Transformer Actor-Critic with Regularization: Automated Stock Trading using Reinforcement Learning

Extended Abstract

Namyeong Lee
Hanyang University
Seoul, South Korea
leeny@hanyang.ac.kr

Jun Moon
Hanyang University
Seoul, South Korea
junmoon@hanyang.ac.kr

ABSTRACT

Recently, with the increasing interest in investments in financial stock markets, several methods have been proposed to automatically trade stocks and/or predict future stock prices using machine learning techniques, such as reinforcement learning (RL), LSTM, and transformers. Among them, RL has been applied to manage portfolio assets with a sequence of optimal actions. The most important factor in investing in stocks is the utilization of past stock price data. However, existing RL algorithms applied to stock markets do not consider past stock data when taking optimal actions, as RL is formulated based on the Markov decision process (MDP). To resolve this limitation, we propose Transformer Actor-Critic with Regularization (TACR) using decision transformer to train the model with the correlation of past MDP elements using an attention network. In addition, a critic network is added to improve the performance by updating the parameters based on the evaluation of an action. For an efficient learning method, we train our model using an offline RL algorithm through suboptimal trajectories. To prevent overestimating the value of actions and reduce learning time, we train TACR through a regularization technique with an added behavior cloning term. The experimental results using various stock market data show that TACR performs better than other state-of-the-art methods in terms of the Sharpe ratio and profit.

KEYWORDS

Reinforcement Learning; Sequence Modeling; Portfolio Allocation

ACM Reference Format:
Namyeong Lee and Jun Moon. 2023. Transformer Actor-Critic with Regularization: Automated Stock Trading using Reinforcement Learning: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

Recently, various machine learning methods such as RL, LSTM, and transformers have been studied for investments in financial stock markets. The main purpose of such studies is to predict future stock prices and/or automatically trade stocks based on specified optimization criteria. In fact, since stock prices exhibit patterns, such algorithms show remarkable performance. Specifically, the RL methods [8, 15], the transformer, and LSTM [3, 10, 16] prove the necessity of considering past stock prices for stock market analysis.

We adopt the RL algorithm to automatically manage assets in various portfolio allocation problems by performing optimal actions on a daily basis [8, 15]. However, when an agent takes an action in the current state using RL in [8, 15], historical information cannot be considered due to the inherent Markov property of RL. Although the algorithms in [11, 14] are proposed as a combination of LSTM and transformers with RL, only state is considered among MDP elements, and LSTM has a vanishing gradient problem. Hence, there exists a limit when the model learns a long sequence. Moreover, the transformer is not used as a decision model to predict actions. To address this problem, the decision transformer [1] is a suitable model, which uses the GPT-2 [12]. By combining the GPT-2 with the RL, the GPT-2 is able to predict the current action based on historical MDP elements through the attention mechanism. Thus, when modeling based on a decision transformer in stock investment, it can be trained using historical data.

The decision transformer updates the transformer using the mean squared error (MSE), which is a loss function. Using only MSE, it is difficult to exceed the performance of suboptimal trajectories prepared with training data. Hence, in this paper, we propose TACR (Transformer Actor-Critic with Regularization), a new RL algorithm, which applies the critic to the decision transformer to improve the overall performance of RL using past MDP elements. Additionally, we train the model offline using pre-generated suboptimal trajectories for imitating good actions and reducing learning time. However, when only the critic is applied, the value function is inaccurate, leading to overestimation and reducing overall performance [7], since the actor does not interact with the environment. Hence, we apply the state-of-the-art regularization method [4] to prevent actions from being overvalued by the critic.

TACR guarantees the highest returns compared to other RL and transformer algorithms [2, 5, 6, 8, 13, 15, 16] using various datasets from Dow Jones, US 50, HighTech, NDX, MDAX, and CSI.

Our code is available at: https://github.com/VarML/TACR

2 TRANSFORMER ACTOR-CRITIC WITH REGULARIZATION

We construct the state space of MDP representing meaningful stock information such as opening/closing/high/low stock prices and technical indicators. The action space is the set of the allocation weights $\boldsymbol{a}_t = \{a_{0,t}, a_{1,t}, \ldots, a_{J,t}\}^\top$ satisfying $\sum_{j=0}^{J} a_{j,t} = 1$. Here, $a_{j,t}$ represents the size of allocation weight to invest in the jth stock for the current period t. The reward is scalar and is defined as the sum of the rate of daily returns for each stock $r_{t+1} = r(s_t, a_t) =$

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom. © 2023 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

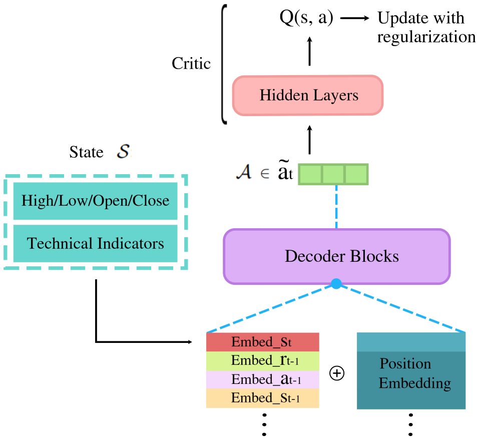

Figure 1: Framework of TACR.

$\boldsymbol{a}_t \cdot (\boldsymbol{\rho}_t - \mathbf{1})$, where ρ_t denotes the ratio of the next day's stock prices to current stock prices.

In TACR depicted in Figure 1, the decision transformer is considered as an actor predicting an action and then evaluates the action with a critic network adding a regularization method to improve performance. The mechanism of the actor network π is to map the previous MDP elements to the current action and consists of hidden layers and several decoder blocks that use the attention mechanism to train the correlation of each MDP element as follows:

$$h_0 = MW_e + W_p \tag{1}$$

$$h_l = \texttt{Decoder\_block}(h_{l-1}), \ l = 1, \ldots, L \tag{2}$$

$$\tilde{a} = \texttt{Softmax}(h_L W_L + b_L), \tag{3}$$

where $M = (r_{-u}, s_{-u}, a_{-u}, \ldots, r_{-1}, s_{-1})$ is a matrix consisting of the token vectors of the previous MDP elements with length u. In (1), the MDP elements are computed by the weights W_e and position embedding weights W_p to represent the hidden state as an input. Through L decoder blocks, the correlation of the embedding inputs is learned via (2). Finally, the action is predicted through a linear transformation layer and a Softmax function in (3).

We train the model using an offline method rather than the off-policy method commonly used in stock trading. The agent imitates prepared suboptimal actions for expecting better performance and reducing learning time. In order to train the model in the offline method, suboptimal trajectories must be created by pairing the suboptimal actions corresponding to each state. We generate trajectories with high rates of action according to the rate of increase in stock prices on a daily basis.

In the offline RL algorithms, there is also a limitation of not interacting with the environment when updating a policy. In this case, agents tend to incorrectly estimate the value of actions for unseen states. To evaluate more accurately out-of-distribution actions, the regularization method is required [4] as follows:

$$\pi = \underset{\pi}{\operatorname{argmax}} \, \mathbb{E}_{(s,a,r)\sim D}\left[\lambda Q(s, \pi(M)) - (\pi(M) - a)^2\right], \tag{4}$$

where π corresponds to the transformer actor and the sequence M indicates that a certain number of MDP elements is stacked. Intuitively, the behavior cloning regularization term is added to DDPG.

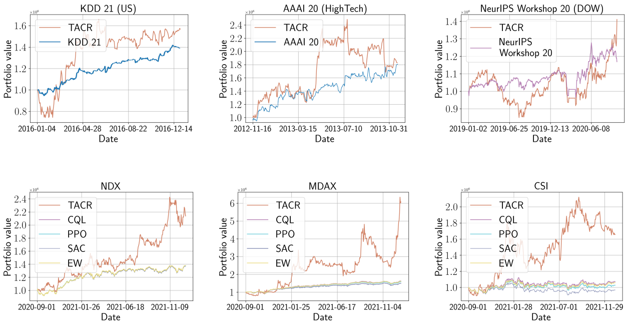

Figure 2: Comparison of the profit of TACR with other methods. TACR yields 13% **(KDD 21),** 1.1% **(AAAI 20),** 20.7% **(NeurIPS Workshop 20),** 54.8% **(NDX),** 274.8% **(MDAX), and** 54.5% **(CSI) higher portfolio values compared with other algorithms.**

This term causes the transformer actor to follow the distribution of actions in the suboptimal trajectories included in the dataset. λ is a hyperparameter that simultaneously controls maximizing Q and minimizing the behavior cloning term.

3 EXPERIMENT

We use the same datasets as the state-of-the-art methods [8, 15, 16] for comparison of TACR. Furthermore, to demonstrate the general applicability of TACR, we provide the additional experiment results using NDX, MDAX, and CSI datasets which are the stock indices of the US, Germany, and China, respectively.

Transactions in all datasets are made on a daily basis. We adopt the most commonly used Portfolio value and Sharpe ratio [9] as metrics. We compare the performance of our model with other papers on portfolio allocation [8, 15, 16], and further include offline RL [6], off-policy RL [5], on-policy RL [13], and classic method (Equal weight strategy) as baselines. We construct baselines with other papers, all model-free RL algorithms to show the superiority of TACR, and EW to show whether RL is practical for stock trading application. As a result, Figure 2 shows that TACR has the highest performance compared with various baselines. Furthermore, when comparing the Sharpe ratio with our model and other algorithms, it is also high at least 13.1%, up to 177.7%, excluding the MDAX dataset. Furthermore, when increasing the sequence length u, most datasets show good results. This means that the longer the past MDP elements are considered, the closer to optimal action is taken.

4 CONCLUSION

In stock trading, it is essential to analyze the market using historical data. We propose TACR, which includes the critic network to the decision transformer and the regularization method. In addition, we train our model efficiently using suboptimal trajectories offline. TACR enables the RL method to consider not only past stock data but also previous MDP elements when taking optimal actions, which is different from standard RL techniques formulated based on the MDP framework. Furthermore, TACR prevents the overestimation of the values of the actions. Compared with state-of-the-art methods, TACR shows good performance for various datasets.

ACKNOWLEDGMENTS

This work was supported in part by the Technology Innovation Program (20018112) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea), in part by the National Research Foundation of Korea (NRF) Grant funded by the Ministry of Science and ICT, South Korea (NRF-2021R1A2C2094350), and in part by Institute of Information & communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-01373).

REFERENCES

[1] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems* 34 (2021).

[2] Stephen Dankwa and Wenfeng Zheng. 2019. Twin-delayed ddpg: A deep reinforcement learning technique to model a continuous movement of an intelligent robot agent. In *Proceedings of the 3rd International Conference on Vision, Image and Signal Processing*. 1–5.

[3] Qianggang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Jian Guo. 2020. Hierarchical Multi-Scale Gaussian Transformer for Stock Movement Prediction.. In *IJCAI*. 4640–4646.

[4] Scott Fujimoto and Shixiang Shane Gu. 2021. A minimalist approach to offline reinforcement learning. *Advances in Neural Information Processing Systems* 34 (2021).

[5] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. PMLR, 1861–1870.

[6] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

[7] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643* (2020).

[8] Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, and Christina Dan Wang. 2020. Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance. *arXiv preprint arXiv:2011.09607* (2020).

[9] Andrew W Lo. 2002. The statistics of Sharpe ratios. *Financial analysts journal* 58, 4 (2002), 36–52.

[10] David M. Q. Nelson, Adriano C. M. Pereira, and Renato A. de Oliveira. 2017. Stock market's price movement prediction with LSTM neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)*. 1419–1426. https://doi.org/10.1109/IJCNN.2017.7966019

[11] ES Ponomarev, Ivan V Oseledets, and AS Cichocki. 2019. Using reinforcement learning in the algorithmic trading problem. *Journal of Communications Technology and Electronics* 64, 12 (2019), 1450–1457.

[12] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[13] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[14] Jingyuan Wang, Yang Zhang, Ke Tang, Junjie Wu, and Zhang Xiong. 2019. Alphastock: A buying-winners-and-selling-losers investment strategy using interpretable deep reinforcement attention networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1900–1908.

[15] Yunan Ye, Hengzhi Pei, Boxin Wang, Pin-Yu Chen, Yada Zhu, Ju Xiao, and Bo Li. 2020. Reinforcement-learning based portfolio management with augmented asset movement prediction states. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 1112–1119.

[16] Jaemin Yoo, Yejun Soun, Yong-chan Park, and U Kang. 2021. Accurate multivariate stock movement prediction via data-Axis transformer with multi-level contexts. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2037–2045.