How to Cite:
Mohan, A., Soni, T. K., Vamshisai, P., & Sateesh, B. (2022). A deep Q network framework on stock price prediction. *International Journal of Health Sciences*, *6*(S5), 3050–3061.
https://doi.org/10.53730/ijhs.v6nS5.9318

A deep Q network framework on stock price prediction

A. Mohan
Assistant Professor, Chaitanya Bharathi Institute Of Technology Hyderabad
Email: amohan_cse@cbit.ac.in

Tarun Kumar Soni
UG Student, Chaitanya Bharathi Institute Of Technology Hyderabad
kumartarunsoni@gmail.com

P. Vamshisai
UG Student, Chaitanya Bharathi Institute Of Technology Hyderabad
Vamshisaisai38@gmail.com

B. Sateesh
Assistant Professor, Chaitanya Bharathi Institute Of Technology Hyderabad
bsateesh_cse@cbit.ac.in

***Abstract*---**Stock price forecasting seeks to forecast future changes in a trading exchange's share price. More profit can be made by investors with the help of correct stock price prediction. Artificial Intelligence methods such as Deep reinforcement learning has become more important for sound decision-making across a range of areas. Deep reinforcement learning has been utilized in a range of fields, including education, healthcare, transportation, finance, video games, robotics, computer vision and natural language processing. This model has proven efficient in handling a variety of challenging decision-making tasks that were previously beyond of the machine's capabilities. As a consequence, it is a model that can be used in creating smart frameworks that assists us in predicting when to buy or sell a specific stock. With limited historic information, reinforcement learning performs quite well. Within this work, we present a Deep Q-Network architecture that makes use of Q-Learning, off-policy reinforcement learning algorithm to determine which action an agent should perform in response to an action-value function. Using an action-value function, Q learning determines the value of being in a certain state and taking a particular action at that state. The Deep Q-Network model is trained using historical stock data to anticipate the holding, purchasing, and selling strategies for companies. Performance is

Manuscript submitted: 18 Feb 2022, Manuscript revised: 27 April 2022, Accepted for publication: 9 June 2022

assessed and analyzed while the model is tested using omitted data from the later time period.

***Keywords*---**Deep reinforcement learning, Deep Q-Network, and Double Deep Q-Network

1. Introduction

The stock market in today's world is defined by massive gains and severe risks, as well as intricate linkages between the primary and secondary markets, resulting in a large number of stock investors and traders, including individuals of all ages. Stock prediction has therefore become a big source of concern for all stockholders and investors[1].

A lot of work has gone into developing strategies and algorithms for predicting stock values and making the best trading decisions. Numerous indicators, machine learning, and deep learning approaches, including neural networks, recurrent neural networks, and reinforcement learning have been developed to predict stock and financial price values and approaches. In comparison to traditional indications and approaches, modern artificial neural network techniques have demonstrated superior performance [2]. Stock price prediction is a difficult endeavour since the stock market fluctuates frequently and data is sometimes partial or insufficient. One way for solving such complicated choice issues is Q-learning which is a well-known strategy of reinforcement learning. Reinforcement learning is one of the methods to solve such complex decision problems. One way for resolving such difficult decision-making problems is reinforcement learning. For predicting stock prices and maximising anticipated return, reinforcement learning may show to be a more effective alternative strategy. High dimensional data can be processed using deep learning approaches to extract the features [3][4]. However, it is incapable of developing conclusions. Deep Reinforcement Learning combines the Deep Learning methodology with Reinforcement Learning's capability for making decisions [5].

Building a trading strategy and extracting features are the two most significant tasks in stock trading when employing machine learning approaches to obtain long-term benefits. Several ways for designing trading strategies using trade signals to optimize profits have been offered. In Indian markets, stock trading tactics and investment choices are subject to the notion of deep reinforcement learning. Deep Q-Network, Target Deep Q-Network, and Double Deep Q-Network are utilized in this project's primary studies. The models performance is evaluated, and a comparison is produced.

2. Related work

Deep Reinforcement learning combines the approaches of both Reinforcement learning and Deep learning. Deep learning refers to the use of neural networks fashioned after structure of the brain. Neural networks proved to be far superior in solving complex decision problems than traditional machine learning methods.

This section explains in detail the Reinforcement learning, Q-learning and the Deep Reinforcement Learning algorithms[6].

Reinforcement Learning

Reinforcement learning is a form of machine learning approach in which a smart agent (computer programmer) interacts with its surroundings and learns how to act in it. The agent interacts with the environment and explores the world on its own. In reinforcement learning, an agent's primary aim is to increase performance by obtaining the maximum positive rewards.[6].

Without any human interaction, the agent learns from its own experiences. It is necessary to understand the elements involved in reinforcement learning. The elements here play a significant role in determining the future actions taken by the agent. The elements are policy (set of rules that govern how an agent acts at any particular time), reward signal (an instantaneous signal which environment gives to the learning agent at each state), value function (which indicates how favourable the circumstance and action are), model of the environment (which predicts how the environment will behave when a given state and action are provided), and Bellman Equation (central element method for computing value functions in an environment)[7].

Q-Learning

Q-learning is a popular model-free reinforcement learning technique. The main goal of Q-learning is to develop a policy that can tell the agent what actions to do in order to maximize reward under certain conditions. It's an off-policy reinforcement learning algorithm that tries to figure out the optimal course of action in the present situation. It is model-free as it the model does not mimic the environment. Q-learning utilizes a data structure called Q-table which store Q-values which are state-action value pairs that define the quality of an action to be taken at a particular state[8].

Deep Reinforcement Learning Algorithms

Deep Q-Network, Target Deep Q-Network and Double Deep Q-Network are discussed in the following sections.

A) Deep Q-Network

An agent that utilises Deep Neural Networks to establish the relationships between states and actions, much as how Q-Learning utilizes a Q-Table. Deep Q-network is a reinforcement learning method that is model-free and can solve sequential decision-making tasks. Finding the best technique to maximise long-term benefit or profit is the learning objective[9]

In a nutshell, Q-Learning develops the state-action value function for a specified goal policy, which then selects the optimum action for a constrained state and action space. On the other hand, storing millions of entries in programme memory may be required for a large set of action space. This causes memory volume to expand, resulting in the curse of dimensionality or an unstable representation of a Q-Function. The correlations present in the series of observations cause the instability in Q-Learning. Small changes in the Q-value can cause a significant change in the agent's policy, as well as the correlation between the target and the Q-Value[10]

Iterative updates and experience replay are the two ways Deep Q-Network uses to get over these drawbacks. By gradually bringing the Q-values closer to the target values, iterative updates lower the correlation between the target and the Q-values. While experience replay appears to address the correlation problem by smoothing out the variations in data distribution, data randomization tends to do the same for the correlation problem[11]

The agent takes action and receives a reward r_t from the environment depending on the environment's current states_t. In order to learn from earlier events, the experience replay is utilised to record previous states, actions, rewards, and forthcoming states. The data from the replay memory is taken at random and given to the train network in tiny batches to prevent overfitting.

$$Q_{target} = r_{t+1} + \gamma \; max_{a'} \; [\mathbf{Q}(s'_t , a'_t ; \Theta)]$$

Where Qtarget is the target Q value calculated with the Bellman Equation, and signifies the Q-Network parameters.

B) Target Deep Q-Network

There are two Q-Networks in the Target DQN: the main Q-Network, also known as the policy Q-Network, and the target Q-Network. The target Q-Network differs from the primary Q-Network except that the Q-values of the target DQN are clones of the policy Q-Network Q-values and are updated on a regular basis. When the data incoming frequency is very high and the training data is strongly correlated, using a single Q-Network in the model results in delayed or sub-optimal convergence and may also cause the target function to become unstable. The Q-Network becomes more stable when two distinct Q-Networks are used.

C) Double Deep Q-Network

Double Deep Q-Network emphasizes more on choosing an optimal future decision. The target network determines the Q-value of performing that action at that state, while the policy network assists in choosing the action for the following state. To choose and evaluate the actions, the optimal Q-value or action-value pair is computed. As the number of iterations increases and the mistakes continue to accumulate, DQN performs the maximum of all actions that result in an overestimation of the Q-value. Using Double DQN, the overestimation of Q-value problem is resolved.

$$Q_{target} = r_{t+1} + \gamma \; \mathbf{Q}(s_t , argmax_{a'} Q(s'_t , a'_t ; \Theta) ; \Theta')$$

where,
γ is the discounting factor (gamma)

Θ is the parameters of policy network
Θ' is the parameters of target network
Q_{target} is the target Q value calculated with the Bellman Equation

3. Algorithm

Step - 1: Create a network with random weights and a replay memory capacity
Step - 2: Perform Steps 3-10 for every step
Step - 3: Set the starting state.
Step - 4: Perform Steps 5-10 for each time step
Step - 5: Select an action according to epsilon-greedy policy
Step - 6: In an emulator, perform the desired action, then observe the reward and the following state
Step - 7: Store experience in reply memory, and random batches are sampled from the replay memory and pre-processed
Step - 8: Utilizing the aforementioned batch of pre-processed states in the policy network, determine the loss between the output Q-values and the target Q-values
Step - 9: Calculate Q-value for the next state be sending the next state as the input to the network
Step -10: Minimize the loss through updation of weights using policy network

4. Implementation

Data Set Description

Date	Open	High	Low	Close*	Adj Close**	Volume
Dec 30, 2021	1,458.50	1,466.85	1,445.00	1,461.50	1,444.70	3,607,959
Dec 29, 2021	1,456.05	1,461.00	1,447.40	1,453.85	1,437.14	3,834,351
Dec 28, 2021	1,460.70	1,466.90	1,453.40	1,460.80	1,444.01	2,725,339
Dec 27, 2021	1,428.90	1,454.00	1,422.15	1,450.80	1,434.13	2,352,549
Dec 24, 2021	1,445.50	1,452.00	1,428.00	1,438.90	1,422.36	2,780,512
Dec 23, 2021	1,453.25	1,457.80	1,436.85	1,444.10	1,427.50	4,889,583
Dec 22, 2021	1,452.30	1,456.00	1,435.70	1,445.20	1,428.59	4,288,742
Dec 21, 2021	1,439.50	1,450.75	1,430.00	1,441.80	1,425.23	4,559,375
Dec 20, 2021	1,452.00	1,458.00	1,414.10	1,425.65	1,409.26	10,578,334
Dec 17, 2021	1,497.00	1,506.00	1,467.70	1,473.05	1,456.12	7,497,384
Dec 16, 2021	1,512.80	1,517.00	1,494.00	1,500.10	1,482.86	6,605,216
Dec 15, 2021	1,503.00	1,523.35	1,495.65	1,500.00	1,482.76	5,057,574
Dec 14, 2021	1,502.50	1,516.00	1,497.05	1,502.45	1,485.18	4,595,550
Dec 13, 2021	1,526.00	1,537.65	1,507.20	1,511.15	1,493.78	5,161,071
Dec 10, 2021	1,524.90	1,528.00	1,508.45	1,522.55	1,505.05	3,889,692

Fig 1: Dataset of Google Stock

The datasets are obtained from Yahoo Finance. It consists of Open, High, Low, Close, Adjusted Closing values along with the Volume. The Open column indicates the price at which a stock began trading on a certain day when the market opened. The price of an individual stock when the stock exchange closed the market for the day is referred to in the Close column. The highest price at which a stock traded over a certain period is shown in the High column. The Low column indicates the period's lowest price. The overall quantity of trading activity during a certain time period is known as volume. The adjusted closing price adjusts a stock's closing price to reflect its worth after any corporate actions have been taken into account. We consider three datasets. The first dataset, identified as the training dataset, comprises of data collected over a minimum period of seven years. The data recorded for the immediately preceding year make up the evaluation dataset. The testing dataset contains substantially more recent data, such as the most recent stock price.

To forecast when to purchase or sell a certain stock on four stocks of well-known companies, Deep q-network, Target q-network, and Double Deep q-network are employed. The stocks' total profit is computed using training and testing data. The difference between the actual and predicted q-values is measured using the Huber loss and Adam is an optimization algorithm used. Although, experience replay helps us to learn more from individual tuples several times, recall unusual events, and make greater use of our experience in general, in addition to breaking detrimental correlations. Target networks prove to be more effective as the q-values do not seem to approach the required target q-values. Overestimating q values is a common issue for target networks. Double deep-q learning reduces overestimation by dividing the target's max operation into action selection and action assessment[12].

Table 1: Total profits for each DRL algorithm used on stocks of top companies

Stocks	TOTAL PROFIT		
	DQN	T-DQN	DDQN
Google	1111.6603360000001	2480.3696880000007	5970.660216000003
Amazon	-679.1896960000004	106.12011699999994	1385.1303739999994
Microsoft	165.0128980000001	263.189531	480.339325
Facebook	-423.54000600000006	-182.620009	103.6797999999998

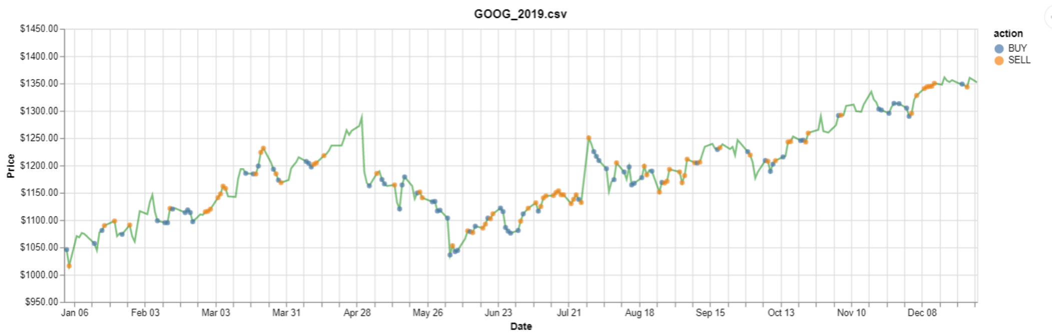

Fig 2: Time market profile of Google stock using DQN

Total profit of DQN algorithm on Google stock from 2019 to 2020 test data sums up to be 1111.66 . Here blue represents BUY and yellow represents SELL.

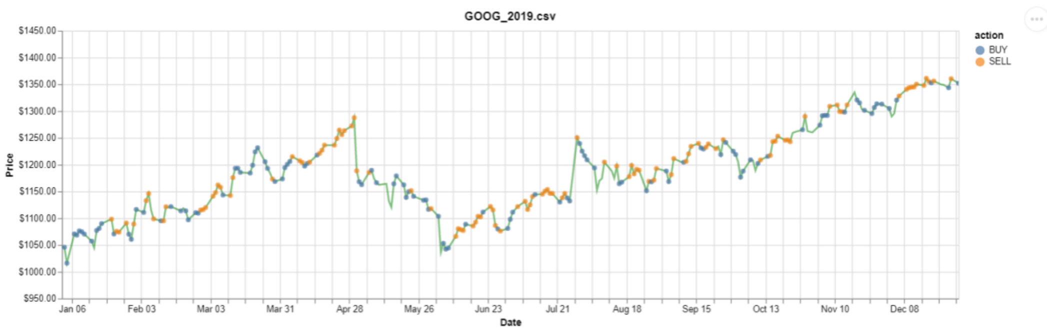

Fig 3: Time market profile of Google stock using T-DQN

Total profit of T-DQN algorithm on Google stock from 2019 to 2020 test data sums up to be 2480.37. Here blue represents BUY and yellow represents SELL.

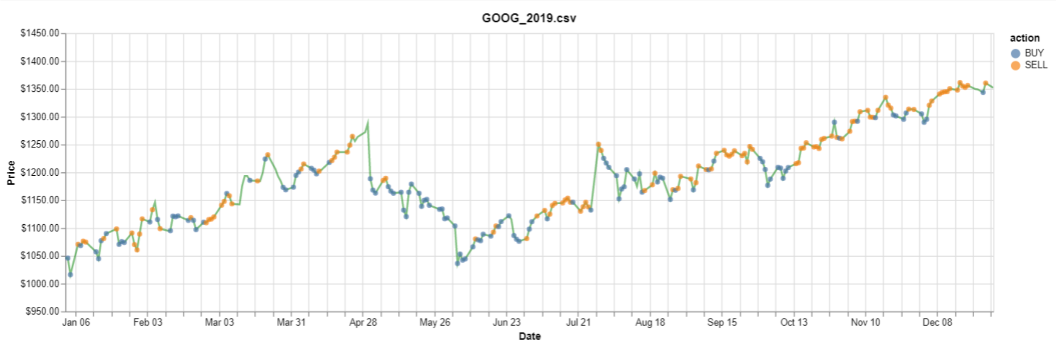

Fig 4: Time market profile of Google stock using DDQN

Total profit of DDQN algorithm on Google stock from 2019 to 2020 test data sums up to be 5970.66 . Here blue represents BUY and yellow represents SELL.

Fig 5: Time market profile of Amazon stock using DQN

Total profit of DQN algorithm on Amazon stock from 2021 to 2022 test data sums up to be -679.19 . Here blue represents BUY and yellow represents SELL.

Fig 6: Time market profile of Amazon stock using T-DQN

Total profit of T-DQN algorithm on Amazon stock from 2021 to 2022 test data sums up to be 106.12 . Here blue represents BUY and yellow represents SELL.

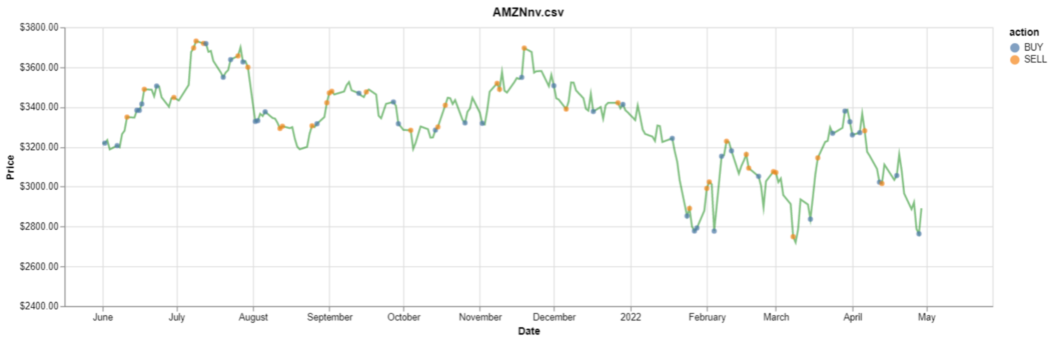

Fig 7: Time market profile of Amazon stock using DDQN

Total profit of DDQN algorithm on Amazon stock from 2021 to 2022 test data sums up to be 1385.13 . Here blue represents BUY and yellow represents SELL.

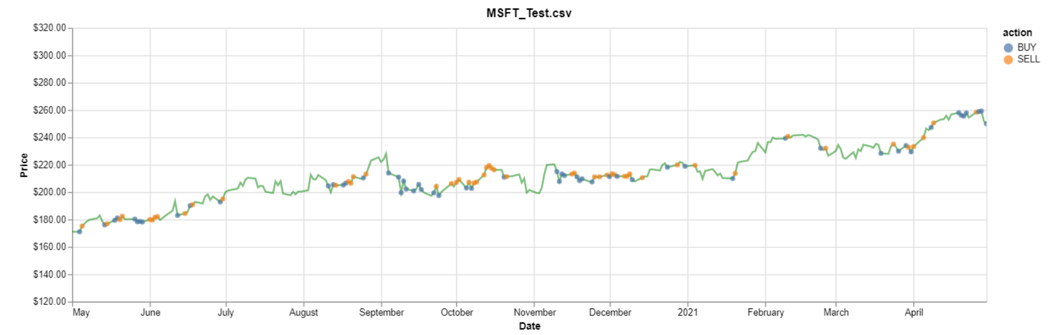

Fig 8: Time market profile of Microsoft stock using DQN

Total profit of DQN algorithm on Microsoft stock from 2020 to 2021 test data sums up to be 165.013 . Here blue represents BUY and yellow represents SELL.

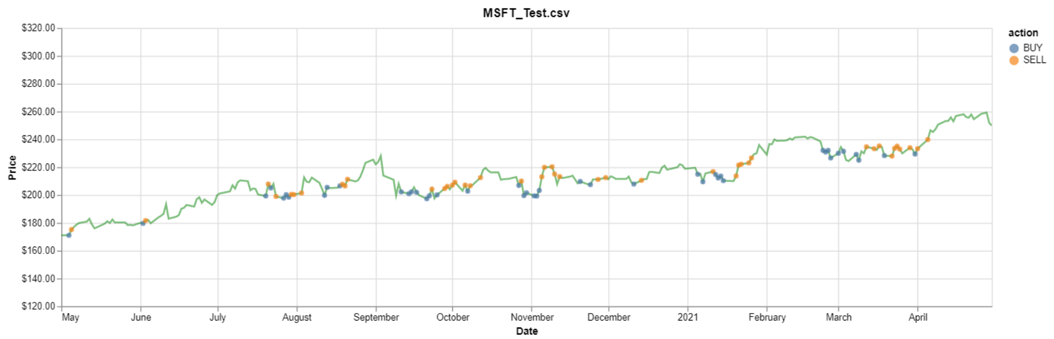

Fig 9: Time market profile of Microsoft stock using T-DQN

Total profit of T-DQN algorithm on Microsoft stock from 2020 to 2021 test data sums up to be 263.19 . Here blue represents BUY and yellow represents SELL.

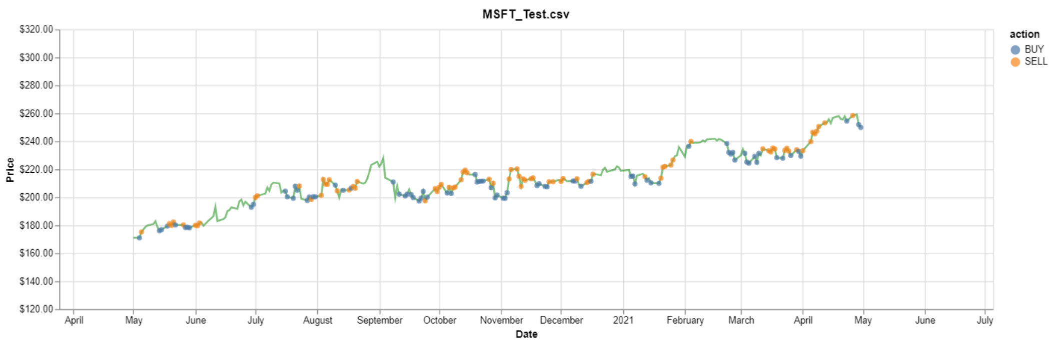

Fig 10: Time market profile of Microsoft stock using DDQN

Total profit of DDQN algorithm on Microsoft stock from 2020 to 2021 test data sums up to be 480.34. Here blue represents BUY and yellow represents SELL.

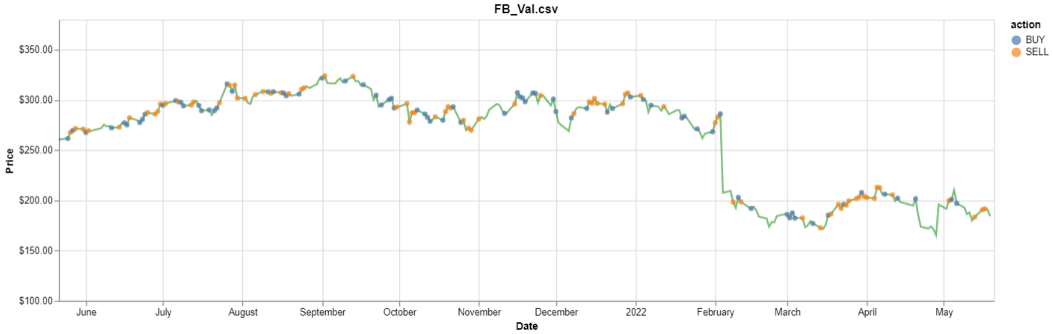

Fig 11: Time market profile of Facebook stock using DQN

Total profit of DQN algorithm on Facebook stock from 2021 to 2022 test data sums up to be -423.54 . Here blue represents BUY and yellow represents SELL.

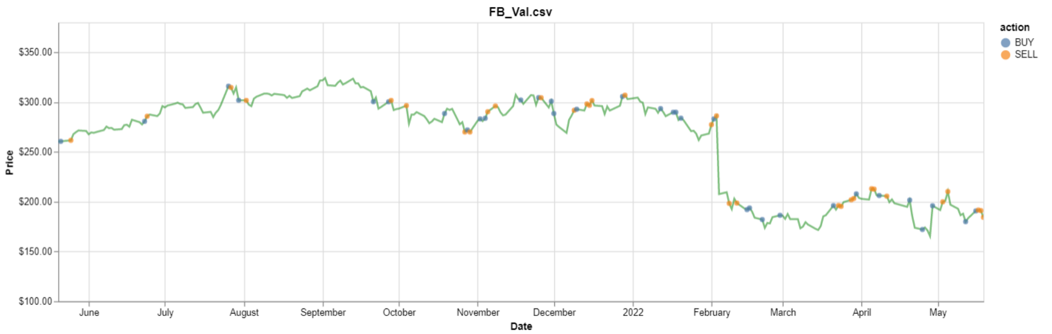

Fig 12: Time market profile of Facebook stock using T-DQN

Total profit of T-DQN algorithm on Facebook stock from 2021 to 2022 test data sums up to be -182.62 . Here blue represents BUY and yellow represents SELL.

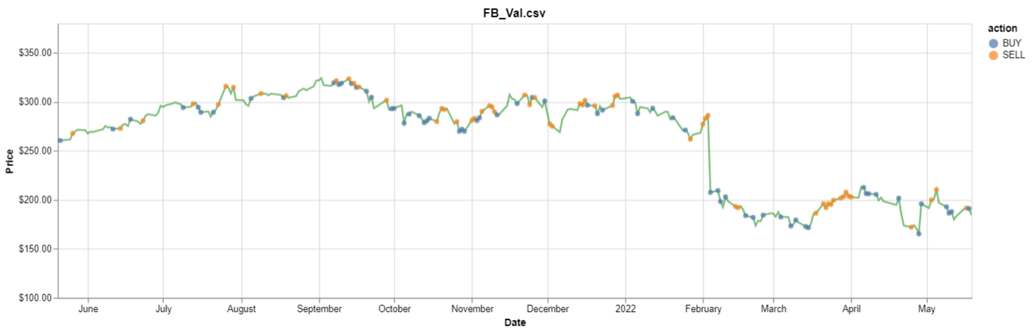

Fig 13: Time market profile of Facebook stock using DDQN

Total profit of DDQN algorithm on Facebook stock from 2021 to 2022 test data sums up to be 103.68. Here blue represents BUY and yellow represents SELL.

5. Conclusion

We employed deep reinforcement learning to automate trade execution and generate profit. In order to evaluate the effectiveness of the several deep reinforcement learning algorithms, we also evaluated DQN, Target DQN, and Double DQN using datasets of well-known stocks. The trials revealed that all three deep learning algorithms are capable of tackling stock market strategy decision-making challenges. Because the stock market is very stochastic and changes rapidly, these algorithms respond swiftly to these fluctuations and outperform traditional approaches. We found that the Double DQN network outperformed Target DQN, and that Target DQN outperformed DQN.

References

[1] I. Parmar et al., "Stock Market Prediction Using Machine Learning," 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018, pp. 574-576, doi: 10.1109/ICSCCC.2018.8703332.

[2] [2] Neeraj Mohan, Pankaj Jha, Arnab Kumar Laha and Goutam Dutta, “Artificial Neural Network Models for Forecasting Stock Price Index in Bombay Stock Exchange,” January 2005

[3] Guoao Duan, Mengyao Lin, Hui Wang, Zhuofan Xu, “Deep Neural Networks for Stock Price Prediction,” 9 January 2022

[4] Yingying Yan and Daguang Yang, “A Stock Trend Forecast Algorithm Based on Deep Neural Networks,” 5 July 2021

[5] Supriya Bajpai, “Application Of Deep Reinforcement Learning For Indian Stock Trading Automation,” July 1, 2021

[6] F. Agostinelli, S. Mcaleer, A. Shmakov, and P. Baldi, “Solving the Rubik’s cube with deep reinforcement learning and search,” *Nature Machine Intelligence*, vol. 1, no. 8, pp. 356–363, 2019.

[7] Junhao Zhang “Deep Reinforcement Learning for Stock Prediction” Next-Generation Optimization Models and Algorithms in Cloud and Fog Computing Published30 Apr 2022.

[8] Z. Xiong, X. Liu, S. Zhong, H. Yang, and A. Walid, “Practical deep reinforcement learning approach for stock trading,” CoRR, vol. abs/1811.07522, 2018.

[9] S. Asadi, E. Hadavandi, F. Mehmanpazir, and M. M. Nakhostin, “Hybridization of evolutionary Levenberg-Marquardt neural networks and data pre-processing for stock market prediction,” *Knowledge-Based Systems*, vol. 35, pp. 245–258, 2012

[10] Rinartha, K., & Suryasa, W. (2017). Comparative study for better result on query suggestion of article searching with MySQL pattern matching and Jaccard similarity. In 2017 5th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-4). IEEE.

[11] R. Hafner and M. Riedmiller, “Reinforcement learning in feedback control,” *Machine Learning*, vol. 84, no. 1-2, pp. 137–169, 2011

[12] P. Yu and X. Yan, "Stock price prediction based on deep neural networks," *Neural Computing and Applications*, vol. 32, no. 6, pp. 1609–1628, 2020

[13] Y. Song, J. W. Lee, and J. Lee, "A study on novel filtering and relationship between input-features and target-vectors in a deep learning model for stock price prediction," *Applied Intelligence*, vol. 49, no. 3, pp. 897–911, 2019.

[14] M Venu Gopalachari, Udumala Vineeth, Zohaib Abdullah Ahmed (2021), "A Deep Learning Approach For Criminal Prediction Analysis", Aegaeum Journal, ISSN: 0776-3808, Volume 9, Issue 6, pp:213-218.