Chapter 2

LITERATURE REVIEW

2.1 S&P introduction

Standard & Poor’s Corporation introduced stock price indices that covering all kinds of industries in 1923. Within 3 years, they are launching the Composite Index which is comprising 90 stocks as initial. The composite index evolved into the Standard and Poor 500 (S&P 500) in 1957 to track the performance of the 500 listed companies in New York Stock Exchange (NYSE). Standard & Poor’s Global Ratings is the agency for U.S.-based credit rating that provides rating of investment credit, data of financial (historical data) and analysis research on various equities which including stocks, bonds, commodities and others.

The companies that selected as the S&P 500 is the companies that can be represent as the economy of the United States because of their compositions in various fields. In reality cases, might be some of the companies will be bankrupt or dismissed for some of reason. The composition and weight of the index are updated/adjusted continuously time-to-time to ensure that its representability of the economy remains. Since the S&P 500 inception, over 900 new companies have been incorporated into the index, while an equal number have been removed from it (Siegel et al., 2006).

The S&P 500 index has consistently outperformed for most active money managers and mutual funds over time as extensively recorded. This performance can be attributed to the newly added that have higher performance than the old, dying companies that were delisted from the S&P 500 index.

2.2 Traditional Analysis in trading

Traditional analysis in trading is the methods or methodologies that have be fully used/utilized until now (passed over 100 years) to predict the movement of market and make the correct decision based on the market information. There are the 2 main process for the traditional analysis in trading are fundamental analysis and technical analysis where fundamental analysis is to evaluates the healthiness of the financial and overall economic environment of assets by analysing economic, financial and qualitative factors that may affect the future price and technical analysis is to predict the trends of stock market price from the historical data (including volume and price) (Murphy, 1999). Traditional analysis methods still exist as foundation in the financial market.

Fundamental analysis is the analysis methods that to determine the background of the asset including companies or stock market. Fundamental analysis involving macroeconomic indicator study, earning of corporate, rate of interest and others data points that can access the intrinsic value of the asset (depends on the trader requirement or the pattern of study/view). Fundamental analysis is the cornerstone/basic of long-term investment strategy that widely used by investors in equity markets. The fundamental factors such as corporate earnings and sentimental of market or others factors that may affect stock prices, by fundamental analysis with modern tools can improve accuracy of prediction in the stock market (Inani et al., 2024). Macroeconomic indicators (GSP growth and inflation) are the main impact factors toward equity market; it was the main information of the fundamental analysis. By analysing and understanding the trends of macroeconomic to address the long-term viability of assets (Inani et al., 2024).

Technical analysis is the method of analysis the pattern of the price movement by using the price charts, volume analysis. The principle of technical analysis is the price reflects all the relevant information. The relevant information including sentimental of market, economic data and news. Traders able to predict the future price movement by analysis the price of market pattern repeat over time. The technical indicators such as moving averages and relative strength index (RSI). By technical indicators can determine the buy/sell signals effectively (Vanguelov, 2016). Technical analysis is the ability to provide real-time insights to allow the trader for react instantly. In the high-frequency of trading, the technical indicators are the main roles in algorithmic models where is combination of traditional analysis and machine learning (Liu and Florin et al., 2023).

Integration of fundamental and technical analysis approaches a balanced view of market. Fundamental analysis is to analysis the long-term viability of the asset and technical analysis is allowing the traders to determine the time of entries and exits by analysis the short-term price movement. Combination of fundamental and technical analysis leads to more informed investment decision. Fundamental analysis able to provide the insights into a value of company while technical analysis is to determine the correct timing for buy and sell points (Levi et al., 2021). The limitation of fundamental analysis unable to capture effectively toward the dynamics market and technical analysis is not familiar toward the long-term economic fundamentals of analysis. By combination of fundamental and technical analysis able to increase the accuracy of prediction but required calibration carefully to prevent overfitting of market signals.

2.3 DRL models in trading

The aims of the DRL are to let the agents to learn optimal action through interactions with an environment with the policy in the Markov Decision Process (MDP) that will maximum the expected cumulative reward over time (Ezgi, 2024). MDP represented by a tuple:

$$M = (S, A, P, r, \rho 0, \gamma)$$

Where:

S is states (all states space availability);

A is actions (all possible action that will be taken by agent);

P is transition function (probability of transitioning from one state to another after action making);

R is reward function (the immediate reward will receive after the action taken in particular state);

γ is Discount factor (factor of discounts future reward);

$\rho0$ is initial state distribution.

The goal of MDP is to find optimal policy that can bring the cumulative reward over time.

Deep Reinforcement Learning (DRL) has revolution of the trading algorithms by allowing/train an agent to learn optimum trading policy from the dynamic and complex financial market through trial and error. The reward mechanism was the marking system that based on the policy set to give the reward or penalty toward an agent's actions. There are the summary table of DRL model including their function, strength, limitation and finding for selecting the best 3 model to further research.

Table 2.1 Summary of DRL model in trading

No	Model Name	Function of model in trading	Strength of model	Limitation of model	Results of paper	Citation
1	Deep-Q-Network (DQN)	By performing the deep neutral networks to calculate the Q-values, discrete trading actions (Sell, Hold, Buy) based on the mapping status (observations of market)	- Able to handles the high dimensional state spaces. - Without the handcrafted features, able to learn directly from the raw data (price of stock data)	- Trends to overestimate Q-values that may affect the grade of policy quality. - Only suitable for the discrete action spaces.	DQN-based trading outperformed rule-based strategies with higher cumulative returns and Sharpe ratios on historical stock data (Otabek et al., 2024).	Otabek, S., & Choi, J. (2024). Multi-level deep Q-networks for bitcoin trading strategies. *Scientific Reports (Nature Publisher Group), 14*(1), 771. doi: https://doi.org/10.1038/s41598-024-51408-w
2	Proximal Policy Optimization (PPO)	By the surrogate clipped objectives to balance the exploration and exploitation. It will help in stable policy update for	- Efficient of sample and robust to hyperparameter choices. - Well handle of stochastics policy and helps in improvement of exploration.	- intensive computation - May achieve the local optima without the adequate exploration.	PPO agents adapt well toward the dynamics market and outperformed than DQN and A2C models in portfolio management (Sun., 2023)	Sun, Q. (2023). *Reinforcement learning algorithms for stock trading* (Order No. 31765482). Available from ProQuest Dissertations & Theses Global. (3186188497). Retrieved from https://vpn.utm.my/dissertations-theses/reinforcement-learning-algorithms-stock-trading/docview/3186188497/se-2

		discrete and continuous spaces.				
3	Soft Actor-Critic (SAC)	SAC is the off-policy actor-critic algorithm that can help in optimize the maximum entropy objective and will helps in exploration and robustness	- Robust toward hyperparameter variations - Have a strong exploration in continuous action spaces	- Complexity of computational - Tunning issues for financial data.	SAC agents overperforming than PPO and DDPG in the trading multiple assets with lower risk and high stability (Kong et al., 2023)	Kong, M., & So, J. (2023). Empirical analysis of automated stock trading using deep reinforcement learning. *Applied Sciences, 13*(1), 633. doi:https://doi.org/10.3390/app13010633

Based on the Table 2.1, there are 3 DRL model that can used in trading but still not practical yet. Those 3 DRL models illustrate the rich diversity of approaches in automated trading. DQN model is values-based models which is excel in discrete decision but limitation in scalability. Actor-critic methods which are PPO, A2C and SAC are balance policy and value learning, handling continuous actions with the better flexibility. Twin delayed methods address the stability and market complexity issues. Despite advances, there are some of the common limitations including instability of training, sensitivity toward 'Noisy' data and computational overhead.

The Advantage Actor-Critic (A2C) and Twin Delayed DDPG (TD3) are also included in the similar research paper, which main focus on Jointed policy (optimal actor) and value (critic) networks to reduce the variances of gradient and increase the speed of training convergence, and TD3 improved DDPG by using the double Q-learning clipped to reduce the overestimation bias and delayed of policy update for stability. But they will not be further elaborated in this research.

The strengths, limitations and functions for each of the models are main consideration of the model selection for further research. The models in automated stock market trading are selected as the models that will used in the research. The 3 DRL models are PPO, DQN and SAC.

Table 2.2 Summary of the three models

No	Models	Reason
1	PPO	- Balance training stability and sample efficiency. - Able to handle discrete and continuous action spaces
2	DQN	- Able to handles the high dimensional state spaces - Basic DRL model
3	SAC	- Efficient Exploration - Sample efficient

Based on Table 2.2, there are the 3 models that selected as the research model. PPO able to balance the training stability and sample efficiency which is adaptive toward ‘Noisy’ data and non-stationary nature of stock market data. PPO able to handle discrete and continuous action spaces which is versatile toward different trading decision such as portfolio adjustments, buy or sell signals. Clopped objective able to prevent the large destructive policy updates which is able to reduce the risk of performance collapse. DQN model created as the basic of the DRL model that able to handles the high dimensional state spaces for the discrete action spaces with high effectively. With the Actor-critic structure allows agents to learn the complex policy while evaluating for the next better updates. SAC able efficient exploration which is important to avoid the premature convergence in the dynamics/volatile financial market. With the off-policy training, sample efficient where allowing agents learn faster from the historical data.

2.3.1 Proximal Policy Optimization (PPO) model

Proximal Policy Optimization (PPO) is a reinforcement learning algorithm that created/designed to improve the policy gradient methods. The ability of PPO is to balance exploitation and exploration by using a novel objective function. Exploration is the strategy of testing by using the new dataset and exploitation is capitalizing on the existing strategy. By using clipped probability ratio as the part of the function of loss to avoid the large update extremely and ensure the stability of learning. The primary objective function for PPO is

$$L_{CLIP}(\theta)=E_t[\min(r_t(\theta)A_t, \text{clip}(r_t(\theta),1-\epsilon,1+\epsilon)\ A_t)]$$

Where:

$r_t(\theta)$ is the probability ratio of the new policy to the existing policy;

A_t is the advantage estimate at time step *t*.

ϵ is the range for the clipping hyperparameter.

By using the function, able to prevent the policy from making unproductive update and help an agent to learn efficiently and stable (Schulman et al., 2017). PPO applied in stock trading and the aims is optimize the strategy of trading by allowing the agent to learn from the historical data (financial data). In the trading, PPO able to optimizes the decision of entry, exit and hold by interacting with the real-time stock market environment. PPO also using the MDP where State (S), Action(A) and Reward (R) as the basic of fundamental. State represents the condition of stock market. Action is the action taken such as buy, sell and hold. Reward is the profit or loss for every action taken. PPO showing the better performance in cumulative return over time and high stability (Sun, 2023). In the dynamics stock market, PPO able to optimize the policy by alternating between sampling data and the objective function (Schulman et al., 2017).

Attached some of the finding from the journal paper in Alireza et al., 2024. The results performance of PPO in stock market over time as following:

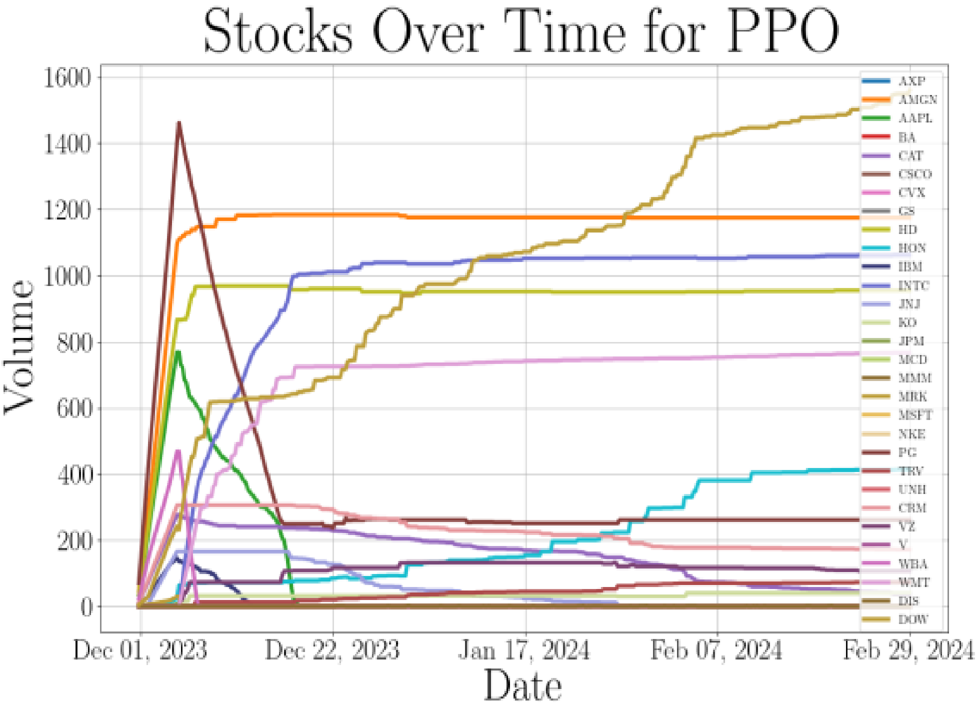


Figure 2.1 Stock Over Time for PPO (Alireza et al., 2024)

Based on the Figure 2.1, the graph showing the volume of trading of different stocks over time for PPO model in financial market. The stock behaviour in the graph showing that each of the companies represent each line for the volume of stock traded from 01 Dec 2023 until 29 February 2024. Volatility of the vary stock market such is the sharp rise in the volume during the early of Dec 2023. The significant changes will trigger the volume of trading activity. PPO model will be affected by the timing and the magnitude of the volume of trading. Larger volume of stock market will lead to PPO model more active and capitalizing responding toward signals. The PPO model performance in the varying level of volume was working fine and optimizing cumulative returns rates. In the summary of the Figure 2.1, PPO model able to work in dynamic stock market and adjust the trading strategy based on the real-time financial data. The increase of volume in trading as the model able to determine the opportunity for profit which means PPO able to adapt the dynamic dataset and finalize the aggressive investment strategy based on the data given.

2.3.2 Deep-Q-Network (DQN) model

Deep Q-Network (DQN) model is the combination of deep neutral network and Q-learning to solve the complex decision-making which is highly suitable for stock market dataset. The DQN model able to handle the large state and action spaces where the dataset of stock market is large. By using deep neural network to approximate the Q-value function, DQN able to allow agent to learn the optimal strategy of trading that can obtain the optimum of cumulative reward over time. Traditional analysis focus on the prediction only, DQN integrate the prediction of stock price and the optimization of trading strategy (Kabbani et al., 2022). The DQN formula as following:

$$Q\ (s_t,\ a_t) \leftarrow Q\ (s_t,\ a_t) + \alpha\ [r_{t+1} + \gamma\ \max_{a'} Q\ (s_{t+1},\ a') - Q\ (s_t,\ a_t)]$$

Where:

$Q\ (s_t,\ a_t)$ is the function of action-value;

r_{t+1} is the reward from the environment;

γ is the discount factor (linked with future reward); α is rate of learning;

$\max_{a'} Q\ (s_{t+1},\ a')$ is the maximum of the expected future reward at the state (s_{t+1}).

In the DQN model, allow to train the agents by minimize the temporal different error between the predicted and actual Q-values, to ensure the agent by learning the optimal policy to explore the stock market environment (Sun, 2023). Key challenges in DQN for trading including the balance of exploration and exploitation and reward function design. DQN model will overfitting toward the specific market if the exploration is not balanced properly (Ezgi, 2024). The exploration is exploring the new strategy and exploiting is the profitable part. If the exploration is not balanced with exploitation which means ratio of exploration is lower than exploitation, the decision making of the agents that train by DQN model will hard to get the proper or correct decision. The traditional reward functions are not sufficient enough to capture the complexness of the financial market. In the traditional reward function only using the profit as the reward and it will lead to the biases especially when the agent main focus on selling actions. By using the self-rewarding mechanisms able to address the traditional reward function issues by allowing the agent to adapt to changing market condition effectively (Huang et al., 2024).

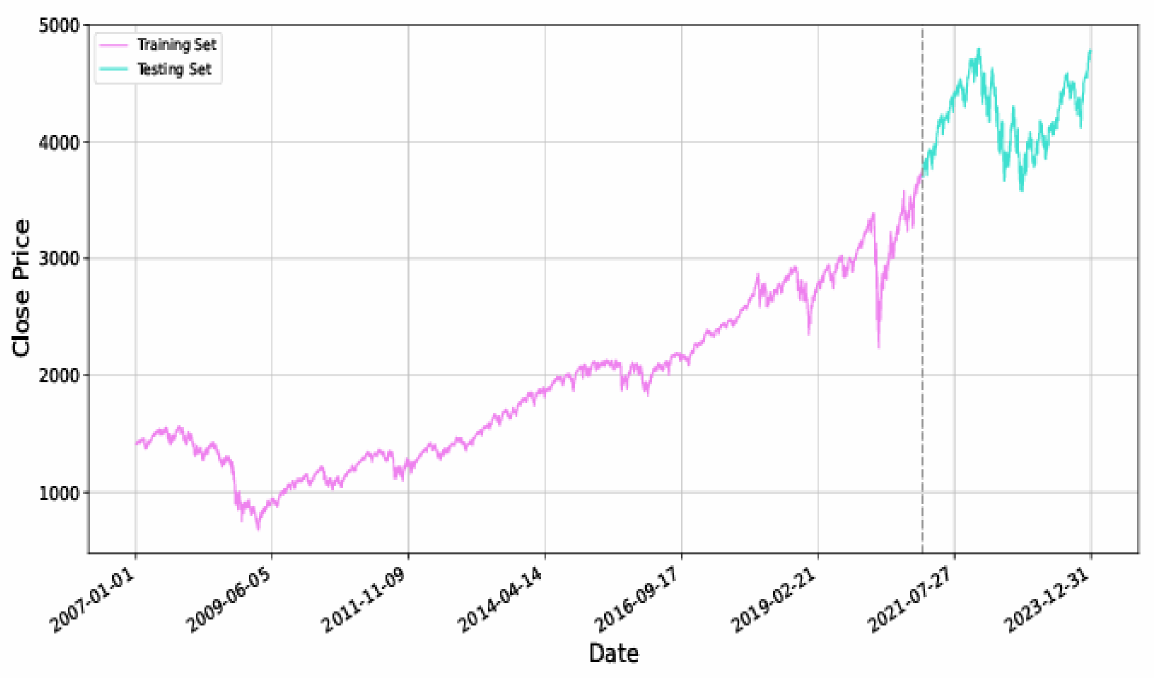


Figure 2.2 Close price over Data for DQN model (Huang et al., 2024)

The finding for the Figure 2.2 are the performance of training and testing, generalization, adaptability toward dynamics stock market and impact of model overfitting. In the training set of DQN where the agent will learn to make the decision based on the historical data given. The agents will learn the optimize actions (buy, hold and sell) based on the pass price trends. For the testing set is the agents to predict and adapt toward the future price movement. After 27 July 2021, the close price increased which is overperformed than previous pattern in training set. The ability of model to react toward the trend of stock data in testing periods will determine the success of the model. In the training set, DQN model able to well-generalize to unseen data in the testing set. If the model is overfitted toward the historical data will lose the ability of the model to adapt toward the unseen data.

In short, by enable the agent to learn optimal trading strategy autonomously DQN model able to handle high-dimensional state and action spaces which means make it more suitable for the dynamics financial stock market. However, the challenges such as trade-off of exploitation and exploration and reward function design are the items that can improve the model of DQN. Future improvement for the challenges of DQN are self-rewarding mechanisms and balancing the exploration and exploitation (Huang et al., 2024).

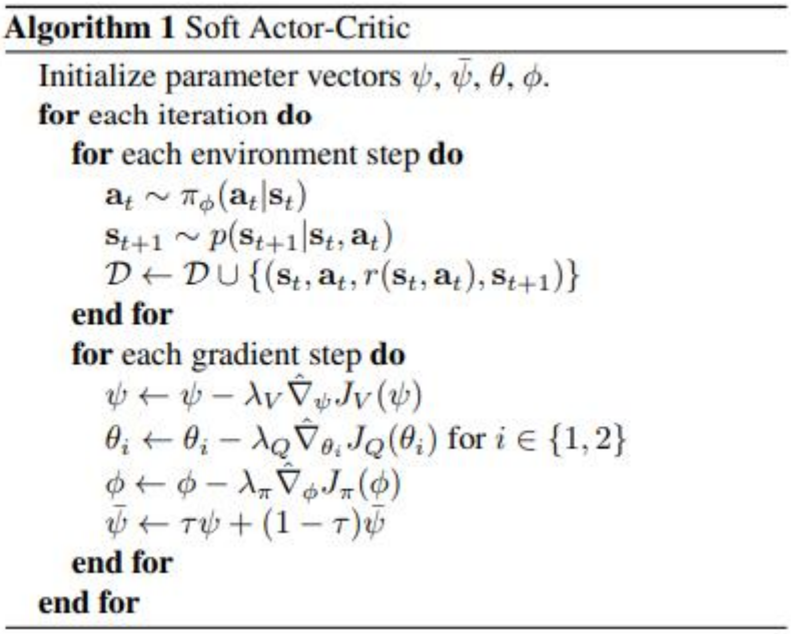
2.3.3 Soft Actor-Critic (SAC) model

SAC is an off-policy algorithm (Sarthak Singh et al., 2022) that applied stochastic policy optimization with Deep Deterministic Policy Gradient (DDPG). It uses feature of entropy regularization but optimizes the trade-off between return and entropy. Off-policy differs from the normal DRL for it aims to reuse the historical experience, instead of new sample needed for each gradient step, and so the policy learned required increasing number of them.

SAC is also a maximum entropy framework (Tuomas Haarnoja et al., 2018) that augments the maximum reward RL objective with a maximizing entropy. Thus, it gives a significant improvement for its robustness in the face of model and estimation error, and also strengthen its exploration by obtaining different behaviours. So, combine these two features, the off-policy maximum entropy actor critic algorithm is called soft-actor critic algorithm, which is famous on its sample-efficient learning and stability.

Actor-critic algorithm is formed by three components, which are an actor-critic structure, an off-policy formulation that allows the efficiently reuse of old data, and an entropy maximization. This include two alternating steps which are policy evaluation and policy improvement. The policy evaluation means calculating the value function for a policy, while policy improvement is about to get a better policy based on the value function obtained. So, the actor is the policy, and the critic is the value function.

There are many other on-policy algorithm also use entropy but instead of maximizing it, they only use it to regularize the model. The algorithm of SAC can be showed in a snippet of the pseudo code like below:



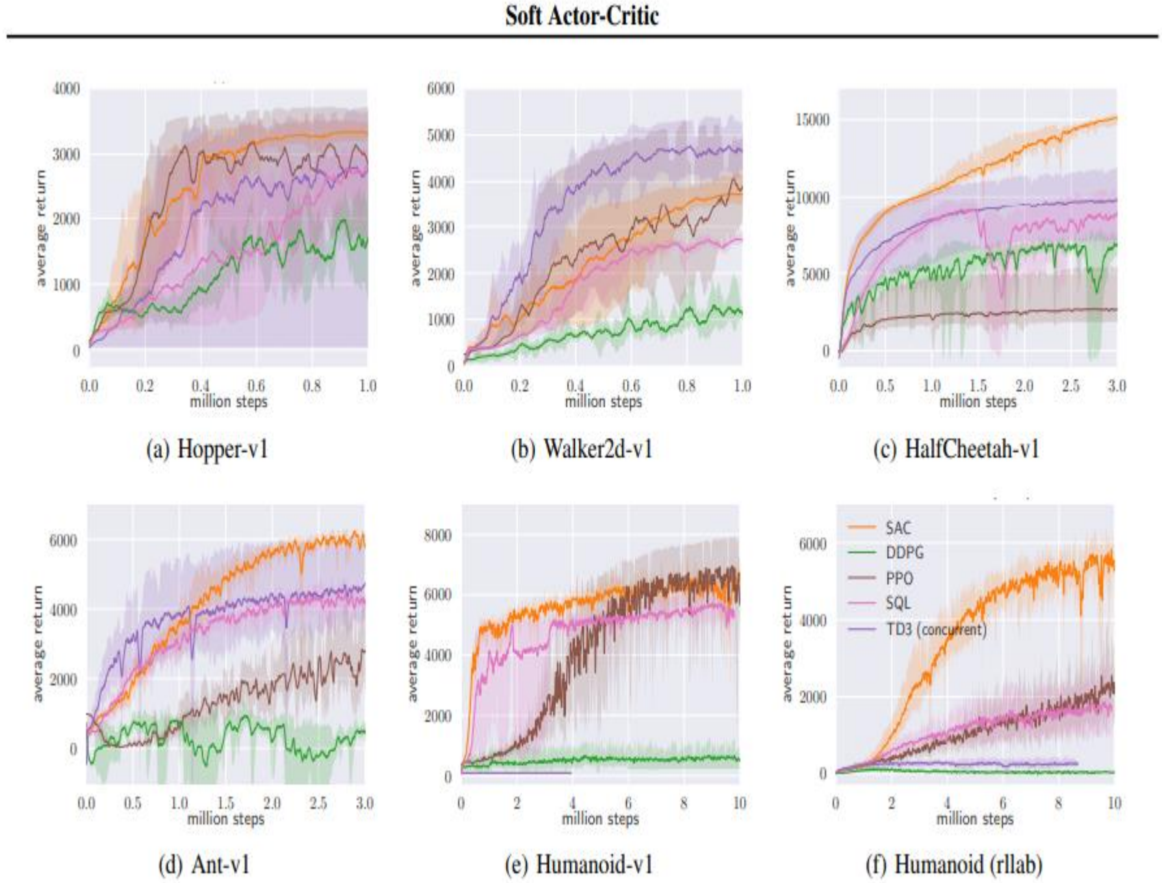
```
Algorithm 1 Soft Actor-Critic
  Initialize parameter vectors $\psi$, $\bar{\psi}$, $\theta$, $\phi$.
  for each iteration do
    for each environment step do
      $\mathbf{a}_t \sim \pi_\phi(\mathbf{a}_t|\mathbf{s}_t)$
      $\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$
      $\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{s}_t, \mathbf{a}_t, r(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1})\}$
    end for
    for each gradient step do
      $\psi \leftarrow \psi - \lambda_V \hat{\nabla}_\psi J_V(\psi)$
      $\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i)$ for $i \in \{1, 2\}$
      $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$
      $\bar{\psi} \leftarrow \tau\psi + (1 - \tau)\bar{\psi}$
    end for
  end for
```

Figure 1: Pseudo code of SAC algorithm

In this pseudo code, the **ψ** is the parameters of the value network Vψ(s), while ψ̄ is the parameters of the target value network Vψ̄(s), θ are parameters of the Q-networks Qθ1(s,a), Qθ2(s,a), and φ is parameters of the policy (actor) πϕ(a|s). The first line is to sample an action from the current policy, the second line is the step that get into the next state, while the third line means the storing of transition in the replay buffer.

The Q-networks is used to estimate the soft Q-value (s, a), the value network is used to estimate soft-state value V(s) for stability, while policy network is applied to learn stochastic policy that maximizes Q-values and entropy. The replay buffer is set to store experience for off-policy updates, and target values are smoothened by the target network to stabilize the Q-learning.

According to the paper written by Tuomas Haarnoja et al. in 2018, several benchmark environments like Hopper-v1, Walker2d-v1, HalfCheetah-v1 and others are used for both the training and testing of the SAC. The outcome of the results are shown as the Figure 2 below.



(a) Hopper-v1

(b) Walker2d-v1

(c) HalfCheetah-v1

(d) Ant-v1

(e) Humanoid-v1

(f) Humanoid (rllab)

Figure 2: Training curves on continuous control benchmarks (Tuomas Haarnoja et al., 2018)

We can see that in the comparison of different models, the SAC indicated as the yellow line did outperform other model in most of the cases for both of the learning speed and final performance.

Besides that, the paper also discussed how the stochastic and deterministic policy of the SAC will affect the training and the testing outcomes. Stochastic policy will output a Gaussian probability distribution over action, and the action is chosen based on the formula of hyperbolic tangent function, which the output is within the range of -1 to 1, and if the output is close to -1, a reverse action will be taken, when it close to 0, no action is being taken, and if it is close to 1, there will be a positive action just based on how we define it. While the deterministic policy will only output a single action for each state. The results of using stochastic policy and deterministic policy in SAC is shown as Figure 3 below.

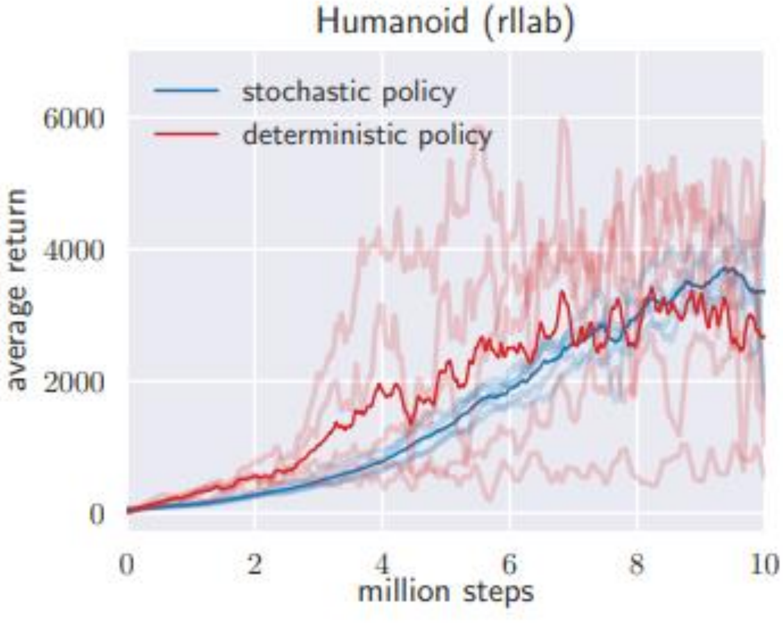


Figure 3: Comparison of stochastic policy and a deterministic variant (red) in the Humanoid (rllab) benchmark.

As what Figure 3 shown above, stochastic SAC samples action and keeps them by maximizing the entropy, while the deterministic SAC only choose the mean action and drop the entropy term, and thus the stochastic SAC yield more stable and efficient learning.

2.4 Reward Function in DRL

The reward mechanism in SAC model is the main process of agent's learning process. The function of reward defines the agent's action taken by evaluate and provide the feedback based on the decision making. For the trading, the reward is based on the profit and returns but SAC model introduces a more sophisticated mechanism by incorporating entropy maximization. The agent is not only maximizing the reward (profit) but encourage exploration by maximizing the entropy within the acceptable bounds. It helps agent from over-deterministic or exploiting a narrow set of strategy (Alireza et al., 2024). The maximum entropy in SAC reward function as following:

$$\text{Objective}=E_{\pi}\,[r\ (s_t,\ a_t)\ +\alpha\cdot H(\pi(\cdot|s_t))]$$

Where:

$r\ (s_t,\ a_t)$ is the immediate reward at the time step *t* for taking action a_t in the state s_t.

$H(\pi(\cdot|s_t))$ is the entropy of the policy, linked with exploration.

α is the parameter of temperature that control the trade-off for exploitation and exploration.

The reward function is to create to capture the trade-off between maximizing profits and controlling risks in the financial stock market. The traditional reward function in trading model relies on the profit and loss calculation. However, SAC model approach to overcome the issues of overfitting toward stock market by encouraging exploration of different trading strategies through the entropy term. The stock trading is dynamics where exploration may lead to discover new or more profitable strategy that might be overlooked (Alireza et al., 2024). In the SAC model, the reward function (Li, 2024) including:

1. Profit and Return – standard of the financial metric where the agent will receive the reward based on the return of the trading action taken.

2. Cost of transaction – The penalty for the trades for incur cost to ensure agent optimize for the net profits instead of gross profits.

3. Adjustment of risk – The additional penalty or reward based on the volatility that can encourage agent to consider the long-term stability alongside short-term return.

The benefits of the reward function for SAC model are improved exploration, stability and adaptability. By encourage the randomness, SAC prevent the situation of suboptimal local optima which is common phenomena in traditional reinforcement learning methods that only focus on the maximizing profits (Haarnoja et al., 2018). Balanced the trade-off exploration and exploitation by the reward structure helps SAC achieve more stable condition for learning with high noise data (financial stock market) (Alireza et al., 2024). SAC able to adapt dynamics stock market data and adjust the trading strategy accordingly which means make it more robust in dynamics environments (Kong et al., 2022). The reward function in SAC is importance for balancing the profitability with exploration. By using traditional reward maximization and entropy term, SAC model able to provide more adaptable and robust trading strategy especially in complex dynamic financial stock market.

The reward function of DQN model can be divided into profit-based reward, dynamic reward shaping, multi-objective reward function and risk and profit trade-off. Profit-based reward is the straightforward reward design in DQN where the reward is change in asset value after a trading action which is buying or selling. The difference between the price of asset at the time of agent take action and the price at a later time. The approach has notable drawbacks. For example, sell-biased reward structures obtain dominate due to the reward is only received when the trade in closed condition, will lead to agent selling excessively over other actions (Sun, 2023). Dynamic Reward Shaping is to address the limitation of the static reward function. The method is to adjust the reward function in real-time with some of the factors affected such as transaction cost, market liquidity and volatility. DQN models can better navigate in the complex trading environment by reducing the risk of the strategy that made by agent trapped in suboptimal condition (Huang et al., 2024). Risk and profit trade-off is the combination of risk management with reward function by integrating risk-adjusted returns. By using Sharpe ratio, DQN agent is to encourage not only maximize profit but also required consider minimize the risk associated with decision of trading. This will lead to risk-averse and stable of seek for the long-term returns (Huang et al., 2024). Due to dynamic of the financial stock market, the single objective like profit maximization insufficient. The integration of multi-objective reward function where balance the short-term profit and long-term stability and risk minimize. The Self-Rewarding Deep Reinforcement Learning (SRDRL) introduce a reward system that can adjust dynamically based on the predicted rewards and expert feedback, combination of human's knowledge with learning process of the agent.

The challenges and the solution in reward function design for DQN is overestimation bias and partial fulfilment problem. Overestimation bias due to the Q-values, it will lead to unstable of leaning especially when the model interacts with the dynamics financial stock market. Self-rewarding DQN (SRDQN) able to overcome the overestimation bias issues by separating the action selection process from the value estimation process, it will lead to more stable and reliable trading strategy (Huang et al., 2024). Due to market constraints, agent may take action (buy or sell) but only partial fulfilment. By using action retention mechanism to improve the ability of agent to handle real-time dynamics financial stock market (Sun, 2023). In short, the reward function of DQN is the importance role to guiding the agent to make the decision in the trading environment. The traditional reward function in DQN have limitation due to only focus on profit, SRDQN is the advanced techniques that can overcome the issues such as overestimation bias and partial order fulfilment in the complex, non-stationary stock market.

The main challenge in PPO is to ensure the reward function encourage desirable behaviours without lead to large and unstable update of policy, PPO operates with a surrogate objective that is optimized iteratively. PPO works to minimize the discrepancies in action taken while optimize the long-term profit which is the cumulative reward over time. The involvement of advantage function of reward function in PPO where the difference between the expected return and the average return. It allowing the PPO model to prioritize the action that led to the higher expected reward. The reward function of PPO helps reduce the variance of advantage estimates by value function which is estimates the expected reward over time (Schulman et al., 2017).

The optimization of PPO reward function by the bonus of entropy. This design is to encourage exploration by penalty deterministic policy where it will promote more diverse action sequences in training of agent process. By using the agent to explore in the different state-action pairs more thoroughly, the entropy bonus can prevent premature convergence to suboptimal solution (Schulman et al., 2017). The reward function of PPO is created/designed to adapt toward the different problem domains such as the high-dimensional continuous control tasks. The reward function designed to guide the agent to learning effectively without overwhelming with the complexity. In the algorithmic trading, the reward function is based on the profit and loss. This reward function is sensitive toward the timing of entry and exit actions where only for completing a transaction (selling stock) trigger the reward. The reward is scalable to reflect the profitability of the trade (Sun, 2023).

In the summary, the design of the reward function in PPO model is importance to remain the stable and performance of the process of learning for the agent. The balance of exploration and exploitation where exploration is to explore the new actions and exploitation is policy learning. By using the advantage function and entropy bonuses, PPO able to optimize the policy effectively over the dynamics financial stock market.

2.5 Motivation

Even the advancements of the deep reinforcement learning in algorithmic trading, there are the challenges that prevent trading strategy optimization. Due to the high-dimensional and non-stationarity of financial stock data, the traditional deep reinforcement learning techniques is hard to capture the complex market dynamics. Those models might be overfitting by the historical data and it will be limited to generalize to unseen market conditions. The balance of exploration and exploitation in trading strategy is importance because the high trade-off toward exploration may lead to loss of large financial and low trade-off toward exploration can lead to poor strategic performance (Huang et al., 2024).

To overcome the challenges, the research optimizes the trading strategy by introduce the self-reward mechanism into the deep reinforcement learning (DRL). By modify the reward mechanism based on the trading performance and market condition, may lead to increase the adaptability of the agent to shifting condition and reduce the chances of overfitting. The aims is to train agent with the high efficiency and stable in sentimental. The establish policy and reward mechanism is needed to train the agent that can fully eliminated the sentimental and the decision-making will be more discipline and the profit will be optimized.

REFERENCES

Jeremy J. Siegel & Jeremy D. Schwartz (2006). The Long-term Returns on the Original S&P 500 Firms, Financial Analysts Journal, vol.62(1), pp. 18-31, Taylor & Francis Ltd.

Murphy, J. J. (1999). Technical analysis of the financial markets: A comprehensive guide to trading methods and applications. Prentice Hall Press.

Inani, S. K., Pradhan, H., Kumar, S., & Biswas, B. (2024). Navigating the technical analysis in stock markets: Insights from bibliometric and topic modeling approaches. *Investment Management & Financial Innovations*, *21*(1), 275-288. doi:https://doi.org/10.21511/imfi.21(1).2024.21

Vanguelov, K. (2016). *Integration of technical trading behaviour in asset pricing* (Order No. 27821930). Available from ProQuest Dissertations & Theses Global. (2351278827). Retrieved from https://vpn.utm.my/dissertations-theses/integration-technical-trading-behaviour-asset/docview/2351278827/se-2

Florin, C. D., Turcaș, F., Ștefania, A. N., Bențe, C., & Boiță, M. (2023). The impact of sentiment indices on the stock Exchange—The connections between quantitative sentiment indicators, technical analysis, and stock market. *Mathematics, 11*(14), 3128. doi:https://doi.org/10.3390/math11143128

Levi, Sriyank & P, Prathima & Merlyn, Sarah. (2021). "FUNDAMENTAL AND TECHNICAL ANALYSIS LEADS TO A SYSTEMATIC INVESTMENT DECISION IN STOCK MARKET EQUITIES". 3. 39-42.

Korkmaz, E. (2024). *A survey analyzing generalization in deep reinforcement learning*. Ithaca: Retrieved from https://vpn.utm.my/working-papers/survey-analyzing-generalization-deep/docview/2910701950/se-2

Otabek, S., & Choi, J. (2024). Multi-level deep Q-networks for bitcoin trading strategies. *Scientific Reports (Nature Publisher Group), 14*(1), 771. doi: https://doi.org/10.1038/s41598-024-51408-w

Sun, Q. (2023). Reinforcement learning algorithms for stock trading (Order No. 31765482). Available from ProQuest Dissertations & Theses Global. (3186188497). Retrieved from https://vpn.utm.my/dissertations-theses/reinforcement-learning-algorithms-stock-trading/docview/3186188497/se-2

Goluža, S., Kovačević, T., Bauman, T., & Kostanjčar, Z. (2024). Deep reinforcement learning with positional context for intraday trading. Ithaca: doi: https://doi.org/10.1007/s12530-024-09593-6

Majidi, N., Shamsi, M., & Marvasti, F. (2022). Algorithmic trading using continuous action space deep reinforcement learning. Ithaca: Retrieved from https://vpn.utm.my/working-papers/algorithmic-trading-using-continuous-action-space/docview/2723274890/se-2

Kong, M., & So, J. (2023). Empirical analysis of automated stock trading using deep reinforcement learning. Applied Sciences, 13(1), 633. doi:https://doi.org/10.3390/app13010633

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal Policy Optimization Algorithms. *ArXiv, abs/1707.06347*.

Mohammadshafie, A., Mirzaeinia, A., Jumakhan, H., & Mirzaeinia, A. (2024). *Deep reinforcement learning strategies in finance: Insights into asset holding, trading behavior, and purchase diversity*. Ithaca: Retrieved from https://vpn.utm.my/working-papers/deep-reinforcement-learning-strategies-finance/docview/3081452987/se-2

Kabbani, T., & Duman, E. (2022). Deep reinforcement learning approach for trading automation in the stock market. Ithaca: doi:https://doi.org/10.1109/ACCESS.2022.3203697

Huang, Y., Zhou, C., Zhang, L., & Lu, X. (2024). A self-rewarding mechanism in deep reinforcement learning for trading strategy optimization. Mathematics, 12(24), 4020. doi:https://doi.org/10.3390/math12244020

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). *Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor*. Ithaca: Retrieved from https://vpn.utm.my/working-papers/soft-actor-critic-off-policy-maximum-entropy-deep/docview/2071194268/se-2