Comparative Analysis of Reinforcement Learning Agents for Portfolio Management

Mummidi Devi Siva Rama Saran, Akshayaa B K, R Sai Raghavendra, Poornima N
Amrita School of Artificial Intelligence, Coimbatore, Amrita Vishwa Vidyapeetham, India
{ramasaranmummidi, akshayaabk1908, sairaghavendra179, poornima.n2425}@gmail.com

***Abstract*—The research objective of this paper is to evaluate and compare the effectiveness of various reinforcement learning (RL) algorithms in optimizing portfolio management strategies. Using the NIFTY 50 dataset from 2018 to 2023, we assess four prominent RL algorithms: DDPG, PPO, TD3, and A2C, focusing on financial metrics like the Sharpe ratio, cumulative return, and risk-adjusted returns. Our findings show that the TD3 algorithm outperformed the others, achieving a Sharpe ratio of 1.249, indicating superior risk-adjusted returns. This study highlights TD3's potential for robust portfolio management and provides insights into the strengths and weaknesses of these RL algorithms in financial applications.**

***Index Terms*—Reinforcement Learning, Portfolio Management, Financial Technology, Sharpe Ratio.**

I. Introduction

The management of financial portfolios is a critical endeavor in investment strategy, where the effective allocation of assets can significantly impact returns and mitigate risks. With the advent of machine learning and artificial intelligence, particularly reinforcement learning (RL), portfolio management has entered a new era of optimization and sophistication. This paper explores the application of RL algorithms—specifically, Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), Advantage Actor-Critic (A2C), and Twin Delayed Deep Deterministic Policy Gradient (TD3)—in the domain of portfolio management using the NIFTY 50 dataset spanning from 2018 to 2023.

The objective of this research is to compare and evaluate the performance of these RL algorithms in optimizing portfolios within the dynamic and complex landscape of financial markets. Each algorithm offers distinct methodologies for learning optimal decision-making strategies, ranging from continuous action spaces to stable training trajectories. By leveraging historical market data and evaluating metrics such as cumulative return, annual return, maximum drawdown and Sharp ratio, this study aims to provide insights into the strengths and weaknesses of each algorithm in the context of portfolio management.

In addition to evaluating performance metrics, this paper also examines the computational efficiency and robustness of each RL algorithm in handling real-world financial datasets. Considering the computational demands and scalability of these algorithms is essential for their practical application in portfolio management, particularly in scenarios involving large-scale investment portfolios and high-frequency trading environments. Thus, this research contributes to a comprehensive understanding of RL techniques in portfolio management and their potential implications for the financial industry.

Reinforcement Learning (RL) is a machine learning paradigm where an agent learns to make decisions by interacting with an environment to maximize cumulative rewards. RL algorithms, inspired by behavioral psychology principles, enable agents to learn optimal decision-making strategies through a process of trial and error. By iteratively adjusting its policy based on feedback from the environment, the agent gradually improves its decision-making abilities and learns to make optimal choices in different situations. This paper explores the application of RL techniques to the domain of portfolio management, highlighting their potential to enhance investment strategies and optimize asset allocation.

This paper's next sections are organised as follows: A brief summary of significant earlier works is provided in Section II. The dataset used is described in Section III, which discusses about the alogorithms used.Section IVwhich also offers a thorough synopsis of the suggested methodological strategy. The experimental results are presented in Section V. The work is finally concluded in Section **??**, which also provides directions for future research studies.

II. Literature Survey

Liang et al. (2018) explored the performances of Deep Deterministic Policy Gradient (DDPG) and Proximal Policy Optimization (PPO) algorithms in portfolio management, highlighting their potential for optimizing investment strategies [1]. Joshi (2022) investigated the performance of A2C, PPO, and DDPG models, with the A2C model demonstrating superior cumulative returns and Sharpe ratio. These studies emphasize the importance of evaluating RL algorithms' performances in managing investment portfolios and provide insights into their effectiveness in real-world financial applications [2].

Santos et al. (2023) evaluated the effectiveness of RL techniques in portfolio optimization, reporting an average increase of 12% in returns without a commensurate increase in risk. This study underscores the advantages of RL algorithms in dynamically adjusting portfolio allocations to maximize returns while mitigating risks [3].In addition to RL-based approaches, traditional optimization techniques such as mean-variance analysis and Sharpe ratio-based portfolios have also been widely studied (Jin et al., 2016). However, these methods

may lack the adaptability and flexibility offered by RL algorithms in handling complex and dynamic market conditions. [4]

Wang et al. (2019) addressed this challenge by developing RL agents capable of achieving significant returns in daily trading scenarios, showcasing the potential of RL techniques in dynamic portfolio management [5].Pawar et al. (2024) proposed a portfolio management system using Deep Q-Network (DQN) and Advantage Actor-Critic (A2C) models, aiming to surpass the risk-adjusted returns of conventional portfolio managers. [6]

Gao et al. (2021) developed an online optimal investment portfolio model using the DDPG algorithm and convolutional neural networks to enhance perception and address data correlation. Their focus was on improving income acquisition in portfolio management through the utilization of Deep Reinforcement Learning (DRL) techniques [7]. On the other hand, Zhai (2021) proposed an ensemble strategy that combined Proximal Policy Optimization (PPO) and Twin Delayed Deep Deterministic Policy Gradient (TD3) methods, demonstrating superior performance over benchmarks in terms of cumulative return, annualized return, and Sharpe ratio, underscoring the robustness of their approach [8].

III. Materials and Methodology

A. Dataset Description

For the purposes of this research paper, we utilized historical data from the NIFTY 50 and Dow Jones Industrial Average (DJIA) indices, obtained from Yahoo Finance. The dataset for each index includes the following columns: Date, Open, High, Low, Close, Adjusted Close, and Volume. The "Date" column records the specific trading date, while the "Open," "High," "Low," and "Close" columns capture the respective prices of the index at the start, peak, trough, and end of the trading session. The "Adjusted Close" column reflects the closing price adjusted for corporate actions such as dividends and stock splits, providing a more accurate reflection of the stock's value over time. The "Volume" column records the total number of shares traded during the session. These columns collectively provide a comprehensive view of market activity, facilitating detailed trend analysis, return calculations, and risk management for portfolio analysis and management. This structured data is crucial for conducting a thorough examination of market behaviors and portfolio performance across different periods.

B. Proposed Method

Our research endeavors to systematically explore the efficacy of reinforcement learning (RL) algorithms in the domain of portfolio management. The proposed methodology encompasses a structured approach encompassing data acquisition, algorithmic implementation, training, validation, and performance analysis.

1) Data Acquisition and Preparation: The foundation of our investigation lies in the meticulous acquisition and preparation of historical stock market data. Leveraging the `yfinance` library, we developed a robust data loader class'Loader', tasked with retrieving stock price information for constituents of the Dow Jones Industrial Average (DJIA) and NIFTY 50. This class facilitates the seamless retrieval of historical stock data within predefined date ranges, ensuring consistency and reproducibility in our experiments.

2) Algorithmic Implementation: Central to our methodology is the implementation of various RL algorithms tailored to portfolio optimization tasks. Specifically, we employ Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), Advantage Actor-Critic (A2C), and Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithms. Each algorithm is meticulously configured with suitable hyperparameters and neural network architectures optimized for portfolio management objectives.

3) Training Procedure: The training process entails iterative interactions between the RL agent and the financial environment. Through sequential decision-making, the agent learns to allocate portfolio assets based on observed market dynamics and historical data. Training is conducted using historical stock market data, with RL algorithms optimizing their policies to maximize cumulative wealth or other predefined performance metrics.

4) Validation and Testing: Validation serves as a critical step in assessing the generalization and robustness of trained RL agents. Separate validation datasets are employed to evaluate the performance of trained algorithms under diverse market conditions. Subsequently, testing procedures are conducted on unseen data to gauge the adaptability and efficacy of RL-based portfolio management strategies in real-world scenarios.

5) Performance Analysis: Comprehensive performance analysis techniques are employed to quantitatively evaluate the effectiveness of RL algorithms in portfolio management. Performance metrics such as Sharpe ratio, drawdowns, and annualized return are computed using established methodologies. `Pyfolio`, a widely-used performance analysis library, is utilized to facilitate the quantitative comparison of RL-based strategies against traditional investment approaches.

C. Comparative Algorithms

The deep reinforcement learning (DRL) methodologies employed in this work for comparison include:

1) Deep Deterministic Policy Gradient (DDPG)
2) Proximal Policy Optimization (PPO)
3) Twin Delayed Deep Deterministic Policy Gradient (TD3)
4) Advantage Actor-Critic (A2C)

Deep Deterministic Policy Gradient (DDPG) is an actor-critic algorithm designed for environments with continuous action spaces. DDPG utilizes a deterministic policy to select actions, while a critic network evaluates these actions, enabling efficient handling of high-dimensional state and action spaces in portfolio management applications [9].

Proximal Policy Optimization (PPO) is known for its training stability and reliability. PPO introduces a novel objective function that maintains a balance between exploring new strategies and exploiting known ones. This is achieved by using a clipped surrogate objective, preventing excessive policy updates and ensuring stable training, making it suitable for financial markets where stability is crucial [10].

Twin Delayed Deep Deterministic Policy Gradient (TD3) enhances the DDPG algorithm by addressing its overestimation bias. TD3 employs two critic networks to provide a more accurate estimation of the value function and introduces a delay in the policy update to further stabilize training. This results in improved performance and robustness in continuous action spaces typical of portfolio management [11].

Advantage Actor-Critic (A2C) synchronizes multiple actor-learners to interact with their environment in parallel, accelerating the learning process. Each actor interacts with its own copy of the environment, and their experiences are used to update a shared critic network. This approach leads to faster and more robust policy updates, beneficial for dynamically adjusting asset allocations in portfolio management [12].

D. Experimental Setup and Evaluation Metrics

In our experimental setup, each RL algorithm is trained and tested separately using the datasets obtained from historical stock market data. The training process involves iteratively optimizing the algorithm's policies to maximize cumulative return while effectively managing risk. The trained models are then evaluated using a comprehensive set of performance metrics.

Evaluation metrics such as cumulative return provide insights into the overall profitability of the portfolio strategies employed by each algorithm. Annual return standardizes the performance measure on a yearly basis, facilitating comparisons across different time frames. The Sharpe ratio evaluates the risk-adjusted return, considering both the portfolio's profitability and its volatility. Additionally, maximum drawdown offers valuable insights into the potential downside risk associated with each algorithm's portfolio management strategy.

TABLE I
EVALUATION METRICS FOR PERFORMANCE ASSESSMENT

S.No	Evaluation Metric	Formula
1	Cumulative Returns	$\frac{P_{end} + D - P_{start}}{P_{start}} \times 100\%$
2	Sharpe Ratio	$\frac{\mathbb{E}[R_p - R_f]}{\sigma_p}$
3	Annual Return	$(1 + \text{Cumulative Returns})^{\frac{1}{n}} - 1$
4	Maximum Drawdown	$\frac{P_t - P_{peak}}{P_{peak}}$

where:

- P_{end} is the portfolio value at the end of the period.
- P_{start} is the portfolio value at the beginning of the period.
- D is the total dividends received during the period.
- $\mathbb{E}[R_p - R_f]$ is the expected excess return of the portfolio over the risk-free rate.
- σ_p is the standard deviation of the portfolio returns.
- n is the number of trading days.
- P_t is the portfolio value at time t.
- P_{peak} is the maximum portfolio value prior to time t.

TABLE II
RISK-ADJUSTED EVALUATION METRICS

S.No	Evaluation Metric	Formula
1	Sortino Ratio	$\frac{\bar{R} - R_f}{\sigma_d}$
2	Omega Ratio	$\frac{\int_{R_f}^{\infty} (1 - F(r))\, dr}{\int_{-\infty}^{R_f} F(r)\, dr}$
3	Kurtosis	$\frac{N \sum_{i=1}^{N} (R_i - \bar{R})^4}{\left(\sum_{i=1}^{N} (R_i - \bar{R})^2\right)^2} - 3$
4	Calmar Ratio	$\frac{\text{Annualized Return}}{\text{Maximum Drawdown}}$

where:

- $\bar{R}$ is the mean return.
- R_f is the risk-free rate.
- σ_d is the downside deviation.
- $F(r)$ is the cumulative distribution function of returns.
- N is the number of returns.
- R_i is an individual return.
- Annualized Return is the average yearly return.
- Maximum Drawdown is the maximum observed loss from peak to trough.

The Sortino Ratio measures the risk-adjusted return of an investment, taking into account only the downside risk, which is the deviation below a certain target or risk-free rate. A higher Sortino Ratio indicates a better risk-adjusted performance.

The Omega Ratio evaluates the risk-return profile of an investment by comparing the probability-weighted returns above and below a threshold, typically the risk-free rate. It provides a comprehensive view of downside risk and upside potential.

Kurtosis quantifies the degree of heaviness or lightness in the tails of a distribution compared to a normal distribution. Positive kurtosis indicates heavier tails, implying a higher probability of extreme returns, while negative kurtosis suggests lighter tails.

The Calmar Ratio measures the risk-adjusted performance of an investment by dividing the annualized return by the maximum drawdown, representing the largest peak-to-trough decline in value. A higher Calmar Ratio signifies better risk-adjusted returns relative to the maximum loss experienced.

IV. Experimental Results

In the evaluation of various algorithms using NIFTY 50 and DJIA data, the experimental results provide crucial insights into their performance and effectiveness. Across both datasets, the algorithms exhibit distinct strengths and weaknesses, offering valuable implications for portfolio management strategies. Specifically, the analysis unveils notable differences in risk-adjusted returns, drawdowns, and cumulative gains among the algorithms. These findings underscore the importance of selecting an appropriate algorithm tailored to specific investment objectives and risk preferences.

TABLE III
Performance evaluation for each algorithm obtained using NIFTY 50

Algorithms	Evaluation Metrics			
	drawdown	sharpe ratio	annual returns	Cumulative returns
TD3	-0.0697	1.249	0.153	0.09149
DDPG	-0.1227	1.094	0.136	0.0821
A2C	-0.096	1.00	0.117	0.0706
PPO	-0.080	1.091	0.108	0.065

TABLE IV
Performance evaluation for each algorithm obtained using DJIA data

Algorithms	Evaluation Metrics			
	drawdown	sharpe ratio	annual returns	Cumulative returns
TD3	-0.08	1.57	0.197	0.280
DDPG	-0.092	1.36	0.21	0.31
A2C	-0.106	1.535	0.225	0.321
PPO	-0.071	1.51	0.162	0.229

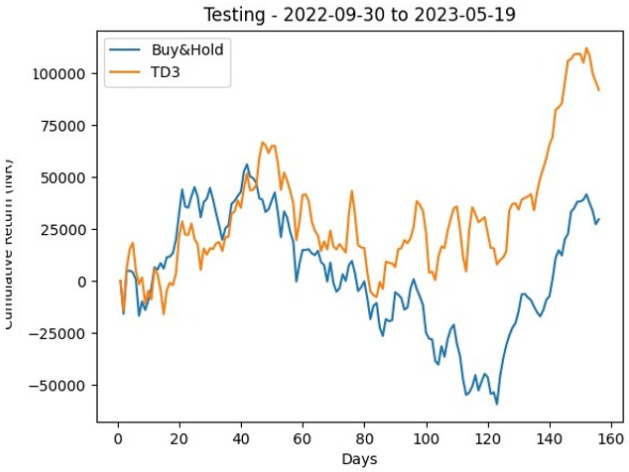

Fig. 1. Performance of TD3 on test data for NIFTY 50

The experimental results from the performance evaluation of various algorithms using NIFTY 50 and DJIA data reveal significant insights into their efficacy. For the NIFTY 50 data, the TD3 algorithm demonstrates superior performance with the highest Sharpe ratio of 1.249 and annual returns

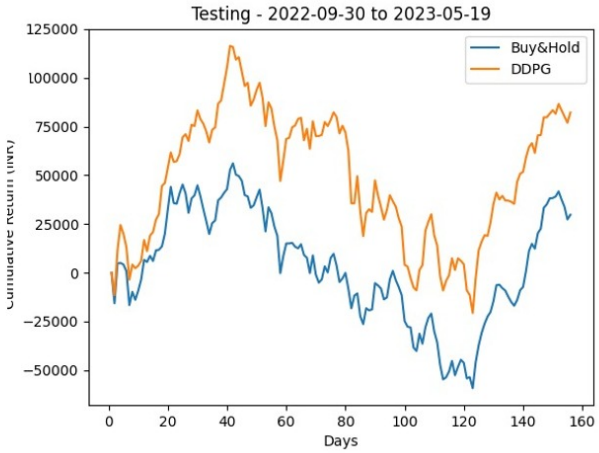

Fig. 2. Performance of DDPG on test data for NIFTY 50

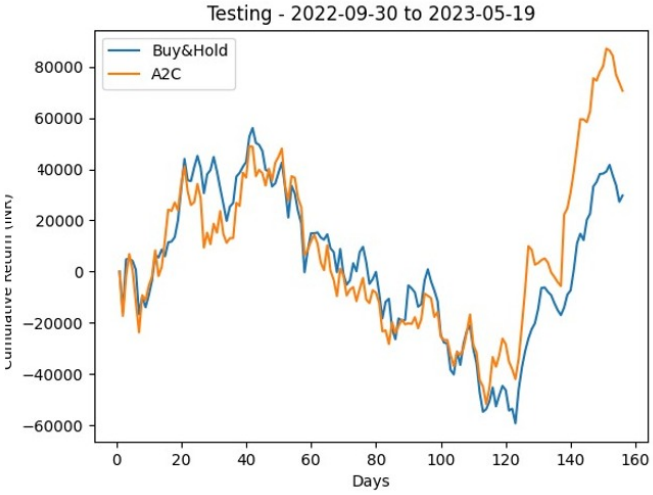

Fig. 3. Performance of a2c on test data for NIFTY 50

of 0.135, indicating strong risk-adjusted returns and yearly gains. However, its drawdown of -0.697 suggests potential for higher losses compared to others. DDPG also performs well, with the highest cumulative returns of 0.0821 and a Sharpe ratio of 1.049, showcasing its strength in long-term investment strategies. The PPO algorithm stands out with the lowest drawdown of -0.080, making it ideal for risk-averse strategies, despite having the lowest annual and cumulative returns.

For the DJIA data, TD3 again leads with the highest Sharpe ratio of 1.57 and annual returns of 0.197, along with a minimal drawdown of -0.081, reinforcing its robustness across different datasets. The a2c algorithm, while not leading in Sharpe ratio or annual returns, achieves the highest cumulative returns of 0.31, indicating its potential for significant long-term gains. PPO also maintains a low drawdown of -0.071, similar to

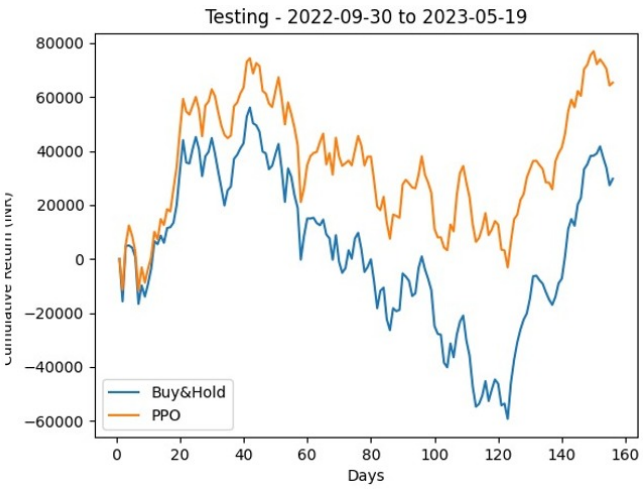

Fig. 4. Performance of ppo on test data for NIFTY 50

its performance with NIFTY 50 data, further establishing its capability to minimize losses.

For the dataset used, the RL algorithms PPO, A2C, and TD3 consistently outperformed the traditional Buy and Hold strategy. The TD3 algorithm (Figure 1) and DDPG (Figure 2) exhibited the highest cumulative returns, surpassing $100,000, while Buy and Hold ended with a significant loss of around $50,000. Among the RL strategies, TD3 demonstrated the most robust performance, maintaining a steady upward trend with higher returns and better volatility management compared to PPO (Figure 4) and A2C (Figure 3). The results indicate that TD3 is particularly effective in generating higher and more consistent returns, highlighting its potential in portfolio management.

TABLE V
RISK ADJUSTED PERFORMANCE EVALUATION FOR NIFTY 50

Algorithms	**Evaluation Metrics**			
	sortino ratio	**omega ratio**	**kurtosis**	**calmar ratio**
TD3	1.929	1.226	0.075	2.205
DDPG	1.590	1.195	0.594	1.115
A2C	1.553	1.176	0.634	0.221
PPO	1.621	1.188	0.106	0.1.502

In comparing the performance of reinforcement learning algorithms TD3, DDPG, A2C, and PPO, as evaluated through risk-adjusted metrics, notable differences emerge. TD3 demonstrates superior risk-adjusted returns, boasting the highest Sortino Ratio (2.205) and Calmar Ratio (1.929) among the algorithms. Conversely, DDPG, A2C, and PPO exhibit varying degrees of performance across these metrics. To enhance the performance of DDPG, adjustments in parameters related to risk management and downside protection may prove beneficial, such as increasing the exploration noise or adjusting the target policy update frequency. Similarly, optimizing risk management strategies, alongside potential fine-tuning of learning rates or exploration techniques, could augment the performance of A2C and PPO, both of which display lower Sortino and Calmar Ratios compared to TD3. These findings underscore the nuanced interplay between algorithmic design and performance outcomes, highlighting the importance of parameter optimization tailored to specific algorithmic frameworks and objectives.

V. CONCLUSION

In this study, we explored the application of reinforcement learning (RL) algorithms, including DDPG, TD3, PPO, and A2C, for portfolio optimization using historical DJIA data. Our experiments demonstrated that RL algorithms significantly improve cumulative returns during training and exhibit competitive performance in validation, with key metrics such as cumulative return, Sharpe ratio, and annualized return confirming their efficacy in asset management. Testing on unseen data highlighted the robustness and consistency of these algorithms across varying market conditions. Comparative analysis revealed distinct strengths among the algorithms in risk mitigation and return maximization. Overall, our findings underscore the practical utility of RL algorithms in enhancing investment strategies and decision-making processes in financial markets, paving the way for further research and practical implementation.

REFERENCES

[1] Adversarial Deep Reinforcement Learning in Portfolio Management. arXiv.org.
[2] Joshi, D. (2022). Portfolio Optimization using Reinforcement Learning. INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT.
[3] Santos, G. C., et al. (2023). Management of investment portfolios employing reinforcement learning. PeerJ Computer Science.
[4] Jin, O.,and Jin, O. (2016). Portfolio Management using Reinforcement Learning. arXiv.org.
[5] Wang, J., et al. (2019). Dynamic Portfolio Management with Reinforcement Learning. arXiv.org.
[6] Pawar, A. A., et al. (2024). Portfolio Management using Deep Reinforcement Learning.
[7] Gao, N., et al. (2021). Online Optimal Investment Portfolio Model Based on Deep Reinforcement Learning. International Conference on Machine Learning and Computing.
[8] Zhai, J. (2021). Study and Improvement on a Reinforcement Learning Framework for the Financial Portfolio Management Problem.
[9] Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., ... and Wierstra, D. (2015). Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971.
[10] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.
[11] Fujimoto, S., Hoof, H., and Meger, D. (2018). Addressing function approximation error in actor-critic methods. arXiv preprint arXiv:1802.09477.
[12] Mnih, V., Badia, A.P., Mirza, M., Graves, A., Harley, T., Lillicrap, T., ... and Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. In International conference on machine learning (pp. 1928-1937).