An AI-based Approach to Stock Market Prediction

Connor Langlois, Abdullah Karim, Eli Legere

Computer Science Department, University of Maine
Orono, ME 04469

connor.langlois@maine.edu
abdullah.karim@maine.edu
john.legere@maine.edu

***Abstract* — This paper describes the creation of different artificial intelligence implementations, specifically neural networks, for stock market prediction. It includes write-ups of the different types of neural networks used and their behavior when interacting with stock market data as well as some statistical analysis of the results gathered.**

***Keywords* — AI: Artificial intelligence**
RNN: Recurrent neural network
LSTM: Long short-term memory
GRU: Gated recurrent unit
ODE: Ordinary differential equation
EPS: Earnings per share
BPS: Book value per share
P/E: Price-to-earnings ratio
P/B: Price-to-book ratio
MACD: Moving average convergence divergence
EMA: Exponential moving average

I. Introduction

The purpose of this paper is to describe how well different neural networks can predict the movement of certain stocks in the U.S. stock market, specifically whether the stock will move up or down from the previous day. We trained and tested these different neural networks on stock market data for Apple Inc. (Ticker: AAPL) and General Electric Corp. (Ticker: GE) to test if their prices would go up or down on a given day. Statistical analysis was run on different models of the neural networks to gather insight on their behaviors and predictive abilities.

II. Background

Trained analysts regard predicting the behavior of a stock as an incredibly difficult task. They take in many factors when making their predictions and so we must do the same with our research while also tying it back to the field of artificial intelligence. An understanding of both finance and artificial intelligence is key when creating a prediction AI for the stock market.

A. The Stock Market

Markets are a place where people can come together and trade goods. At farmers markets, artisan goods are sold for money, and haggling for better prices is not uncommon when buying goods. Capital markets work in the same way as other markets, such as farmers market and pawn shop. The varying factor between these markets is the good that is being bought or sold. In a farmers market, a plethora of fresh products is provided for sale, such as fresh produce. Pawn shops might sell used goods and antiques that are subject to bargain between the buyer and seller. In capital markets, money is traded as an investment with the goal of generating a return on that investment. There are various different capital markets such as the stock market, the bonds market, and the foreign exchange market. This project is mainly focused on stocks in the secondary stock market, where investors exchange stock with each other.

In the stock market, equity or stake in a company is traded in primary and secondary markets via fiat currency through various stock exchanges throughout the world. As the company operates, its stock price will go up or down because of how the company is doing and how investors feel the company will do in the future. The goal is to sell stake in an invested company as it is doing well and its stock price is going up past the price that was invested at, in order to generate a return on investment. If an investment has done well, then a return greater than or equal to the initial investment has been achieved. If it has done poorly, then a negative return is produced and money is lost. This has led many people to find ways to find publicly traded companies to buy stock in to produce a return on their money that is positive. In order to do so, stable companies need to be invested in that operate well. We created this project as an attempt to achieve this goal using various methods rooted in finance as a base for running our models to see if it could predict fluctuations in a stock's price.

When evaluating a company and buying or selling its stock, it is important to combine research from an analysis of a company in various ways. The strategies implemented in this project are ideas that we thought would help in predicting if the stock price of a company would go up or down. The three main types of company analysis we used are fundamental analysis, technical analysis, and sentimental analysis.

1) *Fundamental Analysis*: Companies usually provide financial statements to investors and the public that show how they are performing from a financial standpoint. Fundamental analysis aims to evaluate the financials of a company. Using this type of analysis, we can see how a company is doing in relation to its operations, investments, and financing, which gives us a way to assess if a company is doing well with the money it has by purely looking at its records. Fundamental analysis provides us with many useful ratios that we may use to determine how a stock price will do based on how a company's operations and investments pan out. In this project, we used the fundamental ratios Price-to-Earnings (P/E) and Price-to-Book (P/B) to get an idea of how well a company is growing. The formulae for these are as follows:

$$\frac{Price\ per\ share}{Earnings\ Per\ Share} = \frac{P}{E} \text{ (Price-to-Earnings)}$$

$$\frac{Price}{Shares\ Outstanding} = \text{EPS (Earnings Per Share)}$$

$$\frac{Price\ per\ share}{Book\ Value\ Per\ Share} = \frac{P}{B} \text{ (Price-to-Book)}$$

$$\frac{Book\ Value}{Shares\ Outstanding} = \text{BPS (Book Value Per Share)}$$

$$Tangible\ Assets\ -\ Liabilities\ =\ Book\ Value$$

$$\frac{Market\ Cap}{Shares\ Outstanding} = \text{Price Per Share}$$

The two ratios capture how a company is doing relative to the price it is at currently.

Price-to-Earnings shows a trend in the growth of earnings for a company if it is compared to what the P/E has historically been. It can also be compared to other stocks in the industry to see how it compares to the industry, but this is outside the scope of the project as we are only trying to predict stock price fluctuations. The general idea is that lower P/E is better, within reason, since the P in P/E essentially represents the price paid for a dollar of earnings. Lower P/E's are better, but not always. There could be reasons as to why the stock price is so low, such as investors getting exiting their position because of bad news, tanking the stock price, and showing a lower P/E. Fundamental ratios are not the end of the analysis, because we as investors should not base our analysis solely on one metric because just one metric can't capture everything about a stock and tell where its price will move overall.

We use P/B to relate the growth in a company to its stock price. It is hard to say whether P/B is a good metric for finding undervalued companies to generate a return in, but we believe that over time a high P/B would mean that the valuation of the company is good and would indicate a good buy relative to itself. As with other indicators, using solely this ratio would not create a good metric for if a stock is a good buy or sell and must be paired with other research on how the company is doing.

2) *Technical Analysis*: Technical analysis aims to evaluate the price of a stock based on historical trends that the stock creates. Using technical analysis, we can predict the future value of a stock based on statistics and data gathered about the stock and its price. This analysis uses indicators, usually visualized through stock charts, to predict these stock price movements. The two main indicators used in this project are the Relative Strength Index (RSI) and Moving Average Convergence Divergence (MACD).

The purpose of RSI is to measure how the volume of a stock moves. It's set up like a scale, with it being in the middle. We keep this middle at around 50 percent. This represents an average trading day, where the stock trades exactly as it would on paper with no other factors. We then set an upper bound to the scale which is usually around 70 percent and a lower bound around 30 percent. As money is invested in the stock, the RSI line of the scale will trend towards the upper bound which indicates that more and more investors are buying into the stock. If the RSI line crosses this boundary, it indicates that many investors have bought into the stock and it is a good time to sell if shares are held by an investor. If the line goes in the opposite direction, then that means many investors are selling off shares which could indicate a good time to sell since a mass selloff would indicate the price is coming down. The RSI usually tries to stay at the equilibrium 50 percent, which represents an average amount of trades at any given time.

To understand MACD, knowledge of Exponential Moving Averages is required. Think of exponential moving averages (EMA) as a sliding window over provided stock prices. The sliding window slides over previous daily prices and takes the weighted average of this data, giving more weight to more recent days to produce the moving average for the current day that's being viewed. For example, a 14 day EMA will take the weighted average of the last 14 days, giving preference to the days closer to the present, to produce a moving average for the present day. MACD uses three EMAs, usually having a slower 26 day EMA, a 12 day EMA, and a faster 9 day EMA. The difference between the 12 day and 26 day EMAs is known as the MACD line, whereas the faster 9 day EMA is known as the signal line, whose value is created when crossing over and under the MACD line. If the signal line is above the MACD line, this indicates a buy, and the farther the signal line goes above the MACD line, the stronger the buy signal. Similar is true for the opposite, where the farther below the MACD line the signal line goes, the stronger of a sell signal it becomes. This measures the speed at which prices rise and fall and generates a good metric to track for taking advantage of sell-offs and mass buys in a company's stock.

3) *Sentimental Analysis*: Sentimental analysis aims to measure the behavior of an investor. Here, we look at the publications made about the company being researched. We check the volume of the articles posted and have measurements as to whether the articles published are negative or positive. A net positive sentiment shows that investors like the company while negative sentiment shows that investors do not like the company. Investor sentiment in a company fluctuates for a variety of reasons. For example, a company such as Apple might have positive sentiment if they come out with high sales from a newly released product with new features. They might have negative sentiment if the company produces phones through cheaper labor that leads to bad phone quality. This type of analysis aims to capture the market sentiment created by the various articles that might talk about the qualities of the company that people like or dislike and chooses to quantify that data in a way that is useful for investors.

B. *Neural Networks*

In the field of Computer Science, the term neural network refers to an algorithm loosely modeled on how humans learn. Neural networks are capable of modeling any function and so should also be capable of discerning patterns from stocks in the stock market.

Typically, a neural network consists of layers of neurons, or nodes, where training data is passed through the network to produce a value. Each node between the layers is connected by edges, each with an associated weight. These weights are the key to the network's ability to predict. By giving the network thousands of example input and output data, the weights of the net should progress toward a value that minimizes the network's loss, or error. These layers between the input and the output layers are called the hidden layers

There are many different classes of neural networks. The subclass of neural networks implemented in this project fall under RNN's. RNN's allows sequential data to be modeled.

A. *LSTM*

An LSTM network is a derivation of RNNs where each hidden state builds off the previous hidden states. This is the fundamental architecture of an LSTM; the key difference is how the hidden states are altered.

In an RNN, a neuron performs a single operation on the information. However, in an LSTM a neuron is composed of 3 gates. Gates themselves are composed of a sigmoid, or tanh function, and a vector transformation. A sigmoid function will output a value between 0 and 1 and a tanh function will output a value between -1 and 1.

The first gate in the cell is the so-called 'forget gate.' The forget gate looks at the hidden inputs of the previous cell and the final state of the previous cell and performs a sigmoid operation for each number in the previous cell state. The output determines how much information is going to be retained or forgotten.

The second gate is the input gate. The input gate is composed of two functions. The first is a sigmoid layer that updates the values that were retained from the previous state. The next function takes the previous state values and performs the tanh function on them. Those two sets of values are then multiplied.

The output gate runs the final values through over sigmoid layer. Then these values are run through a tanh layer. These are the values the cell passes on to the next cell.

In performing these operations, LSTM derives the ability to learn sequential data where it may be important to remember long sequences of related data or forget past no-longer-relevant ones.

B. GRU

This network is similar to an LSTM. In a GRU, there are only two gates, the reset and update gates.

The reset gate is a sigmoid function with the input of the previous hidden state. The sigmoid values along with the previous state's values are then passed as inputs into a tanh function. The previous state values are passed into a sigmoid function. The two sets are multiplied by each other. Then the values are passed to the next cell.

III. Methods

A. Task

The extremely important question to ask even before gathering data is: would a regression or classification model better suit this problem? The answer to this is not entirely clear, but due to practical observations seen in the field, we chose classification as the model to learn. Classification is regarded as an "easier" problem than regression due to the limited number of choices the model has to make versus an infinite amount. Thus, classification should allow the model to learn much better than regression and therefore produce better predictions.

To be able to predict the movement of a stock price, classification can be used to divide the movement into two classes:

1.) The stock moves upward
2.) The stock remains stationary or moves downward

By dividing the movement into these two classes, the problem can be cast as a binary classification problem, therein simplifying the problem and in theory producing a better prediction accuracy for these classes.

B. Data Collection

The data collection process is an important part of the process. Data needs to be accurate in order to produce meaningful results. To gather the information needed for the neural networks, we went to the Capital Markets Training Lab at the University of Maine in Orono. They have an educational software called provided by the Bloomberg company that is essentially a huge database of financial knowledge. With the help of this database, we thought about our different types of analysis and created data tables for Apple (AAPL) and General Electric (GE).

While we built our data, we took into account a proper organizational structure and what we needed for the strategy to be input into the neural network. Our data can be seen in the "aapl.csv" file and the "ge.csv" files. We organized the data for AAPL and GE into columns. Our organizational piece is the timestamp that we put as the date column, showing when each piece of data was recorded and creating a series of data over time. We then implemented a "Last Price" column to create our corrections and other indicators associated with our data. We list volume to see how many transactions occurred in a given day and then moved on to technical indicators.

The two technical indicators we used were RSI and MACD. We used these since we thought it would be a good way to incorporate technical analysis while also taking into account trading volume as stocks traded over time. These two indicators are represented in the "RSI" and "MACD" tabs. An important factor to note about the MACD column is that the MACD column only shows the difference between the signal line and the MACD line. We excluded the raw signal and the MACD lines as they were used to produce this difference and because most analysis done with MACD in practice uses the difference to show its direction and strength.

The next two columns show our sentimental analysis. It was extremely difficult to find any news sentiment data that was impactful and had a good amount of historical data. In the end, we decided the length of our other columns based on how much data we could get on our news as it was so scarce. The first of the two columns show the average news sentiment on a given day, which gives an idea of how the stock was performing at that time. The next column shows the publication count on a given day, interpreted as a warning signal if the publication count skyrocketed on a certain day.

The last few columns showed an implementation of ideas from a fundamental view of the company, as well as a correction column for the neural network to use to see if it predicted the stock movement correctly. This data was by far the easiest to get from the Bloomberg database because it is so plentiful and widely used by practical investors.

C. Models

The models that we chose to develop were two key types of neural networks: LSTM and GRU. These networks were chosen as they are specifically geared toward sequential data. Stock market data is strictly time series, meaning that each data point relies on the previous point. In the stock market, the price of one day has a large impact on the price of the next day. Thus, a simple neural network could not be used. Recurrent networks capable of handling this type of data are needed.

The LSTM and GRU neural networks are two well known and widely implemented types of recurrent networks. Due to significant applications backing their implementations and performance, we chose to use these two networks in the prediction of stock market behavior and test their differences in predictive ability.

Specifically, in terms of network architecture, we chose multiple variants of the number of layers and nodes per layer. One keen network developed, as described more thoroughly in the Discussion section below, consisted of three layers: one input layer, one recurrent layer, and one output layer, with the recurrent layer consisting of 10 nodes. This network performed sufficiently for the problem of predicting the movement of the stock price. In each adaptation, however, the input layer consisted of as many nodes as the features, and the output layer consisted of 1 node, representing the binary problem as either a 0 or 1 (or in between as a probability of 1).

D. Implementation

In the implementation phase, Keras backed by Tensorflow-GPU and Python was chosen to be used to develop the network architectures. Python provides an easy high-level language in which to test frequently changing and prototypical network architectures. Furthermore, Keras allows for simple integration with different neural network backends, such as Tensorflow or Theano. Tensorflow, or more specifically Tensorflow-GPU, was chosen due to its involvement with Google's deep learning successes and its speed of learning with the use of a GPU. These factors allowed for different networks to be built swiftly and easily.

For data storing, Pandas combined with Numpy was used to store and manipulate the data. Stock market fundamental, technical, and sentimental data were stored in CSV files and read with Pandas. Pandas provides the necessary data frames required for time series data and allows easy data extraction. Numpy provides speed with its high-level library written in Python that invokes low-level C libraries.

Furthermore, data preprocessing was done in order to allow the neural networks to train as efficiently as possible and overcome local minima. Thus, data was scaled through the use of StandardScaler from the Python package sklearn. StandardScaler normalizes data such that the resulting data has unit variance. By doing so, the network does not give more emphasis to features with a larger range of values and is able to learn more effectively and evenly across features. Additionally, as the data is stored in a data frame, TimeseriesGenerator from the Keras preprocessing package was used to reorganize the data into the correct shape expected by the LSTM and GRU networks. That is, it reformats the time series data into the shape (samples, steps, features), where "samples" is the number of total samples of length "steps" with "features". We chose 14 as the number of steps due to its prevalence in the financial technical analysis world as a sufficient amount of lookback.

Additionally, the data was split into train and test sets. The train set consisted of 90% of the training data, leaving 10% for the test set. This was done to ensure that the networks could train on a large enough portion of the data to be able to effectively learn, while still leaving enough data to test their accuracy. Considering only 3 years of data was able to be gathered, it is important to allow the networks as much data to train on as possible.

For the network models described above, each was implemented through the use of Keras. The activation function chosen for each layer was ReLU due to its ability to remove the effect of the vanishing gradient problem and its popularity in the deep learning community. Additionally, a sigmoid activation was done on the output layer to bound the network's output to the range [0, 1]. This is important as the response variable is either a 0 or 1, a binary classification problem. Thus, sigmoid ensures that the response is binary.

On top of that, the binary cross entropy loss function was used as this is a binary classification problem. Also, an Adam Optimizer was chosen due to its practical prevalence and benefits of updating the learning rate for each parameter with an exponential moving average. This allows for more efficient learning compared to traditional methods such as stochastic gradient descent.

One interesting feature of the network architectures was the inclusion of a BatchNormalization layer after each recurrent layer. This layer's purpose is to mimic that of the StandardScaler, which was done on the inputs, but on the output of the layer above it instead. By doing so, the network not only receives normalized data but continually normalizes it (0 mean and unit variance) after each layer to ensure values are scaled evenly when input into the next layer. Once again, this gives the advantage of increased learning efficiency and accuracy.

For training the networks, 300 epochs were used incrementally in order to determine the best cutoff point. This was determined where the testing accuracy was at its maximum over the 300 epochs. Therefore, in deploying the models, these numbers of epochs should be used in the training phase.

IV. Results

TABLE I
AAPL LSTM Prediction Accuracy

Epoch	Train Accuracy	Test Accuracy
1	0.51	0.61
2	0.51	0.6
4	0.52	0.6
7	0.54	0.61
13	0.56	0.61
24	0.58	0.57
45	0.61	0.62
84	0.64	0.62
159	0.68	0.6
300	0.73	0.59

GRAPH I

AAPL LSTM Prediction Accuracy

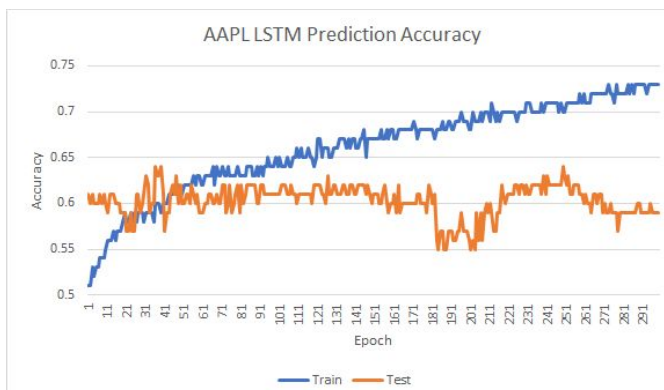


TABLE II
AAPL GRU Prediction Accuracy

Epoch	Train Accuracy	Test Accuracy
1	0.49	0.49
2	0.5	0.46
4	0.5	0.49
7	0.53	0.47
13	0.56	0.47
24	0.6	0.44

45	0.63	0.43
84	0.67	0.51
159	0.72	0.52
300	0.76	0.48

GRAPH II

AAPL GRU PREDICTION ACCURACY

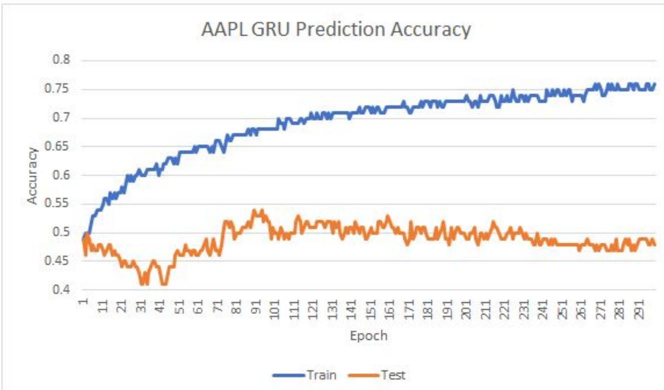


TABLE III
GE LSTM PREDICTION ACCURACY

Epoch	Train Accuracy	Test Accuracy
1	0.52	0.53
2	0.52	0.53
4	0.52	0.54
7	0.55	0.56
13	0.57	0.58
24	0.58	0.55
45	0.62	0.57
84	0.66	0.56
159	0.72	0.53
300	0.76	0.54

GRAPH III

GE LSTM PREDICTION ACCURACY

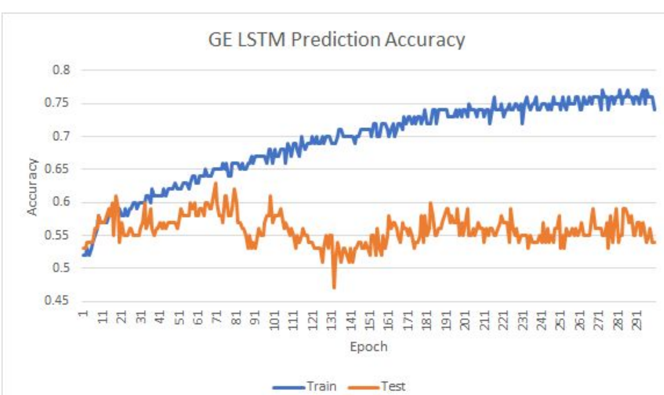
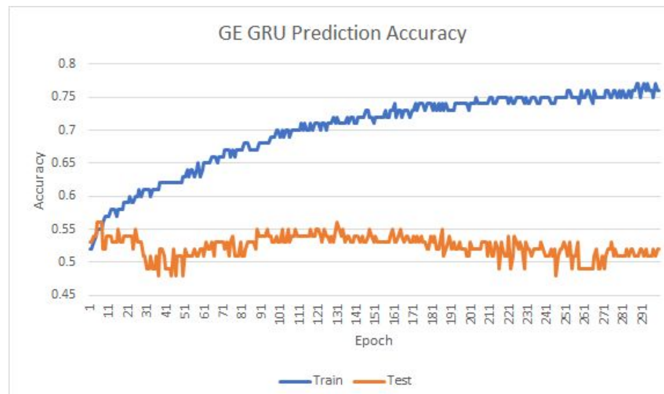


TABLE IV
GE GRU PREDICTION ACCURACY

Epoch	Train Accuracy	Test Accuracy
1	0.52	0.53
2	0.52	0.53
4	0.54	0.54
7	0.55	0.56
13	0.58	0.53
24	0.59	0.52
45	0.62	0.52
84	0.68	0.53
159	0.73	0.54
300	0.76	0.52

GRAPH IV

GE GRU PREDICTION ACCURACY



V. DISCUSSION

One interesting result we gathered was that simple networks that consisted of only three layers performed the best. That is, networks will 1 input layer, 1 recurrent layer, and 1 output layer performed significantly better in terms of testing accuracy than those will a higher number of recurrent layers.

This can be attributed to the fact that given a more complex network, there is a much higher number of weights that need to be learned. Thus, since the data we collected was not large, the networks were unable to learn these higher level weights at early layers of the network. Even with an increase in epochs, the network was unable to learn effectively.

A. LSTM Network

The LSTM network that we employed performed rather well on the data supplied. Overall, as can be seen in *Table I* above, the training accuracy increased from around 50% to 74%, and the testing accuracy stayed rather stationary at 60% accuracy throughout. This is an interesting effect encountered. It would be expected that the testing accuracy would increase relatively monotonically with the number of epochs (before the model becomes overfitted). However, the testing accuracy seemed to oscillate around 60% accuracy, jumping above and below at times. Thus, the best cutoff point we determined to be around 64% at 36 epochs.

B. GRU Network

The GRU network, on the other hand, performed poorly. As seen in *Table II*, the training accuracy increased from around 50% to 74%, while the testing accuracy, similar to the LSTM, stayed relatively stationary at 50%. Different than the LSTM, however, the GRU's testing accuracy did not oscillate as strictly as the LSTM. Instead, it produced a rather smooth descent and climb until ultimately leveling off around 50%. Thus, the best cutoff point is determined to be 90 epochs.

C. Apple vs. GE

In creating neural networks for both Apple and GE stock, it appears both have similar outcomes in terms of prediction accuracy. The LSTM network seemed to fit better to Apple by a slight margin, resulting in its test accuracy hovering around 60%, while GE hovered around 55%. On the other side, the GRU for GE did not dip as much as for Apple, but both reached a similar peak prediction accuracy.

D. Scope and Limitations

This research was limited by a few factors:

1. The amount of data we were able to gather was not necessarily sufficient. For instance, the news sentiment data was scarce. We were only able to find news sentiment on stocks dating back to 2015 and thus had to cut a significant amount of data that originated prior to 2015.
2. Because the data was difficult to gather, we were only able to test the networks against two stocks. Therefore, we do not know for certain if they would work similarly on other stocks.
3. A model will most likely only work on the stock that it is trained with. Each stock behaves slightly different and so a model trained on one stock would have inaccurate predictions for another.

E. Future Research

The work we accomplished set the groundwork for further studies. Those studies and research can be centered around the questions that this paper did not address.

Although we are confident that the LSTM and GRU networks would produce similar results on different stocks, this needs to be tested. The testing procedure would be identical to the process described by this paper.

Future research will focus on generating long-term predictions. Currently, the LSTM and GRU networks predicate whether or not the price will go up or down on a day by day basis. This is not necessarily useful for long-term investments. Therefore the next step in development will be creating a neural network that can generate buy-sell signals for a long-term strategy. For that task, we may need to resort to other methods of data gathering and other types of networks.

The next steps will also relate to refining the current networks to be faster, more efficient, and, more accurate. To achieve those goals we will be implementing a neural ODE network. A neural ODE is based on the fact that RNNs, in general, can be generalized by the equation.

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t, \theta_t)$$

In this equation *h* is the hidden state, *t* is the timestep and theta represents the weights. This is similar to Euler's method. Therefore by taking infinitesimally small timesteps, we can rewrite the equation to be

$$\frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t, \theta)$$
.

This equation is an ODE. Therefore, if *h(0)* is known then *h(T)* can be deduced. With the *adjoint-sensitivity method,* we can compute the gradient of the loss function. This process is described in the paper, 2018, by T. Q. Chen et al.

With an ODE it might be possible to generate a continuous-time generative model in spite of the irregular data gathered from the stock market. If successful, an ODE may prove to be a more effective method of predicting stock price movement. Such a model would prove useful in long term predictions.

Acknowledgments

The authors wish to acknowledge Roy M. Turner for his teachings and guidance in Artificial Intelligence. We would also like to acknowledge Bloomberg for aiding us in gathering this data, and the Maine Business School offering classes that taught us preliminary financial knowledge about finance.

References

[1] Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed., Upper Saddle River, New Jersey, United States: Prentice Hall Press, 2009.

[2] Roy M. Turner, *MaineSAIL*, http://mainesail.umcs.maine.edu/COS470

[3] Bloomberg L.P. "Stock price and fundamental, technical, and sentimental analysis data for Apple Inc. and General Electric 01/02/15 to 04/12/19." (2006). *Bloomberg database*. The University of Maine at the Capital Markets Training Lab, Orono, ME. 12 April 2019.

[4] Chen, James. "Relative Strength Index - RSI." *Investopedia*, Investopedia, 18 Apr. 2019, www.investopedia.com/terms/r/rsi.asp.

[5] Chen, Ricky T.Q., et al. "Neural Ordinary Differential Equations." University of Toronto, Vector Institute, 15 Jan. 2019.

[6] Hayes, Adam. "Moving Average Convergence Divergence (MACD)." *Investopedia*, Investopedia, 13 Apr. 2019, www.investopedia.com/terms/m/macd.asp.

[7] Hayes, Adam. "What the Price-to-Book Ratio (P/B Ratio) Tells Us." *Investopedia*, Investopedia, 20 Apr. 2019, www.investopedia.com/terms/p/price-to-bookratio.asp.

[8] Hayes, Adam. "What the Price-to-Earnings Ratio Tells Us." *Investopedia*, Investopedia, 20 Apr. 2019, www.investopedia.com/terms/p/price-earningsratio.asp

[9] Olah, Christopher. "Understand LSTM Networks." *Colah's Blog*, Github, 2015, Christopher Olah.

[10] Kostadinov, Simeon. "Understanding GRU Networks." *Towards Data Science*, Towards Data Science, 16 Dec. 2017, towardsdatascience.com/understanding-gru-networks-2ef37df6c9be.