Maximizing Returns with Reinforcement Learning: A Paradigm Shift in Stock Market Portfolio Management

Anirudh Bhakar[1], Priyanshu Senabaya Deori[2], Yash Vardhan Gautam[3] and Srinivasa KG [4]

***Abstract*—This report introduces an innovative stock market portfolio management approach, utilizing reinforcement learning and sentiment analysis techniques. Unlike traditional methods, which rely on time-consuming fundamental and technical analysis susceptible to human biases, our proposed approach leverages machine learning advancements such as DQN, DDQN, and Dueling DQN algorithms, combined with sentiment analysis to automate and enhance portfolio decisions. This study introduces a novel framework that combines reinforcement learning with multiple methods and sentiment analysis for stock market portfolio management.We have used Sensex NSE stock data comprising of historical data of open ,low , high , close prices from 2010 to 2023 . Experimental results demonstrate that our approach outperforms traditional portfolio management methods regarding returns and risk management.**

***Index Terms*—Portfolio management, stock market, reinforcement learning, sentiment analysis, DQN, DDQN, Dueling DQN**

I. Introduction

Traditional trading techniques rely on predetermined and subjective indicators to provide trade signals, such as moving averages, the relative strength index, and opening range breakout. However, most indicators, regardless of position size or risk management, can only offer long (buy) and short (sell) indications. Given the position and investment risk, portfolio management is a crucial and practical topic for investors. Among the different types of portfolio strategy, the hedging strategy known as equity market neutral (EMN) offers the strongest risk management capabilities. You could balance buying and selling relatively strong stocks with the aid of EMN. As a result, the investment risk can be significantly reduced, but it is difficult to quantify and categorise comparably strong and weak companies in practical situations.

To solve the previously mentioned investment constraints an adept economical portfolio management system (PMS) is constructed over the reinforcement learning (RL) architecture, that is used to help human decision-making in the real-world trading circumstances.[10]

Reinforcement learning is an effective machine learning technique that enables an agent to learn optimal actions by interacting with its surroundings and receiving incentives based on its performance. An agent can be trained to make investment decisions based on market data and sentiment research in the context of stock market portfolio management, to maximize returns while minimizing risks. Another machine learning technique that may extract significant information from social media and news sources is sentiment analysis, which allows portfolio managers to analyze market sentiment and make informed investment decisions.

The combination of reinforcement learning and sentiment analysis results in an effective toolkit for automated and intelligent stock market portfolio management.[6] Portfolio managers can use these strategies to make more informed investment decisions and better control risks while attaining higher returns. This study suggests and tests a unique approach to stock market portfolio management that incorporates these techniques. Our strategy paves the way for automated and intelligent portfolio management in a complex and dynamic market environment.

II. Literature review

A. Reinforcement Learning (RL)

One of the oldest and most prominent works in reinforcement learning is the Q-learning algorithm, proposed by Watkins and Dayan in 1992[15]. Q-learning includes estimating the worth of each action in a given state and updating these estimates based on the rewards obtained.

More recently, deep reinforcement learning (DRL) has developed as a strong technique for learning optimum policies from high-dimensional input fields. DRL employs deep neural networks to approximate the Q-values or policy functions, enabling agents to learn complex and non-linear policies.

The simple concept of RL is shown below:

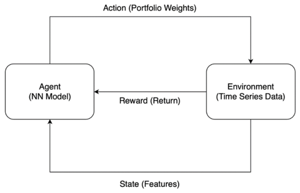


Fig. 1: A simple architecture of Reinforcement Learning

Moody and Saffell (2001)[8] applied RL to stock trading using a Q-learning algorithm, showing positive returns. Recent studies integrated deep reinforcement learning (DRL) into stock market prediction and trading.

Chang et al. (2018) developed a DRL-based portfolio management algorithm using a policy gradient algorithm for

[1] Anirudh Bhakar is with the Data Science and Artificial Intelligence Department, IIIT-Naya Raipur, Chhattisgarh 493661, India (e-mail: anirudh21102@iiitnr.edu.in).

[2] Priyanshu Senabaya Deori is with the Data Science and Artificial Intelligence Department, IIIT-Naya Raipur, Chhattisgarh 493661, India (e-mail: priyanshud21102@iiitnr.edu.in).

[3] Yash Vardhan Gautam is with the Data Science and Artificial Intelligence Department, IIIT-Naya Raipur, Chhattisgarh 493661, India (e-mail: yashg21102@iiitnr.edu.in).

[4] Srinivasa K. G. is with the Data Science and Artificial Intelligence Department, IIIT–Naya Raipur, Chhattisgarh 493661, India (e-mail: srinivasa@iiitnr.edu.in).

optimal asset allocation. The results showed that the DRL-based method surpassed traditional strategies in returns and risk management.[5]

B. Natural Language Processing (NLP)

Natural Language Processing (NLP) is a field of computer science and artificial intelligence that enables computers to understand and generate human language. It has been applied to sentiment analysis of financial news stories to guide stock market prediction and trading. Recent studies have integrated deep reinforcement learning (DRL) techniques into sentiment analysis and stock market prediction.

Li et al. (2020) proposed a DRL-based algorithm that incorporates sentiment analysis of financial news to predict stock prices. Their results showed that the sentiment-aware DRL algorithm outperformed traditional DRL algorithms in terms of accuracy and profitability.[11]

Min-Yuh (2016) suggested a Deep Learning based approach that tries to extract sentiment elements from financial news. These sentiments were then used to inform trading decisions. The sentiment-aware Deep Learning algorithm demonstrated better returns and risk management compared to standard trading methods.[2]

C. Reinforcement Learning along with Natural Language Processing (RL+NLP)

The integration of Reinforcement Learning (RL) into stock market prediction has shown promising results in providing accurate forecasts for various stocks. However, it is essential to acknowledge that the stock market dynamics are not solely governed by RL-based agents, as human sentiment plays a significant role in shaping market movements. Relying solely on sentiment analysis of news is also insufficient, as the market behavior is influenced by a multitude of factors.

In response to these challenges, N. Darapaneni (2022) proposed a sophisticated approach that combines Deep Learning techniques, specifically Long Short-Term Memory (LSTM) networks, with sentiment analysis of news data for predicting stock prices in the Indian market. This fusion of methodologies has exhibited superior performance compared to previous works.[1]

Nevertheless, the merging of Natural Language Processing (NLP) and Reinforcement Learning for stock market analysis does present certain obstacles. Issues such as the requirement for extensive data volumes and the potential risks of overfitting demand careful consideration. To overcome these challenges, future research endeavors to develop more efficient algorithms and explore alternative machine learning techniques, such as recurrent neural networks and attention mechanisms.

In conclusion, the convergence of NLP and Reinforcement Learning holds tremendous potential in enhancing stock market prediction and trading strategies.

III. Dataset

The Yahoo Finance[3] dataset for SENSEX NSE is a significant financial analysis and market research resource. The dataset includes information such as daily stock prices, trading volumes, market capitalization, and sector-wise makeup of the index. This data can examine market trends, conduct technical analyses, develop trading strategies, and perform risk assessments. Researchers, traders, and investors can leverage the Yahoo Finance dataset for SENSEX NSE to acquire insights into the Indian stock market and make informed judgments.

For the sake of our project, we have used the stock data of two companies:

- Reliance Industries Limited (RELIANCE.NS)
- Tata Motors Limited (TATAMOTORS.NS)

And for analysis, we have considered the following factors:

- *Open*: The opening price is when a stock or other financial instrument starts trading at the beginning of a trading session or a specific time period.
- *High*: The high price is the highest level a stock or financial instrument reaches during a specific trading period.
- *Low*: The low price is the lowest level a stock or financial instrument reaches during a specific trading period.
- *Close*: The closing price is the final price at which a stock or financial instrument trades at the end of a trading session or a specific time period.

We have ignored other features and restricted our analysis to the four features listed above. This strategy can be attributed to the training of our RL (Reinforcement Learning) agent, which is trained to decide whether to buy, hold, or sell shares on a daily basis. The opening price, closing price, maximum price (High), and minimum price (Low) for each trading day serve as the primary factors influencing the agent's decisions. Other indicators, like as trade volume and market capitalization, which are more indicative of long-term investment prospects than of immediate activities on a given day, are subordinate to these variables. Our RL agent can make intelligent and effective decisions in the quick-paced and dynamic world of daily trading activities by giving priority to these crucial aspects.

IV. Methodology

The methodology for the recommendation of whether to buy, sell or hold a particular stock was as follows:

A. Data Pre-processing

Data pre-processing is a crucial step in any data analysis project. In this part of the solution, we collected and compiled Indian stock market data from 2010 to 2022. The dataset included information about selected stocks and features, such as the opening price, closing price, high and low price, and trading volume. Also, the dataset had null values, which we removed as they were empty and useless for data imputation.

B. Custom Stock Trading Environment in GYM

We created a custom stock trading environment in GYM, including an action space, observation space, reward function, and other necessary functions like step and reset function.

- *Action Space:* The set of possible actions that our agent took in our environment was buying, holding, or selling stocks.
- *Observation Space:* Our observation space included the stockposition and position values (Current Close - Yesterday Close) for 85 days in a tuple. When the data for the 86$^{\text{th}}$ day was to be stored in the observation space, the data for day 1 was removed and replaced by the 86$^{\text{th}}$ day. This procedure continued for the rest of the data.

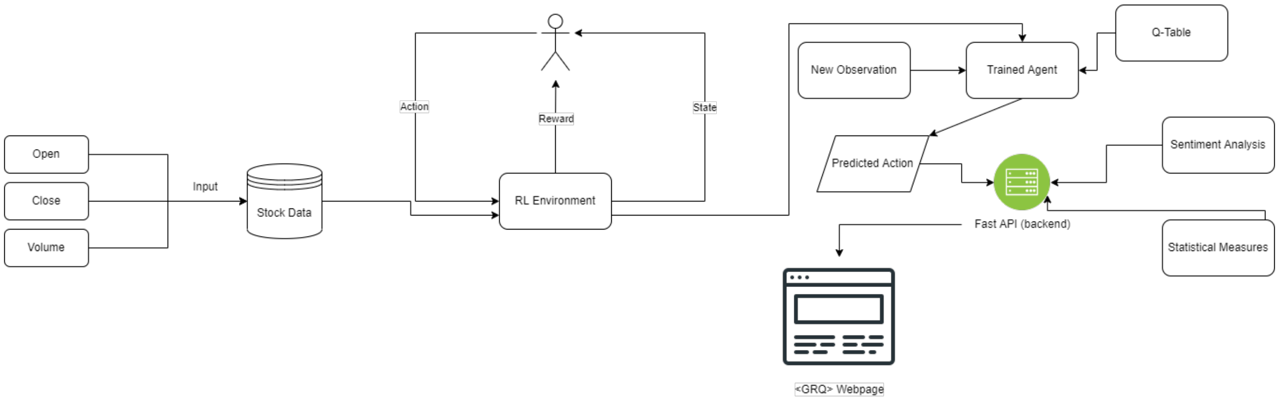


Fig. 2: Block Diagram

- *Reward Function:* The reward function developed calculated a reward value for the agentbehaviors. When the agent bought a stock, it added the closing price of the stock to its list of positions. When the agent sold a stock, it checked if it held any positions. If it did, it calculated the profit earned by comparing the selling price to the buying price and added it to the reward. If the agent tried to sell without holding any positions, it paid a penalty (reward of -1).
- *Step Function:* The step function developed determined the outcome of each interaction or ”step” the trading agent performed within the environment.
- *Reset Function:* The reset function developed was responsible for initializing and preparing the trading environment for a new trading session.

C. Deep Q-Network (DQN)

Deep Q-Network (DQN) is a reinforcement learning technique that combines the capability of deep neural networks with Q-learning. By leveraging a deep neural network as a function approximator, DQN learned to estimate the ideal action-value function in our environment, which allowed our agent to make intelligent decisions in difficult contexts.[7]

Two important features of the DQN algorithm were the use of a target network and the use of experience replay. The target network had the same parameters as the online network except it copied parameters after every τ step from the online network. Target used:

$$y_i = r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) \tag{1}$$

And in experience replay, we stored the observed transitions and updated them regularly. The *loss function* used here is:

$$L_i(\theta_i) = E_{s,a \sim \rho(\cdot)}[(y_i - Q(s, a; \theta_i))^2] \tag{2}$$

The full algorithm, which we call *Deep Q-Network*, is presented in Algorithm 1.

Algorithm 1 Deep Q-Network Architecture

```
Initialize Stock price replay memory $M$ to capacity $N$
Initialize the action-value function $Q$ with some random weights
for episode = 1, $M$ do
    Initialize sequence(current state) $s_1 = x_1$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
    for $t$ = 1, $T$ do
        With probability $\epsilon$ select a random action $a_t$ (Buy, Hold Or Sell) otherwise select $a_t = \max_a Q^*(\phi(s_1), a; \theta)$
        Execute action $a_t$ in custom trading environment and observe reward $r_t$ and price $x_{t+1}$
        Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
        Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $M$
        Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $M$
        if terminal $\phi_{j+1}$ then
            Set $y_j = r_j$
        else if non-terminal $\phi_{j+1}$ then
            Set $y_j = r_j + \gamma \max_{a'} Q^*(\phi(s_1), a'; \theta)$
        end if
        Perform a gradient descent step on $(y_j - \max_a Q(\phi_j, a; \theta))^2$ on equation 2
    end for
end for=0
```

D. Dual Deep Q-Network (DDQN)

Dual DQN (Double Deep Q-Network) is a reinforcement learning system that improved upon the original DQN by addressing its overestimation issue as the max operator used the same values to both select and evaluate action. In Dual DQN, two distinct neural networks were employed to estimate the action values: the primary network for selecting actions and the target network for evaluating the action values[12]. The target used by DDQN is:

$$y_i = r + \gamma Q(s', \arg\max_a Q(s', a; \theta_i); \theta_{i-1}) \tag{3}$$

By decoupling the action selection and action evaluation, Dual DQN eliminated over-optimistic value estimates and led to more steady and accurate learning. This strategy boosted the performance and convergence of the reinforcement learning agent, making it more successful in decision-making tasks.

E. Dueling Deep Q-Network (Dueling DQN)

Dueling Deep Q-Network (Dueling DQN) is an upgrade to the Deep Q-Network method that separated the estimation of the state value function from the advantage function. By decoupling these two components, Dueling DQN enabled the agent to learn the value of remaining in a certain state and the advantages of taking different actions independently. This separation allowed the agent to prioritize acts based on their prospective advantages, even when the values of several actions were identical.[14] By leveraging a shared feature representation for both the value and advantage streams, Dueling DQN efficiently learned and assessed state-action values, leading to enhanced performance and faster convergence in reinforcement learning tasks. The integration of Dueling DQN has proved its usefulness in several fields, including game-playing, robotics, and decision-making systems.

F. Natural Language Processing (NLP) Sentiment Analysis

We integrated sentiment analysis into our project to provide users with a comprehensive understanding of the current state of a stock. This feature went beyond simple buy, sell, or hold recommendations and aimed to offer insights into the overall sentiment surrounding the stock at any given time. By combining sentiment analysis with other data, we helped users gauge market mood and make more informed decisions about their assets. Our sentiment analysis component utilized news from *newsapi.org* and leveraged the power of *GPT-3* to provide users with a holistic perspective on the stockperformance.[9]

A basic example is given below:

News: *Mcap of 8 of top 10 valued firms soars Rs 1.26 lakh cr; Reliance, Hindustan Unilever shines*[4]

Results:

Sentiment Analysis: Positive

Score: 8/10

Stocks Affected: Reliance, Hindustan Unilever

Analysis:

The sentiment analysis of this news suggested a positive sentiment. The significant increase in the market capitalization of the top-valued firms indicated a positive market trend. The rise in market capitalization reflected investor confidence and indicated that these companies were performing well.

V. Results & Observations

A. For Reliance Stock

1) DQN results: We first trained our RL agent with DQN using the Reliance stock data and plotted the graphs of loss and reward.

Fig.3 shows the loss and reward the agent got after its training using DQN. As we can see that after 200 epochs the loss tends to approach to zero while the rewards keep on increasing.

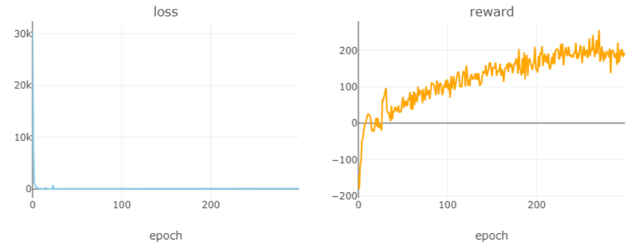


Fig. 3: Loss - Reward of DQN

2) DDQN results: Next we trained our RL agent with DDQN using the previous Reliance stock data and plotted the graphs of loss and reward.

Fig.4 shows the loss and reward the agent got after its training using DDQN. As we can see that after 200 epochs the loss tends to approach to zero but reward also tends to zero. So we can say that this model does not fit well for this stock.

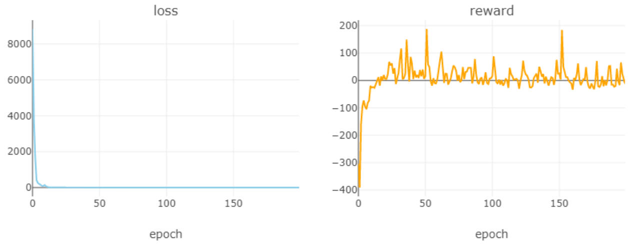


Fig. 4: Loss - Reward of DDQN

3) Dueling DQN results: Next, we trained our RL agent with Duelling DQN using the previous Reliance stock data and plotted the graphs of loss and reward.

Fig. 5 illustrates the loss and reward outcomes resulting from the training of the agent using Dueling DQN. It is evident that as the training progresses through 200 epochs the loss tends to approach zero but the reward also tends to zero. So we can say that this model also does not fit well for this stock.

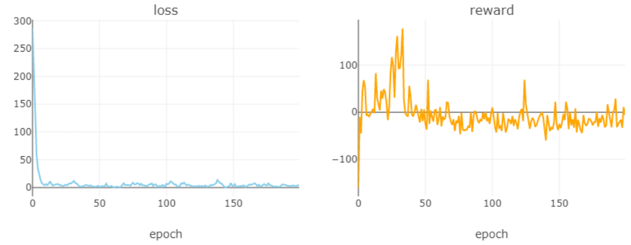


Fig. 5: Loss - Reward Dueling DQN

After comparing DQN, DDQN, and Dueling DQN, we concluded that DQN (Least loss and greater reward) gave the best result for recommending users to sell, buy, and hold Reliance stock shares.

B. For Tata Motors Stock

1) DQN results: We first trained our RL agent with DQN using the Tata Motors stock data and plotted the graphs of loss and reward.

Fig.3 shows the loss and reward the agent got after its training using DQN. As we can see that after 200 epochs the loss tends to approach to zero while the reward also aprroaches to zero. So we can say that this model does not fit well for this stock.

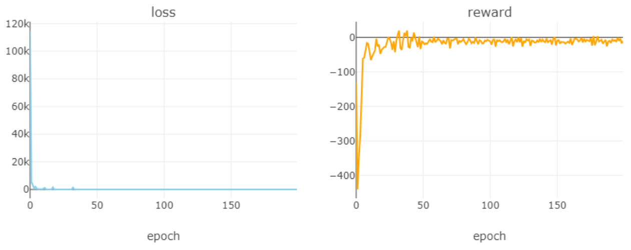


Fig. 6: Loss - Reward of DQN

2) DDQN results: Next we trained our RL agent with DDQN using the previous Tata Motors stock data and plotted the graphs of loss and reward.

Fig.4 shows the loss and reward the agent got after its training using DDQN. As we can see that after 200 epochs the loss tends to approach zero but the reward is on the positive side with significant magnitude.

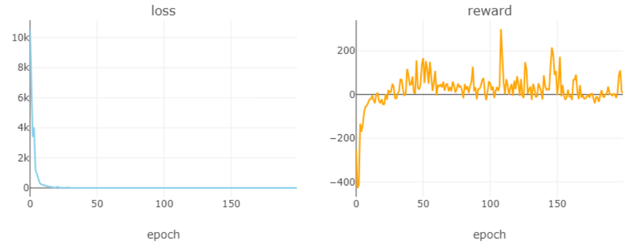


Fig. 7: Loss - Reward of DDQN

3) Dueling DQN results: Next, we trained our RL agent with Duelling DQN using the previous Tata Motors stock data and plotted the graphs of loss and reward.

Fig. 5 illustrates the loss and reward outcomes resulting from the training of the agent using Dueling DQN. It is evident that as the training progresses through 200 epochs the loss tends to approach to zero but reward is on the positive side but after certain epochs it is also approaching zero.

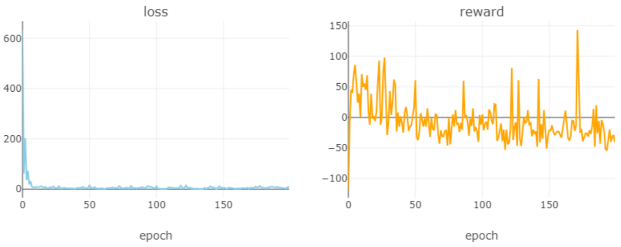
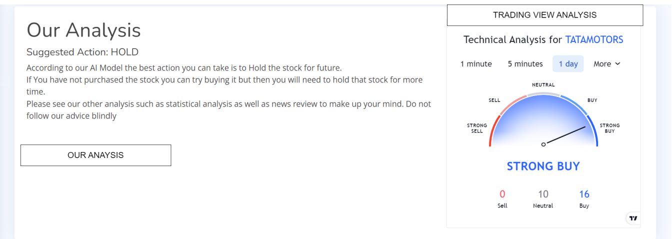


Fig. 8: Loss - Reward Dueling DQN

After comparing DQN, DDQN, and Dueling DQN, we concluded that DDQN (Least loss and greater reward) gave the best result for recommending users to sell, buy, and hold Tata Motors stock shares.

For comparison purposes, we have also embedded the TradingView Widget[13] that recommends buying, selling, or holding the stock so that the user can compare our recommendation and the results from an online free widget.



Our Analysis

Suggested Action: HOLD

According to our AI Model the best action you can take is to Hold the stock for future.
If You have not purchased the stock you can try buying it but then you will need to hold that stock for more time.
Please see our other analysis such as statistical analysis as well as news review to make up your mind. Do not follow our advice blindly

Fig. 9: Results page (COMPARISON)

VI. Limitations

Our paper introduces an innovative approach that combines reinforcement learning and sentiment analysis techniques for automated portfolio decisions. However, its limitations include a restricted capability to generalize to different stock markets, potential bias in data selection resulting from the use of a particular dataset, computational complexity, and difficulties in understanding reinforcement learning models. Concerns can arise from the limits of sentiment research, overfitting risk, shifting market dynamics, and the potential for market manipulation.

VII. Conclusion and Future Works

In conclusion, this study presents a novel stock market portfolio management method by harnessing the power of reinforcement learning and sentiment research. The standard methods of portfolio management, which mainly rely on time-consuming fundamental and technical analysis, can be enhanced and automated utilizing our proposed approach.

By integrating reinforcement learning algorithms such as DQN, DDQN, and Dueling DQN, together with sentiment analysis, our platform helps portfolio managers to make optimal investment decisions. Reinforcement learning helps the agent to learn from market interactions and maximize its reward function. At the same time, sentiment analysis gives valuable insights from social media and news sources to evaluate market sentiment.

Our technique displays improved performance through considerable experimentation compared to traditional portfolio management strategies regarding returns and risk control.

This symbolizes the effectiveness and promise of combining reinforcement learning with sentiment analysis for portfolio management.

In future works, the effectiveness of the model can be enhanced by increasing computational capacity and evaluation methods. Our model processes numerous hyperparameters that must be tuned, and due to the high computational requirements, we have only trained our agent on a limited number of stocks. In addition, we believe that using all of the content, as opposed to just passing along news headlines, may increase overall performance and experience.

VIII. Acknowledgment

We offer our heartfelt appreciation to Dr. Srinivasa K. G. for his guidance and support during this effort. His invaluable knowledge, insightful comments, and constructive input were critical in refining our grasp of the subject area and assisting us in overcoming the hurdles we encountered during the project. We are grateful to our institution faculty and staff for providing us with the resources and facilities to complete this project successfully. This project has been a fantastic learning experience for us, and we are happy for the chance to work on it.

References

[1] Narayana Darapaneni, Anwesh Reddy Paduri, Himank Sharma, Milind Manjrekar, Nutan Hindlekar, Pranali Bhagat, Usha Aiyer, and Yogesh Agarwal. Stock price prediction using sentiment analysis and deep learning for indian markets. *arXiv preprint arXiv:2204.05783*, 2022.

[2] Min-Yuh Day and Chia-Chou Lee. Deep learning for financial sentiment analysis on finance news providers. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1127–1134. IEEE, 2016.

[3] Yahoo Finance. Yahoo Finance - Stock Market Live, Quotes, Business Finance News. https://finance.yahoo.com/, 2023.

[4] Times Of India. News About Reliance and Hindustan Unilever. https://rb.gy/0zx5g, 2023.

[5] Caiyu Jiang and Jianhua Wang. A portfolio model with risk control policy based on deep reinforcement learning. *Mathematics*, 11(1), 2023.

[6] Taylan Kabbani and Ekrem Duman. Deep reinforcement learning approach for trading automation in the stock market. *IEEE Access*, 10:93564–93574, 2022.

[7] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[8] John Moody and Matthew Saffell. Learning to trade via direct reinforcement. *IEEE transactions on neural Networks*, 12(4):875–889, 2001.

[9] OpenAI. ChatGPT : Large-scale Language Model. https://openai.com/chatgpt, 2021.

[10] Jia-Hao Syu, Mu-En Wu, and Jan-Ming Ho. Portfolio management system with reinforcement learning. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 4146–4151. IEEE, 2020.

[11] Thibaut Théate and Damien Ernst. An application of deep reinforcement learning to algorithmic trading. *Expert Systems with Applications*, 173:114632, 01 2021.

[12] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[13] Trading View. Free Stock Widgets - Financial Web Components. https://in.tradingview.com/widget/, 2023.

[14] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pages 1995–2003. PMLR, 2016.

[15] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8:279–292, 1992.