# Design a Reinforcement Learning Agent for Making Stock Trading Decisions

DR. ARUN PANDIAN
School of Computer Science and Engineering
*Vellore Institute of technology*
Vellore, India
arunpandian.j@vit.ac.in

YUKENDHIRAN
School of Computer Science and Engineering
*Vellore Institute of technology*
Vellore, India
yukendhiranelango@gmail.com

***ABSTRACT*—This study delves into automating stock trading decisions to enhance efficiency and profitability. Automated trading's appeal lies in its quick decision-making, reduced emotional bias, and rapid data processing, capturing overlooked market opportunities. The proposed approach utilizes a reinforcement learning (RL) agent, chosen for its continuous learning and adaptability in dynamic markets. Unlike supervised learning relying on labeled data, RL learns through trial and error, dynamically adjusting to market changes. In comparison to unsupervised learning, RL actively explores and exploits market patterns, providing a more adaptive and proactive strategy. Implementing RL empowers the agent to autonomously navigate and capitalize on market dynamics, optimizing decision strategies over time. This research introduces an innovative and adaptive model, a significant contribution to stock trading decision-making for both companies and individual investors, promising more informed and potentially profitable decisions in the ever-evolving stock market.**

***Keywords*—Reinforcement Learning, DDQN, Stock Trading, Machine Learning.***

## I. INTRODUCTION

The stock trading landscape has historically been defined by manual decision-making processes, susceptible to emotional biases and constrained by the limitations of human data processing speed. These challenges hinder efficiency and often lead to missed opportunities within the dynamic and rapidly evolving markets. Despite the availability of automated trading systems, there exists a notable gap in leveraging advanced technologies, particularly reinforcement learning (RL), to enhance decision-making processes in stock trading. This research addresses this gap by proposing the development and implementation of an innovative RL-based model tailored specifically for stock trading decisions. By utilizing standard, high-quality data sourced from Fortune 500 companies and integrating advanced algorithms such as the Double Deep Q-Network (DDQN) within the reinforcement learning framework, this study aims to revolutionize traditional stock trading practices. The choice of Python programming language and the chainer framework for implementation is driven by their established effectiveness in refining decision strategies over time, ensuring robustness and reliability in real-world trading scenarios. Through rigorous evaluation and comparison with existing systems, this study seeks to provide valuable insights into the efficacy of the proposed RL-based approach in navigating the dynamic nature of stock markets. Automating stock trading decisions and reducing reliance on manual intervention are crucial objectives of this research. By doing so, the aim is to enhance efficiency, mitigate emotional biases, and capitalize on overlooked market opportunities. Ultimately, the goal is to foster more reliable and profitable stock trading decisions for both companies and individual investors in their pursuit of success in the ever-evolving stock market landscape

## II. LITERATURE SURVEY

[1] A Fresh Approach This research suggests employing a scaling convolutional neural network, based on SARSA to automatically identify features from weekly and daily financial data to enhance the stock trading process.

[2] This study introduces an optimal execution agent that employs order splitting to impact market dynamics while acquiring the most effective trading strategies within a simulated reactive market environment.

[3] By utilizing deep reinforcement learning techniques like policy gradient methods, deep Q learning, and deep SARSA this study presents a framework, for simulating stock trading and training an agent to automate trading operations.

[4] This research demonstrates the importance of using reinforcement learning (DRL) to make decisions in trading by developing a profitable trading model. The study highlights how DRL surpasses the limitations of learning methods.

[5] The study applies DRL techniques, like Advantage Actor Critic (A2C) and Deep Deterministic Policy Gradient (DDPG) to enhance a trading model focused on stocks.

[6] In order to showcase the efficiency of this automated stock trading approach Wang and colleagues trained DRL agents. Combined them into an ensemble using three actor critic algorithms; A2C, PPO and DDPG.

[7] When the agent consistently makes choices the Double Q Learning Network proves to be more effective than both the Dueling Double Q Learning Network and the Deep Q Learning Network and, in terms of performance.

[8] To assess how well five reinforcement learning (RL) algorithms perform in managing stock portfolios this research employs a three vector to depict the movement of individual stocks and other market attributes.

[9] By integrating trading methods, like time portfolio protection (TIPP) and constant proportion portfolio insurance (CPPI), into multi agent deep deterministic policy gradient (MADDPG) this study introduces two MARL algorithms for making strategic trades in quantitative markets.

## III. METHODOLOGY

### A. Research Objective:

The primary objective of this research is to develop and implement an innovative reinforcement learning (RL) based model for automating stock trading decisions. This model aims to improve efficiency, reduce emotional bias, and capitalize on overlooked market opportunities compared to traditional methods.

### B. Research Design:

This research will employ a computational approach using reinforcement learning techniques. Here's a breakdown of the design:

- Agent: An RL agent will be designed to interact with a simulated stock trading environment.
- Environment: A simulated trading environment will be created using historical stock data. This environment will provide the agent with rewards based on its trading decisions.
- Learning Algorithm: A DQN, DDQN and DDDQN algorithms will be used to train the agent. This algorithm allows the agent to learn through trial and error by interacting with the environment and receiving rewards for profitable decisions.
- Evaluation: The performance of the RL agent will be compared to existing trading data by splitting data into train and test data.

### C. Data Collection:

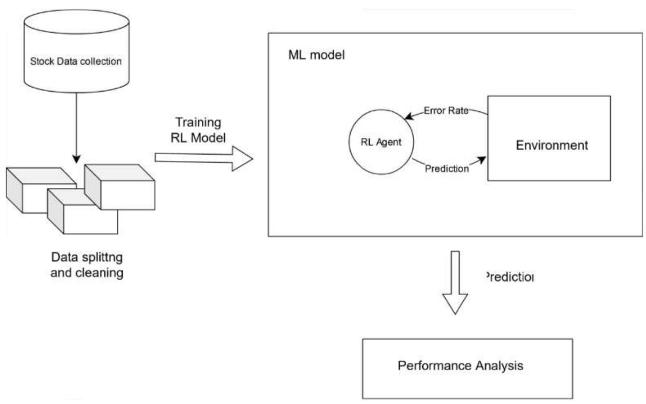
Historical Stock Data: High-quality historical stock data will be collected from reliable sources like Fortune 500 companies. This data will include opening, closing, high, and low prices, volumes, and other relevant financial indicators.

### D. Data Analysis:

The collected data will be preprocessed to ensure its quality and consistency. This might involve handling missing values, normalization, and feature engineering. Statistical analysis might be performed on the data to understand market trends and identify potential trading signals.

### E. Aechitectural Design:

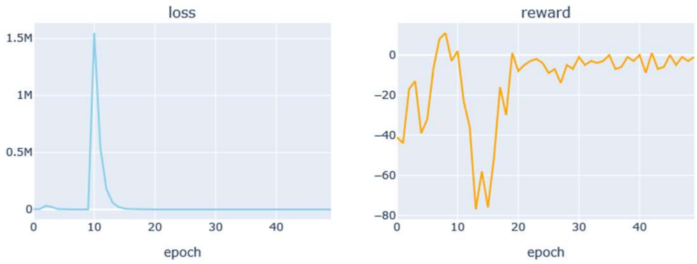
**Architecture Diagram:**




## IV. Results and Discussion

This research proposes a system to predict short-term stock prices by looking at historical data. The system uses past daily prices and trading volumes of all US stocks and ETFs for its analysis. To test the system, I trained a special learning models on data from July 2014 to January 2016. Then, tested the model's accuracy using data from January 2016 to July 2017.


DQN (Deep Q-Network) trains AI agents by letting them learn the best actions to take in an environment through trial and error, using deep learning to handle complex situations. Our DQN model trained on historical data (July 2014 - Jan 2016) showed promising results in predicting short-term stock prices. During both training and testing (Jan 2016 - Jul 2017), the model's predictions closely followed actual prices. This success is reflected in the loss function steadily decreasing during training, signifying effective learning. Additionally, the reward function stabilized after training, suggesting the model's ability to make optimal decisions within the simulated environment. These results demonstrate the DQN's potential for short-term stock price prediction




.
Fig-1:loss and reward graph

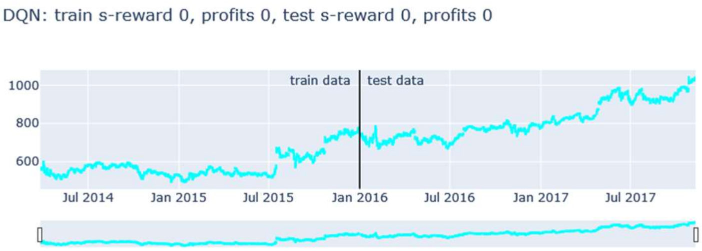



Fig-2:DQN  graph

Double DQN is an improvement on DQN (Deep Q-Network) that addresses a specific issue. Regular DQN can overestimate the value of actions, leading to inefficient learning. Double DQN uses two networks to tackle this, resulting in more accurate Q-value estimates. DDQN is better than DQN at spotting the general direction stock prices are moving. Even though it learns with some bumps along the way (loss function in Figure 4), it quickly figures out how to get the best results (reward function). This suggests DDQN is good at making accurate predictions about stock price trends, even if its learning process isn't always smooth. Overall, DDQN seems like a promising approach for predicting stock price movement

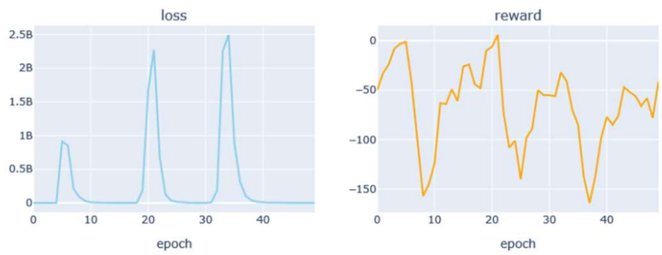



Fig-3:loss and reward graph of DDQN

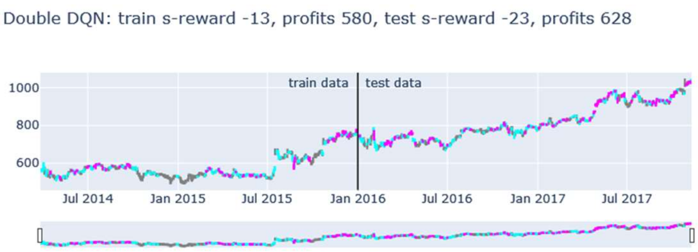



Fig-4:DDQN  graph

Dueling Double DQN combines the strengths of both Double DQN and Dueling DQN. Double DQN: Fixes overestimation issues in regular DQN by using separate networks for action selection and evaluation. It Improves efficiency by learning the state's value and the advantage of each action separately within the same network. This combination allows for more accurate Q-value learning and potentially better performance in complex environments.

Dueling Double DQN Performance: Building upon the success of DQN, the Dueling DQN experiment focused on maximizing profits through stock price prediction. Similar to DQN, its performance is evaluated using train and test s-rewards, likely representing average profits during training and testing. it suggests the model prioritizes achieving optimal rewards (accurate predictions even with some fluctuations in the learning process, potentially leading to improved overall profitability compared to DQN. This approach warrants further investigation to solidify its effectiveness in stock price prediction

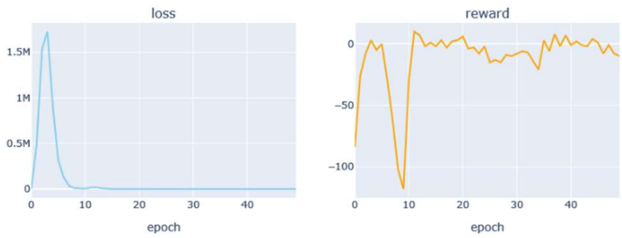



Fig-5:Dueling DDQN  loss and reward graph


Fig-6: Dueling DDQN   graph

## V. CONCLUSION

In conclusion, this reinforcement learning (RL) study investigated the potential for automating stock trades. The findings demonstrate promising results, with Deep Q-Network (DQN) excelling in short-term predictions, while Double DQN (DDQN) and Dueling DQN (DDDQN) exhibited strengths in capturing long-term trends and maximizing profits, respectively. All three models displayed effective learning capabilities, evidenced by decreasing loss functions and stable reward generation within the simulated environment.

It is important to acknowledge that real-world application might necessitate the incorporation of high-frequency data to achieve more comprehensive training. Future research should involve testing these models with live market data to assess their generalizability. Additionally, exploring advanced RL methodologies holds promise for the development of a robust trading strategy.Overall, this research contributes to a growing body of knowledge regarding the potential of RL to overcome inherent biases and capitalize on market opportunities. The findings offer valuable insights for enhancing automated stock trading decision-making in the dynamic and complex financial markets.

## REFERENCES

[1] Yuling, Huang., Yunlin, Song., Zongren, Chen. (2023). A Multi-Scaling Reinforcement Learning Trading System Based on Multi-Scaling Convolutional Neural Networks. Mathematics,
11(11):2467-2467. doi: 10.3390/math11112467

[2] Matthew, Dicks., Tim, Gebbie. (2022). A simple learning agent interacting with an agent-
based market model. arXiv.org, abs/2208.10434 doi: 10.48550/arXiv.2208.10434

[3] Yunfei, Luo., Zhan-Sheng, Duan. (2023). Agent Performing Autonomous Stock Trading
under Good and Bad Situations. arXiv.org, abs/2306.03985 doi: 10.48550/arXiv.2306.03985

[4] Taylan, Kabbani., Ekrem, Duman. (2022). Deep Reinforcement Learning Approach for Trading Automation in the Stock Market. IEEE Access, 10:93564-93574. doi:10.1109/ACCESS.2022.3203697

[5] Neda, Yousefi. (2022). Deep Reinforcement Learning for Tehran Stock Trading. 1(02):3742.doi:10.56741/jnest.v1i02.171

[6] Minseok, Kong., Jungmin, So. (2023). Empirical Analysis of Automated Stock Trading Using Deep Reinforcement Learning. Applied Sciences, 13(1):633-633. doi: 10.3390/app13010633

[7] Shymala, Gowri, S., Hema, Priya, N., Rathika, Pd. (2022). Stock market Prediction using Reinforcement Learning Technique. Ymer, 21(07):1022-1036. doi: 10.37896/ymer21.07/83

[8] Zheng, Hao., Haowei, Zhang., Yipu, Zhang. (2023). Stock Portfolio Management by Using Fuzzy Ensemble Deep Reinforcement Learning Algorithm. Journal of risk and financial management, 16(3):201-201. doi: 10.3390/jrfm16030201

[9] Hengxi, Zhang., Zhendong, Shi., Yuanquan, Hu., Wenbo, Ding., Ercan, E., Kuruoglu., Xiao-Ping, Zhang. (2023). Strategic Trading in Quantitative Markets through Multi-Agent Reinforcement Learning. arXiv.org, abs/2303.11959 doi: 10.48550/arXiv.2303.1195