Independent University **Bangladesh (IUB)**

IUB Academic Repository

Research Articles 2023

2023-06

A study of forecasting stocks price by using deep Reinforcement Learning

Khan, Razib Hayat

Independent University, Bangladesh

https://ar.iub.edu.bd/handle/11348/598

Downloaded from IUB Academic Repository

A study of forecasting stocks price by using deep Reinforcement Learning

Razib Hayat Khan
Department of Computer Science and Engineering
Independent University Bangladesh
Dhaka, Bangladesh
rkhan@iub.edu.bd

Jonayet Miah
Department of Computer Science
University Of South Dakota
South Dakota, USA
Jonayet.miah@coyotes.usd.edu

Md Minhazur Rahman
Department of Computer Science
University of South Dakota
South Dakota, USA
minhazur.rahman@coyotes.usd.edu

Md Maruf Hasan
Department of Computer Science
University Of South Dakota
South Dakota, USA
Maruf.hasan@coyotes.usd.edu

Muntasir Mamun
Department of Computer Science
University Of South Dakota
South Dakota, USA
Muntasir.mamun@coyotes.usd.edu

Abstract- Financial investors are so concerned now about the future of the stock market and how the market will behave next decade because the world economy is now in an alarming condition which leads to losses in the stock market. That is why traders want to know a little bit about the future forecast of the stock market. So, in this paper, we approach a bit to predict the stock market using the deep reinforcement learning method. Traditional methods for stock price prediction often rely on statistical models or technical indicators, which may struggle to capture the non-linear patterns and sudden shifts in stock prices. In recent years, deep reinforcement learning (DRL) has emerged as a promising approach for predicting stock prices, as it can learn complex patterns from raw data and make decisions based on sequential actions. n this study, we propose a novel framework for stock price prediction using DRL. The framework incorporates a deep neural network as a function approximator, which is trained using the Q-learning algorithm to learn optimal actions for buying, selling, or holding stocks. The neural network takes historical stock price data as input and outputs Q-values, which represent the expected rewards for different actions at each time step. The best course of action to pursue in each market state is then determined using the Q-values. We performed a sensitivity analysis to investigate the effects of various network designs and hyperparameters on the effectiveness of our DRL-based strategy. We found that the choice of hyperparameters, such as learning rate and exploration rate, had a significant impact on the performance, and tuning these hyperparameters could further improve the prediction accuracy. Our experimental results showed that our DRL-based approach outperformed the traditional methods in terms of predicting stock prices, with higher accuracy and lower prediction errors.

Keywords: Reinforcement learning, Q-values, Markov, GRU network, MTC-R.

I. Introduction

There is so much work done to predict stock prices, but the problem is nobody could propose a sustainable forecast because the stock price is a continuous process, and the market is not stable it's very fluctuating over time. In reinforcement learning, the stock market is considered an environment in which the agent (the reinforcement learning algorithm) interacts. The environment is defined based on the preprocessed data, including the state space, action space, and reward function. The state space represents the current state of the market, which can include features such as stock prices, trading volumes, and technical indicators. The action space represents the available actions that the agent can take, such as buying, selling, or holding stocks. The reward function defines the immediate feedback the agent receives for each action taken, which can be based on changes in stock prices or other relevant indicators.

A frequent concern among financial analysts and investors revolves around the ability to make precise predictions, regarding stock prices [1]. Deep reinforcement learning (DRL) is a cutting-edge approach to artificial intelligence. (AI) that combines two powerful techniques: deep learning and reinforcement learning. It is a kind of machine learning approach where an agent learns to make decisions in an environment to maximize a specific objective through trial and error. DRL allows agents to learn directly from raw sensory input, such as images or sounds, without relying on handcrafted features or domain-specific knowledge. This makes DRL highly

flexible and capable of tackling complex tasks in a wide range of domains, including robotics, autonomous driving, game playing, recommendation systems, and more. As per the prevailing value investment theory, the intrinsic value of a stock is typically determined by the market valuation of the corresponding company. However, there are instances where stock prices deviate from rational market expectations. The unpredictability and fluctuating nature of stock prices often challenges the assumptions and foundations of statistical models, leading to inaccurate or even detrimental prediction outcomes [2,3]. DRL has achieved remarkable breakthroughs in various domains, surpassing human-level performance in games like Go, chess, and video games, and demonstrating capabilities in autonomous driving, robotics, and more. DRL has also shown promise in addressing real-world problems, such as optimizing energy usage, personalized medicine, and financial decision-making [4]. Despite its successes, DRL still faces challenges, including sample inefficiency, exploration-exploitation trade-offs, and safety concerns. Researchers are actively working on developing new algorithms, architectures, and methodologies to address these challenges and unlock the full potential of DRL. As DRL continues to evolve, it has the potential to revolutionize industries, create new applications, and shape the future of AI and machine learning.

II. Literature review

Jae Won Lee. [7] This paper introduces a novel approach that applies reinforcement learning to predict stock prices. The authors view stock price prediction as a Markov process and propose using the TD (0) algorithm, which learns from experiences, for optimization. To approximate the values of states representing different stock price trends, an artificial neural network is employed for function approximation. The proposed method aims to model and learn interactions in real-world situations for improved stock price prediction.

K. Seo et al. [8] The authors propose a new model for predicting stock price movements called Multi-Time Contexts and Randomness (MTC-R). The model assumes that noisy traders' trading and time-based momentums, which add randomness to the stock market, both have an impact on stock prices. To model these hypotheses, MTC-R has three essential steps. First, a GRU network is used to encode different time viewpoints after learning time representations using time2vec. Additionally, it employs a multi-head attention method to construct multi-time contexts. while incorporating randomness. The proposed model demonstrates enhanced prediction performance in accuracy and Matthew’s correlation coefficient when compared to baseline models on six benchmark datasets. Additionally, MTC-R proves its effectiveness in portfolio trading simulations by showcasing improved prediction results through cumulative returns.

M. S. Roobini et al. [9] The main aims to evaluate and compare various supervised classification machine learning techniques to determine the most effective approach. The model aims to forecast stock price increases or stable states. This work also examines the effectiveness of various machine learning techniques, such as logistic regression, random forest, decision trees, and naive Bayes theorem, in generating stock price forecasts. To evaluate these methods, the study uses data from the transport traffic department and conducts a comparative analysis.

Roy, S. [10] This study employs the Auto-ARIMA and Holt-Winters models to forecast future stock prices. Additionally, linear programming is utilized for portfolio optimization. By constantly training the model with up-to-date market data, it becomes capable of capturing the latest market trends, making it applicable to a wider range of portfolios in the future. The obtained trend analysis can aid in predicting better investment strategies in the stock market, enabling investors to make optimal decisions based on current market analysis. This approach aims to maximize returns and minimize losses, ultimately helping investors make informed investment choices.

A. Gondkar et al. [11] This article's main goal is to compare various machine learning forecasting algorithms for stock prices and to suggest a new model for anticipating stock price index values. With the help of information from the transportation and traffic departments, the study evaluates and compares the effectiveness of various algorithms. To produce stock price forecasts, the study employs four machine learning techniques, including naive Bayes theory, decision trees, random forests, and logistic regression. Utilizing techniques like calculating the confusion matrix, prioritizing data categorization, and contrasting it with desired accuracy, recall, and F1 score levels, the proposed system is evaluated.

S. Merello et al. [12] This paper presents a regression-based approach to forecasting stock market movements by estimating market returns. The approach considers the magnitude of price fluctuations by assigning different weights to samples and addresses data scarcity through transfer learning. The results of real-world trading simulations demonstrate that the proposed approach can be used to strategically invest in high-performing stocks, resulting in higher profits with fewer trades, given a limited amount of capital.

A. Pastore et al. [13] The author of this paper analyze data from a financial market online game, involving forty-six

players, to investigate whether reinforcement learning (specifically, Q-Learning) can capture their behavior using a riskiness measure based on financial modeling. The hypothesis that players are "naïve" or shortsighted is also tested. The findings suggest that the decision-making process of the player involves reinforcement learning as one of its components. Furthermore, it is found that using a full reinforcement learning model, as opposed to a reduced version that only values immediate rewards (myopic), improves the fit for some players, suggesting that not all players are naïve in their decision-making approach.

III. Methodology

In reinforcement learning, an agent gathers data from its surroundings and modifies its behavior in response to the environment's state. Each action has an impact on the environment and alters the overall reward points, and the agent chooses actions depending on connected rewards. The new status is instantly considered when updating rewards and punishments for activities. The agent and environment continue to interact until the agent discovers a decision-making method that maximizes total return. Rewards and environment are two of the four crucial elements that influence reinforcement learning, according to Sutton and Barto. Policies in reinforcement learning are like self-imposed rules that guide the agent's behavior within the environment. In the training process, the primary objective is the reward function, which represents the ultimate goal to be achieved. On the other hand, the value function evaluates the long-term desirability of states or state-action pairs in terms of their overall goodness.

1. Data Collection and Processing

It is the initial process of this work we have done the most struggling work. Acquiring high-quality financial data can be costly, and free sharing of such data is rare. However, we have successfully collected comprehensive historical daily price and volume data for all US-based stocks and ETFs traded on the NYSE, NASDAQ, and NYSE MKT, making it one of the most valuable datasets available in this domain. The collected data is then preprocessed to prepare it for further analysis. This can involve cleaning the data by handling missing values, correcting errors, and normalizing or standardizing the data to a common scale. Data preprocessing also includes feature engineering, which involves selecting relevant features or indicators that are expected to have predictive power for stock price movements.

After preprocessing the data, the training, validation, and testing sets are created. During the training process, the validation set is utilized to adjust hyperparameters and assess the model's performance. In contrast, the testing set is reserved for evaluating the final trained model's performance, while the training set is used to train the reinforcement learning algorithm. Overall, the data collection process for stock price prediction using reinforcement learning involves gathering historical stock price data, preprocessing the data, defining the environment, splitting the data into training, validation, and testing sets, collecting data during training and evaluation, analyzing the collected data, and iterating the process to refine the model's performance. The flowchart, presented in Figure 1, outlines the procedure of reinforcement learning (RL) for agent-environment interaction in a concise manner. On this event, the agent takes an action in response to the current state, and because of this action, the system receives a signal, known as a reward (Rt+1), which guides the By employing the probability function f(at, st), the state is updated in the subsequent time step to reflect the behavior of the chosen action.

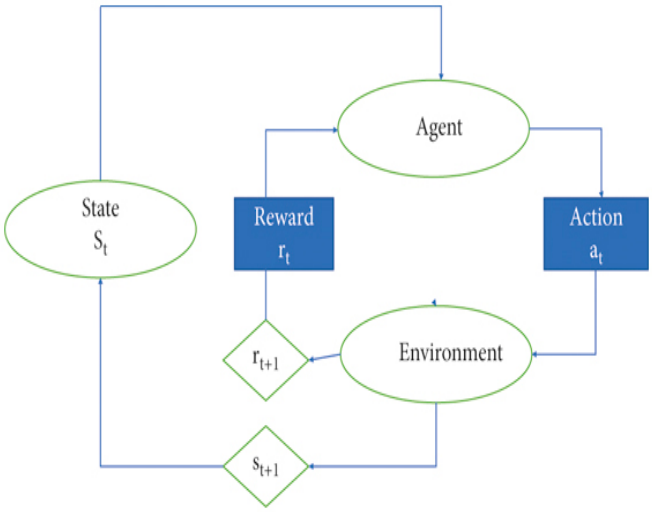

Fig 1: The Full process of reinforcement learning (RL) for agent-environment interaction

2. Relation between the agent and the environment

Reinforcement learning (RL) aims to maximize the total reward received by an agent within a finite number of actions by mapping environmental states to actions. The Markov decision process (MDP) is commonly used as a mathematical model to represent the state of the RL problem, denoted as Quad (S, A, p, f). In this notation, S represents the set of possible states, A represents the possible actions, p represents the probability of state transitions, and f represents the reward function.

3. A value function-based approach to deep reinforcement learning

Convolutional neural networks (CNNs) from deep learning and the Q learning algorithm from conventional reinforcement learning are combined in Minh's novel deep Q network (DQN) model. This model stands out for its use of convolutional layers, which significantly improve learning effectiveness and performance. Four previously processed photos that come before the current moment are used as input in the DQN model. These images undergo a nonlinear modification after being passed through numerous convolutional layers and fully linked layers. The Q value corresponding to each action is generated by the output layer. Figure 2 shows the DQN model's organizational
structure.

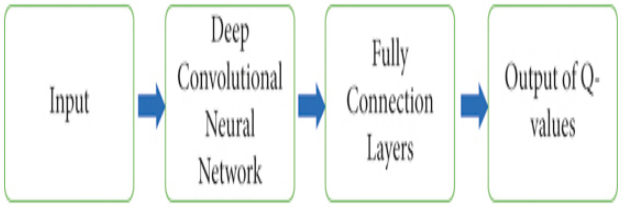

Fig 2: The structure of the DQN model

Our research employed a single-agent training technique that focuses on how quickly the price of a particular stock fluctuates in the market. By focusing on the rate of change, the training method aims to increase the capacity to forecast changes in the stock price. We assumed that the data would encompass all relevant market information, such as information about the surrounding area and the current condition of the stock. Despite the complex and dynamic nature of the market and the fact that the market price is influenced by the behaviors of multiple investors, we believe that after a prolonged reinforcement learning the results might offer a thorough reflection of the data present in the market during the reinforcement learning (RL) training period. It is appropriate to think about this assumption as a result.

The state parameter is a crucial component in the policy equation, as it influences the decision-making process. In our analysis, we assumed that all investors have rigid policies and that their choice of action is solely determined by the current state. In this context, the policies of other investors can be considered as a representation of the state of the market, especially when analyzing data in a 5-minute time frame.

The stock information indicators we considered in our research include Six state indicators for the stock market that reflect several data points, such as the starting and closing prices of a day, the highest and lowest prices reached, the day's average price, and the number of profitable trades executed. These metrics are essential for comprehending a stock's performance on a particular day.

IV. Result

a) Training and testing data Simulation

The objective of this article is for the authors to include past price movements to create a framework for forecasting short-term stock values. To test their suggested methodology, they used historical daily price and volume data for U.S. equities and ETFs. Data ranging from JUL 2013 to JAN 2016 were used to train the DRL model, while data from JAN 2016 to JUL 2018 were used to evaluate its performance. Figure 3 depiction of the findings shows a striking degree of agreement between the model's predictions and real stock prices during both the training and testing stages.



Fig 3: Testing and training data result

b) Loss function

During the model's training process using the dataset, the loss function gradually decreases as the epoch data increases. Figure 4 shows the training results of the loss function, revealing that the smallest loss is observed after approximately 10 epochs.

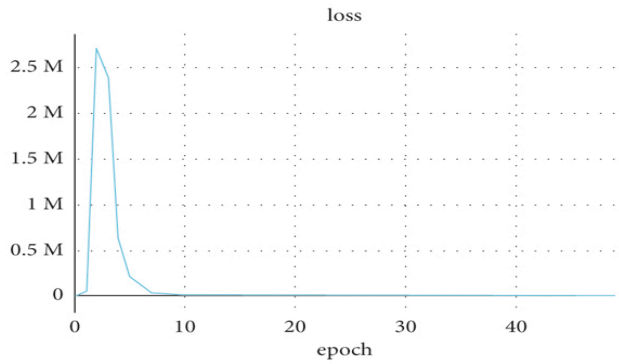

Fig 4. Loss function simulation

c) Reward Convergence

Furthermore, in Figure 5 illustrates how the dataset's training and testing phases involved tracking modifications in rewards within the DRL model. The DRL model was effectively trained, as shown by the graph in Figure 3, which shows that rewards fluctuated significantly at the start of the training and testing procedures but stabilized as the number of epochs rose [18].



Fig 5. Reward Convergence flow

d) DQN training and testing results.

Subsequently, we utilized the policy gradient method based on DRL as proposed in this paper for stock price prediction, as illustrated in Figure 6. The findings, also depicted in Figure 6, highlight that the approach proposed in this paper demonstrates superior accuracy in forecasting the trend of stock price data. In comparison to the results obtained from DRL, it is noted that although the method proposed in this paper may exhibit less stability in terms of the loss function, it rapidly stabilizes on the reward curve.

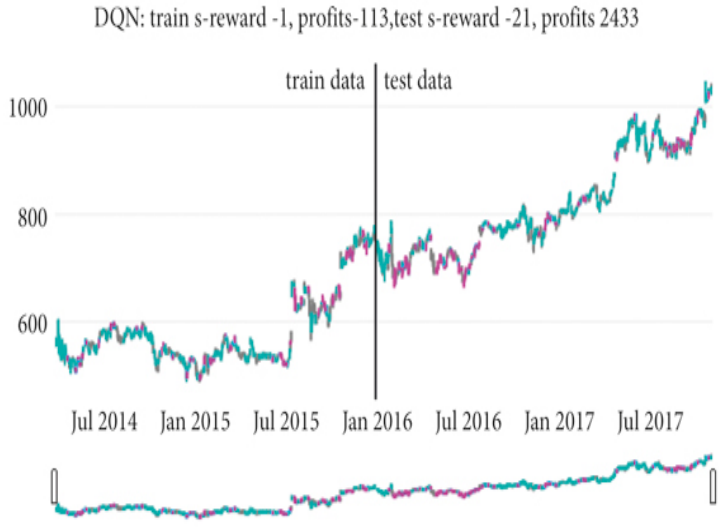

Fig 6. DQN training and testing results.

Discussion and Future Work

This paper proposes a new reinforcement learning method that incorporates policy gradients to improve the accuracy of daily stock price predictions. A comparison was made the comparison between two methods of predicting daily changes in stock prices: the basic Deep Reinforcement Learning (DRL) method and a newly proposed DRL-based policy gradient method. However, there is still room for improvement in this research. Intraday stock price volatility can be high, and using the results of this research for trading may result in reduced returns for investors who hold many stocks during periods of reduced holdings. Additionally, the data volume for daily stock price prediction often spans a long time, and market information from different years may not be generalized. We, therefore, propose that it could be more advantageous for investors to increase the frequency of stock price projections to 5- or 1-minute intervals. In order to help investors, forecast future price changes and boost returns, shorter periods offer a more accurate picture of stock price swings.

References

[1] 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

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

[3] V. Konda and J. Tsitsiklis, “Actor-critic algorithms,” Advances in Neural Information Processing Systems, vol. 12, 1999

[4] J. W. Lee, E. Hong, and J. Park, “A Q-Learning Based Approach to Design of Intelligent Stock Trading Agents,” In Proceedings of the 2004 IEEE International Engineering Management Conference, pp. 1289–1292, 2004

[5] M. P. Naeini, H. Taremian, and H. B. Hashemi, “Stock market value prediction using neural networks,” In Proceedings of the 2010 International Conference on Computer Information Systems and Industrial Management Applications, pp. 132– 136, IEEE, 2010.

[6] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” 2015, https://arxiv.org/abs/1506. 02438.

[7] Jae Won Lee, "Stock price prediction using reinforcement learning," IEEE International Symposium on Industrial Electronics Proceedings, pp. 690-695, 2001, Doi: 10.1109/ISIE.2001.931880.

[8] K. Seo and J. Yang, "Exploring Multi-Time Context Vector and Randomness for Stock Movement Prediction," In Proceedings of the International Conference on Big Data, pp. 1114-1123, 2022, Doi: 10.1109/BigData55660.2022.10020373.

[9] M. S. Roobini, K. Babu, J. Joseph, and G. Ieshwarya., "Predicting Stock Price using Data Science technique," In proceedings of the Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2022, pp. 1013-1020, 2022, Doi: 10.1109/ICAIS53314.2022.9742772.

[10] S. Roy, “Stock Market Prediction and Portfolio Optimization Using Data Analytics”. In Computational Intelligence in Data Mining: Proceedings of the International Conference on ICCIDM, pp. 367-381, 2018,

[11] A. Gondkar, J. Thukrul, R. Bang, S. Rakshe and S. Sarode, "Stock Market Prediction and Portfolio Optimization," In proceedings of the 2nd Global Conference for Advancement in Technology, 2021, pp. 1-10, 2021, Doi: 10.1109/GCAT52182.2021.9587659.

[12] S. Merello, A. P. Ratto, L. Oneto and E. Cambria, "Ensemble Application of Transfer Learning and Sample Weighting for Stock Market Prediction," In Proceedings of the International Joint Conference on Neural Networks, pp. 1-8, 2019, Doi: 10. 1109/IJCNN.2019.8851938.

[13] A. Pastore, U. Esposito, and E. Vasilaki, "Modelling stock-market investors as Reinforcement Learning agents," In Proceedings of the IEEE International Conference on Evolving and Adaptive Intelligent Systems, pp. 1-6, 2015, doi: 10.1109/EAIS.2015.7368789.

[14] Miah, J., et al. "Mhfit: Mobile health data for predicting athletics fitness using machine learning model." 2nd International Seminar on Machine Learning, Optimization, and Data Science, (Preprint).

[15] R.H. Khan, J.Miah M.Tayaba M.M.Rahman, “A Comparative Study of Machine Learning Algorithms for Detecting Breast Cancer,” 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), USA, (Preprint)

[16] R.H. Khan, J.Miah, Shah Ashisul Abed nipun, M. Islam, “A Comparative Study of Machine Learning Classifiers to Analyze the Precision of Myocardial Infarction Prediction,” 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), USA, (Preprint)

[17] S. Kayyum et al., "Data Analysis on Myocardial Infarction with the help of Machine Learning Algorithms considering Distinctive or Non-Distinctive Features," 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1-7, Doi: 10.1109/ICCCI48352.2020.9104104

[18] R. H. Khan, Poul H. Heegard, “Software Performance Evaluation Utilizing UML Specification and SRN model and Their Formal Presentation”, Journal of Software, Vol. 10, No. 5, pp. 499 – 523, 2015