



## Academic Research Review

### Algorithmic Trading using Neural Networks

#### EXECUTIVE SUMMARY

In this paper, we attempt to use a neural network to predict opening prices of a set of equities which is then fed into an AI system to make optimal trade decision. The data for the neural network is collected on basis of three parameters namely fundamental analysis, technical analysis and sentiment analysis thereby mimicking the way in which discretionary traders make their trading decisions.

#### INTRODUCTION

Artificial neural networks (ANN) is a branch of machine learning inspired by the structure and function of the brain. ANNs are capable of learning high-level abstractions by using a deep graph with multiple processing layers trained with large amount of data. One of the real world applications of neural networks is image recognition which has undergone a revolution due to ANN's ability to find distinctive patterns and features from noisy data. Unlike other machine learning algorithms, which tend to plateau in performance with more data, the performance of neural networks scales well with the volume of data.

#### DATA FORMULATION AND MODEL PARAMETERS

We resorted to a popular machine learning quote, "Don't model the World; Model the Mind." for data formulation. We wanted to emulate the way in which discretionary traders work and hence we explored different indicators they take into consideration while making a trading decision. There are three aspects, which stands out, and we formulated the data for our system based on these:

- Fundamental Analysis (FA), which refers to the core financials of the company.
- Technical Analysis (TA), which refers to different ways in which a stock is moving in the market with indicators like trend, momentum etc., and
- Sentiment Analysis (SA), which refers to the positive/negative sentiment that is prevalent for a company or a country usually, obtained from news/social media.

For the purpose of the study, we have considered seven stocks from S&P 500 taking market capitalization and industry diversification into mind. They are Berkshire Hathaway (BRK.A), Exxon Mobil (XOM), Microsoft (MSFT), General Electric (GE), Walt Disney (DIS), Johnson&Johnson (JNJ) and JP Morgan (JPM).

The table below explains all the input features that is considered for the model,

<b>Aspect</b>	<b>Indicator Type</b>	<b>Feature Name</b>	<b>Formula</b>
<i>Fundamental Analysis</i>	Balance Sheet	Market Valuation	Outstanding Shares * Current Price for one share
	Valuation Ratio	Price-Earning (PE) Ratio	Adjusted Close Price/ Earnings Per Share
	Valuation Ratio	Price-to-Book (PB) Ratio	Adjusted Close Price / Book Value Per Share
<i>Technical Analysis</i>	Operation Ratio	Financial Leverage	Total Assets/Common Equity
	Balance Sheet	Accounts Receivable	Accounts owed to the company within a year
	Momentum	Relative Strength Index (RSI) – 14 period model	RSI = $100 - (100 / (1 + RS))$  Where RS = Average Gain/Average Loss
	Volume	Volume Rate of Change (ROC) oscillator	$((Volume - Volume\ n\ periods\ ago) / Volume\ n\ periods\ ago) * 100$
	Volatility	Bollinger Bands	20 day simple moving average
<i>Sentiment Analysis</i>	Trend	Moving Average Convergence Divergence (MACD)	12 day exponential moving average – 26 day exponential moving average
	Sentiment Index	Bloomberg News Sentiment Index	Provides a numerical value for the news sentiment for a stock according to its current news polarity

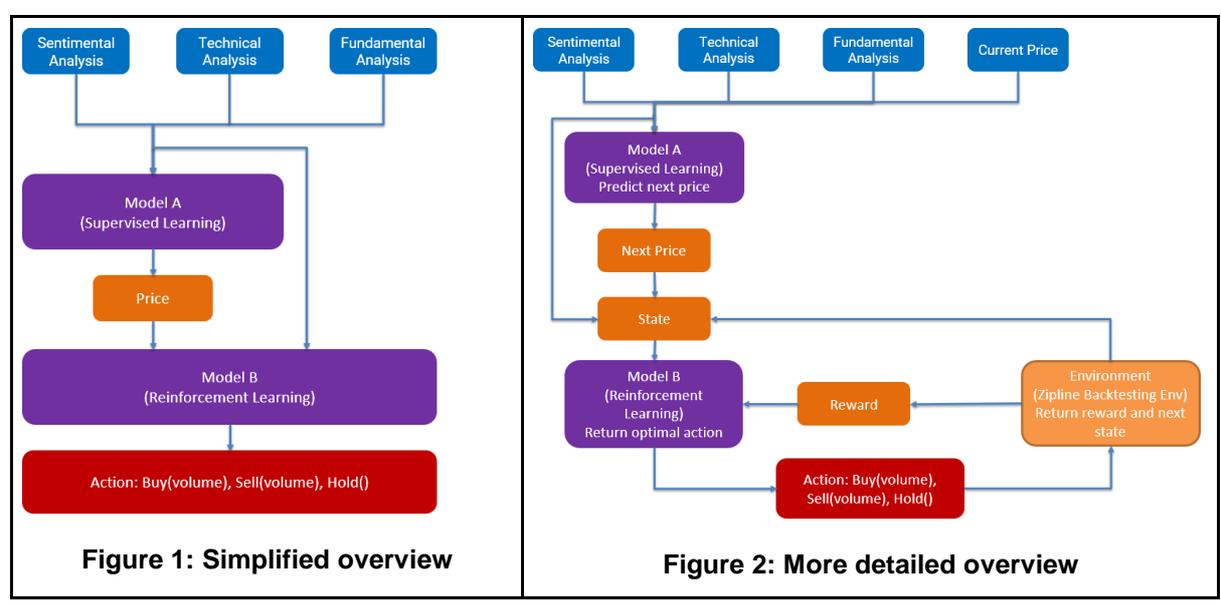
**Table 1: Input Feature for Price Model**

We collected data for the period from 5/9/2016 until 9/12/2016 for the seven asset classes mentioned in the preceding paragraph. We gathered the technical indicators and sentiment index from Bloomberg whereas Quantopian, which provides Morningstar Fundamentals, gave us the fundamental indicators. The first four months was taken as the training period for the model while the last two months were left for back testing. The predicted prices that were obtained after training the model were back tested for the last two months and the returns obtained during the period will be presented as final results.

## METHODOLOGY

The system described below is used to generate the optimal trading decision (buy, hold, sell) for a single security. The system can be generalized to trading a basket of securities.

### System Overview



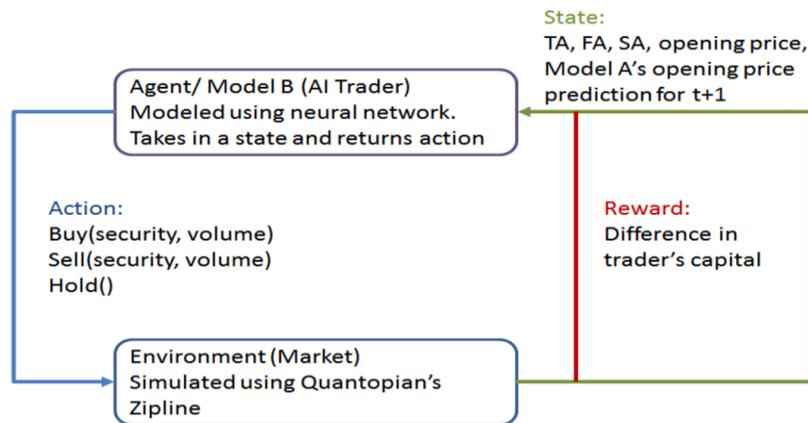
### Model A – Price Model

The role of model A is to generate a prediction for the security’s price in the next time step(minute). Model A will be trained with supervised learning algorithms using historical pricing data. While having a prediction of the next price is useful, it does not tell us the optimal action to take. A simple strategy based on model A would be to buy when price is predicted to rise and sell otherwise. However, we can do better by providing the prediction to another model which decides on the optimal action for us. This brings us to model B.

### Model B – Decision Model

Unlike the training of model A, we cannot train model B (the AI agent) using supervised learning approach as it is not possible to know the correct action to take in any given state. Therefore, the agent has to be trained using reinforcement learning where the agent can learn the optimal action to take by interacting with the environment. We will be using Quantopian’s Zipline backtesting to simulate the environment as it provides a realistic trading conditions for our agent to learn the optimal actions. The next state returned by the environment will be augmented with model A’s prediction of the next price, the agent’s current capital and

the features from the three feature classes (technical analysis, fundamental analysis and sentimental analysis).



**Figure 3: Interaction between model B and the environment**

In reinforcement learning, the agent's goal is to learn an optimal policy that allows it to collect as much reward as possible in the long run. Therefore, it is very important that the reward is defined such that it aligns with our goal of maximizing profit. As of now, the reward is defined as the difference in trader's capital from the current time to the next time step.

### Model Architecture

The model has to be able to learn long term dependencies as a change in price could be caused by events that happened long ago. To effectively model the market behaviour, we used long short-term memory (LSTM) network for the price model. LSTM networks are capable of learning long term dependencies which is something traditional recurrent neural networks have troubles learning.

## IMPLEMENTATION AND RESULTS

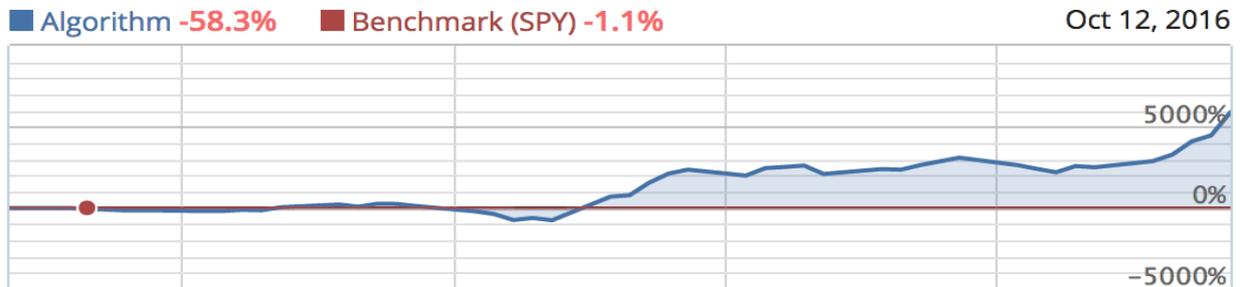
The methodology mentioned above was implemented in Python. The neural network was implemented with Keras while backtesting was done in Quantopian, an open-source algorithmic trading library for Python. The price model (Neural Network Model) provided us with predictions for (t+1) day for each security which was utilized for back testing in Quantopian's environment.

Due to unfamiliarity with the Quantopian library, we were unable to implement the decision model which requires close interaction with the backtester. Therefore, we opted for a simpler rule based decision model. The trading strategy that was adopted during back testing varies according to the risk appetite of the trader/investor. We tested a risk-seeking and risk-averse strategy for the seven stocks for the back-testing period from 07/10/2016 to 09/12/2016. In the risk averse scenario, the trader/investor will make a BUY call for any security, only when the predictions of the stock is positive. When the predicted prices go negative, the trader/investor does not make any decision. In a risk seeking scenario, the trader/investor will make a BUY call for any security when the predictions of the stock is positive or when it the negative predictions are within a threshold value, -0.5 in our case. When the predictions go beyond -0.5, no trade call is made.

We obtained the below results in back-testing time period by using these strategies,

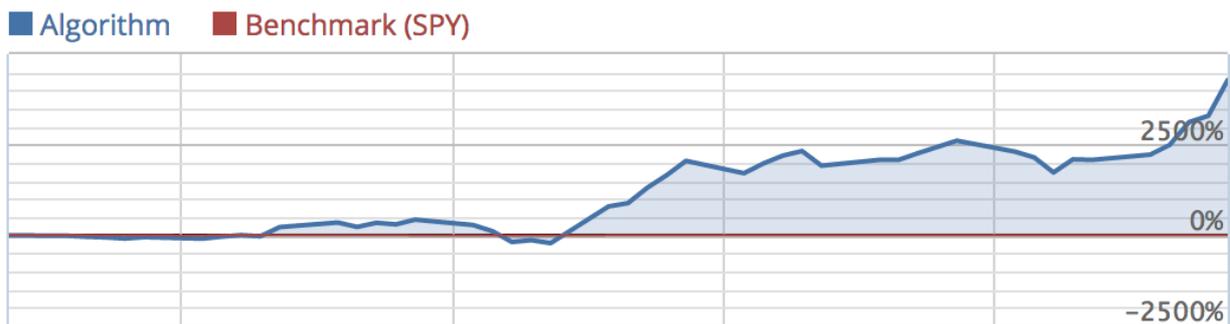
### Risk Seeking Strategy

RETURNS	ALPHA	BETA	SHARPE	DRAWDOWN
5903.2%	61.09	-55.49	1.74	-73.7%



### Risk Averse Strategy

RETURNS	ALPHA	BETA	SHARPE	DRAWDOWN
4282.5%	173.34	-8.39	2.42	-97.8%



Using the initial capital of \$1,000,000, the backtester invests a proportionate capital for every minute on each trading day in a security that will give the highest return for a 30-minute window. The trading algorithm will rebalance its investments after every 30 minutes and then reinvest the gains on the same basket of securities. This strategy was repeated for a period from 07/10/2016 to 09/12/2016 and the above results were obtained.

Both the strategies tend to be profitable, with risk-seeking investor getting better returns than the risk-averse investor does. The transaction costs are set to be \$7 per trade and the slippage spread has been set to be 0.01, both taken in accordance with FINRA's recommendation.

### REASONS FOR EXORBITANT RETURNS

The high returns that we get through our trading strategy can be possibly explained through the below factors:

- **High Frequency Strategy** – the trading algorithm executes about 25-30 trades per minute on average during the two month backtesting period.

- **Portfolio Rebalancing** – the algorithm optimizes the portfolio weights every 30 minutes to ensure that high returns are obtained from every trade. As a result, the algorithm takes note of various market movements and ensures that its investment at that point in time can fetch maximum returns
- **Long-only Strategy** – we only “buy” different quantities from our basket of securities depending on its market movement. So, the gains obtained through these trades are aggregated and is continually reinvested in the portfolio.
- **Diversity in Securities** – the basket of securities are chosen diligently to ensure that there is no significant correlation between them. Therefore, at majority of occasions, the algorithm had an option to make a trade decision that would give a high return.
- **Backtesting Period** – the time period in which the strategy was tested was from October to December 2016. There were a lot of macro-economic changes during this period that greatly influenced the returns obtained. The algorithm begins to gain significantly after US elections during which the S&P 500 also increased considerably on positive market outlook.

## CONCLUSION AND IMPROVEMENTS

While the prediction from the price model were accurate enough for us to turn a profit by utilizing a simple rule based decision model, there are areas for improvements. First, the model could use a larger dataset for training. This would allow us to use a more complicated model and avoid over-fitting. Another improvement would be to use the neural network decision model that was planned but not implemented.

## REFERENCES

1. Applying Deep Learning to Enhance Momentum Trading ... (n.d.). Retrieved October 29, 2016, from <http://cs229.stanford.edu/proj2013/TakeuchiLee-ApplyingDeepLearningToEnhanceMomentumTradingStrategiesInStocks.pdf>
2. Arévalo, A., Niño, J., Hernández, G., & Sandoval, J. (2016). High-Frequency Trading Strategy Based on Deep Neural Networks. Intelligent Computing Methodologies Lecture Notes in Computer Science

## Algorithmic Trading using Neural Networks

*Research Analysts: Ram Thilak Prem Kumar, Vincent Seng Boon Chin*

---

This research material has been prepared by NUS Invest. NUS Invest specifically prohibits the redistribution of this material in whole or in part without the written permission of NUS Invest. The research officer(s) primarily responsible for the content of this research material, in whole or in part, certifies that their views are accurately expressed and they will not receive direct or indirect compensation in exchange for expressing specific recommendations or views in this research material. Whilst we have taken all reasonable care to ensure that the information contained in this publication is not untrue or misleading at the time of publication, we cannot guarantee its accuracy or completeness, and you should not act on it without first independently verifying its contents. Any opinion or estimate contained in this report is subject to change without notice. We have not given any consideration to and we have not made any investigation of the investment objectives, financial situation or particular needs of the recipient or any class of persons, and accordingly, no warranty whatsoever is given and no liability whatsoever is accepted for any loss arising whether directly or indirectly as a result of the recipient or any class of persons acting on such information or opinion or estimate. You may wish to seek advice from a financial adviser regarding the suitability of the securities mentioned herein, taking into consideration your investment objectives, financial situation or particular needs, before making a commitment to invest in the securities. This report is published solely for information purposes, it does not constitute an advertisement and is not to be construed as a solicitation or an offer to buy or sell any securities or related financial instruments. No representation or warranty, either expressed or implied, is provided in relation to the accuracy, completeness or reliability of the information contained herein. The research material should not be regarded by recipients as a substitute for the exercise of their own judgement. Any opinions expressed in this research material are subject to change without notice.