Automated Stock Trading using Reinforcement Learning

Elise Kim SKIM2665@UWO.CA
Master of Data Analytics
Western University
London, ON N6A 3K7, Canada

Prerna Sharma PSHAR228@UWO.CA
Master of Data Analytics
Western University
London, ON N6A 3K7, Canada

Raj Tulluri RTULLURI@UWO.CA
Department of Computer Science
Western University
London, ON N6A 3K7, Canada

Editor: Raj Tulluri

Abstract

The task of maximizing returns from single stock trading in financial markets presents significant challenges due to the inherently stochastic nature of market dynamics, compounded by a multitude of unpredictable factors. Traditional financial models often fail to capture the complex dependencies and non-linear relationships inherent in market data, resulting in sub-optimal predictive performance. Similarly, while existing machine learning approaches have shown promise, they frequently fall short in effectively managing the sequential decision-making process required for real-time stock trading. This paper introduces a comparison between reinforcement learning (RL) frameworks - Deep Q-Networks (DQN), Advantage Actor-Critic (A2C), and Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithms. By facilitating direct interaction between the RL agent and market data, the approach enables the agent to learn and adapt optimal trading strategies dynamically. This study details the architecture of each algorithm, modifications for the financial context, and the training process involving real-world data from LnT stock provided by Zerodha from August 2011 to February 2024. Performance is evaluated based on profit percentage and the Sharpe ratio, aiming to offer an empirical basis for the effectiveness of RL.

Keywords: Reinforcement Learning, Deep Q-Networks, Advantage Actor-Critic, Twin Delayed Deep Deterministic Policy Gradient

1 Introduction

A stock represents an ownership share in a corporation, making the stockholder a partial owner of that company. Stocks are fundamental units of ownership that are traded on various financial markets, such as stock exchanges. The process of stock trading is influenced by a wide array of factors including economic indicators, company performance, industry trends, and broader geopolitical events, which together contribute to the stock's

price volatility. Successfully trading stocks requires the ability to predict price movements, a task complicated by the markets' complex and stochastic nature. Traditional trading strategies often rely on historical data and statistical models to make predictions; however, these methods can fall short when unexpected market changes occur. The dynamic and often unpredictable environment of stock trading makes it a challenging domain, where traditional approaches may not always capture the full scope of market behaviors.

The inherent limitations of traditional stock trading methods have spurred interest in more adaptive and sophisticated technologies. Among these, reinforcement learning (RL) emerges as a particularly promising approach. In the context of financial trading, an RL agent learns to execute trades based on the state of the market with the goal of maximizing cumulative financial return over time. Unlike traditional models that rely heavily on predefined rules and historical patterns, reinforcement learning adapts its strategies based on ongoing feedback from the market. To address the challenges of financial trading through reinforcement learning, this study focuses on three distinct algorithms. Deep Q-Networks (DQN) are an extension of Q-learning enhanced with deep neural networks (Mnih et al., 2015). Advantage Actor-Critic (A2C), on the other hand, introduces a separate structure for the policy (actor) and value function (critic), which operate concurrently to evaluate and improve the policy (Mnih et al., 2016). This separation helps in reducing the variance of the updates and speeds up learning. Finally, Twin Delayed Deep Deterministic policy gradient (TD3) builds on the success of the deterministic policy gradient by using twin critics and delayed policy updates to mitigate the overestimation of Q-values (Fujimoto et al., 2018a).

2 Related Works

The intersection of reinforcement learning (RL) and financial trading has gained substantial attention, with various methodologies being explored to enhance trading strategies through automated systems.

The study by Ponomareva et al. (2019) discusses the application of reinforcement learning specifically in the context of algorithmic trading. The paper employs an asynchronous advantage actor-critic (A3C) method, integrating several neural network architectures to optimize trading strategies on Russian futures markets. The authors demonstrate that their approach, which incorporates recurrent neural networks (RNNs) with LSTM units, can achieve significant profitability, with results showing a 66% annual return after commissions. However, the study's limitation lies in its focus on a single financial market, which may not generalize across different market environments or financial instruments. Huang (2018) take a different approach by implementing a deep recurrent Q-network (DRQN) to tackle the financial trading game. This paper extends the DRQN framework to manage the trading decisions in the spot foreign exchange market, employing a novel action augmentation technique to enhance learning efficiency by reducing the necessity for random exploration. The results are promising, showing positive returns on 12 different currency pairs under transaction costs. Fujimoto et al. (2018b) in another study, developed a framework treating financial trading as a game, which uniquely positioned RL agents to learn and adapt trading strategies effectively by simulating trading as a sequential game against the market. The study's findings indicated potential for scalable and adaptive trading strate-

gies, but also acknowledged challenges in dealing with extreme market events and anomalies that are not often well-represented in training datasets.

Pricope (2021) comprehensive review spans various RL models such as DQN, A2C, and policy gradient methods, applied in quantitative algorithmic trading. The review meticulously outlines each model's theoretical framework and potential applications, stressing their adaptability and learning capabilities in non-stationary market environments. While providing a thorough theoretical foundation, Pricope's review does not extend into empirical validations, indicating a significant gap where future research could contribute by implementing these theoretical models in practical trading scenarios to verify their effectiveness and practicality.

The selection of the three reinforcement learning algorithms DQN, A2C, and TD3 for this study is grounded in the extensive review of related works and the unique advantages each method offers in addressing the complexities of financial trading. DQN has demonstrated proficiency in handling high-dimensional state spaces through the integration of deep learning techniques. It is also the simplest and the first algorithms in Deep Reinforcement learning, hence it can be used as a baseline in comparing other complex algorithms. The A2C algorithm enhances the traditional actor-critic method by allowing simultaneous updates to the policy and value functions, offering faster and more stable convergence essential for adapting to the evolving financial data. TD3 improves upon the foundational actor-critic framework by introducing mechanisms such as twin Q-networks and delayed policy updates to mitigate common issues like overestimation bias, thereby ensuring more reliable and stable performance.

3 Dataset & Pre-processing

The dataset employed in this study comprises daily stock price data of Larsen & Toubro (LnT) spanning from August 2011 to February 2024 (as shown in Figure 1), obtained from Zerodha, a prominent stock trading platform. This comprehensive dataset includes key variables crucial for stock price analysis and prediction such as open, high, low, close prices, and trading volume for each trading day. The extensive time span covered by this dataset allows for an analysis of long-term trends and patterns, as well as the examination of the algorithms' performance under varying market conditions, including periods of significant economic events and market volatilities.

To augment the dataset's utility for predictive modeling, several technical indicators widely utilized in financial analysis were computed. These indicators help in understanding market trends and dynamics more effectively, aiding the algorithms in making informed predictions. The simple moving average (SMA) is a critical indicator in financial analysis, providing a smoothed average of the stock prices over a specific period. It is calculated by taking the arithmetic mean of a given number of closing prices. For this study, the SMA over a 20-day window is defined in Equation (1), where P_i denotes the closing price on day t

$$\mathrm{SMA}_{20} = \frac{1}{20} \sum_{i=0}^{19} P_{t-i} \tag{1}$$

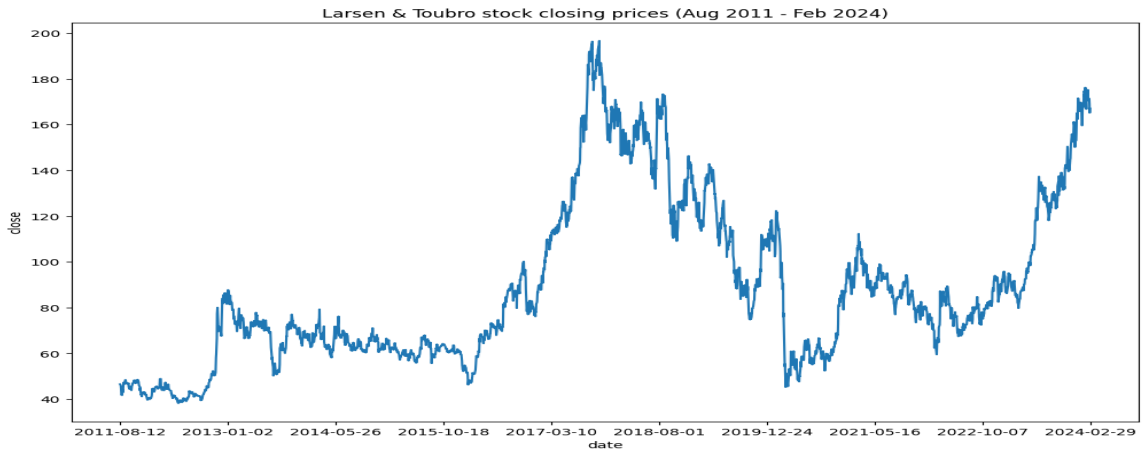

Figure 1: Larsen & Toubro stock closing price from August 2011 to February 2024

Unlike the SMA, the exponential moving average (EMA) places more significant weight on recent data points, which is believed to react faster to price changes. The EMA for the same 20-day period is formulated as defined in Equation (2), where k = $\frac{2}{21}$ represents a smoothing factor and P_i denotes the closing price on day t

$$\text{EMA}_{20} = (P_t \times k) + (\text{EMA}_{yesterday} \times (1 - k)) \tag{2}$$

Bollinger Bands provide a graphical representation of the volatility around the price of a stock. Bollinger Bands include a middle band being a moving average alongside an upper and lower band set at standard deviations from this middle band (Equation (3)).

$$\begin{aligned} \text{Middle Band} &= \text{SMA}_{20} \\ \text{Upper Band} &= \text{SMA}_{20} + 2 \times \sigma_{20} \\ \text{Lower Band} &= \text{SMA}_{20} - 2 \times \sigma_{20} \end{aligned} \tag{3}$$

In trading, these bands serve several functions. The area between the upper and lower band represents the normal price range. Prices touching or crossing the upper band may signal overbought conditions, suggesting a potential selling point, while touching or crossing the lower band might indicate oversold conditions, hinting at a buying opportunity. Moreover, a sudden widening of the bands can signify the start of a new trend given the increased volatility.

The Volume Oscillator is a technical indicator that highlights the difference between a fast-moving average and a slow-moving average of trading volume, showcasing trends in volume before they are reflected in price movements. Equation (4) calculates the oscillator, where the fast volume might be over 5 days and the slow volume over 20 days. A positive value indicates that the short-term volume is higher than the long-term volume, suggesting increased trading activity that could precede upward price movements. Conversely, a negative value implies a decrease in trading activity which might be an early signal of a downward trend or price correction.

$$\text{Volume Oscillator} = \text{SMA}_{fast\_volume} - \text{SMA}_{slow\_volume} \tag{4}$$

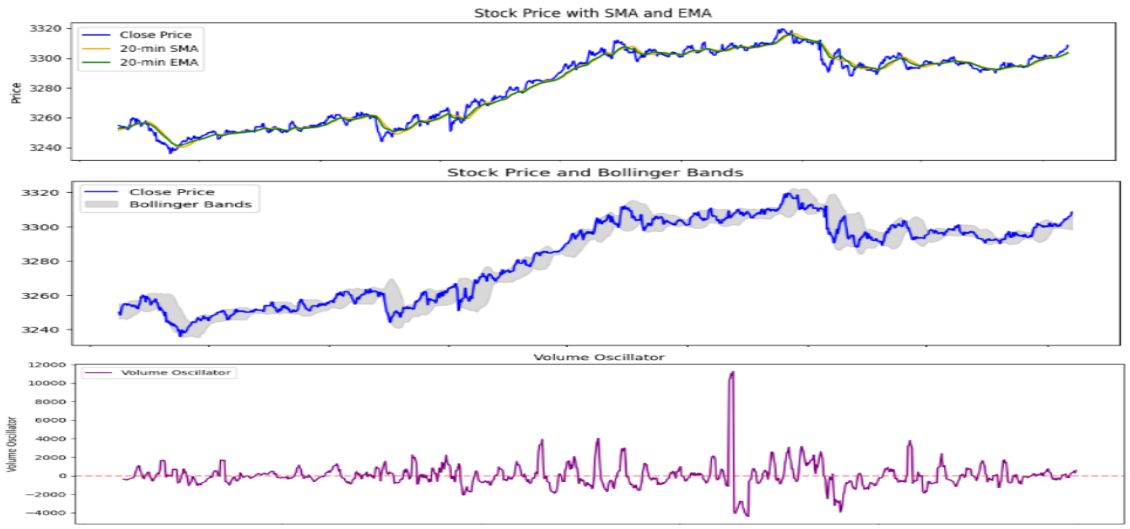

Figure 2: Simple Moving Average, Exponential Moving Average, Bollinger Bands and Volume Oscillator on closing price

The Figure 2 shows the moving averages on a rolling window of 20, and the bollinger band widths on the closing price along with the volume oscillator for the stock.

4 State, Action and Reward

In the development of the reinforcement learning environment for stock trading, defining a precise state representation, action set, and reward function is crucial. The state in the trading environment encapsulates key features that provide the RL agent with necessary market insights at each timestep. The chosen features include the open, high, low, and close prices of the stock, along with technical indicators such as the 20-day Simple Moving Average (SMA_20) and the 20-day Exponential Moving Average (EMA_20), Upper/Lower Bollinger bands and the Volume Oscillator (as shown in Equation (5)).

$$\text{State} = \begin{bmatrix} \text{open}_{t-20:t} \\ \text{high}_{t-20:t} \\ \text{low}_{t-20:t} \\ \text{close}_{t-20:t} \\ \text{SMA}_{20} \\ \text{EMA}_{20} \\ \text{upper\_band} \\ \text{lower\_band} \\ \text{volume\_osc} \end{bmatrix} \tag{5}$$

Each feature is appropriately normalized to ensure model stability and to facilitate the learning process. The dataset is divided into training and testing segments based on a date offset, ensuring that the model is trained on historical data and tested on unseen future data.

The action space in this trading environment is discrete, consisting of three possible actions:

- Buy (action = 0)
- Hold (action = 1)
- Sell (action = 2)

The reward function is designed to encourage the agent to make profitable trades while penalizing losses and unnecessary holding costs. The agent buys stock at the current closing price. The reward is the difference between the maximum of the SMA and EMA values and the current price if the price is below these indicators (see Equation (6)).

$$\text{reward}_{\text{buy}} = \begin{cases} \max(\text{SMA}_{20}, \text{EMA}_{20}) - \text{price} & \text{if price} < (\text{EMA}_{20}, \text{SMA}_{20}) \\ 0 & \textit{otherwise} \end{cases} \tag{6}$$

The agent sells stock that was previously bought. The reward is the profit made from the sale after deducting the purchase price and transaction costs (as shown in Equation (7)).

$$\text{reward}_{\text{sell}} = \text{current\_price} - \text{last\_purchase\_price} - \text{transaction\_cost} \tag{7}$$

The holding cost is intentionally designed to be negative, ensuring that the agent is penalized for merely holding onto stocks without making trades. This discourages the agent from adopting a passive strategy that delays active decision-making (Equation (8)).

$$\text{reward}_{\text{hold}} = -\text{holding\_cost} \tag{8}$$

These rewards are structured to mimic real trading scenarios, where strategic buying and selling are key to profitability, and holding is sometimes necessary but less desirable due to opportunity costs.

5 Results

Our empirical study applied three distinct reinforcement learning algorithms— TD3, DQN, and A2C —to the realm of algorithmic trading, and the results were evaluated based on portfolio value compared to the initial investment of 1000 units and the Sharpe ratio, a measure of risk-adjusted return.

The TD3 algorithm demonstrated superior performance with a final portfolio value of 726.61 and a Sharpe ratio of 0.992 (see Table 1). The graph of TD3's trading activity (see Figure 3) illustrates a strategy capturing favorable entry and exit points, corresponding to dips and peaks in the stock price, respectively. It is observed that the agent initiates a buy signal in mostly conditions where the stock signal is either rising or dipped to a local

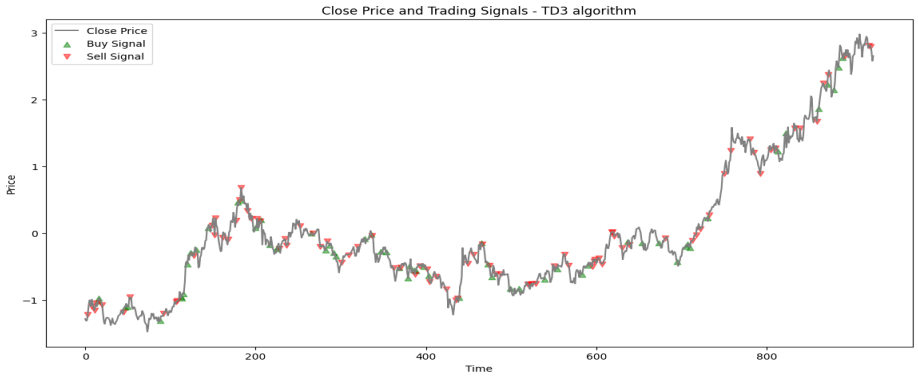

Figure 3: TD3 trading signals aligned with the closing price.

low. Similarly, the sell signal is initiated at peaks of the signal, suggesting a sophisticated pattern recognition capability.

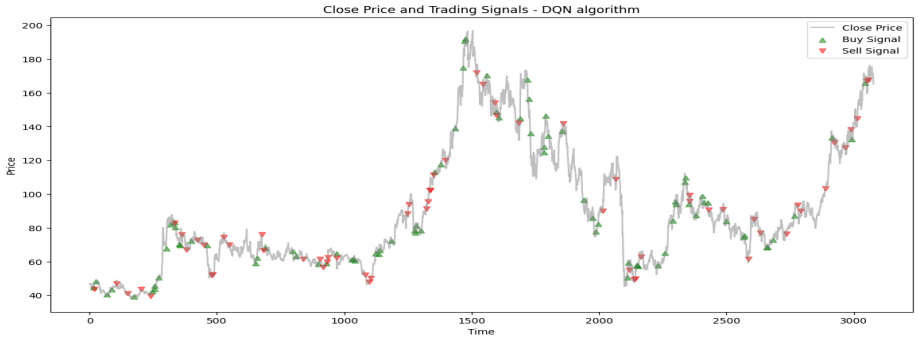

Figure 4: DQN trading signals aligned with the closing price.

In comparison, the DQN algorithm's strategy yielded a slightly lower portfolio value of 623.15 and a Sharpe ratio of 0.980. As illustrated in Figure 4, the algorithm adopted a trend-following approach for identifying trading moments. Drawing comparisons to TD3, the DQN agent, however, did not exhibit the same kind of pattern recognition. The marginally lower Sharpe ratio and portfolio value, as compared to TD3, indicate a less optimal risk-adjusted performance.

The A2C algorithm, adopting an actor-critic approach, resulted in a portfolio value of 126.79 and a Sharpe ratio of 0.6334 (see Table 1). The trading signals, depicted in Figure 5, were less frequent and more conservative. The agent exhibits some non-ideal decisions when initiating an action, as seen by some buy actions at peaks of the signal and some sell actions at the dips. This is further corroborated by the lower Sharpe ratio, highlighting a potential gap in the algorithm's strategic cautiousness.

To summarize, the TD3 algorithm exhibited the most proficient risk-adjusted return, with DQN following closely. In contrast, A2C lagged behind. These insights, quantified by the reported portfolio values and Sharpe ratios, illuminate the trading behavior of the algorithms.

Figure 5: A2C trading signals aligned with the closing price.

Algorithm	Portfolio Value	Sharpe Ratio
TD3	726.61	0.992
DQN	623.15	0.980
A2C	126.79	0.6334

Table 1: Portfolio values and Sharpe ratios for TD3, DQN, and A2C algorithms.

6 Future Work and Conclusion

Future enhancements of the models will focus on increasing training duration and episodes to amplify the algorithms' robustness, particularly in recognizing and adapting to long-term financial trends. This, along with the diversification of stock types within training datasets, is aimed at improving the generalizability of the algorithms Incorporating a wider array of financial indicators into the state data will provide a richer context for decision-making, enhancing the potential for informed and profitable trading strategies.

The study's analysis of TD3, DQN, and A2C algorithms in algorithmic trading demonstrated TD3's superior performance in portfolio value and risk-adjusted returns, with DQN following closely and A2C showing areas for potential refinement, as detailed in the results section. In summary, these findings contribute to the knowledge that underscores the potential of reinforcement learning to enhance and potentially transform trading strategies. The study's insights suggest that reinforcement learning models, when provided with extensive market data and subjected to a variety of market scenarios, are capable of learning and effectively navigating the intricacies of the stock market.

References

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. page pages, 2018a.

Scott Fujimoto et al. Using deep reinforcement learning for financial trading as a game. *Journal of Computational Finance*, 22(4):100–117, 2018b.

Chien-Yi Huang. Financial trading as a game: A deep reinforcement learning approach. *Journal of Financial Markets*, 21:1–25, 2018.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. page pages, 2016.

E. S. Ponomareva et al. Using reinforcement learning in algorithmic trading problems. *Journal of Trading Algorithms*, 15(3):45–65, 2019.

Tidor-Vlad Pricope. Deep reinforcement learning in quantitative algorithmic trading: A review. *Quantitative Finance*, 22(8):1305–1320, 2021.