Using Reinforcement Learning for Stock Market Trading

A Thesis

presented to

School of Applied Computing, Faculty of Applied Science and Technology

of

Sheridan College, Institute of Technology and Advanced Learning

by

Chelsea

in partial fulfilment of the requirements

for the degree of

Honours Bachelor of Computer Science (Mobile Computing)

September 2023

Using Reinforcement Learning for Stock Market Trading

by

Chelsea

Submitted to the School of Applied Computing, Faculty of Applied Science and Technology
on November 21, 2023, in partial fulfillment of the
requirements for the degree of
Honours Bachelor of Computer Science (Mobile Computing)

Abstract

In this research, we employ a novel rule-based strategy to train a successful machine learning algorithm, specifically Reinforcement Learning (RL). This choice is motivated by the challenges associated with error-prone programming and tedious debugging in the decision-making process of when, at what price, and in what quantity to trade. The prevalence of Artificial Intelligence (AI) is increasing across diverse fields and applications, with automatic trading emerging as a promising area. Artificial Intelligence (AI) empowers traders to enhance their trading strategies for financial assets, addressing challenges like price fluctuations and market dynamics. Successful investors must judiciously apply optimal strategies to maximize returns while minimizing risks. The primary objective is to gain a competitive edge in quantitative finance. However, quantitative traders encounter a steep learning curve in developing an agent capable of positioning itself strategically in the market for consistent wins.

This paper leverages the Reinforcement Learning for Finance Framework (FinRL framework) to comprehensively analyze the stock market. Our approach involves the integration of various models and state variables to identify the most effective trading strategy.

Keywords: Stock Market, Reinforcement Learning, Reinforcement Learning for Finance Framework (FinRL framework), Artificial Intelligence (AI)

Thesis Supervisor: Dr. Ghassem Tofighi
Title: Professor, School of Applied Computing

Acknowledgments

I want to express my deepest gratitude to everyone who has contributed to the successful completion of this thesis. This journey has been challenging and rewarding, and I am thankful for the support and encouragement I received along the way.

First, I sincerely appreciate my thesis advisor, Prof. Ghassem Tofighi, for their guidance, expertise, and unwavering support throughout the research process. Their valuable insights and constructive feedback have been instrumental in shaping the direction and quality of this work.

I am grateful to my thesis committee members for their time and commitment to reviewing and evaluating this research. Their input has enriched the depth and breadth of the study.

I thank my colleagues and peers who provided a collaborative and stimulating academic environment. The exchange of ideas and discussions greatly contributed to developing and refining this thesis.

Special thanks go to my family and friends for their understanding, encouragement, and patience during the demanding phases of this academic endeavour. Their belief in my abilities has been a driving force, and I am fortunate to have their unwavering support.

Last but not least, I appreciate the resources and facilities provided by Sheridan College. The conducive research environment and access to relevant materials significantly contributed to the overall success of this thesis.

This work would not have been possible without the collective efforts of all those mentioned above, and I am truly thankful for the collaborative spirit that has characterized this research journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The constantly changing financial markets present challenges for accurate prediction and effective trading. Traditional methods, relying on statistical models and fundamental analysis, struggle with the intricacies of the stock market, which is influenced by many unpredictable factors. As financial markets become more complex, crafting robust and adaptive trading strategies becomes increasingly difficult.

In response to these challenges, experts are exploring advanced techniques to enhance stock market trading strategies. One such technique is reinforcement learning, which has gained attention from a growing body of research. Notable works include Moody and Saffell's study of "Reinforcement learning for trading systems and portfolios" [2], which laid the groundwork for understanding how reinforcement learning could be applied to trading systems and portfolios.

Sutton and Barto's comprehensive overview of reinforcement learning principles[3], provides a broad understanding of these principles and serves as a cornerstone for applying them to real-world scenarios, including financial markets.

In addition, Gu et al.'s study, "Reinforcement learning for trading"[4], adds a practical dimension to the discourse by demonstrating the applicability of reinforcement learning in trading strategies. Their work contributes insights into the potential enhancements in accuracy and effectiveness achievable by incorporating reinforcement learning in financial decision-making.

These works highlight the increasing recognition of reinforcement learning as a promising approach for navigating the complexities of stock market dynamics. This research builds upon this foundation to explore and contribute to applying reinforcement learning in the context of evolving financial markets. By examining traditional methods, challenges faced, and the

potential of advanced techniques, this research aims to provide valuable insights into developing sophisticated and adaptive trading strategies.

1.1.1 Reinforcement Learning

Reinforcement Learning(RL) is a subset of machine learning which focuses on decision-making through interaction with an environment. It is a type of learning where an agent learns to make sequential decisions by taking actions in the environment to maximize a cumulative reward. In this process, the agent learns to make decisions through trial and error. (It is similar to training your dog; when your dog performs the trick, you reward him, and if he does not, you punish him.) The terminology used in reinforcement learning (RL) is explained in the table below. To understand more efficiently, we can think of the PacMan game.

Term	Definition
Agent	An agent is a computational system or robot that takes actions in an environment for Reinforcement Learning (RL).
Environment	The external system in which the agent operates provides feedback to the agent's decisions through rewards or penalties.
Action (a)	The choices or decisions available to the agent to interact with the environment.
State (s)	A representation of the current environment help the agent decide which action to take.
Reward (r)	The rewards given by the environment to the agent after each action serve as a way to measure the success or desirability of the agent's actions. The agent's objective is to maximize the cumulative reward over a period of time.
Policy (π)	The policy is a set of rules that govern how an agent chooses actions based on different states. It can be either deterministic or stochastic, depending on the level of uncertainty in the environment.
Value Function (V or Q)	Functions that estimate the expected cumulative reward or value of being in a particular state (V) or taking a specific action in a particular state (Q).

Table 1.1: Reinforcement Learning Terminology

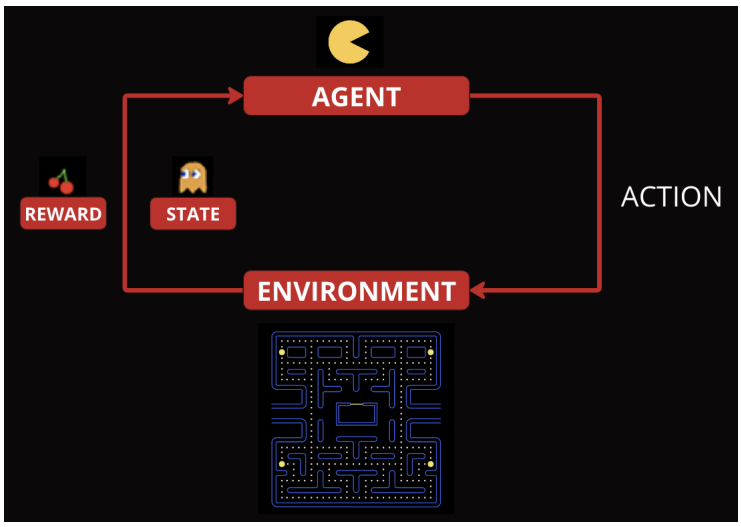

Figure 1-1: Terminologies presented using PacMan Game[1]

1.2 Motivation

The motivation behind delving into the realm of reinforcement learning for stock market trading stems from the inherent intricacies of the financial markets. The stock market, characterized by its high complexity, dynamism, and unpredictability, poses significant challenges for traders and investors. Traditional methods often struggle to navigate this intricate landscape effectively.

Reinforcement learning, however, presents a promising avenue for addressing these challenges. Its ability to learn from market data and adapt to changing market conditions positions it as a potentially transformative approach for stock market trading. The motivation to explore this topic arises from the desire to contribute to developing more effective and efficient stock market trading strategies.

The motivation encapsulates the industry's recognition of the need for advanced techniques, particularly reinforcement learning, to navigate the complexities of stock market dynamics. The goal is to understand the potential of reinforcement learning and actively contribute to the evolution of more adaptive and successful trading strategies in the ever-changing financial landscape.

1.3 Contributions

This research bridges the gap between finance and technology by showcasing the practical application of machine learning in financial markets. Its methodologies and findings contribute to the advancement of FinTech. The core objective is to provide actionable insights and guidelines

for traders, investors, and financial institutions. These recommendations will empower market participants to make better decisions and navigate the complexities of the stock market more effectively.

1.4 Organization

This paper is divided into five sections for clarity and coherence. The first section reviews existing research on stock market endeavours, while the second section outlines the research approach, including the model and state variables being studied. The third section deals with the implementation of code and algorithms necessary to derive meaningful results from the research process. The fourth section evaluates the findings and conclusions, while the final section comprehensively summarizes the research undertaken.

1.5 Keyword Description

This section will explain the key terms that our thesis relies on.

1. Reinforcement Learning: It is a machine learning technique where an agent learns through trial and error in an interactive environment by receiving feedback from its actions and experiences.[5]

2. Stock Market: The stock market is a constellation of marketplaces where securities like stocks and bonds are bought and sold[6].

3. FinRL Framework: A specialized framework tailored for applying reinforcement learning in financial markets, offering a suite of tools and methodologies for stock market trading.

1.6 Thesis Statement

This study explores the efficiency of Reinforcement Learning in guiding ongoing decisions related to holding, buying, or selling stocks. The primary objective is to train and evaluate Reinforcement Learning models, identifying those most effective in the stock market. Additionally, the research delves into potential performance enhancements by incorporating additional variables, such as the State variable.

1.7 Significance of the Study

This study is significant beyond academia and represents an important step in solving the challenges of stock trading. The research used modern techniques, like reinforcement learning, and offered insights that could change how traders, investors, and financial institutions navigate complex financial markets. The findings have both theoretical and practical implications for stock trading, potentially influencing decision-making. This study could reshape stock market strategies, offering valuable insights to make better decisions in this ever-evolving world.

1.8 Chapter Summary

This introduction provides a foundation for the research ahead. It covers the background, explains why reinforcement learning is being studied in stock market trading, lists clear research objectives shows how the research can help finance and technology, and gives an overview of the paper.

The following chapters will go deeper into existing research on stock market trading, explain the chosen research approach and implementation, evaluate findings, and conclude with a summary of the research. Each chapter thoroughly explores the topic, contributing to a better understanding of reinforcement learning in stock trading.

Chapter 2

Literature Review

Aspiring investors and traders constantly strive to make sound and lucrative decisions in the continually evolving Stock market. Traditional techniques utilized for stock market analysis have primarily consisted of fundamental and technical analysis. However, Reinforcement Learning (RL) has emerged as a promising tool for improving decision-making in stock trading. This section will delve into historical implementations, challenges and strategies of Reinforcement Learning (RL). It is important to note that while traditional techniques may provide a strong foundation for analysis, Reinforcement Learning (RL) can supplement and enhance decision-making by adapting to the dynamic nature of the stock market.

2.1 Historical Perspective

Reinforcement Learning (RL) use in stock trading has not yet been extensively researched. Initial studies have primarily focused on basic models and small datasets. In 2001, researchers Moody and Stafell conducted a study on implementing Q-Learning as a Reinforcement Learning (RL) technique for stock trading. The primary objective of their research was to investigate how Reinforcement Learning(RL) could be leveraged to enhance trading strategies and make more informed trading decisions in the financial market. Their findings demonstrate the potential for Reinforcement Learning(RL) to improve the efficacy of stock trading practices significantly.

Reinforcement Learning (RL) in trading can be categorized into three main methods: critic-only, actor-only, and actor-critic approaches, according to the current literature [7]. Deep Q-Networks (DQN) are often used in publications on the critic approach, which is based on constructing a state-action value function, Q, to evaluate the goodness of an action in a given state. Discrete action spaces are commonly used in these works, where agents are trained to take a full position, either long or short. Studies by [8], [9], [10], [11], and [12] are examples of this

method. Due to the limitations of having discrete action spaces, Deep Q-networks (DQNs) may not be suitable for problems that require continuous action spaces. Another major drawback of DQNs is balancing exploration and exploitation in high-dimensional state spaces.

The actor-only approach is the second most commonly used method [13], [14], [15], [16]. This approach concentrates primarily on the learning policy of the actor to directly map states to actions without explicitly estimating values functions. The techniques used in the actor-only approach primarily rely on gradient ascent methods to improve performance. The policy is adjusted to maximize the expected cumulative reward over time. However, this approach can be inefficient due to its dependence on a large amount of data, which requires numerous interactions with the environment to learn a good policy. Furthermore, determining the optimal policy can be a challenge with this approach.

A popular technique for problem-solving is the actor-critic approach, which involves two key elements: an actor and a critic. The actor is tasked with learning and updating the policy and adjusting the variant to explore the state-actions space better. Meanwhile, the critic estimates the value of being in a particular state and offers feedback to the actor regarding the error. This error, known as the temporal difference (TD) error, is utilized to update the actor policy. Despite its efficacy, the actor-critic approach remains the most under-researched method.

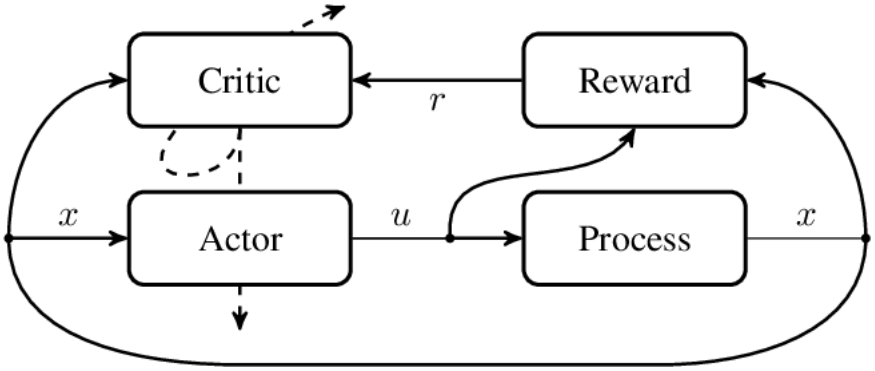

Figure 2-1: Schematic overview of an actor-critic algorithm [17]

2.2 Algorithmic Advancement

Over time, reinforcement learning has undergone significant developments through continuous research and practice in the financial sector, focusing on the advantages of trading strategies. In 2015, Mnih [18]introduced the Deep Q-network (DQN), which demonstrated remarkable success in playing Atari games and sparked interest in utilizing reinforcement learning in the financial market.

In 2017 [19], the Deep Q-network (DQN) is a type of artificial intelligence that can create automated trading strategies that adjust to the unpredictable and intricate stock market. A research paper has expanded on the DQN by incorporating Recurrent Neural Networks (RNNs) into its design. This addition enables the model to capture the time-dependent relationships in stock price data, which allows the Reinforcement Learning (RL) agent to make decisions based on past patterns. The Deep Q-network (DQN) agent shows more promising results than the one with Recurrent Neural Networks (RNNs). However, the author acknowledges the challenges that come with this model, such as data noise, strategies, extensive training, and cost. The author also provides future works, such as optimizing trading strategies in the real world.

Various researchers have explored the application of Reinforcement Learning in the stock market. For instance, in 2019, Yang conducted an experiment using the Proximal Policy Optimization (PPO) model [17]. The study employed Deep Deterministic Policy Gradient neural networks to approximate functions and an actor-critic architecture to learn and enhance portfolio allocation policies. Subsequently, in 2020, Jiang introduced meta-learning for the stock market to adapt reinforcement learning (RL) agents to different stock market scenarios. However, practical Implementation in the real world of the stock market requires additional improvements and specifications for competition. The table below presents more related research conducted in various time spans. The table explains the Focus Area, Application Area, Key Findings, and limitations of each research implemented in the financial field using reinforcement learning.

Table 2.1: Comparison of RL Research Papers in Stock Market Trading

Research Paper	Focus Area	Key Findings	Limitations	Summary
Reinforcement learning for trading systems and portfolios[13]	Trading systems and portfolios	Laid groundwork for RL in trading systems and portfolios.	Limited empirical testing; basic models and small datasets.	Pioneering work establishing the foundation for RL in trading systems and portfolios. Although limited by basic models and small datasets, it set the stage for future exploration in the field.
Reinforcement Learning: An Introduction[20]	Broad understanding of RL principles	Comprehensive RL principles for diverse applications.	More theoretical; focuses on principles rather than specific applications.	A foundational book offering a broad understanding of RL principles. Its theoretical nature makes it suitable for readers seeking a deep understanding of RL but may lack detailed practical applications specific to trading.
Reinforcement learning for trading[21]	Practical application of RL in trading strategies	Demonstrates practicality and applicability of RL in trading strategies.	Limited exploration of algorithmic nuances and complexities; focuses on general concepts.	The study bridges theory and practice by showcasing the practical application of RL in trading strategies. While not delving into algorithmic complexities, it provides valuable insights into applying RL to enhance trading strategies.
Algorithmic trading using Q-learning and Recurrent Reinforcement Learning[22]	Application of Q-learning and recurrent RL in algo trading	Discusses Q-learning and recurrent RL in algorithmic trading.	May lack coverage of other RL approaches.	Focuses on Q-learning and recurrent RL for algorithmic trading, simplifying complexities for clarity. It acknowledges limitations in capturing dynamic market conditions fully, emphasizing the need for further refinement in future studies.
Portfolio management using Reinforcement Learning[23]	RL in portfolio management	Explores RL for optimizing investment decisions in portfolio management.	Limited exploration of real-world applications.	Examines RL's potential in optimizing portfolio management decisions. It could benefit from a more in-depth analysis of scalability and robustness in various market conditions to enhance its practical implications.
Financial portfolio optimization with Reinforcement Learning[24]	RL in financial portfolio optimization	Investigates RL for addressing challenges in financial portfolio optimization.	Challenges in handling large datasets in real-time processing.	Examines RL for solving challenges in financial portfolio optimization. While addressing key aspects, such as optimization, it could benefit from a more comprehensive discussion of transaction costs and practical implementation challenges.

2.3 Data Sources and Preprocessing

The data source is vital for Reinforcement Learning(RL) based stock market analysis. The researchers have employed various data resources such as Historical Stock Price Data, Market Indicators, New and Sentiment Data and Fundamental Data. The data source used was His-

torical Stock Price Data for all the researchers mentioned above. Once data has been stored, it is crucial to preprocess datasets using various methods. These include data cleaning, normalization and scaling, feature engineering, time series data handling, missing data, data splitting, and more.

2.4 Challenges and Concerns

The research histories discussed in this section have shown successes but face challenges and raise concerns about future implementations. There are different challenges which a researcher could come across while working with Reinforcement Learning(RL) models and state variables such as Data Quality, Noise signals, Transaction Costs, Risk Management, Exploration vs. Exploitation, Real-world Implementation and many more. In [14], The researcher faced some difficulties related to the quality of the data and the presence of disruptive signals, which ultimately affected the performance of the trading strategies. In addition, overfitting the historical data was another drawback faced by the author. Furthermore, the authors cited in [18, 19] encountered data quality and sample efficiency difficulties when training their Reinforcement Learning (RL) models. An extensive database is required, which can be costly, and the data quality remains an additional challenge. Therefore, from the above reading, the most common challenges in Reinforcement Learning(RL) are data efficiency, Data quality and real-world Implementation.

2.5 Real-World Applications

The research shows that Reinforcement Learning (RL) techniques like Deep Q-Networks (DQN), Deep Deterministic Policy Gradients (DDPG), Trust Region Policy Optimization (TRPO), and Proximal Policy Optimization (PPO) are being used in actual stock trading scenarios. Experts are focusing on overcoming challenges like data accuracy, transaction expenses, and risk management to develop better and more effective trading strategies using RL technology. While RL has promise, it must be evaluated carefully regarding ethical, regulatory, and risk management factors when implemented in real-life trading situations.

2.6 Chapter Conclusion

In conclusion, Reinforcement Learning (RL) has shown great potential in improving decision-making in stock trading. While traditional techniques like fundamental and technical analysis provide a strong foundation for analysis, RL can supplement and enhance decision-making by adapting to the dynamic nature of the stock market. RL in trading can be categorized into three main methods: critic-only, actor-only, and actor-critic approaches, with the actor-critic approach being the most popular one. Over time, RL has undergone significant developments through continuous research and practice in the financial sector, focusing on the advantages of trading strategies. However, challenges such as data noise, strategies, extensive training, and cost need to be addressed, and future work needs to focus on optimizing trading strategies in the real world.

Chapter 3

Methodology

This section will discuss the methods used in this study to obtain results on Stock Market analysis through Reinforcement Learning.

3.1 Research Design

The study uses quantitative research to analyze historical stock market data and develop a reinforcement learning-based trading strategy. This approach emphasizes collecting and analyzing numerical data, enabling statistical and quantitative evaluations of the stock market's behaviour. Using different state variables, indicators, and agents aligns with the belief that a data-driven quantitative approach can capture and exploit financial market patterns, trends, and anomalies, leading to evidence-based trading decisions.

3.2 Data Collection

3.2.1 Data Sources

We used a Yahoo Finance dataset to trade and test reinforcement learning models. The dataset includes the stocks of the Dow Jones Industrial Average (DOW 30), a stock market index of 30 large, publicly traded companies in the United States. We chose Yahoo Finance as it is a comprehensive financial website that provides various information related to finance and investment. The Dow 30's composition is periodically adjusted to reflect economic and stock market changes. Well-known companies traditionally part of the Dow 30 include Apple, Microsoft, Coca-Cola, and IBM. The Dow 30's performance is often used to indicate the overall health and direction of the U.S. stock market.

To obtain historical data, we used various APIs, and the downloaded data is in OHLCV format, which stands for open, high, low, close, and volume. These values provide crucial information for the time series of a stock market and help traders gain insights and make predictions about the market. We used the 'config tickers DOW 30 TICKER' variable to specify the selected tickers.

We will use the Unified Data Processor provided by FinRL to retrieve, clean, and extract data features. This involves consolidating fragmented data into a central view and accessing them as per project requirements.

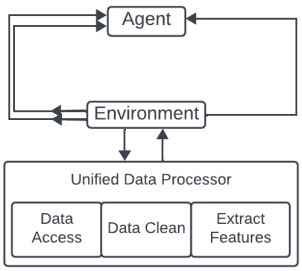

Figure 3-1: Unified Data Processor

3.2.2 Data Variables

Independent Variables

1. OHLCV

   (a) Open: The price of a stock when it starts trading for the day.

   (b) Close: The price of a stock when the trading day ends.

   (c) High: The highest price at which a stock was traded during a specific period.

   (d) Low: The lowest price at which a stock was traded during a specific period.

   (e) Volume: The total number of shares traded for a particular security during a specific period.

2. Technical Indicators

   (a) Moving Average Convergence Divergence (MACD): It shows the relationship between the two moving averages of the security's price.

   (b) Bollinger Bands (Upper and Lower Band): It is the standard deviation of the price from the moving average.

(c) Relative Strength Index (RSI): It is a momentum oscillator that measures the speed and change of price movements. In the code, we are using the period of 30.

(d) Commodity Channel Index (CCI) with a period of 30: It is a momentum that helps determine when the stock is being oversold and overbought.

(e) Directional Movement Index(DX) with a period of 30: It is used to quantify the strength and direction of a trend.

(f) Close Price 30-day/60-days Simple Moving Average (close 30 sma): the average closing prices over the last 30 days/60 days. It helps smooth out price data to identify the trends over a specific period.

(g) KDJ Indicator: It analyzes and predicts changes in stock trends and price patterns. For KDJ, values are being computed using the function calculated KDJ with parameters having default values.

3. Additional Features

(a) Volatility Index(VIX): It represents the market's expectation of future volatility. Adding this feature can help the model account for overall market volatility.

(b) Turbulence: The model can consider the broader economic environment when making predictions.

Dependent Variables

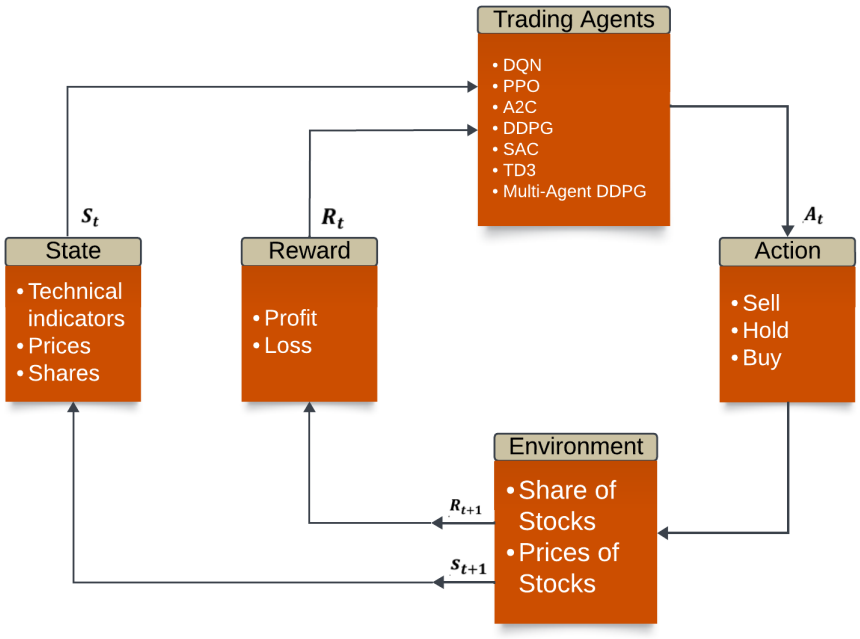

Figure 3-2: Overview of automated trading

1. State Space(S) In the context of an interactive market, a trading agent relies on its perception of the environment to make sequential decisions. The state space is the set of features the agent observes and uses to inform its actions. Allowing various levels of granularity in the time step t, such as daily, hourly, or minute intervals. For the Open-high-low-close (OHLC) prices

$$o_t, h_t, l_t, p_t \in \mathbb{R}_+^n$$

and trading volume

$$v_t \in \mathbb{R}_+^n$$

Technical indicators such as Moving Average Convergence Divergence(MACD)

$$M_t \in \mathbb{R}^n$$

, Relative Strength Index(RSI)

$$R_t \in \mathbb{R}_+^n$$

2. Action Space(A) This text explains the various actions that an agent can perform while in a particular state. It comprehensively describes all the possible steps the agent can

take for that specific state. The act of multiple shares is

$$a \in -k, ..., -1, 0, 1, ..., k$$

where k represents the maximum number of shares to buy or sell.

3. Reward Function(R) An agent utilizes this function to learn an improved policy.

4. Performance Metrics

   (a) 1. Cumulative Return: The total percentage change in an investment's value over a specified period is a metric used to evaluate the performance of the investment. A higher percentage indicates better performance.

   (b) Annualized Return: The average annual rate of return over a period of time. (Extremely high values may indicate potential issues or anomalies)

   (c) Sharpe Ratio: It measures the risk-adjusted return of an investment. (The higher the value implies better risk-adjusted return.)

   (d) Max Drawdown: It represents the maximum loss from a peak to a trough of an investment before a new peak is attained.

3.3 FinRL Framework Implementation

3.3.1 Overview of FinRL

The FinRL framework is popular for applying reinforcement learning techniques to financial markets. It is relevant to our research as it provides a robust and flexible platform for exploring various investment strategies and evaluating their performance. One of the key features of the FinRL framework is its ability to handle complex financial data, including time-series data, which is critical in financial markets. The framework also incorporates powerful tools, such as deep learning algorithms, that enable it to learn patterns and make predictions based on past market trends.

Another important feature of FinRL is its ability to handle multiple trading environments simultaneously, essential for creating and testing different investment strategies. Moreover, the framework is designed to be highly scalable, allowing it to handle large datasets and multiple users concurrently. Overall, the flexibility, scalability, and powerful tools the FinRL framework

provides make it an ideal platform for applying reinforcement learning techniques to financial markets.

3.3.2 Customization for Stock Market Trading

In this section, we present several strategies that can be considered to customize or extend the FinRL framework for stock market trading. The primary objective of this research is to develop a robust and profitable trading strategy by adapting the FinRL framework to suit specific requirements. We explore different techniques to optimize performance metrics and increase the framework's robustness to achieve this.

Firstly, we propose adapting each model's hyperparameters based on the financial data's characteristics. This involves experimenting with different configurations to optimize performance metrics for stock trading. We adjust hyperparameters such as the learning rate, batch size, or number of steps to better suit the financial data's characteristics.

Secondly, we explore the incorporation of ensemble learning techniques to combine predictions from multiple models. Ensemble learning can improve overall performance and robustness by leveraging the strengths of various models. We combine the predictions from a deep neural network with those from a random forest model to create a more accurate and reliable trading strategy.

Lastly, we propose expanding the technical indicators used in the state space. Technical indicators are mathematical calculations based on the price and/or volume of a security that can help traders identify patterns and trends in the market. We add more indicators to the state space to capture a more comprehensive view of market conditions, leading to more informed trading decisions.

In conclusion, customizing or extending the FinRL framework for stock market trading involves experimenting with different strategies to find the best approach for a given dataset or trading scenario. The proposed strategies in this section can create a more effective and profitable trading strategy by adapting the framework to suit specific requirements.

3.3.3 Model Architecture

Models

The models selected for this study encompass a range of reinforcement learning architectures, such as Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), Twin Delayed Deep Deterministic Policy Gradients (TD3), and

Soft Actor-Critic (SAC). Each architecture is chosen for its specific advantages and suitability for addressing the complexities of financial markets.

Number of Layers

The deep learning models are characterized by multiple layers, each serving a distinct purpose in the learning process. The number of layers may vary based on the selected architecture. For instance, A2C and PPO typically consist of actor and critic networks, while DDPG involves actor and critic networks with additional exploration noise. TD3 employs twin critics for reduced overestimation bias, and SAC emphasizes entropy maximization.

Novel Components

For this paper, we will consider Open, High, Low, and Close(OHLC) prices for the stock, volume, Simple Moving Average (SMA), Exponential Moving Averages (EMA), and the Moving Average (MACD), Bollinger upper band, Bollinger lower band, Relative strength index (30 periods), Convergence Divergence (MACD) indicator.

3.3.4 Hyperparameter Tuning

We use different Deep Reinforcement Learning (DRL) algorithms to train the trading agents and evaluate their performance. The Overview of the hyperparameters according to each model is shown below:

Time

- Frames Per Second(fps): It represents the number of steps in the environment processed per second.
- Iterations: The number of batches of data processed by the algorithm performed during training.
- Time Elapsed: The total time that has passed since the start of training, measured in seconds.
- Time Timesteps: The total number of timesteps (environment steps) the agency has experienced so far.
- Episodes: The interactions between the agent and the environment begin with the agent's initial state and end when a specific condition is met.

Train

- Actor Loss: The difference between the predicted actions and those that maximize the expected cumulative reward.
- Critic Loss: The difference between the predicted values and the actual observed rewards.
- Entropy Loss: A regularization term that encourages the agent's policy to be more random.
- Explained Variance: A measure of how well the value function predicts the rewards. It indicates how much of the variance in the returns is explained by the predicted values.
- Learning Rate: It determines the step size in the parameter space during optimization.
- n updates: The number of times the model has been updated. Each update involves optimizing the policy and value function based on the collected data.
- Policy Loss: It measures how much the predicted actions diverge from the action taken.
- Reward: The average reward the agent obtains in the specified time frame or over the training period.
- Standard deviation(std): It shows how much the advantages vary during the training.
- Value Loss: The difference between the predicted value and actual return.
- Approximate Kullback Leibler divergence(Approx kl): The measure of how much the current policy has changed from the previous policy.
- Clip Fraction: The fraction of clipped samples(ratio of new policy to old policy is clipped) in the training batch.
- Clip Range: It is a range that limits how much the new policy is allowed to deviate from the old policy during an update.
- Entropy Coefficient(Ent Coef): It controls the importance of the entropy term in the total loss function.
- Entropy Loss Coefficient(Ent Coef Loss): It ensures that the entropy of the policy is balanced, preventing it from becoming overly deterministic or too random.

The range of these hyperparameter values is from lowest to the highest. There are two run cases in which the values are kept to the lowest, and the second case makes it to the highest.

Training Process

We have selected Dow Jones 30 stocks as our trading stocks as of January 1, 2020. To train our agent and test its performance, we have used historical daily prices from January 1, 2013, to September 30, 2022, which we obtained from Yahoo Finance. Our experiment consists of three stages: training, validation, and trading. In the training stage, we have generated a well-trained trading agent. The validation stage involves adjusting key parameters such as learning rate and number of episodes. Finally, in the trading stage, we evaluate the profitability of the proposed scheme.

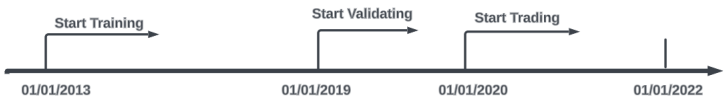

Figure 3-3: Data Splitting

We have divided the complete dataset into three parts for our analysis, as illustrated in Figure 3.1. The data ranging from January 1, 20013, to December 31, 2019, has been used for training purposes, while data from January 1, 2019, to January 1, 2020, has been used for validation. Our agents have been trained on training and validation data to maximize the available information. Finally, we tested our agents' performance on trading data from January 1, 2020, to December 31, 2022.

The dataset has been split into a 70:30 ratio, where 0.07 of the data has been used for training, and 0.03 has been used for testing. To leverage the trading data better, we have continued training our agents during the trading stage. This will help enhance the agent's ability to adapt to market dynamics and perform better.

3.4 Evaluation Metrics

1. Cumulative Return: The total percentage change in an investment's value over a specified period is a metric used to evaluate the performance of the investment. A higher percentage indicates better performance.

2. Sharpe Ratio: It measures the risk-adjusted return of an investment. (The higher the value implies better risk-adjusted return.)

3.5 Summary

The study utilizes quantitative research, which emphasizes collecting and analyzing numerical data, to develop a reinforcement learning-based trading strategy. The study employs the FinRL framework to gain valuable insights into stock market dynamics and create an automated trading strategy. Data collection for this research involved sourcing historical stock price and trading volume data from financial databases such as Yahoo Finance. The entire dataset is split into training, validation, and trading. The research is anchored in a robust reinforcement learning framework, which provides a systematic approach to learning from historical market data and making informed trading decisions. The FinRL frameworks offer modularity, simplicity, and improved market environment modelling. Several Reinforcement Learning (RL) models were trained as trading agents, and their performance was evaluated. The hyperparameters used for each model are listed in the text.

Chapter 4

Findings (Analysis and Evaluation)

4.1 Case Studies

This section will discuss our experiments to determine the most accurate stock market values.

4.1.1 Indicator: Stochastic Oscillator(KDJ)

In this experiment, we integrated the KDJ indicator into our data. The KDJ is a commonly used momentum indicator in technical analysis that helps evaluate price movement. By adding this indicator to our data, we could conduct a more comprehensive analysis of the model. We also identified additional milestones that had a noticeable impact on the model's performance. This information allowed us to make more informed trading decisions.

	A2C	DDPG	PPO	TD3	SAC	Mean Var	djia
date							
2018-01-02	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.001744e+06	1.000000e+06
2018-01-03	1.000368e+06	1.000139e+06	1.000154e+06	1.000726e+06	1.000538e+06	1.007484e+06	1.003976e+06
2018-01-04	1.000993e+06	1.001737e+06	1.000547e+06	1.001656e+06	1.002225e+06	1.011820e+06	1.010115e+06
2018-01-05	1.003801e+06	1.006577e+06	1.002162e+06	1.009003e+06	1.005387e+06	1.032801e+06	1.019010e+06
2018-01-08	1.002771e+06	1.006606e+06	1.001977e+06	1.007159e+06	1.002072e+06	1.027834e+06	1.018490e+06
...	...	...	...	...	...	...	...
2022-12-22	1.341445e+06	1.419844e+06	1.239449e+06	1.476003e+06	1.641591e+06	1.820354e+06	NaN
2022-12-23	1.345813e+06	1.425963e+06	1.250034e+06	1.484035e+06	1.652390e+06	1.831295e+06	NaN
2022-12-27	1.343412e+06	1.426130e+06	1.254492e+06	1.482023e+06	1.651771e+06	1.830975e+06	NaN
2022-12-28	1.330396e+06	1.410909e+06	1.238620e+06	1.463981e+06	1.634153e+06	1.815133e+06	NaN
2022-12-29	1.345827e+06	1.426518e+06	1.249422e+06	1.483819e+06	1.649827e+06	1.834839e+06	NaN

Figure 4-1: Portfolio Value Table for KDJ

The values in the table represent the portfolio values for each algorithm and the corresponding benchmarks (Mean Var and DJIA) on specific dates spanning from 2018-01-02 to 2022-12-29. A portfolio value of 1.0e+06 is often used as an initial benchmark.

A2C, DDPG, PPO, TD3, SAC: All algorithms have a general upward trend in the portfolio values across the observed period. This suggests that, in general, these reinforcement learning algorithms have successfully adapted and made profitable decisions in the dynamic and complex environment of the stock market. Variations in performance are noticeable, indicating that certain algorithms may outperform others during specific market conditions. For instance, observing periods of rapid market growth or decline could help identify algorithms demonstrating robust performance across different market scenarios.

The Mean Var metric is included in the comparison, providing insight into the volatility of the portfolio values. A higher Mean Var could indicate increased risk, while a lower value might suggest a more stable performance.

DJIA Comparison: DJIA values are also included for benchmarking. Comparing algorithmic performance against DJIA helps to contextualize their effectiveness in the broader market. Deviations from DJIA trends might indicate the algorithms' ability to exploit market inefficiencies or navigate market conditions more effectively.

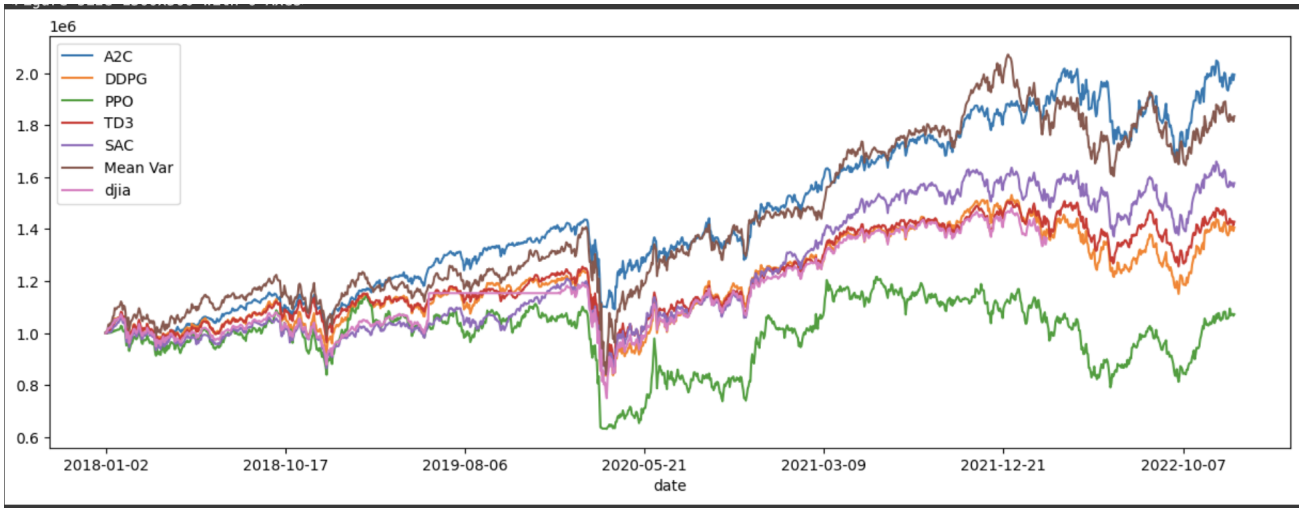

Figure 4-2: Graph created with Stochastic Oscillator(KDJ)

The line plot displays the fluctuations in portfolio values of different models, Mean Var and DJIA - over a specified period. A different colour represents each line. By observing the patterns and trends in these lines, we can gain a preliminary understanding of the relative performance of these models. Based on the Graph, Actor-2-Critic (A2C) outperformed Mean-variance and the Dow Jones Industrial Average (DJIA).

4.1.2 Changing Hyperparameter Values

We experimented to determine the best model for making trading decisions. We modified various hyperparameters such as the learning rate, steps, entropy coefficient, buffer size, and batch size to achieve this. We used different parameter values within each variable's minimum and maximum range to evaluate how the RL model responds to other configurations. The lowest parameter values were used to establish a baseline for the model's performance under minimal complexity. In contrast, the highest values helped us explore the upper bounds of the model's capacity and performance. However, in the real world, these values are based on historical or present data analysis.

PyPortfolioOpt, a powerful Python library for portfolio optimization, was utilized to compute the portfolio values shown in the snapshot above. This calculation considered various state variables, reward functions, action space, and other crucial parameters.

Output with Value1

For this section, we took the following values of the hyperparameter:

1. n steps = 128
2. entropy coefficient = 0.001
3. learning rate = 0.0001
4. buffer size = 100000
5. batch size = 64
6. learning start = 100

The values mentioned above provided us with the following Sharpe Ratio Values:

1. A2C: 1.030
2. DDPG: 1.467
3. PPO: 0.827
4. TD3: 1.166
5. SAC: 1.344

A higher Sharpe ratio represents superior risk-adjusted performance. Deep Deterministic Policy Gradients (DDPG) is the agent that is currently outperforming other models based on this ratio. However, other factors must also be considered to determine which models best suit trading decisions.

date	A2C	DDPG	PPO	TD3	SAC	Mean Var	djia
2018-01-02	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.001744e+06	1.000000e+06
2018-01-03	1.000067e+06	1.000008e+06	1.000534e+06	9.998225e+05	1.000245e+06	1.007484e+06	1.003976e+06
2018-01-04	1.000164e+06	1.002261e+06	1.001280e+06	1.001732e+06	1.001597e+06	1.011820e+06	1.010115e+06
2018-01-05	1.000446e+06	1.008982e+06	1.003861e+06	1.003409e+06	1.007027e+06	1.032801e+06	1.019010e+06
2018-01-08	1.000516e+06	1.009608e+06	1.003869e+06	1.002844e+06	1.007467e+06	1.027834e+06	1.018490e+06
...	...	...	...	...	...	...	...
2022-12-22	1.478158e+06	1.381410e+06	1.544894e+06	1.483012e+06	1.345558e+06	1.820366e+06	NaN
2022-12-23	1.486048e+06	1.386527e+06	1.551355e+06	1.491477e+06	1.351569e+06	1.831307e+06	NaN
2022-12-27	1.485666e+06	1.387331e+06	1.552265e+06	1.493529e+06	1.353234e+06	1.830987e+06	NaN
2022-12-28	1.471673e+06	1.370091e+06	1.531759e+06	1.480543e+06	1.339401e+06	1.815145e+06	NaN
2022-12-29	1.493263e+06	1.389026e+06	1.545319e+06	1.497823e+06	1.350727e+06	1.834851e+06	NaN

Figure 4-3: Portfolio Value table of Low Hyperparameter Values

The snapshot above displays the portfolio values for different reinforcement learning agents (A2C, DDPG, PPO, TD3, SAC) on various dates. The five columns represent each agent's portfolio values, and the next column displays the mean-variance value calculated from the data of that particular date. The last row represents the Dow Jones Industrial Average values. By analyzing the values in these columns over time, the Graph can help assess the performance of each reinforcement learning agent in managing the portfolio compared to the mean-variance and the broader market represented by the Dow Jones Industrial Average(DJIA). This evaluation helps understand the effectiveness of each agent in making investment decisions.

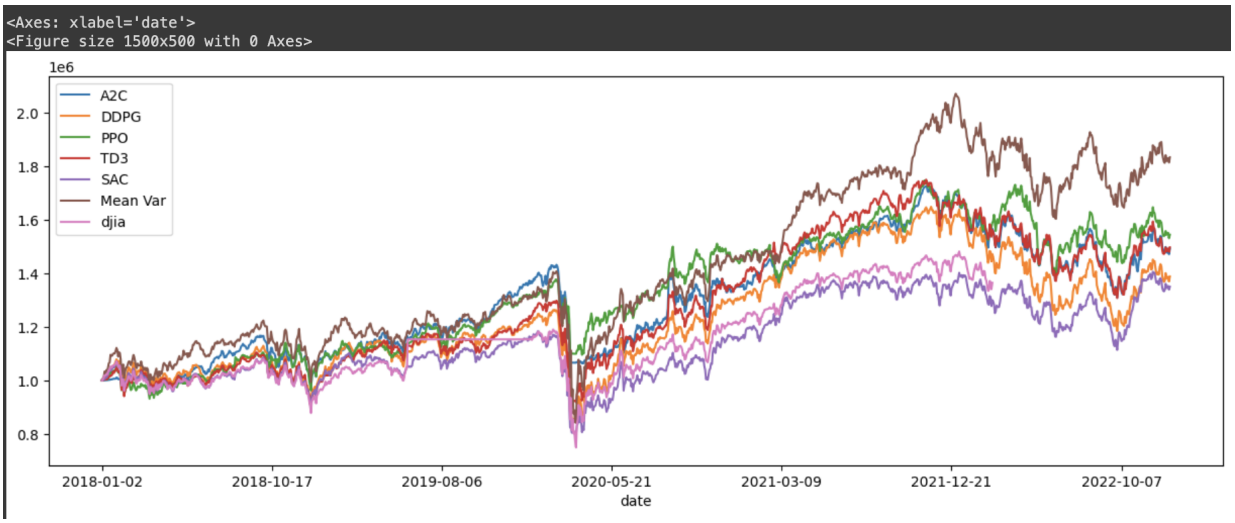

Figure 4-4: Portfolio Value Representation using Graph

After evaluating the other parameters and state variables, it was found that the PPO agent outperformed the others and is considered the best model for trading decisions compared to the Dow Jones Industrial Average. However, the figure shows that using the lowest hyperparameter values will not yield satisfactory results. The mean variance outperforms all other agents and the Dow Jones Industrial Average. This ultimately leads to the conclusion that the PPO agent is the best model for trading decisions compared to the Dow Jones Industrial Average.

Ouput with Value2

For this section, we took the following values of the hyperparameter:

1. n steps = 2048
2. entropy coefficient = 0.01
3. learning rate = 0.001
4. buffer size = 1000000
5. batch size = 256
6. learning start = 100

The values mentioned above provided us with the following Sharpe Ratio Values:

1. A2C: 1.387
2. DDPG: 1.375
3. PPO: 1.205
4. TD3: 1.169
5. SAC: 1.169

date	A2C	DDPG	PPO	TD3	SAC	Mean Var	djia
2018-01-02	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.000000e+06	1.001744e+06	1.000000e+06
2018-01-03	1.000223e+06	1.000884e+06	1.000098e+06	1.000337e+06	1.000263e+06	1.007484e+06	1.003976e+06
2018-01-04	1.002107e+06	1.004181e+06	1.000182e+06	1.001139e+06	1.000600e+06	1.011820e+06	1.010115e+06
2018-01-05	1.007728e+06	1.009540e+06	1.004165e+06	1.008338e+06	1.002652e+06	1.032801e+06	1.019010e+06
2018-01-08	1.007338e+06	1.009051e+06	1.004099e+06	1.008954e+06	1.002311e+06	1.027834e+06	1.018490e+06
...	...	...	...	...	...	...	...
2022-12-22	1.559099e+06	1.680935e+06	1.199510e+06	1.307982e+06	1.528699e+06	1.820366e+06	NaN
2022-12-23	1.561061e+06	1.688579e+06	1.204714e+06	1.314796e+06	1.537983e+06	1.831307e+06	NaN
2022-12-27	1.556510e+06	1.690680e+06	1.206783e+06	1.319448e+06	1.540895e+06	1.830987e+06	NaN
2022-12-28	1.520795e+06	1.673064e+06	1.195198e+06	1.302076e+06	1.525846e+06	1.815145e+06	NaN
2022-12-29	1.554598e+06	1.691702e+06	1.208365e+06	1.314034e+06	1.535027e+06	1.834851e+06	NaN

Figure 4-5: Portfolio Value Table for Highest Hyperparameter Value

Actor-2-Critic(A2C) is the agent that is currently outperforming other models based on this ratio. However, other factors must also be considered to determine which models best suit trading decisions.

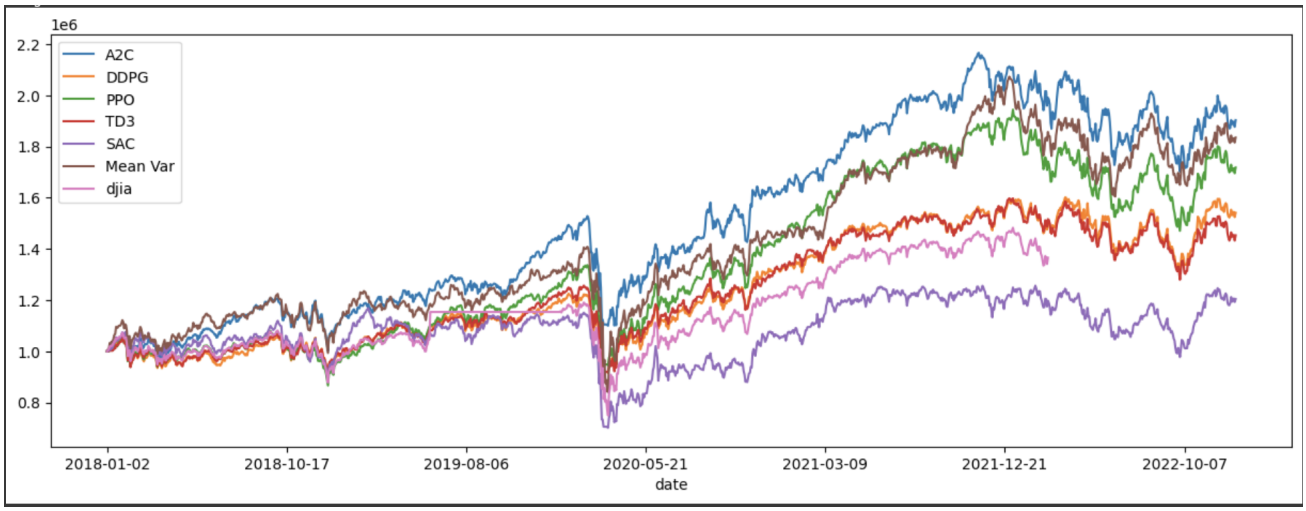

Figure 4-6: Portfolio Value Representation using Graph

According to the above graph representation, it is found that Actor-2-Critic(A2C) is the agent that has outperformed all the agents, mean-variance, and Dow Jones Industrial Average values. This means Actor-2-Critic (A2C) is the best agent for trading decisions in the stock market.

4.2 Summary of Findings

After a thorough experiment, it was concluded that the Actor-2-Critic model outperformed other models in making informed market stock trading decisions. Its effectiveness can be attributed to its unique approach and ability to accurately analyze and interpret market data.

Therefore, it can be considered a reliable and valuable tool for traders looking to maximize their profits while minimizing risks. Deep Deterministic Policy Gradient (DDPG) is another model that can be used for implications in the stock market.

4.3 Discussion

This part briefly explains the results obtained from our research paper, including all the outcomes derived from running the models. Furthermore, we will evaluate the limitations of our study in the following chapter. Additionally, we will discuss potential future studies that can be conducted in the same field.

4.3.1 Interpretation of Results

After comprehensively evaluating our research, we concluded that the Actor-2-Critic model was the highest-performing model across all factors. By adjusting the hyperparameters and adding various state variables, we found that Actor-2-Critic consistently outperformed all other models. Notably, the addition of the Stochastic Oscillator (KDJ) significantly improved the performance of the Actor-2-Critic model, making it a promising choice for use in the stock market. Additionally, we discovered that utilizing hyperparameter values closest to their highest values resulted in more accurate predictions. In these situations, the Actor-2-Critic model consistently outperformed all other models, demonstrating its superiority in stock market prediction.

4.3.2 Implication of Results

The findings of this research have profound implications for the field of stock market prediction using reinforcement learning. The consistent outperformance of the A2C model suggests its practical applicability in real-world financial decision-making. The enhanced performance with the addition of the KDJ indicator underscores the importance of incorporating relevant technical indicators in reinforcement learning models for stock market applications. Practitioners and decision-makers in the financial sector can leverage these insights to improve the accuracy of their predictions and make more informed investment decisions.

4.3.3 Limitations of the study

While our research has yielded promising results, it is essential to acknowledge its limitations. These include constraints in data availability, the specificity of the chosen technical indicators,

and the inherent challenges in predicting stock market behaviour. The limitations provide context for understanding the scope of the study and highlight areas where future research could further refine and expand the current findings.

4.3.4 Areas of Future Research

Building on the insights gained from this study, several areas for future research emerge. Future investigations include investigating the impact of external factors, such as market news or economic indicators, which could provide a more comprehensive understanding of stock market dynamics. The identified limitations suggest avenues for refining the methodology and exploring new directions for more accurate and reliable stock market prediction models.

Chapter 5

Conclusion

5.1 Summary

This study investigates the effectiveness of Reinforcement Learning (RL) for continuous decision-making in stock trading, specifically focusing on holding, buying, or selling stocks. The primary goal is to train RL models and identify the most effective ones within the stock market context. The A2C model, particularly when combined with the Stochastic Oscillator (KDJ) indicator, emerges as a strong performer, showcasing significant potential in financial decision-making. The study underscores the superiority of the A2C model, especially when hyperparameter values are optimized near their upper limits.

The study's implications extend beyond its immediate scope, offering practical applications for practitioners and decision-makers in the financial sector. The A2C model, coupled with relevant technical indicators, proves valuable for enhancing the accuracy of stock market predictions. This aligns with existing literature recognizing the effectiveness of RL models, notably A2C, in capturing intricate market dynamics.

5.2 Limitations

Despite the successes, it is crucial to acknowledge the limitations of this research. Constraints related to data availability, the specificity of chosen technical indicators, and the inherent challenges in predicting stock market behaviour constrain the study's scope. These limitations provide valuable insights for future research, highlighting areas requiring refinement or alternative approaches.

5.3 Recommendation

Looking forward, several promising areas for future exploration emerge. Researchers could delve into alternative technical indicators, assess the impact of external factors like market news, and refine methodologies to overcome identified limitations. The dynamic nature of financial markets creates opportunities for continuous improvement and enhancement of predictive models. Addressing these recommendations could contribute to a more comprehensive understanding of RL applications in stock trading.

5.4 Future Works

The study contributes to the evolving knowledge in stock market prediction using RL. The success of the A2C model, particularly with the incorporation of the KDJ indicator, underscores its practical applicability and introduces new avenues for advancing the field. Future research could explore diverse technical indicators, analyze the influence of external factors, and refine methodologies to overcome identified limitations. The study is a foundation for ongoing advancements at the intersection of artificial intelligence and financial markets, emphasizing the potential for continued growth and innovation in predictive modelling.

Bibliography

[1] Bechir Trabelsi. Safe and efficient off-policy reinforcement learning., Jul 2021.

[2] John Moody and Matthew Saffell. Reinforcement learning for trading systems and portfolios. *Journal of Forecasting*, 2001.

[3] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[4] Shuyang Gu, John Kelly, Dacheng Xiu, and Paul Algoet. Reinforcement learning for trading. *SIAM Journal on Financial Mathematics*, 2018.

[5] Lucarelli, Giorgio and Borrotti, Matteo. A deep Q-learning portfolio management framework for the cryptocurrency market. *Neural Computing and Applications*, 32:1–16, 12 2020.

[6] Tretina, Kat. What Is The Stock Market? How Does It Work? *Forbes Advisor*, Mar 2023.

[7] Fischer, T G. *Reinforcement Learning in Financial Markets-A Survey*. 2018.

[8] Tan, Zhiyong and Quek, Chai and Cheng, Philip Y K. Stock trading with cycles: A financial application of ANFIS and reinforcement learning. *Expert Syst. Appl.*, 38(5):4741–4755, May 2011.

[9] Bertoluzzo, Francesco and Corazza, Marco. Testing different reinforcement learning configurations for financial trading: Introduction and applications. *Procedia Econ. Finance*, 3:68–77, 2012.

[10] Jin, O and El-Saawy, H. *Portfolio Management Using Reinforcement Learning. Technical report working paper*. 2016.

[11] Ritter, Gordon. Machine learning for trading. *SSRN Electron. J.*, 2017.

[12] Huang, Chien Yi. Financial trading as a game: A deep reinforcement learning approach. Jul 2018.

[13] Moody, John and Wu, Lizhong and Liao, Yuansong and Saffell, Matthew. Performance functions and reinforcement learning for trading systems and portfolios. *J. Forecast.*, 17(5-6):441–470, Sep 1998.

[14] Moody, J and Saffell, M. Learning to trade via direct reinforcement. *IEEE Trans. Neural Netw.*, 12(4):875–889, 2001.

[15] Deng, Yue and Bao, Feng and Kong, Youyong and Ren, Zhiquan and Dai, Qionghai. Deep direct reinforcement learning for financial signal representation and trading. *IEEE Trans. Neural Netw. Learn. Syst.*, 28(3):653–664, Mar 2017.

[16] Lim, B and Zohren, S and Roberts, S. Enhancing Time Series Momentum Strategies Using Deep Neural Networks. *The Journal of Financial Data Science*, 1(4):19–38, 2019.

[17] Jiang, Zhengyao and Xu, Dixing and Liang, Jinjun. A deep Reinforcement Learning framework for the financial portfolio management problem. 2017.

[18] Mnih, V., Kavukcuoglu, K., Silver, D., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, Feb 2015.

[19] Jiang, Z and Zhao, H and Li, H. Stock trading with Recurrent Reinforcement Learning (RRL). In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2017.

[20] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[21] S. Gu, J. Kelly, D. Xiu, and P. Algoet. Reinforcement learning for trading. *SIAM Journal on Financial Mathematics*, 2018.

[22] M. Deviaene. Algorithmic trading using q-learning and recurrent reinforcement learning. 2017.

[23] C. Liu. Portfolio management using reinforcement learning. 2019.

[24] J. Papenbrock and Q. Song. Financial portfolio optimization with reinforcement learning. 2020.