Deep Reinforcement Learning for Stock Recommendation

Yifei SHEN\textsuperscript{1, a}, Tian LIU\textsuperscript{1, b}, Wenke LIU\textsuperscript{3, c}, Ruiqing XU\textsuperscript{4, d}, Zhuo LI\textsuperscript{5, e} and Jia WANG\textsuperscript{6, f}

1The School of Mathematical and Physical Sciences, The University of Sussex, UK; 2The Department of Business School, The University of Southampton, UK; 3Microport EP MedTech Cp., Ltd; 4The Department of Computer Science, The University of Sheffield; 5The School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, China; 6Department of Intelligent Science, Xi’an Jiaotong-Liverpool University, China.

\textsuperscript{a}ys266@sussex.ac.uk; \textsuperscript{b}tl1y19@soton.ac.uk; \textsuperscript{c}liuwk@microport.com; \textsuperscript{d}rxu22@sheffield.ac.uk; \textsuperscript{e}lizhuo@cqupt.edu.cn; \textsuperscript{f}Jia.wang02@xjtlu.edu.cn.

Abstract. Recommending stocks is very important for investment companies and investors. However, without enough analysts, no stock selection strategy can capture the dynamics of all S&P 500 stocks. Nevertheless, most existing recommending strategies are based on predictive models to buy and hold stocks with high return potential. But these strategies fail to recommend stocks from different industrial sectors to reduce risks. In this article, we propose a novel solution that recommends a stock portfolio with reinforcement learning from the S&P 500 index. Our basic idea is to construct a stock relation graph (RG) which provide rich relations among stocks and industrial sectors, to generate diversified recommendation result. To this end, we design a new method to explore high-quality stocks from the constructed relation graph with reinforcement learning. Specifically, the reinforcement learning agent jumps from each industrial sector to select stock based on the feedback signals from the market. Finally, we apply portfolio allocation methods (i.e., mean-variance and minimum-variance) to test the validity of the recommendation. The empirical results show that the performance of portfolio allocation based on the selected stocks is better than the long-term strategy on the S&P 500 Index in terms of cumulative returns.

1. Introduction

With the rapid development of the financial market, organizations and individuals have begun to invest in the stock market [1]. The growth of investment makes the market turmoil becomes even intense, which increases the risk of investment. By estimating the trend of stock prices, investors try to realise the expected return with the portfolio investment. Although earnings reports provide insights on how to value companies and select stocks, there are not enough human analysts to capture all the dynamics in the market to select stocks. Therefore, recommending stock is very important for investment companies and individuals [2].

Traditional stock recommendation methods can be classified into two categories. A single standard-based approach makes the portfolio contain stocks selected by the most critical factors (e.g., the lowest price/earnings to growth (PEG) ratio) [3]. Although these methods are effective, they often lead to an unstable performance in practice. For example, selecting top revenue stocks may result in less exploration ability. Other approaches [4, 5, 6] try to rank the stocks based on multiple criteria (e.g., P/E
ratio, price to sales (P/S) ratio, price/earnings to growth (PEG) ratio) from the financial domain knowledge. However, these approaches cannot consider the correlation between different criteria. Consequently, independently assigning the weights to each factor decreases the accuracy of stock ranking.

Nowadays, many studies have tried to develop a stock recommendation system by considering the characteristics of different investors. [7] provides general investors with recommendations that are more appropriate to their actual conditions. However, the analysis of the investors’ characteristics has brought new uncertainty to the stock recommendation. Moreover, several works [8, 9] try to value the companies by modeling investment portfolios with financial time series models. These recommendation models are built by estimating prospective revenues and costs. Typically, a cross-sectional model [10, 11] was proposed to estimate the implied cost of capital (ICC), and it plays an important role in firm valuation. Meanwhile, numerous research works [12, 13] consider using regressions models to estimate the revenue and net income. Recently, deep learning methods appeared to model stock data.

The deep neural network models [14, 15] can be used to predict future market fundamentals such as market volume and P/E ratio. As a result, the investors can select the stocks based on the predicted fundamentals. Considering that stock price is a dynamic Gaussian noise that is affected by a combination of complex factors and random factors. It is hard to make an accurate prediction of a stock’s future price. Also, the predicted top quantile stocks may fall in the same industrial sector, which increases the potential risks.

Figure 1. A toy example of selecting stocks by StockRL. Along with the constructed relation graph, the agent first picks the industrial sector and then selects the stocks based on the trading feedback. Under the covid-19 epidemic, at the state of investing in the Technology, Media, and Telecom (TMT) industry, the agent would further consider investing in the consumer packaged good (CPG) industry rather than in the aviation industry (AI).

In this paper, we propose a novel scheme called StockRL, which constructs a relation graph to represent the affiliation of stocks with different industries and recommends stocks portfolio based on reinforcement learning (RL) [16]. Reinforcement learning is one of the most general machine learning paradigms, alongside supervised learning and unsupervised learning. At each time step, a reinforcement learning agent receives the observation of the stock market and takes an action. After that, the agent will receive a signal which also calls the reward that evaluates the value of taking such action. With this feedback, the StockRL will label each state with correct action. The design of StockRL is based on the following principles: (1) all stocks affiliated to the same industry may share similar latent states to avoid information isolation; and (2) With reinforcement learning, the agent can explore and select stocks from different industrial sectors. The diversified result of stock recommendations can reduce the investment risk. We use Figure 1 to illustrate StockRL. The graph represents the relation of stocks, where the red circle represents stocks, and the yellow circle represents the industrial sectors. When applying our StockRL model, the agent learns to navigate in each sector and then select stocks associated with it. However, in many stocks, it is a challenge for RL agents to learn from “valuable” trial and error samples, which is also known as the RL exploration problem [17]. We claim that high-quality learning samples should satisfy two requirements: 1) informative, meaning that currently these samples are representative, such that training the strategy with them will change model parameters significantly, and 2) counterfactual,” ignore” the stocks that have been considered before but cause a loss, as a result, the
strategy can avoid similar mistakes.

To address this problem, we construct a stock relation graph (RG) which provides rich relations among stocks and industrial sectors, where the weight on the edge represents the correlation between stocks. With a knowledge graph, the agent can effectively learn from hundreds of stock samples. In particular, if one stock makes a huge profit, the agent will move along the edge of the graph, trying to find other similar stocks. Finally, we apply portfolio allocation methods (i.e., equally weighted, mean-variance, and minimum-variance) to test the validity of the recommendation. The empirical results show that the performance of portfolio allocation based on the selected stocks is better than the long-term strategy on the S&P 500 Index in terms of Sharpe ratio and cumulative returns.

In summary, the main contributions of this work are as follows:
- We incorporate knowledge graphs into reinforcement learning to select “valuable” trial-and-error samples to learn stock recommendation strategies.
- We develop a novel method, by applying a multi-hop navigation strategy, a reinforcement learning agent can quickly learn from knowledge entities on the graph.
- We conduct extensive experiments on the S&P 500 datasets, demonstrating the strength of StockRL in recommending high-margin, low-risk (i.e., diversified) stocks.

2. Task Formulation and Preliminaries

We first formally introduce the problem definition of stock recommendation. Then, we give an explanation and prepare RL settings to solve the problem.

In a stock market, an investor is able to buy an asset at time \( t_i - 1 \) and sell it at a fixed time \( t_i \geq 0 \). Let \( p_{ti} \) be the price of the asset \( i \) at time \( t \). Then the total return, \( R_t \), on the investment at time \( t \) is given by \( R_t = p_{ti} / p_{ti-1} \).

**Definition 1**: Portfolio. Assume there are number of \( n \) different assets and each of them corresponds a revenue return \( R_i \), \( i = 1, ..., n \). The portfolio research how to invest a total amount of money \( M_t \) into different asset \( i \) with weight \( w_i \), \( i = 1, ... n \) with high-revenue a low-risk, such that \( \sum_{i=1}^{n} M_{ti} = M_t \) and restricts that \( M_{ti} \geq 0 \), \( i = 1, ... n \), where \( M_{ti} \) is the amount of investments in asset \( i \) and \( M_t = \sum_{i=1}^{n} w_i M_{ti} \).

**Problem Definition**: Stock recommendation. Given a set of \( k \) stocks \( S = \{s_1, s_2, ..., s_k\} \) covered by the S&P 500 index, our goal is to select \( n \) stocks, where the portfolio allocation methods can obtain the largest cumulative return.

2.1. Reinforcement Learning Setup

Stock recommendation requires an agent to select stocks based on the dynamic market changes. This problem can be formulated as a Markov decision process (MDP) and solved using the reinforcement learning framework. Typically, a MDP contains 6-tuple \((I, A, T, R)\), where \( I, A, T, R \) are the states, actions, transitions, and rewards of the system. In the following, we formulate the stock recommendation problem in the context of reinforcement learning:

- **Observation space I**: An observation \( I_t \in I \) is defined as the representation of the stock market that includes the volume, macro market situation, and holding positions.
- **Action space A**: An action \( a_t \in A \) indicates stocks and sectors that the agent selects at the time \( t \).
- **Reward R**: After the agent takes action \( a_t \) under observation \( o_t \), it receives the reward \( r_t \) from the market as follows:
  \[
  r_t = \alpha R_t - \beta V_t - \gamma T_t, \tag{1}
  \]
  where \( R_t \) is the total return at time \( t \), and \( V_t \) is the variance which indicates the risk of investment, and \( T_t \) is the transaction cost. The transaction cost is calculated as follows.
  \[
  T_t = |S_t - S_{t-1}| \times P_t \times 0.1%, \tag{2}
  \]
  where \( S_t \) is the stock that needs to be bought or sold at the time \( t \) according to the weight of the portfolio, and \( S_{t-1} \) is the remaining share of stock at the time \( t - 1 \). \( P_t \) is the stock price at time \( t \).
3. The Model Framework

The framework of the proposed StockRL is illustrated in Figure 2, which mainly consists of the two modules, weight evaluator and recommender. In addition, the relation graph is built to introduce the extra relations among stocks and sectors, which is useful to infer recommendation results. The weight on the relationship graph refers to the trend similarity of stocks. The weight evaluator updates the graph weight with the trading feedback of the selected stocks in the market. Then, the agent learns to generate the stock selection result according to the structure of the graph. Recursively performing such graph weight updating, stock selection, StockRL ultimately able to yield the recommendation results that adapt to market dynamics.

![Figure 2. The system framework. The weight evaluator and stock recommender promote each other. Weight evaluator updates the weight on relation graph with the stocks sampled by recommender. While, the accurate relation graph helps StockRL to select stocks with high-margin and low-risk.](https://example.com/figure2.png)

3.1. Relation Graph

Inspired by recent works [18], we organize stock attributes, interrelations as well as stock sectors in the form of relation graph (KG). The first layer identifies the relation of the sectors, and the second layers is the stocks. The establish RG is formalized as $G = \{(e^i, e^j) | e^i, e^j \in E\}$, where $E$ is the entity set unifying the sector set $C$ and the stock set $S$. For example, (sector: TMT, stock: AAPL) describes the fact that AAPL belongs to the TMT sector. As the result, a learning agent can generate the recommendation result by navigating the entities along with the structure of the relation graph.

3.2. Weight Evaluator

The weight of the relation graph represents the trend similarities of entities at the two ends of the edge. Since updating the weight $\theta$ on the graph $G$ involves the sampling operation, which cannot be directly optimized by the gradient-based algorithm [19], we adopt the policy-gradient-based reinforcement learning to approximate the underlying weight value. In particular, the objective is to maximize the long-term return revenue, and we can update the weights by following the ascending by the following gradient:

$$\nabla_\theta L(G) = \nabla_\theta \sum_{|v|} \sum_{i=1} E_{v \sim G(v|\theta)} \log[R_t(v)],$$

$$= \frac{1}{|v|} \sum_{i=1} \nabla_\theta \log G(v|\theta) \log[R_t(v)]. \quad (3)$$

The gradient $\nabla_\theta L(G)$ is the expectation summation over the selected samples $v \sim G(v|\theta)$ weighted by the reward signal $R_t(v)$. As the result, the weights on the graph will be updated depending on the market dynamics. For example, under the market conditions of the covid-19 epidemic, the weight of the edge representing the relation between Technology, Media, and Telecom (TMT) sector and customer package good (CPG) sector will be increased and the weight of the TMT and aviation (AI) sector will be decreased.

3.3. Recommender

Based on the constructed relation graph, the recommender conducts the two-layers navigation. As
shown in Figure 2, we define the exploration operation involving two successive layer edges. The recommender first selects sectors on the first layer and proposes the stocks on the second layer. Formally, at step $t$, $a_t = (c_t \rightarrow s_t \rightarrow s_t')$ denotes a two-hop path rooted at the current sector $c_t$ towards proposing the stock $s_{jt}$ via the historical navigating result $s_t$, where $c_t$ and the $(s_t, s_{ti})$ are connected via the edge with the weight updated by the weight evaluator. With this exploration operations, the recommender will sequentially generate a set of results $(s_1, s_2, ..., s_t)$ by the policy $\pi$ is updated as $\max_\pi E_{a_t \sim \pi}(\sum_{i=1}^{T} \gamma^i r_i)$, where $\gamma$ is the decay factor; the expectation of $\pi$ is to make the selected stocks earning the maximum revenue as possible.

4. Experiment
In this section, we apply portfolio allocation methods (i.e., equally weighted, mean-variance, and minimum-variance) to valid the effectiveness of our proposed recommendation method. Then we analyse the performance of portfolio allocation in terms of Sharpe ratio and cumulative returns.

4.1. Parameters and environment
All the experiments are carried out on a computer with i7-5820K CPU, Nvidia GTX 1080-Ti GPU, and 8GB RAM. Our algorithm is implemented in Python 2.7 with PyTorch 1.0. For the model optimization, we perform stochastic gradient descent with the learning rate 0.05. Moreover, the smooth parameters $\alpha$, $\beta$, and $\gamma$ in the reward function Eq. (1) are set to 0.6, 0.2, and 0.2, respectively. Through cross-validation, we set the hyper-parameters in our model as follows. We set $m$ as 10 and run D-steps 200 times, G-steps 200 times. The parameters for all the baselines are set as described in the corresponding papers. The experimental data contains the extra attributes including market capitalization, dividend, price/sales, price value, price/earnings, and EBITDA. In order to analysis whether the proposed model has made effective recommendation, we consider the stocks in S&P 500 index with four periods. The four periods include bottom period (2018/11–2019/02, see A in Figure 3, middle period (2019/03–2019/09, see B in Figure 3, top period (2019/10–2020/01, see C in Figure 3 of a wave with rising trend, and top period (2020/01–2020/04, see D in Figure 3 of a wave with falling trend.

![Figure 3](image1.png)

(a) Data analysis. The price of stocks in each sector. (b) The price changes of the recommended stocks.

![Figure 4](image2.png)

(a) Experimental Results Analysis. (a) Return comparing of two portfolio method with the S&P 500 in the A period. (b) Return comparing of two portfolio method with the S&P 500 in the A period.

4.2 Experimental Results Analysis
In this section, we briefly analyze and explain the experimental results. After the learning agent
jumps from each industrial sectors to select stock based on the feedback signals from the market, StockRL selects a set of stocks. The recommended stock prices for 2018-2020 are shown in Figure 3(b). One can see that, BOND is less volatile than other stocks, which helps reduce risks.

In order to prove the effectiveness of our proposed recommendation method, we apply the portfolio allocation method to the selected stock set, and back-test the return rate in the A and C periods. Figure 3(a)(b) shows the returns of comparing two portfolio allocation methods (i.e., mean-variance and minimum-variance) with the S&P 500 Index within 12 weeks of trading. Note that A and C periods are the start period of the S&P 500 Index fall and raise. Generally, stock prices will fluctuate along with the index. If the recommended stocks get better returns in the falling and rising period, it clearly shows that the proposed method can select stocks with high-margin and low-risk.

From Figure 4(a)(b), one can observe that both of the two portfolio allocation methods get the better returns and greatly better than the revenue return of S&P 500 Index. With mean-variance, the portfolio can get 54.96% better returns at the 10th trading week in Figure 4(a), and the portfolio method with minimum-variance can gets the 23.14% more returns at the 11th trading week in Figure 4(b). Certainly, the recommended stocks can let experienced investors to get much better returns than the S&P 500 index baseline.

Reference
[1] Kewei Hou, Chen Xue, and Lu Zhang. Digesting anomalies: An investment approach. The Review of Financial Studies, 28(3):650–705, 2015.
[2] Yun-Jung Lee and Gyun Woo. Analyzing the dynamics of stock networks for recommending stock portfolio. Journal of Information Science & Engineering, 35(2), 2019.
[3] Radim Gottwald. The use of the p/e ratio to stock valuation. European Grants Projects Journals, 2012:21–24, 2012.
[4] Hrishikesh D Vinod, DF Hsu, and Y Tian. Combining multiple criterion systems for improving portfolio performance. Fordham University Department of Economics Discussion Paper, (2008-07), 2008.
[5] Sharon K Parker and Catherine G Collins. Taking stock: Integrating and differentiating multiple proactive behaviors. Journal of management, 36(3):633–662, 2010.
[6] Keith R Cridle and Andrei Y Streletski. Multiple criterion management of a sequential fishery. Annals of Operations Research, 94(1):259–273, 2000.
[7] Tomer Geva and Jacob Zahavi. Empirical evaluation of an automated intraday stock recommendation system incorporating both market data and textual news. Decision support systems, 57:212–223, 2014.
[8] Yunchuan Sun, Menting Fang, and Xinyu Wang. A novel stock recommendation system using guba sentiment analysis. Personal and Ubiquitous Computing, 22(3):575–587, 2018.
[9] Weina Wang and Krishn Kumar Mishra. A novel stock trading prediction and recommendation system. Multimedia Tools and Applications, 77(4):4203–4215, 2018.
[10] Kent Daniel and Sheridan Titman. Evidence on the characteristics of cross-sectional variation in stock returns. Princeton University Press, 2005.
[11] Richard Frankel and Charles MC Lee. Accounting valuation, market expectation, and cross-sectional stock returns. Journal of Accounting and economics, 25(3):283–319, 1998.
[12] Paul So´derlind. Predicting stock price movements: regressions versus economists. Applied Economics Letters, 17(9):869–874, 2010.
[13] C Emre Alper, Salih Fendoglu, and Burak Saltoglu. Forecasting stock market volatilities using midas regressions: An application to the emerging markets. 2008.
[14] Hongyang Yang, Xiao-Yang Liu, and Qingwei Wu. A practical machine learning approach for dynamic stock recommendation. In 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pages 1693–1697. IEEE, 2018.
[15] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Deep learning for event-driven stock prediction. In Twenty-fourth international joint conference on artificial intelligence, 2015.
[16] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. Journal of artificial intelligence research, 4:237–285, 1996.
[17] Ashvin Nair, Bob McGrew, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Overcoming exploration in reinforcement learning with demonstrations. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 6292–6299. IEEE, 2018.
[18] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering, 29(12):2724–2743, 2017.
[19] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.