Development of Strategies for Use of Reinforcement Learning in Financial Trading

Harris S. Rose, MD, MBA
Artificial Intelligence Masters Degree Candidate
Johns Hopkins University
Baltimore, MD
hrose10@jh.edu

***Abstract*—Reinforcement Learning provides an opportunity for investors to make better decisions in the face of uncertainty. The natural instincts of human traders are subject to a myriad of biases that can be reduced or eliminated through augmented intelligence or artificial intelligence. This paper explores the potential advantages associated with the technology then looks at potential issues with deployment and finally makes suggestions as to how to improve on past results.**

***Index Terms*—finance, reinforcement learning, artificial intelligence**

I. Introduction

Making decisions in the face of uncertainty is a challenge for people that can be optimized with artificial intelligence (AI). One of the more obvious applications for this is in the trading of financial instruments on open markets. While it is human nature to evaluate decisions based on outcomes, the presence of uncertainty in data and variability in outcomes for a given decision make this unwise. Optimally, decision making would be more objective and structured to have the highest chance of success. In other words, in making a series of decisions in the face of uncertainty, one should strive to provide the maximum chance of achieving the overall goal, regardless of the outcome of any individual decision.

Reinforcement learning (RL) is the third field of machine learning(ML), after supervised learning and unsupervised learning. RL uses iterative techniques to develop a policy map. A policy map is a list of the actions to be taken given a set of conditions. An optimal policy map maximizes the likelihood of achieving a desired outcome. One might consider an optimized policy map to be the Holy Grail of decision making. Determined in advance, without emotional bias, and without the interference of needing to know the actual outcome, the optimized policy map provides an agent with the best odds of achieving the agents previous defined goals.

While the use of ML to assist in trading has been around for quite some time, RL is just starting to develop its potential in trading. There are several advantages that RL may offer over supervised learning. Supervised learning performs very well for producing classifications of data and regression models; these can be very helpful for predicting stock prices. Successful portfolio management, however, requires additional strategic considerations; one must account for diversification, risk management, and exogenous events that affect stock prices. There are longstanding principles for managing these issues but it is exciting to consider that allowing an AI agent to explore the state space of a portfolio may lead to confirmation of these principles or to entirely different means of efficiently managing risk and reward.

II. Background

The use of algorithms and ML for investment decision making is not novel. In 1974, Jim Simons, a mathematician, gave up a successful academic career to start Renaissance Technology Fund and generated billions of dollars in returns by finding patterns in market data [1]. In the 1980's, Richard Dennis and his "turtle traders" used relatively simple, hand calculated algorithms to generate hundreds of millions of dollars in gains [2]. As computing power and algorithms have evolved since that time, many others have attempted to automate the process of generating returns. Currently one can find a plethora of textbooks [3], [4] and articles on implementing ML for trading.

Understandably, firms that have significant success using ML for trading are unlikely to publish details on their success as mass adoption would reduce the value of their techniques. Nevertheless, a variety of academic and amateur scientist/traders have published results of

their testing. While these results generally limited to backtesting, they do provide a substantial benefit. Examples include feature engineering for predicting stock price movement [5] as well as the use of TensorFlow and long short-term memory (LSTM) models for the same purpose.

As for the application of RL to trading, the field is still nascent but there is a growing body of literature. Jansen includes a chapter on the topic in his text [3]. Pricope provides an excellent review of the current state of the field [6]; other authors go into more details about specific applications of techniques. Yang describes deploying an ensemble strategy for RL models [7] while Zhang tests RL applications on commodities and equities [8]. Although these articles report positive results, neither compares their results to performance of an index or the baseline equity; they use other models as the benchmark. This is discussed below in the section on how to evaluate trading algorithms.

III. Method

This paper reviews topics germane to development of RL solutions for trading. This consists of a literature review of selected topics that would be important for creation of such a system followed by an analysis of these topics and suggestions for how such a system could be constructed. Future areas for research are then addressed. The topics addressed include:

- Optimal Decision Making
- Human Trading Biases
- Reward Setting
- Data Sources
- RL Model Evaluation

IV. Analysis

A. Optimal Decision Making

To make an optimal investment decision in the face of uncertainty, one must choose the action that has the highest expected return while avoiding decisions that can result in catastrophe. For traders, catastrophe means experiencing losses that cause a depletion of the portfolio and therefore no way to have subsequent gains that restore the portfolio.

Unfortunately, people have a tendency to evaluate their individual decisions based on individual outcomes of the decision as opposed to assessing progress toward the overall goal. Consider, for example, an individual who buys her first lottery ticket and wins. There is a temptation to call this a good decision despite the fact that lotteries all have negative expected returns and in the long term it is not possible to profit by repeatedly playing the lottery. A more prudent investor would focus on opportunities with positive expected value; this is the likely reason that up to 70% of lottery winners spend or lose all of their winnings within 5 years [9].

Reinforcement learning is related to optimal decision making in that the output of RL is a policy map or a neural network that functions as a policy map. These policy maps, if properly ascertained, recommend the action to be taken from a given state that will result in the highest expected reward. That does not mean the recommended action is guaranteed to have a good outcome but it does provide the means to have the best chance of repeatedly best outcomes.

B. Human Trading Biases

The psychology of human trading has been well studied and there are a vast number of identified biases that prevent people from making optimal decisions. A complete review of these classifications is beyond the scope of this paper but certain biases will help to understand the value provided by RL in trading.

Prospect theory, for which Daniel Kahneman and Amos Tversky were awarded a Nobel Prize, describes how most people have a preference for avoiding losses over receiving gains of similar value [10]. For traders, this can be quite harmful to performance as choosing to focus on avoiding individual losses within a portfolio can reduce overall performance. An algorithmic approach to investing would avoid this bias.

Overconfidence bias refers to the tendency of most people to overestimate the quality of their decision making. This can have multiple adverse consequences for traders, from simply choosing investments with negative expected returns to failing to properly diversify as a result of overconfidence in a limited number of assets. In a highly relevant study, Haghani and Dewey provided 61 finance students and professionals with a test [11]. Each participant was given a stake of $25 and asked to bet on a coin that would land heads 60% of the time. The prizes were capped at $250. The participants were not informed of the existence of the Kelly Criterion, a formula that determines the optimal size for a bet given the expected returns of the bet.

> Remarkably, 28% of the participants went bust, and the average payout was just $91. Only 21% of the participants reached the maximum. 18 of the 61 participants bet everything on one toss, while two-thirds gambled on tails at some stage in the experiment. [12]

Had the students deployed the Kelly criterion, 95% of the participants would have been expected to obtain the cap of $250. Due to overconfidence bias and bet sizes that were too large, many people ran out of money, even with a positive expected return. These and other biases provide the impetus for developing AI solutions that make more objective investment decisions.

C. Reward Setting

In training RL models to avoid human biases and to develop an optimal policy map, appropriate rewards for the agent are essential. These rewards should mimic the goals of the investor in that they should maximize returns, but perhaps more importantly, minimize risk. There are many proposed models for this in the literature.

One approach would be to argue that an RL algorithm designed to maximize return would take risk into account by default. The argument here is that an agent that repeatedly explores the environment obtains high average returns will be an agent that has minimized risk. This is the approach taken by Huang when describing the Reward Hypothesis as "All goals can be described by maximization of expected future reward." [13] Zhang follows this approach as well by assuming a "risk-insensitive trader" as an agent [8]. While this simplifies the reward function, it is perhaps unreasonable to deploy in real markets due to the risk-sensitivity all investors should possess.

A more comforting, and perhaps quicker to converge, approach would be to set certain parameters regarding risk. These parameters are well defined in conventional finance and translate well to algorithms. These could include adding goals for diversification, adding rewards for improving metrics of diversification such as the Sharpe ratio, or setting limits on actions to ensure diversification occurs. This approach is advantageous over a pure RL approach as it will reduce the size of the state space to be explored for an agent. Wu and others employed this approach by using a reward function which incorporates risk reduction, specifically, they optimized for return and for an improved Sortino ratio [14]. The Sortino Ratio is similar to the Sharpe ratio, but uses downside risk in place of volatility which may be a better goal for these models.

D. Data Sources

Signals for trading prediction can be obtained from many sources. These sources can be divided into price histories, fundamental data, and alternative data sources. Price histories are time series representing market history. Fundamental data includes metrics like P/E ratios as well as corporate SEC filings, and alternative data sources which are those that seem unrelated but may provide predictive values. A fascinating example of alternative data is the use of data on housing starts to predict Home Depot revenues and stock prices.

Pricing data is the cleanest and most readily available source of information available for trading predictions. This data is available for download from many sources, both free and paid. The advantages of paid services such as AlphaVantage over free sources such as Yahoo Finance is that the data is cleaner and is available preprocessed for many factors, including dividends and stock splits. Downloaded as a time series, price history can be used immediately with techniques designed for such data such as LSTM.

One has to wonder, however, if pricing data contains the optimal amount of signal for trading. The efficient markets hypothesis (EMH) states share prices reflect all publicly available information and therefore future changes in price must be caused by exogenous sources and are not contained in price history. If one accepts the EMH, then it can be concluded that consistent outperformance of the market is not possible. Although markets tend to be efficient, there are many examples of consistent market outperformance (Berkshire Hathaway, Renaissance Technology Fund, etc.) which have generated outsize return over decades, such that one can be certain that inefficiencies exist. This results in the conclusion that while price history is one part of the signal for future prices, additional information exists out the EMH.

Fundamental data about stocks certainly correlates with stock price changes and there is signal within this data that should be helpful in predicting prices. This data is generally not available as a time series and is often in text format. Prepossessing, including converting to time series as well as sentiment analysis and other NLP techniques is often needed to blend this data into an RL model. Decisions as to which data to include is often a balance of anticipated signal gained vs. effort involved in blending to the model.

Alternative data is relatively new to AI stock prediction but has substantial potential. Examples of alternative data sources include sentiment analysis of social media posts, news articles, and macroeconomic data. Exogenous data is likely to provide additional advantageous signal with the cost of increasing model complexity. The challenges of including exogenous data include the

TABLE I
DATA SOURCE SUMMARY

Source	Advantages	Disadvantages
Price History	Easy to obtain Clean Allows comparison between markets	Limited signal Not proprietary
Fundamental Data	Easy to obtain May have stronger signal	Not time series data Requires preprocessing
Alternative Data	May have uniquely strong signal	Harder to Obtain Highly variable in presence of signal Requires substantial signal processing

burden of combining it with endogenous data as well as difficulty creating historical models for backtesting. Despite these difficulties, the prospect of finding unique and proprietary alternative data sources often justifies the required extensive efforts to evaluate these sources.

Table I summarizes advantages and disadvantages of each type of data discussed above.

E. RL Model Evaluation

In designing AI systems for investing, it is important to think in advance about how one will measure success. This encompasses both the method of testing an algorithm as well as the metrics needed to consider the algorithm a success.

The first issue to address when transitioning from model development to model testing is slippage. Slippage represents the losses associated with attempted trades not occurring at the desired price. Sources of slippage include delays in order execution, effects of large trades on market prices, and the bid-ask spread of the market maker. Test results either need to account for slippage by involving live trading with small amounts of money, or must assume a value for the amount of slippage which will occur with each trade. Good values for slippage assumptions are in the range of 1-5% of the targeted share price.

When testing a policy map, backtesting is an ideal place to start but has significant limitations. Backtesting involves withholding a subset of market data from the training process and then applying the policy to determine the success rate on this historical data. Backtesting is a reasonable way to initially evaluate a policy and it does allow comparison of multiple models prior to live deployment. Care must be taken, however, not to rely solely on backtested results prior to proceeding with live trading.

Using RL for trading is an exercise in extrapolation as one is trying to predict future events based on past events. Results from interpolation, or backtesting, are more likely to be successful than extrapolation results. Therefore, live testing is essential as final confirmation of the efficacy of an algorithm. Live testing can be done in the form of "paper trading," where trading is simulated and returns are calculated, or testing can be done with a relatively small test portfolio.

An issue not addressed in any of the papers that were reviewed was the effect of taxes on algorithm performance. Perhaps this is a result of many papers originating outside of the United States and that many types of entities exist with different taxation rules for each entity. Nevertheless, neglecting the tax consequences of investment decisions in a comparison of techniques should be avoided. In the United States, general guidelines for taxation of investments by individuals and partnerships requires that taxes be paid at the end of each year on net capital gains. For long term investments (those held longer than 1 year) the tax rate is approximately 20% and for short term net gains the rate can be above 40%. Any testing models should incorporate these parameters as an algorithm that beats a market index with short term gains but incurs double the tax bill may not be more effective than passively investing in the index.

V. IMPLEMENTATION STRATEGY

Implementation of RL for a trading strategy will require multiple decisions to create a model that balances the desire to incorporate as much data and function as possible with the need to limit complexity in initial designs.

There are many decisions for an agent to make managing an investment portfolio. These include:

- Choosing financial instruments for the portfolio
- Determining allocation of these financial instruments
- Deciding on an entry point for each financial instrument

- Monitoring endogenous and exogenous data for each financial instrument
- Deciding on an exit point for each financial instrument

While an ideal solution would provide stellar performance while making all investment decisions, this is unrealistic as a starting point. A better approach would be to assess an existing human trading strategy for each of these functions, decide which functions would most benefit from an optimized policy map, and then create a model to augment the human traders efforts. As success with this augmented intelligence solution is obtained, the model can be expanded to incorporate additional functions.

Regardless of the functional area on which initial focus is set, the state space and action space will require definition. Any state space will need to include the following as a minimum:

- Current portfolio contents
- Price history on all potential financial instruments
- Additional sources of signal for the agent

The action space for the agent can be of variable complexity. The simplest action space could be to buy, sell, or hold one unit of a financial instrument. This model is relatively simple in that there are only two decisions to be made for each market in which the agent can participate. If the current position for a market is zero units, the action space is {buy, sell}, whereas if the current position holds one unit, actions are {hold, sell} and if the current position is short one unit the action space becomes {hold, buy}. While this definition is simple for an agent that is limited to one financial instrument, it grows linearly with the number of possible instruments. Additionally, assumptions will be required for an initial unit size as a percentage of portfolio size to manage risk through diversification. Table II provides an overview of this simple action space.

TABLE II
SIMPLE ACTION SPACE FOR TRADING

Holdings	**Actions**
+1 unit	{Hold, Sell 1 unit}
0 units	{Buy 1 unit, Sell 1 unit}
-1 units	{Hold, Buy 1 unit}

Although simple, the above action space definition is quite limiting for the agent and therefore likely to reduce returns from their potential maximum. Additional layers of complexity to add could include:

- Allow the agent to acquire a variable number of units of each financial instrument
- Allow the agent to flip from long to short positions in one time step
- Increase the number of markets available to the agent

The resultant increase in parameters is likely to make the agent more effective; however, the additional computational complexity will necessitate additional resources. This reinforces the need to start small and grow, while also leading to a strategy for algorithm development as discussed next.

RL can be accomplished through iterative techniques based upon the Bellman equation, one of which is Q-learning, or through the use of neural networks, referred to as deep reinforcement learning(DRL). Q-learning is a more precise method of finding a policy map but as state and action spaces grow, Q-learning can become computationally infeasible. DRL is more complex to implement and makes many assumptions; therefore DRL may not be as complete as Q-learning but DRL can handle much larger state spaces. AlphaGo is an excellent example of the successful use of DRL for the enormous state space of the game Go [15]. A comparison of these two techniques is included in Table III. This leads to the conclusion that an optimal path to algorithm development would start with using Q-learning for the smallest reasonable state and action spaces and progress to DRL as these spaces grow.

Given the above, Table IV shows our proposed implementation strategy for early stage RL trading algorithm development.

VI. CONCLUSIONS

From the information included in this paper, we believe that following conclusions can be drawn. (1) There exists significant bias amongst human traders that AI may result in improved performance. (2) Reinforcement Learning and the subsequent policy map is produces have potential to reduce the downsides of human bias. (3) Any agent used to guide investment strategies should be evaluated relative to human performance or standardized indices, as opposed to simply based upon return. (4) Development of RL based policy maps may be optimized by starting with augmenting the decision making of human traders as opposed to starting with a full decision making algorithm (5) The state and action spaces for RL can be quite large which may limit their utility; appropriate simplification of state and action spaces will

TABLE III
Q-LEARNING VS DEEP RL

Algorithm	Advantages	Disadvantages
Q-learning	More complete policy map Allows comparison between markets	Complexity grows as state and action spaces grow
DRL	Can handle large state spaces	Less precise Complex to implement Requires extensive computing power

TABLE IV
STEPWISE IMPLEMENTATION STRATEGY FOR RL TRADING ALGORITHM

Implementation Steps
1. Create Q-learning model using:
a. Single financial instrument
b. Buy, sell, hold one unit action space
c. Only price history for data
2. Add in multiple financial instruments
3. Add in incremental until amounts for buy and sell
4. Add in fundamental/alternative data to increase signal
5. As model complexity shifts, switch from Q-learning to DRL

be important in the development of any RL trading solution.

VII. FUTURE RESEARCH POSSIBILITIES

Future research needs in the arena of RL for trading algorithms are nearly limitless; as questions are answered, more questions will occur. One reason for this is that performance goals for market algorithms can be effectively limitless. To understand this better, take blackjack as an example of decision making in the face of uncertainty. Blackjack has finite defined state and action spaces and the probabilities of new cards (state transitions) are known. Due to the finite nature of the game, an optimal policy map for blackjack can be determined and is in fact already known. This optimal policy map for blackjack yields an expected return on any hand of approximately -0.5% and any algorithm that attempts to play blackjack efficiently can be evaluated relative to this known expectation [16].

Markets have no such limit on their performance; a theoretical maximum in a given market does exist but it is the result of perfect timing with agent buying at every local minimum and selling at every local maximum of price. Unlike blackjack, this theoretical maximum is not likely to be obtainable; however, continuous improvement of models by adding information should be possible. Accordingly, the most valuable research is that which can ascertain which information to added provides the most lift to model returns.

Areas of potential research within the scope of the optimal data to include in the state space include the following:

- Choosing fundamental company data to include
- Assessing past performance of company executives
- Incorporating SEC filings
  - Form 4 - Insider trading activity
  - Form 8-K - Adverse Event Reports
- Sentiment analysis on web sources
  - Social media
  - Government reports
    - Housing Starts, etc
    - Consumer Price Reports
    - Federal Reserve Activity Reports

While this in no way an exhaustive list of opportunities to gain signal, it does give the reader an idea of the limitless sources of information that could enhance trading algorithms. It is this ability to continue to find new sources of information and ways to effectively harvest their value that make RL for trading such an exciting and promising field.

REFERENCES

[1] G. Zuckerman, *The Man Who Solved the Market: How Jim Simons Launched the Quant Revolution*. Portfolio, 2019.

[2] M. W. Covel, *The Complete Turtle Trader How 23 Novice Investors Became Overnight Millionaires*. Collins Business, 2007.

[3] S. Jansen, *Machine Learning for Algorithmic Trading : Predictive models to extract signals from market and alternative data for systematic trading strategies with Python, 2nd Edition*. Packt Publishing, 2020.

[4] L. Bernut, *Algorithmic Short-Selling with Python*. Packt Publishing, Limited, 2021.

[5] A. Ntakaris, G. Mirone, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Feature engineering for mid-price prediction with deep learning," *IEEE Access*, vol. PP, pp. 1–1, 06 2019.

[6] T.-V. Pricope, "Deep reinforcement learning in quantitative algorithmic trading: A review," *arXiv:2106.00123v1 [cs.LG]*, May 2021.

[7] H. Yang, X.-Y. Liu, S. Zhong, and A. Walid, "Deep reinforcement learning for automated stock trading: An ensemble strategy," in *Proceedings of the First ACM International Conference on AI in Finance*, pp. 1–8, 2020.

[8] Z. Zhang, S. Zohren, and S. Roberts, "Deep reinforcement learning for trading," *The Journal of Financial Data Science*, vol. 2, no. 2, pp. 25–40, 2020.

[9] G. Lowenstein, "Five myths about the lottery," *Washington Post*, Dec. 2019.

[10] D. Kahneman and A. Tversky, "Prospect theory: An analysis of decision under risk," *Econometrica*, vol. 47, no. 2, pp. 263–291, 1979.

[11] V. Haghani and R. Dewey, "Rational decision-making under uncertainty: Observed betting patterns on a biased coin," *arXiv:1701.01427 [q-fin.GN]*, Jan. 2017.

[12] Buttonwood, "Irrational tossers," *The Economist*, Nov. 2016.

[13] C. Y. Huang, "Financial trading as a game: A deep reinforcement learning approach," *arXiv:1807.02787 [q-fin.TR]*, July 2018.

[14] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, and H. Fujita, "Adaptive stock trading strategies with deep reinforcement learning methods," *Information Sciences*, vol. 538, pp. 142–158, oct 2020.

[15] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–489, jan 2016.

[16] R. R. Baldwin, W. E. Cantey, H. Maisel, and J. P. McDermott, "The optimum strategy in blackjack," *Journal of the American Statistical Association*, vol. 51, pp. 429–439, sep 1956.