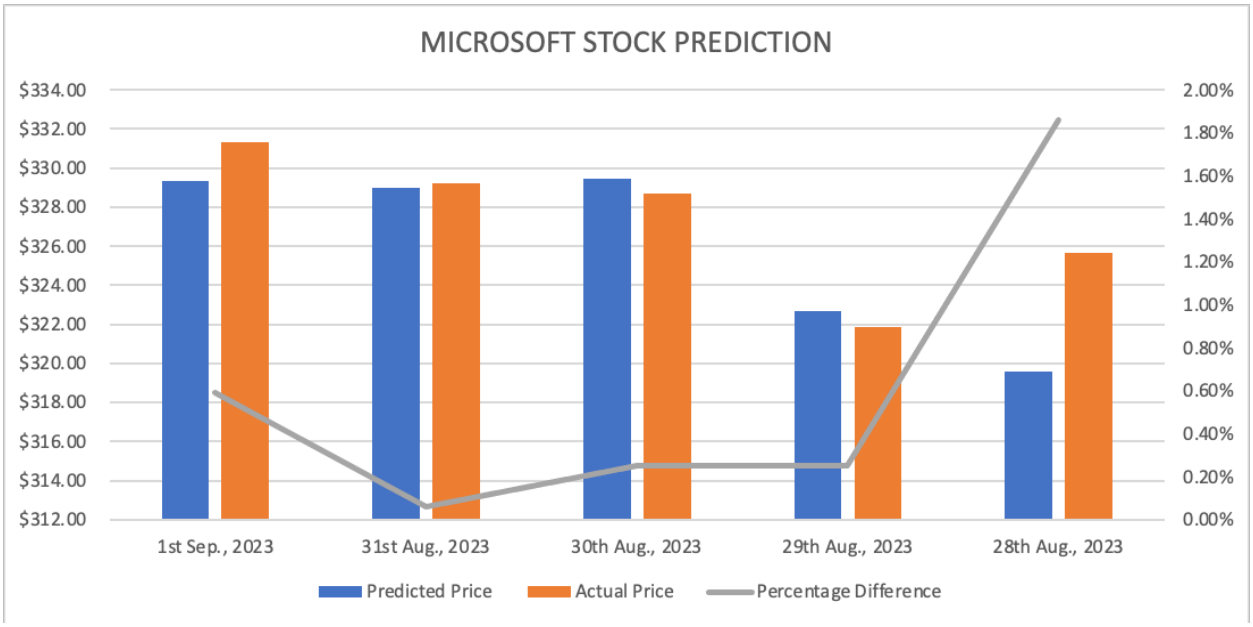
AMAZON

Date	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
Predicted Price	$136.29	$135.16	$134.95	$133.82	$133.59
Actual Price	$139.46	$135.06	$134.93	$133.38	$133.78
Percentage Difference	2.27%	0.07%	0.01%	0.33%	0.14%
MSE	15.09	16.18	9.09	52.97	62.14



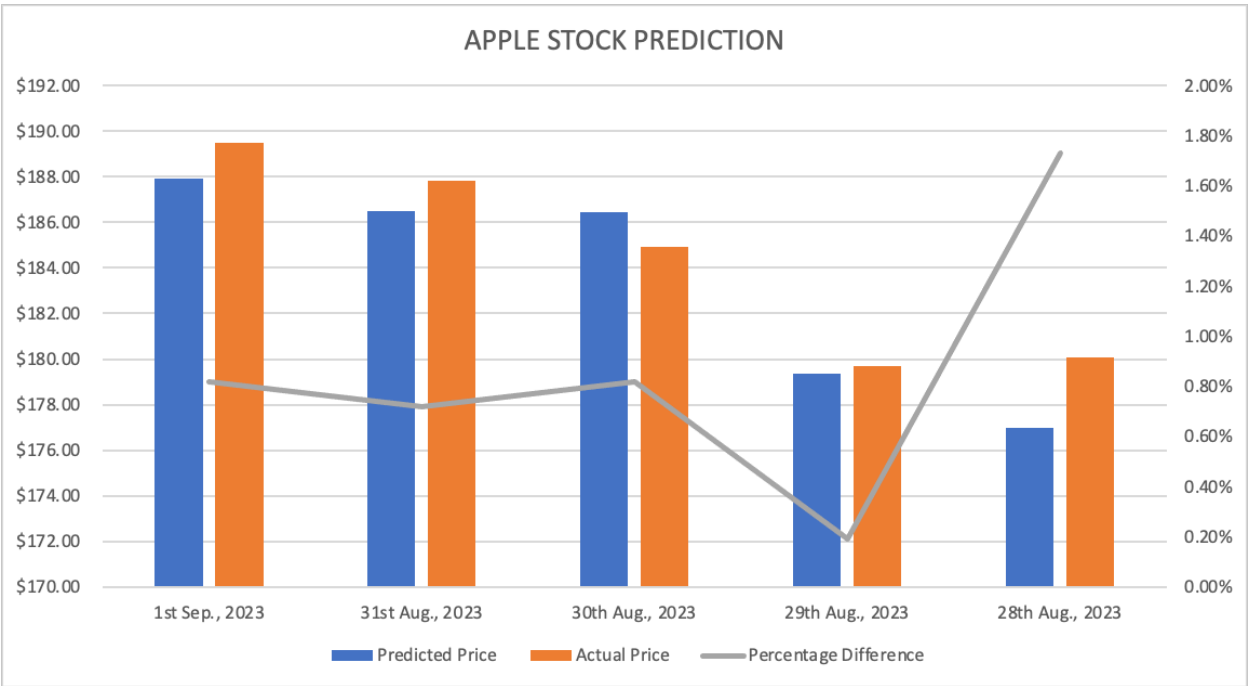
MICROSOFT

Date	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
Predicted Price	$329.36	$329.01	$329.48	$322.69	$319.59
Actual Price	$331.31	$329.20	$328.67	$321.88	$325.66
Percentage Difference	0.59%	0.06%	0.25%	0.25%	1.86%
MSE	33.87	33.2	21.63	319.18	331.74



APPLE

Date	1st Sep., 2023	31st Aug., 2023	30th Aug., 2023	29th Aug., 2023	28th Aug., 2023
Predicted Price	$187.94	$186.49	$186.46	$179.35	$176.97
Actual Price	$189.49	$187.84	$184.94	$179.70	$180.09
Percentage Difference	0.82%	0.72%	0.82%	0.19%	1.73%
MSE	9.85	10.83	6	123.04	128.29



The mean squared error (MSE) was also measured in the table above. It ranged from being single digits to triple digits. The reasons impacting MSE is the size of the companies, which are very large (Apple: $2.7 trillion, Microsoft: $2.45 trillion, Amazon: $1.3 trillion, etc) and the fact that days which had missing data the values were replaced by 0. This indicates that our program did an accurate job in predicting prices.

NEXT STEPS

The following are the possible next steps for improving the project.

A. In our study we used 5 stocks, this can be extended to a larger number of stocks covering large cap, mid cap and emerging companies.
B. The study was using the last 5 years of data. The time period can be extended to cover more years.
C. A study can also be done during bullish or bearish periods for stress testing the portfolio.
D. Sentiment analysis can also be incorporated in the study.
E. The above is the opening price prediction which can be extended for intraday movement such as peak price or lowest price during the day.
F. The neural network can also be enhanced by working on the parameters such as hidden layers and weights.

CONCLUSION

There are many practical implications and potential applications of neural networks in stock price prediction. Large companies like Goldman Sachs are pioneers in using neural networks to predict stock prices. Other players such as Two Sigma, BlackRock, AlphaSense, etc have also been utilizing neural networks in stock price prediction. Therefore this project can act as a great tool in investment decision making. It can provide investors with insight on buying and selling stocks, risk management and investment portfolios. Furthermore, these predictive models offer significant benefits to financial analysts and market regulators, enabling them to monitor market behavior, evaluate economic conditions, and detect irregularities or anomalies in the market. Neural networks can be applied to the wider financial industry. It can forecast exchange rates, yield curves, bond rates and volatility. Additionally, the research contributes to the wider domain of machine learning and data science, providing valuable insights for academic researchers to develop novel methodologies and extend the use of predictive models into diverse fields beyond finance.

BIBLIOGRAPHY

1. "How Do You Calculate the Number of Parameters of an MLP Neural Network?" *Quora*, www.quora.com/How-do-you-calculate-the-number-of-parameters-of-an-MLP-neural-network. Accessed 19 Oct. 2023.
2. Bailey, Regina. "Explore the Anatomy of Neurons." *ThoughtCo*, ThoughtCo, 10 July 2019, www.thoughtco.com/neurons-373486.
3. http://bit.csc.lsu.edu/~jianhua/sam.pdf
4. Fadlalla, Adam, and Chien-Hua Lin. "An Analysis of the Applications of Neural Networks in Finance." *Interfaces*, vol. 31, no. 4, 2001, pp. 112–22. *JSTOR*, http://www.jstor.org/stable/25062724. Accessed 19 Oct. 2023.
5. Saltz, J. S., & Stanton, J. M. (2017). An Introduction to Data Science. Thousand Oaks: SAGE Publications
6. Kotler, Philip. Marketing Management. Upper Saddle River, N.J. :Prentice Hall, 2000.