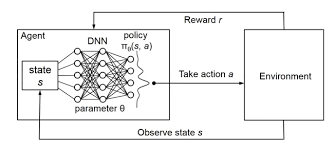
**Deep Reinforcement Learning for Stock Trading**

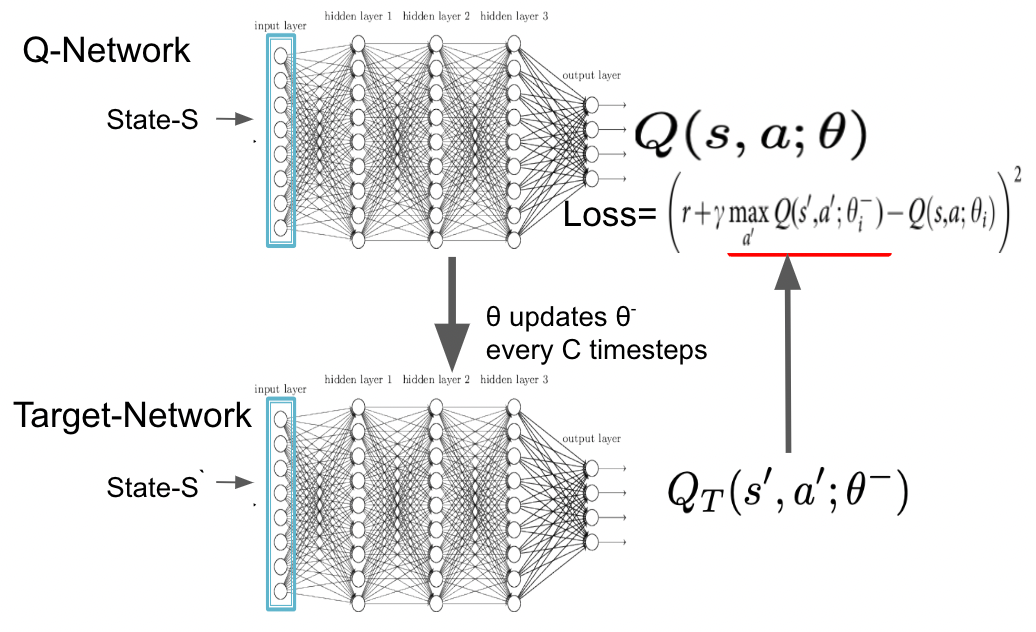
**I. Introduction**

Stock trading is a challenging problem that has attracted significant attention in recent years. With the rapid development of deep learning techniques, reinforcement learning has shown great potential in solving complex decision-making problems, including stock trading. This report presents a study on using deep reinforcement learning (DRL) to develop an effective trading strategy for stocks.

**II. A. Methodology**

The study uses the Deep Q-Network (DQN) algorithm, a classic DRL model, to build a trading agent. The agent learns to make decisions based on historical stock data and rewards. The DQN model consists of four fully connected linear layers, with the input being the state (stock prices) and the output being the Q-value of each action (buy, sell, or hold).





**B. Soft Update**

To stabilize the training process, the study uses a soft update strategy to update the target network. Soft update is a method that updates the target network parameters gradually, rather than directly copying the local network parameters. This approach helps to smooth the training process and improve the convergence of the algorithm.

**III. Experiments**

The study conducts experiments on a simple trading environment, where the agent learns to trade a single stock without considering transaction costs. The reward function is set to the profit earned from selling the stock. The results show that the agent learns a simple "buy low, sell high" strategy and earns a positive cumulative return.

**IV. Results**

The study presents the following results:

1. **Training Process**: The agent learns to make decisions based on historical stock data and rewards, and the cumulative return increases over time.
2. **Backtesting**: The agent's strategy is tested on the training data, and the results show that the agent earns a positive return.
3. **Testing**: The agent's strategy is tested on a separate testing dataset, and the results show that the agent earns a positive return, indicating that the strategy is generalizable.
4. **Complex Trading Environment**: The study also conducts experiments on a more complex trading environment, where the agent learns to trade multiple stocks with transaction costs. The results show that the agent learns a more complex strategy and earns a positive return.

A graph with red and blue dots

Description automatically generated

A blue line graph on a white background

Description automatically generated

A graph with red and blue dots

Description automatically generated

**V Conclusion**

This study demonstrates the effectiveness of deep reinforcement learning in developing an effective trading strategy for stocks. The use of soft update and DQN algorithm helps to stabilize the training process and improve the convergence of the algorithm. The results show that the agent learns a simple "buy low, sell high" strategy and earns a positive cumulative return in both simple and complex trading environments.

**Future Work**

Future work can focus on improving the trading strategy by incorporating more complex features, such as technical indicators, fundamental analysis, and risk management techniques. Additionally, the study can be extended to other financial markets, such as forex and futures.