Alexandru Bordanca

**A Reinforcement Learning and NLP Based Approach to Algorithmic Trading**

**Abstract**

The past year has been markedly different from previous year in terms of financial markets and the events that transpired within them. We have seen a large influx of brand-new investors, wild swings in prices of consistent benchmarks, blue chips, and speculative investments alike, and an unprecedented level of collusion between retail market participants that has driven up the prices of securities that arguably have no real fundamental value. Algorithmic trading has been used both academically and professionally for the last few years on a rather large scale to varying results; it has been praised for taking the emotion out of investing that can both plague novices and hardened investors, but has been criticized for inconsistent results and for having a significant amount of drawdown, that can put portfolios at risk for being completely wiped out in a matter of hours. The general behavior of such trading bots can be succinctly put: they either grossly overperform or grossly underperform. I then became interested on how a trading bot based on different algorithmic trading models could perform in a radical financial environment like the past year or so since the onset of the pandemic, particularly the early part of 2021 seeing the rise of formerly dormant stocks like GameStop and AMC. Given that these are more or less “memes” and are propped up by game-theory like collusion, I thought it would be a challenged. Ultimately the model provided a strong return, doubling portfolio value in a little over two years.

**Introduction**

Though algorithmic trading and reinforcement learning as it applies to finance are barely in their adolescence, the theoretical methods behind them have been around for decades. My exposure to it began when I started trading cryptocurrency and forex where there is an abundance of trading bots that are used both institutionally and on the retail side. It is estimated that the overwhelming majority (92%) of trades in the forex market are placed algorithmically. One of the more significant academic studies of algorithmic trading in a simulated securities market is Ritter in 2017. This uses one of our RL methods of Q learning to properly assign reward function that mitigate the risk involved with these high frequency trades. It focuses on Sharpe ratio, a metric that I focused on heavily in selecting an appropriate model, indirectly in examining the shape of the utility function of investors showing that they prefer a more risk averse portfolio to one with a similar return but higher risk. His model showed potentially high returns by using RL to find and exploit arbitrage while satisfying the risk minimization condition. Another key paper that examined the use of algorithmic trading, this time in the precious metals market, was Dunis and Nathani in 2007. Their research consisted primarily of the use of neural network methods in trading such as Multilayer Perception, Higher Order Neural Networks, and K Nearest Neighbor. Their research similarly showed excess returns in the trading of the gold and silver markets. In light of the wallstreetbets sensation I wanted to see how well an easily put together model could trade the same time period.

Based on the recent financial climate and the previous body of work, I pose the following question: Can a trading bot trade the same stocks as wallstreetbets and outperform a benchmark like the S&P 500 and if so how consistently? I posit that while it can certainly outperform, it’s consistency can certainly be refined.

**Methods**

Data was collected using Yahoo Finance, which has a built-in interface in the FinRL package through YahooDownloader, which while slightly increasing runtime avoids having to clean and store data in csv files. I focused on data for three stocks, GME, SNDL, and AMC as these were big movers in the wallstreetbets movement. These were loaded into data frames, cleaned to focus on a specific date range of January 1st, 2000 to April 1st, 2021. These were then split into test and training data sets for the model to work on. Using the feature engineer from FinRL, the data was then expanded to include technical indicators for each relevant stock on each trading day. These indicators included MACD, RSI, Bollinger bands, etc. An environment was then designed to constrain the action space, so I included costs for buying and selling stocks, to replicate real life trading commissions, and intimal capital of $100,000. I then trained 5 models (TD3, SAC, DDPG, A2C, PPO), all built into the FinRL framework and only moved forward with the one that yielded the highest Sharpe ratio of the 5. This was necessary as the models take a long time to run, greatly adding to total runtime and this cut down time significantly in retesting and modifying the program. Early runs of the program showed almost 90% as maximum drawdown, though while it had stronger returns than the final model, exposed the portfolio to too much overall risk, potentially wiping it out. Thus I incorporated a turbulence variable, which the built in model allows for thus allowing us to reduce risk to the portfolio. I then ran the model on the test data set from January 1st 2019 to April 1st 2021, and then back tested the results, which are discussed below.

**Results**

Overall, I feel that the model that was selected for testing (TD3), which is a twin delayed DDPG performed rather well, considering the market environment that it was trading in. Figure 1 shows a high-level overview of key metrics of the trading results of the model. We see a cumulative rate of return of approximately 95% or approximately 35% annually, which is strong. When compared to the S&P 500 it has higher returns, a higher Sharpe ratio, and about less drawdown, the combination of which indicates the model realizes higher returns for substantially lower risk. The same statistics for the baseline S&P 500 are shown in Figure 2. It is also worth examining not only how well it performed overall, but when it performed well and when it underperformed or took severe losses. Figures 3 highlights this. Figure 3 highlights how the performance was underwhelming for the duration of the test period, however the early part of 2021 shows significant returns. In large part this can be attributed to a general lack of movement in the asset price of the respective stocks (GME in dormancy, and SNDL not public until August 2019). This confirms previously held beliefs that trading bots do poorly in flat markets and can potentially excel in bull or bear markets with a great degree of momentum. Figure 4 shows that rather than a necessarily poor performance, it remained flat for much of the dormant parts of the test period, so on the whole it can be considered a successful trading period. Finally, we can examine Figure 5 that measures the bot’s beta over time. Beta is a measure of how an asset or portfolio moves in comparison to the general direction of the market. Based on this we can see increases in beta in the mid part of 2020 through the remainder of the test period, yet again confirming the bot’s superior performance in the bull market. Based on these metrics, while there is certainly room for refinement of the model and its parameters, I feel that this bot can be considered an overall success.

**Discussion**

While overall, I believe the project was a success, it merits we discuss its limitations. First there is still a significant way to go from this point to a full-fledged bot that can run independently and make trades in real time. Secondly, there is a great deal of room to refine the Sharpe ratio and ultimately the return. Given that there is a tradeoff between returns and drawdown or risk, the model as a whole would benefit from a more thorough optimization of the turbulence threshold.

Third, it is also worth keeping in mind that these returns are realized trading exclusively “meme” stocks, so to the merit of the model, it might be worth exploring and testing its performance in trading more blue-chip stocks. Since those securities are generally more active with high volume and good liquidity, that would create a better training scenario for the model to base future predictions are and potentially better understanding a flat market. In the same light, it is important to consider the performance of the model in the particular market period it acted in. The movements of the stock prices in the early part of 2021 were neither fundamental nor technical, but rather completely artificial and driven by a mass collaboration. Bearing that in mind, while it can certainly be refined, we can accept that the model is remarkable in some sense, and at bare minimum shows a glimpse into what the future has in store.

**Literature**

Ritter, Gordon, Machine Learning for Trading (August 8, 2017). Available at SSRN: https://ssrn.com/abstract=3015609 or <http://dx.doi.org/10.2139/ssrn.3015609>

Dunis, Christian & Nathani, Anup. (2007). Quantitative trading of gold and silver using nonlinear models. Neural Network World. 17.

**Figures**

**Table

Description automatically generated**

Figure Key Model Metrics

**Table

Description automatically generated**

Figure S&P 500 Metrics

Chart, line chart

Description automatically generated

Figure Cumulative Returns

Chart, line chart

Description automatically generated

Figure Daily Returns

Chart

Description automatically generated with medium confidence

Figure Beta over time