



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

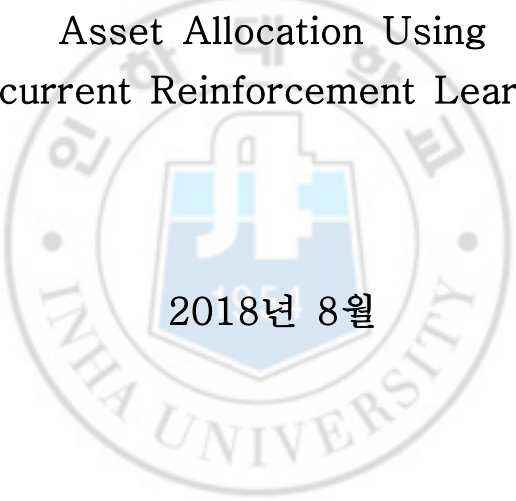
이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사학위 논문

회귀 강화학습을 이용한 포트폴리오 자산 배분의
성능 향상 연구

A Study on Performance Improvement of Portfolio
Asset Allocation Using
Recurrent Reinforcement Learning

The seal of Inha University is a circular emblem. It features a central shield with a stylized 'J' and 'H' in blue and white. The shield is set against a light blue background. The words 'INHA UNIVERSITY' are written in a circular path around the shield. The Korean text '인하대학교' is also visible around the top of the seal.

2018년 8월

인하대학교 대학원

컴퓨터공학과(컴퓨터공학전공)

강 문 주

공학석사학위 논문

회귀 강화학습을 이용한 포트폴리오 자산 배분의
성능 향상 연구

A Study on Performance Improvement of Portfolio
Asset Allocation Using
Recurrent Reinforcement Learning

2018년 8월

지도교수 이 주 홍

이 논문을 석사학위 논문으로 제출함

이 논문을 강문주의 석사학위논문으로 인정함.

2018년 월 일

주심 강상길

부심 이주홍

위원 최동완



요 약

최근, 자산 매매 및 포트폴리오에 인공지능을 활용한 연구들이 활발히 진행되고 있다. 본 논문은 기존에 제시된 회귀 강화학습(Recurrent Reinforcement Learning)을 기반으로 한 운용 모델의 성능을 향상을 위해 자산들의 미래 예측값과 인공적으로 생성한 데이터를 사용하였다. 기존의 회귀 강화학습의 성능과 예측값, 인공 생성 데이터를 함께 사용한 포트폴리오의 성능을 비교 분석하여 어떤 방법을 어떻게 사용했을 때 효과적인지 분석하였다. 실험결과 예측값을 적용한 실험에서 낮은 예측 정확도에도 불구하고 성능이 향상되는 것을 알 수 있었고, 높은 예측 정확도에서는 더욱 향상되었다. 이 결과는 자산 예측이란 요소의 예측 성공, 실패만이 중요한게 아니라 미래에 대한 불확실성을 최소화해 포트폴리오 운용에 큰 도움이 된다는 것을 확인했다. 또한, 실제 관측된 데이터와 그 데이터를 기반으로한 인공 생성 데이터를 함께 학습시켰을 때 거의 모든 경우에서 성능 향상이 되었다. 포트폴리오를 위한 회귀 강화학습에 예측값과 인공 생성 데이터가 포트폴리오 성능향상을 이끌어냄을 알 수 있었다.

Abstract

Recently, researches using artificial intelligence in asset trading and portfolio have been actively conducted. In this paper, we use the predicted values of the assets and the artificially generated data to improve the performance of the operational model based on the existing Recurrent Reinforcement Learning. The performance of the existing regression reinforcement learning, the predictive value, and the performance of the portfolio using artificial generation data were compared and analyzed. Experimental results show that the proposed method improves performance in spite of low prediction accuracy and high prediction accuracy. In addition, when the actual observed data and the artificial generation data were learned together, the performance was improved in almost all cases. We found that predictive and artificial creation data positively influenced portfolio performance in regression reinforcement learning for portfolios.

목 차

제 1 장 서 론	1
제 2 장 관련 연구	3
제 3 장 제안 모델	5
3.1 자산 예측값 데이터 생성	5
3.2 가우시안 프로세스를 이용한 데이터 생성	6
3.3 회귀 강화학습 모델 설명	6
3.4 모델의 상태(State), 행동(Action), 보상(Reward) 정의	8
제 4 장 실험	10
4.1 실험 데이터	10
4.2 실험 환경	11
4.3 최적의 Unfold 수 실험	11
4.4 최적의 상태 길이 실험	12
4.5 다른 알고리즘과의 비교	12
4.6 예측값 적용 실험	13
4.7 인공 생성 데이터 적용 실험	13
4.8 인공 생성데이터 조합에 대한 영향	14
4.9 인공 생성데이터의 비율 증가에 따른 영향	14
제 5 장 결 론	16
제 5 장 참고 문헌	17

제 1 장 서 론

최근, 인공지능분야의 기술적 발전과 고성능 계산력을 갖춘 하드웨어의 발전을 통해 인공지능이 뛰어난 성과를 거두어 다양한 분야에 적용되고 있다. 특히, 금융 분야에서 인공지능과 데이터마이닝 기술을 사용한 산업이 빠르게 발전하고 있다. 금융 분야에서 인공지능이 학습한 알고리즘을 이용해 투자 조언, 투자 결정 및 자산 운용을 할 수 있게 되었다. 인공지능이 적용되는 금융 분야의 세부적인 영역으로는 포트폴리오 최적화, 신용 등급평가, 옵션가격 결정, 주식 투자, 자산 예측 등이 있다. 그중 포트폴리오 최적화는 안정적인 수익 창출을 목표로 하는 투자 전략으로 중요한 가치를 지닌다.

포트폴리오는 마코위츠[1]에 의해 체계화된 이론으로, 주식, 채권, 상품, 통화 및 현금 등가물 등의 여러 자산에 대해 분산투자하는 것을 말한다. 포트폴리오의 목적은 위험을 줄이고 높은 수익률을 얻기 위해 다양한 종목에 대해 투자를 효율적으로 배분하는 것을 목표로 한다. 포트폴리오는 위험을 적절하게 관리하기 위해 주어진 자산배분 비중으로 일정 기간 운용한 뒤 다시 자산배분 비중을 조정하는 포트폴리오 리밸런싱(rebalancing)을 한다[2]. 포트폴리오 이론은 수익과 위험을 각각 평균과 분산으로 표현하였으며, 이러한 평균-분산(mean-variance)모형의 체계를 구축한 것을 시작으로 많은 이론 및 연구들이 진행되었다[3]. 기존의 포트폴리오 알고리즘으로는 통계적 기법, 유전자알고리즘(Genetic algorithm), 선형계획(Linear Programming)이나 비선형계획(Nonlinear Programming), 인공신경망(Artificial Neural Networks) 등의 다양한 방법들이 있다[4].

Moody[5]가 회귀 강화학습을 소개한 뒤로 많은 회귀 강화학습을 사용한 연구가 진행되었는데, 예를 들면, Carl Gold[6]은 회귀 강화학습을 단일 계층과 다중 계층으로 구현해 비교실험을 하였다. Yue Deng[7]는 딥 러닝을 통해 자산의 과거 데이터에서 특징을 추출하고, 회귀 강화학습에 입력으로 사용한 실험을 하였다. Saud Almahdi[8]는 회귀 강화학습의 목표함수로 사용된 샤프지수 대신 칼마 지수(Carmal ratio)와 스티어링 지수(stering ratio)를 사용하여 비교 실험하였다. 하지만 회귀 강화학습과 관련된 여러 기존 연구들은 회귀 강화학습의 입력으로 오직 자산들의 과거 데이터를 사용하였을 뿐 포트폴리오의 목표에 도움을 요소들에 대한 적용이 부족하다는 아쉬움을 가지고 있다.

포트폴리오의 성능을 향상에 도움을 주는 요소들은 자산들의 과거 데이터들 외에도 여러 요소들이 존재한다. 특히 자산의 주가 예측은 주식 시장에서 이익을 얻는 중요한 역할을 한다[9]. 주식 시장의 변동성으로 인해 예측된 결과가 100% 정확하지 않더

라도 투자에 도움이 된다[10]. 수익률이 높고 안정적인 종목들로 포트폴리오를 구성하는 것과 함께 좋은 예측력의 주가예측 모델을 이용하여 매매(trading)하면 포트폴리오를 통해 추구하는 목표에 근접할 수 있다[11][12]. 미래의 주가를 예측하는 것으로 불확실성을 최소화할 수 있기 때문이다[13].

또한, 실제로 관측된 데이터들만으로 훈련하여 각 자산들의 트렌드를 인지하고 포트폴리오의 목적을 위한 결정을 최적화하기에는 부족할 수 있다. 주식 데이터 특성상 초 단위, 일 단위로 많은 데이터 포인트들을 가지고 있다 하더라도, 트렌드의 관점에서 본다면 하나의 주식 데이터 포인트들은 하나의 트렌드를 나타내고 있다. 그렇다고 실제로 하나의 자산과 유사한 트렌드를 가지는 데이터를 충분히 모으는 것도 쉽지 않다. 이러한 문제들은 인공적으로 생성된 데이터를 사용한다면 해결 가능하다. 실제로 관측된 데이터와 유사한 트렌드를 가지지만 다양한 변동성을 지닌 데이터들을 생성해 견고한 포트폴리오 모델학습에 도움을 줄 수 있다.

따라서 본 논문은 안정적이면서도 수익을 최대화시키는 포트폴리오의 최적 자산 배분 비중을 생성하기 위해, 예측값과 인공 생성 데이터의 활용이 효과적일 것이라는 가설을 세웠다. 본 논문에서는 기존에 Moody가 제시한 회귀 강화학습을 사용하였고, 자산들의 과거 데이터, 자산의 미래 예측값, 인공 생성 데이터를 사용하였다. 자산의 미래 예측값은 운용 기간 동안의 예측 정확도에 따른 자산 가격의 상승, 하락에 대한 예측값을 인공적으로 생성해 사용하였다. 인공 생성 데이터는 가우시안 프로세스(Gaussian Porcess)를 사용하여 실제 관측된 데이터와 유사한 동향을 가지면서도 차이를 지닌 인공적 데이터를 생성해 사용하였다. 회귀 강화학습에 대해 자산들의 과거 데이터만을 사용한 방법과 자산 예측을 추가한 방법, 그리고 인공 생성 데이터를 기존 자산데이터와 함께 사용한 학습방법들의 비교 실험을 통해 어떠한 방법이 회귀 강화학습을 이용한 포트폴리오 운용의 성능향상에 효과적인지를 실험하였다.

본 논문은 구성은 다음과 같다. 2장에서 본 논문과 관련된 논문을 소개한다. 3장에서는 본 논문이 제안하는 모델이 서술되어 있다. 4장에서는 본 논문에서 진행한 실험이 서술되어있다. 5장에서 본 논문에 대한 결론이 서술되어있다.

제 2 장 관련 연구

Markowitz[1]는 포트폴리오를 최적화하는 평균-분산(mean-variance)모델을 소개하면서 포트폴리오 이론을 체계화하였다. Markowitz가 제시한 마코위츠 모델은 모든 투자기회 중에서 최적의 수익률, 리스크 조합을 가진 투자기회를 결정하는 이론으로, 각 종목들 간의 과거 자료, 평균 수익률, 분산만을 사용해 분산 투자 하는 이론이다. 위험의 정도인 종목 간의 분산을 최소화, 최소 기대수익률을 달성, 모든 사용 가능한 금액을 투자하는 제약조건 세가지를 가진 비선형계획모델이다[1][11].

Moody[5]는 회귀 강화학습을 사용하여 포트폴리오의 자산 할당 및 거래 시스템을 최적화하는 방법을 제시했다. 또한 Moody와 Saffell[14]은 재귀 강화학습과 Q-learning을 실제 데이터를 사용하여 비교실험을 하였고 Q-learning보다 재귀 강화학습이 더 좋은 결과를 보였다고 소개하였다.

Saud Almahdi[8]은 매매 시그널과 자산 배분 비중을 얻기 위해 Calmar Ratio를 이용한 Recurrent Reinforcement Learning Method(RRL)를 제안했다. 실험은 자주 거래되는 상장 된 펀드로 구성된 포트폴리오를 사용하여, E(MDD) 기반의 목적 함수가 이전에 제안된 RRL 목적 함수와 비교하여 우수한 수익률을 산출한 결과를 발표하였다.

Yue Deng[7]이 제시한 모델은 딥러닝과 강화 학습의 두 가지 학습 개념에서 영감을 얻어 구현했다. 제시된 모델에서 딥러닝 부분은 유익한 기능 학습을 위한 역동적인 시장 상태를 자동으로 감지한다. 그런 다음 강화학습 부분은 deep representations와 상호 작용하고 알려지지 않은 환경에서 최종 보상을 축적하기 위해 거래 의사 결정을 내렸다. 학습 시스템은 심층 구조와 반복 구조를 모두 나타내는 복잡한 신경망으로 구현했다.

박강희[11]는 포트폴리오 운용 알고리즘으로 Max-Return and Min-Risk(MRMR)을 제안하였고, 주가예측모델을 함께 활용하였다. 또한, 좋은 종목들로 포트폴리오를 구성하는 것과 함께 예측력이 좋은 주가예측 모델을 이용하면 포트폴리오를 통해 추구하는 목표에 근접할 수 있다고 주장했다[9].

Du, Xin[15]은 Q-Learning의 방법을 이용하여 거래 시스템을 구현하였다. 또한, 자산 거래 시스템을 위한 Q-Learning 과 Recurrent Reinforcement Learning 방법 중 어떤 방법의 성능이 더욱 우수한지 소개하였다. Q-learning은 모델 없는 강화학습

의 한 형태이다. 포트폴리오에 대한 결정을 내려주는 에이전트는 도메인의 지도를 만들지 않고도 행동의 결과를 경험함으로써 Markovian domains에서 최적으로 행동하는 법을 학습한다. 그러나 Q-Learning은 데이터의 잡음에 매우 민감하게 반응해 잡음이 많은 주식 데이터에 적합하지 못하다. Du, Xin[15]의 연구결과는 RRL이 Q-Learning에 보다 우수한 성능보였다.

Yang Wang[16]는 Q-Learning을 Deep Neural Network를 사용하여 만든 Deep Q-Learning을 사용하여 자산 거래 시스템에 대해 연구 하였다. Deep Q-Learning(DQN)은 Q-Learning이 가지던 데이터의 잡음에 대해 강건하지 못한 단점을 해결하였다. Yang Wang의 연구 결과에서는 DQN이 RRL에 비해서 더 좋은 성능을 보였다. 하지만 DQN은 이산형 행동 공간을 가지기 때문에 포트폴리오와 같은 다차원 행동에 대한 결정을 내리는 문제에 대해 적합하지 않다.



제 3 장 제 안 모 델

3.1 자산 예측값 데이터 생성

예측 정확도에 따른 자산 예측정보가 회귀 강화학습에 적용 시 예측 정확도에 증가에 따라 혹은 어느 정도의 예측 정확도가 포트폴리오 성능 향상에 효과적인지 실험하기 위해서 일정 예측 정확도에 따른 자산 예측 데이터를 인공적으로 생성하였다. 예측 데이터는 운용 기간 동안의 상승, 하락에 대해서 확률적으로 예측해, 운용 시작 시점 기준 대비 상승할 것이라고 예측되면 1, 하락할 것이라고 예측되면 -1로 표현하였다. 예측 성공률을 38%부터 64%까지 2% 단위로 나누어 적용하였다. 각각의 예측 정확도를 통해 생성한 데이터를 회귀 강화학습에 적용해 포트폴리오 성능 향상에 대해 실험하였다. 그림 1은 48% 예측 데이터 생성 예시이다.

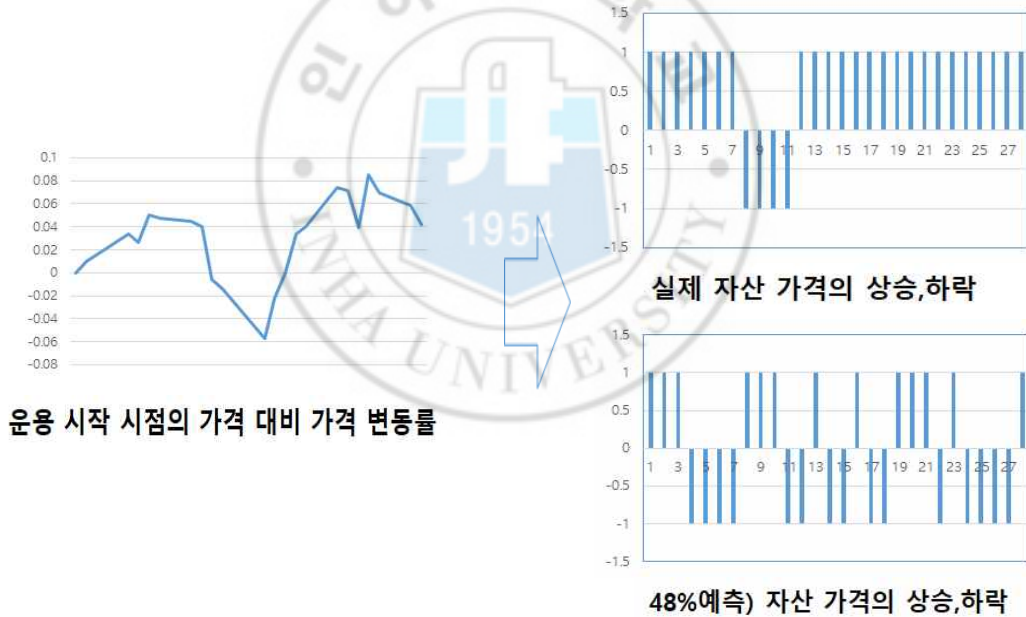


그림 1. 예측 데이터 생성 예시

3.2 가우시안 프로세스를 이용한 데이터 생성

가우시안 프로세스는(Gaussian Process)는 각 시간에서 관측된 자산의 가격으로 이루어진 훈련 데이터를 통해 공분산 함수 커널을 사용하여 함수에 대한 하나의 확률 분포를 정의한다. 커널은 보편적으로 사용하는 제곱 지수 커널을 사용하였고 인공 생성 데이터의 변동성을 부여하기 위하여 노이즈 모델을 사용하였다. 커널의 수식은 수식(1)과 같다.

$$k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2} |x_p - x_q|^2\right) + \sigma_n^2 \delta_{pq} \quad (1)$$

가우시안 프로세스의 하이퍼 파라미터중 σ_n^2 은 훈련 데이터 집합에 대한 불확실성과 관련된 파라미터이다. σ_n^2 값을 조정 하여 큰 동향을 따라가지만 각 시간별로 원본 데이터와 차이가 있는 인공 생성 데이터를 생성할 수 있다. 그림 2는 인공 생성 데이터의 과정 예시이다.

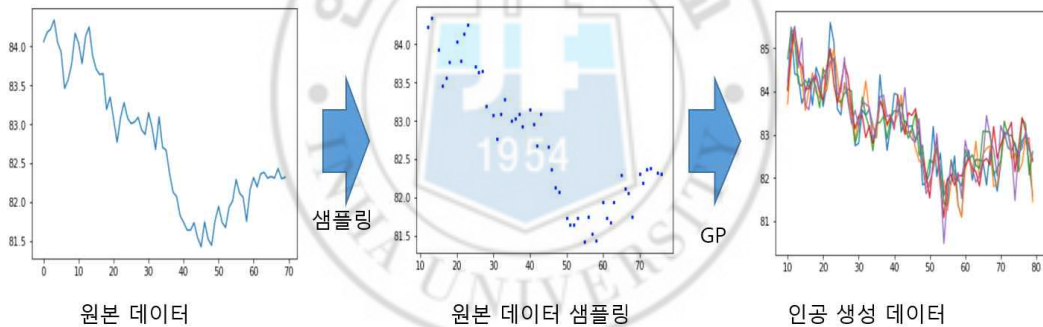


그림 2. 인공 생성 데이터 간략한 생성 과정 및 예시

3.3 회귀 강화학습 모델 설명

포트폴리오를 위한 회귀 강화학습의 목적은 환경(주식 시장)과 상호작용을 통해 보상(Reward)를 최대화하도록 행동(포트폴리오 자산 배분 비중)을 하는 모델을 학습시키는 것이다. 회귀 강화 학습은 연속적인 행동 공간을 가지므로, 다차원 행동의 행동을 선택할 수 있다. 회귀 강화학습의 주요 특징으로는 이전 자산 배분 비중에 관한 정보들을 전달받아 현시점의 입력과 상호작용하여 자산 배분 비중을 출력한다. 또한, 모델이 받는 입력과 목적 함수 설정이 자유로워 다양한 정보들을 추가하거나, 목적 함수를 다양하게 정의할 수 있다.

본 논문에서는 기존에 제시된 회귀 강화학습의 프레임워크를 Unfold된 Long short term Memory (LSTM)을 사용하여 구현하였다. LSTM은 회귀 강화학습과 같이 이전의 포트폴리오 운용에 대한 정보를 LSTM의 Hidden State와 Cell State를 통해서 이전의 정보를 전달받아 입력으로 받은 현재 시점에서의 상태와 상호작용하여 행동을 결정한다.

회귀 강화학습 모델의 훈련은 다음과 같은 순서로 진행된다. Unfold 된 LSTM은 각 시점에 대응되고, t 시점의 LSTM으로부터 자산 배분 비중 $\vec{F}_t(\theta)$ 를 얻는다. $\vec{F}_t(\theta)$ 를 통해 포트폴리오 수익률 \vec{R}_t 를 얻는다. T 시점까지의 \vec{R}_t 로 목적함수를 $U_T(\vec{R}_1, \vec{R}_2, \dots, \vec{R}_T)$ 를 구한다. 최종적으로 U_T 를 최대화하도록 LSTM의 내부 가중치 θ 를 조정한다. 수식 (2)는 U_T 를 최대화하는 미분 수식이다. 수식 (2)를 정확히 계산하고 최적화하기 위해서 LSTM의 BPTT 학습방법을 사용했다[17].

$$\frac{dU_T(\theta)}{d\theta} = \sum_{t=1}^T \frac{dU_T}{dR_t} \left\{ \frac{dR_t}{dF_t} \frac{dF_t}{d\theta} + \frac{dR_t}{dF_{t-1}} \frac{dF_{t-1}}{d\theta} \right\} \quad (2)$$

본 논문에서 U_T 는 대표적인 포트폴리오 성능 척도인 샤프 지수를 사용하였다. 그림 3은 구현한 회귀 강화학습의 모델에 대한 예시이다.

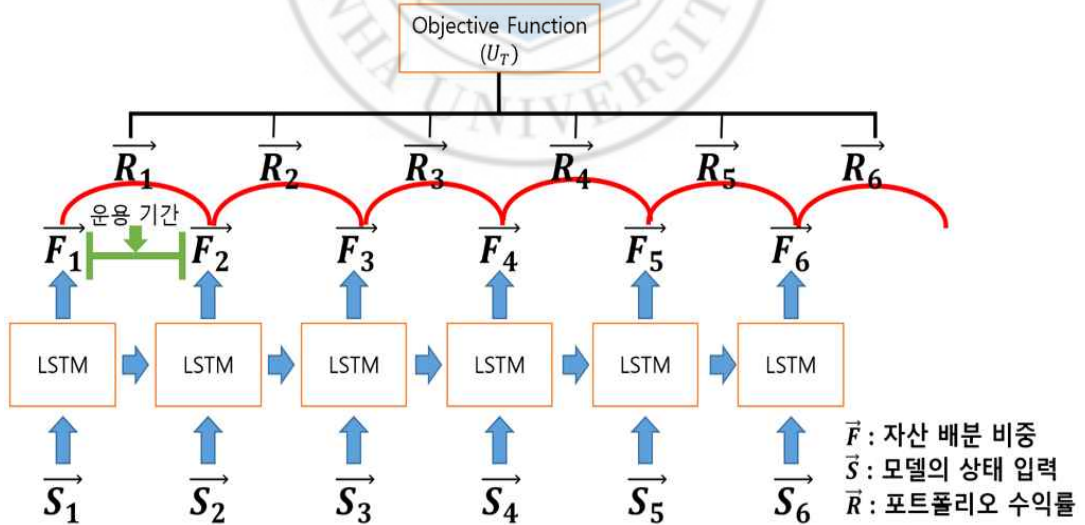


그림 3. LSTM으로 구현된 회귀 강화학습 모델

3.4 모델의 상태(State), 행동(Action), 보상(Reward) 정의

포트폴리오를 위해 구현된 회귀 강화학습을 이용한 포트폴리오 운용 에이전트(agent)는 주어진 상태(State)에 대해 행동(action)을 선택하고 보상(Reward)인 샤프지수(Sharpe ratio)를 최대화시키기 위해 인공지능망을 학습시킨다. 해당 모델의 상태, 행동, 보상에 대한 정의는 아래와 같이 정의하여 사용했다.

기호 1) 에이전트의 액션 F_t

\vec{F}_t : 시간 t 시점에서 포트폴리오의 m 개의 개별자산들에 대한 자산 배분 비중 벡터,

$$\vec{F}_t = (F_t^1, F_t^2, \dots, F_t^m) \quad (2)$$

F_t^i : 시간 t 시점에서 포트폴리오의 i 번째의 개별자산에 대한 자산 배분 비중,

$$F_t^i \geq 0, \sum_{i=1}^m F_t^i = 1 \quad (3)$$

기호 2) 에이전트의 상태 S_t 정의

\vec{S}_t : 시간 t 시점에서 포트폴리오 운용 에이전트의 상태 입력 벡터,

$$\vec{S}_t = (\vec{Sa}_t^1, \vec{Sa}_t^2, \dots, \vec{Sa}_t^m) \quad (4)$$

\vec{Sa}_t^i : Sa_t^i 는 예측값 사용 유무에 따라 두 가지 경우로 나뉜다.

1) 예측값 사용 : 시간 t 시점에서 i 번째 개별자산의 과거 l 일동안의 자산 가격 일 단위 수익률 벡터와 미래 n 일 동안의 예측값 벡터,

$$\vec{Sa}_t^i = (\vec{r}_t^i, \vec{j}_t^i) \quad (5)$$

2) 예측값 미사용 : 시간 t 시점에서 i 번째 개별자산의 과거 l 일동안의 자산 가격 일 단위 수익률 벡터,

$$\vec{Sa}_t^i = (\vec{r}_t^i) \quad (6)$$

\vec{r}_t^i : 시간 t 시점에서 과거 l 일 동안의 과거 일 단위 수익률 벡터,

$$\vec{r}_t^i = (r_{t-l+1}^i, r_{t-l+2}^i, \dots, r_t^i) \quad (7)$$

r_t^i : 시간 t 시점에서 i번째 개별자산의 일 단위 수익률,

$$r_t^i = \frac{z_t^i}{z_{t-1}^i} - 1 \quad (8)$$

z_t^i : 시간 t 시점에서 i 번째 개별자산의 자산 가격.

\vec{j}_t^i : i 번째 개별자산의 시간 t 시점에서 미래 t+n 구간 동안 예측 정확도에 따른 예측값 벡터, 각 예측값들은 상승 예측 시 1, 하락 예측 시 -1로 설정.

$$\vec{j}_t^i = (j_{t+1}^i, j_{t+2}^i, \dots, j_{t+n}^i) \quad (9)$$

기호 3) 에이전트의 보상 $U(\theta)$ 정의

$U(\theta)$: 총 운용 기간 T 시간 동안 에이전트의 행동으로 인해 발생한 포트폴리오의 수익률에 대한 샤프 지수,

$$U(\theta) = \frac{E(P_1, P_2, \dots, P_T)}{\sqrt{Var(P_1, P_2, \dots, P_T)}} \quad (10)$$

P_t : 시간 t 시점에서 포트폴리오의 수익률,

$$P_t = \sum_{i=1}^m W_t^i \cdot r_{t+1}^i \quad (11)$$

W_t^i : 시간 t 시점에서 에이전트의 행동 F_t^i 의 포트폴리오 운용에 따라 변경된 자산 배분 비중.

제 4 장 실험

4.1 실험 데이터

본 논문은 지수데이터 7개와 국내 주식 데이터 8개, 해외 주식 데이터 10개를 사용했다. 지수데이터는 NASDAQ, S&P500, KOSPI200, WTI, GOLD, Dollar/Won, US High Yield를 사용했다. 국내 주식 데이터는 Samsung, POSCO, SK Hynix, LG, CJ CGV, KIA, NAVER, JYP Entertainment를 사용했다. 해외 주식데이터는 APA, AVA, BMY, CHK, CPB, CPK, ETFC, DGX, FCX, MS를 사용했다. 5개의 자산들의 집합을 만들었다. 자산들의 집합은 표1과 같다. 자산 데이터는 모두 일 단위의 데이터를 사용했다. 훈련 데이터의 기간은 2012년 10월 17일부터 2014년 1월 3일까지로 설정하였고 테스트 데이터의 기간은 2014년 1월 6일부터 2015년 3월 26일까지로 설정하였다. 운용 일자는 20일로 고정하였다. 표 2는 각 자산들의 테스트 기간 샤프 지수이고, 표 3은 포트폴리오 구성 자산 집합의 테스트 기간 평균 샤프 지수이다.

포트폴리오 구성 자산 집합					
집합 1	S&P500	KOSPI200	Samsung	AVA	BMY
집합 2	NASDAQ	GOLD	Sk hynix	WTI	LG
집합 3	US High Yield	D/W	Posco	CHK	ETFC
집합 4	MS	NAVER	FCX	CPB	DGX
집합 5	CJCGV	CPK	APA	KIA	JYP

표 1. 포트폴리오 구성 자산 집합

자산 이름	샤프 지수	자산 이름	샤프 지수
S&P500	0.074704832	LG	0.079383
NASDAQ	0.078354463	Posco	-0.01419
KOSPI200	0.000641	Samsung	0.031009
US High Yield	0.057435	SK Hynix	0.040773
GOLD	-0.059	NAVER	0.022871
WTI	-0.02167	KIA	0.005339
Dollar/Won	0.020966	JYP	-0.05148
CHK	-0.00364	CJCGV	0.018301
FCX	-0.08891	BMY	0.057119
ETFC	0.05378	AVA	0.082289
DGX	0.112974	MS	0.063043
CPK	0.084346	APA	-0.02739
CPB	0.081915		

표 2. 자산들의 테스트 기간 샤프 지수

포트폴리오 구성 자산 집합	평균 샤프 지수
집합 1	0.049152
집합 2	0.023568
집합 3	0.022871
집합 4	0.038378
집합 5	0.005824

표 3. 포트폴리오 구성 자산 집합의
테스트 기간 평균 샤프 지수

4.2 실험 환경

실험환경은 Intel Xeon 3.50Ghz CPU, 128G DRAM과 NVIDIA GTX 1080을 사용하여 진행하였다. 실험 프로그램은 Python과 Tensorflow를 사용하여 구현하였다.

4.3 최적의 Unfold 수 실험

회귀 강화학습을 Unfold한 LSTM으로 구현하여 포트폴리오 성능이 가장 뛰어난 Unfold 수를 찾는 실험을 수행했다. Unfold만을 변수로 두고, 상태 길이는 현 시점 과거 60일로 고정해 실험했다. Unfold 수는 3, 5, 8, 10, 12 설정하였다. 실험 결과 Unfold 3에서 가장 높은 성능을 보였으며, 결과는 그림 4와 같다.

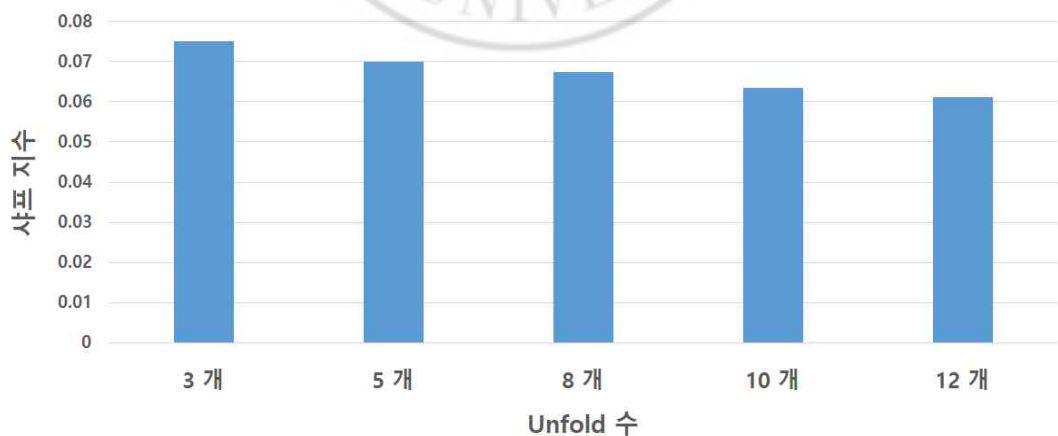


그림 4. Unfold 수에 따른 성능 그래프

4.4 최적의 상태 길이 실험

4.3 에서의 결과대로 Unfold를 3으로 고정한 다음, 최적의 상태 길이를 찾기위해 상태의 길이를 현 시점에서 과거 20, 40, 60, 80, 100, 120일로 설정하여 실험하였다. 실험 결과 상태 길이가 20일에서 가장 좋은 성능을 보였다. 결과는 그림 5와 같다.

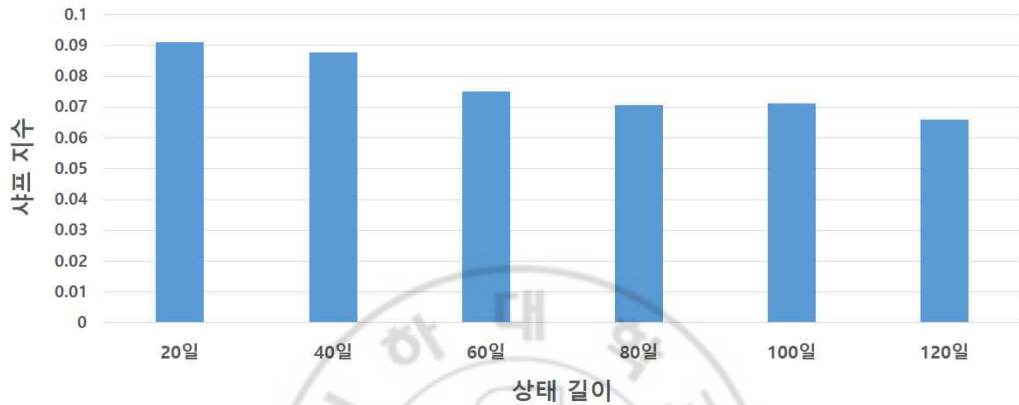


그림 5. 상태 길이에 따른 성능 그래프

4.5 다른 알고리즘과의 비교

회귀 강화학습의 기본적인 성능을 분석하기 위하여, 4.3과 4.4에서 얻은 최적의 Unfold 수와 상태 길이를 적용하여 Markowitz 모델과 1/N 포트폴리오와의 성능을 비교하였다. 실험 결과 회귀 강화학습이 Markowitz 모델과 1/N 포트폴리오보다 높은 성능을 보였다. 결과는 그림 6와 같다.

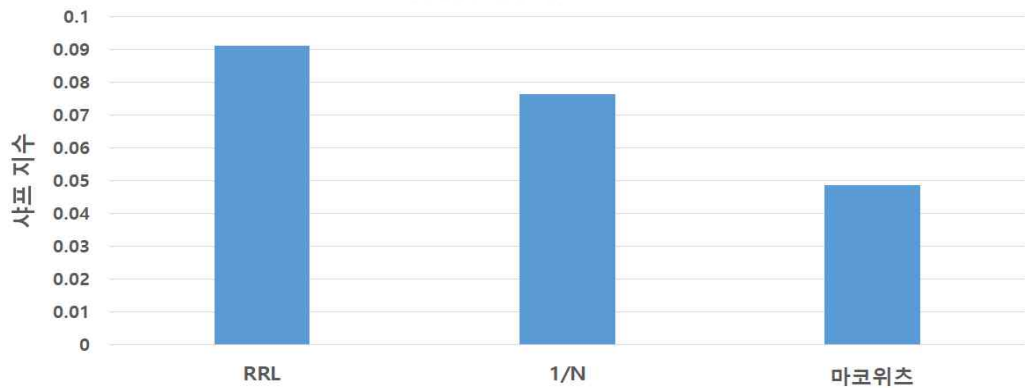


그림 6. Markowitz와 1/N 포트폴리오와의 성능 비교 그래프

4.6 예측값 적용 실험

예측값을 회귀 강화학습에 적용시 성능 향상을 실험하였다. Unfold는 3, 상태 길이는 20으로 설정하였으며 예측 정확도를 38%~64%로 2% 단위로 나누어 실험하였다. 예측값을 사용하지않고 자산의 과거 데이터들만을 사용한 기본 회귀 강화학습의 샤프 지수와의 비를 계산하여 그래프로 표현하였다. 실험 결과 예측 정확도가 높으면 매우 우수한 성능을 보였을 뿐만 아니라 예측 정확도가 높지 않더라도 유의미한 상승을 보였다. 그림 7은 실험 결과이다.

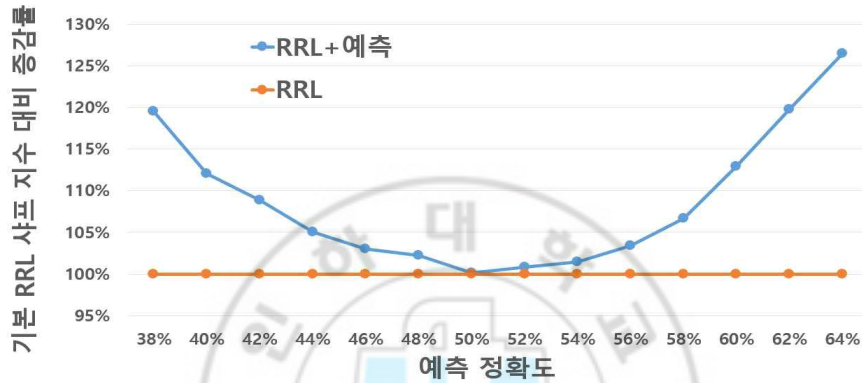


그림 7. 예측 정확도에 따른 성능 비교 그래프

4.7 인공 생성 데이터 적용 실험

가우시안 프로세스를 하이퍼 파라미터인 σ_n^2 을 여러 단계로 나누어 설정해 원본 데이터들과 함께 사용하여 학습시켰다. 원본 데이터,인공 생성 데이터의 비율을 1 : 1로 했다. 결과는 기본 회귀 강화학습의 샤프지수와의 비를 계산하여 그래프로 표현하였다. 실험 결과는 그림 8과 같다.

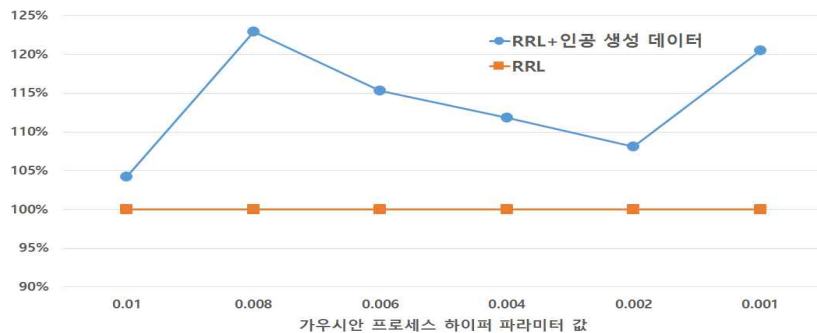


그림 8. 파라미터에 따른 인공 생성 데이터 적용 성능 결과 그래프
(세로축은 기본 RRL 샤프 지수 대비 상승률)

4.8 인공 생성 데이터 조합에 대한 영향

4.7에서는 원본 데이터와 인공 생성 데이터의 비율이 1:1이었지만 4.9에서는 원본 데이터와 2개의 인공 생성 데이터의 조합으로 실험을 진행하였다. 따라서 원본 데이터, 인공 생성 데이터 1, 인공 생성 데이터 2를 1 : 1 : 1 비율로 설정하여 진행했고 실험결과는 표4 와 같다. 결과는 기본 회귀 강화학습의 샤프지수 비를 계산하여 그래프로 표현하였다.

0.01	120%	121%	121%	115%	111%	
0.008	119%	133%	125%	116%		
0.006	127%	122%	110%			
0.004	120%	116%				
0.002	112%					
0.001						
σ_n^2	0.001	0.002	0.004	0.006	0.008	0.01

표4. 두 종류의 인공 생성 데이터의 조합에 따른 성능 결과 표
(가로축: 인공 생성 데이터 1의 파라미터, 세로축: 인공 생성 데이터 2의 파라미터)

4.9 인공 생성 데이터의 비율 증가에 따른 영향

4.7와 4.8에서 찾은 최고의 성능을 보인 경우에 대해서 원본 데이터와 인공 생성 데이터의 비율을 조정하여 실험하였다. 인공 생성 데이터 적용 방법 중 최고의 성능을 보인 경우는 원본 데이터,파라미터1= 0.002, 파라미터2=0.008 일 때이다. 이 경우에 대해 인공 생성 데이터의 비율을 증가시켜 얼마나 성능 향상에 영향을 주는 지 실험하였다. 표5에서 인공 생성 데이터의 비율을 나타내고 있고 실험결과는 그림9와 같다. 결과는 기본 회귀 강화학습의 샤프지수와의 비를 계산하여 그래프로 표현하였다.

	Set1	Set 2	Set 3	Set 4	Set 5	Set 6	Set 7	Set 8	Set 9	Set10
원본 데이터	1	1	1	1	1	1	1	1	1	1
인공 생성 데이터 1	1	2	3	4	5	6	7	8	9	10
인공 생성 데이터 2	1	2	3	4	5	6	7	8	9	10

표5. 인공 생성 비율표

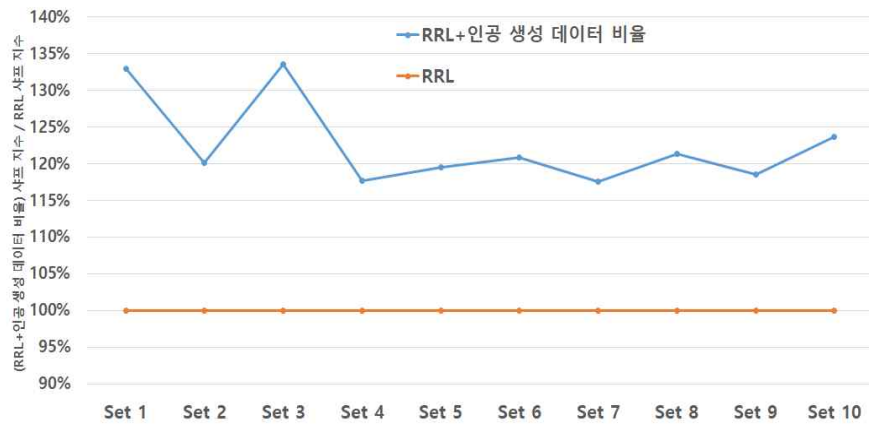


그림9. 인공 생성 데이터 비율에 따른 실험 결과



제 5 장 결 론

최적의 파라미터를 찾은 회귀 강화학습이 포트폴리오의 대표적인 모델인 Markowitz와 1/N 포트폴리오와 비교하여 더 우수한 성능을 보임을 확인 할 수 있었다. 회귀 강화학습에 예측값과 인공 생성 데이터라는 포트폴리오 성능 향상에 도움을 줄 수 있는 요소들을 함께 적용하면 성능의 향상에 도움이 될 것이라는 가정을 실험을 통해 증명하였다. 실제 실험 결과 예측값은 거의 모든 구간에서 기본적인 회귀 강화학습보다 우수한 성능을 보였고 특이점은 높은 예측 정확도에서 아주 우수한 성능을 보인 것만 아니라 낮은 예측 정확도에서도 성능이 향상된 결과를 보였다. 예측에 대한 실험을 통해 미래에 대한 정확한 예측성이 중요한 것이 아니라 일정한 예측 정확도와 미래에 대한 불확실성을 줄이는 것이 포트폴리오 성능 향상에 도움이 됨을 알 수 있었다.

또한, 인공 생성 데이터를 함께 훈련한 실험에서도 모든 경우에서 기본 회귀 강화학습보다 높은 성능을 보였고 최대 약 34%의 증가를 보였다. 실험을 통해 포트폴리오를 위한 회귀 강화학습에 예측값과 인공 생성 데이터의 적용이 성능향상을 끌어냄을 알 수 있었다.

본 논문은 실험을 통해 회귀 강화학습에 예측값과 인공 생성 데이터의 사용이 포트폴리오 성능 향상에 도움이 된다는 결론을 도출하였고, 적용 방법 및 성능 향상 방법의 가이드라인을 제시했다.

제 5 장 참고 문헌

- [1] Markowitz, H., 1992. Portfolio selection. Journal of Finance 7, 77-91.
- [2] 박광수. "한국에서의 포트폴리오 리밸런싱에 관한 실증적 연구." Financial Planning Review 2.1 (2009):
- [3] 황승규, 임형준, and 유시용. "논문 (論文): 기대수익률의 추정에 의한 최적자산배분에 관한 연구-평균-분산 모형과 평균-VAR 모형을 중심으로." 재정정책논집 11.1 (2009): 27-57.
- [4] Byun, H. W., Song, C. W., Han, S. K., Lee, T. K. and Oh, K. J. (2009). Using genetic algorithms to develop volatility index-assisted hierarchical portfolio optimization. Journal of the Korean Data & Information Science Society, 20, 467-478.
- [5] Moody J et al 1997 Performance functions and reinforcement learning for trading systems and portfolios J. Forecasting at press
- [6] Gold, Carl. "FX trading via recurrent reinforcement learning." Computational intelligence for financial engineering, 2003. proceedings. 2003 IEEE international conference on. IEEE, 2003.
- [7] Deng, Y., Bao, F., Kong, Y., Ren, Z., and Dai, Q. (2016). Deep direct reinforcement learning for financial signal representation and trading. IEEE Transactions on Neural Networks and Learning Systems.
- [8] Almahdi, S., & Yang, S. Y. (2017). "An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown", Expert Systems with Applications, 87, 267-279.
- [9] Jain, Vikalp Ravi, Manisha Gupta, and Raj Mohan Singh. "Analysis and Prediction of Individual Stock Prices of Financial Sector Companies in

NIFTY50." International Journal of Information Engineering and Electronic Business 10.2 (2018): 33.

[10] Mohapatra, Avilasa, et al. "Applications of neural network based methods on stock market prediction: survey." International Journal of Engineering and Technology (UAE) 7.2.6 (2018): 71-76.

[11] 박강희, 신현정, “포트폴리오 최적화와 추가예측을 이용한 투자 모형”, 대한 산업공학회지 , 제39권, 제6호(2013), pp.535-545.

[12] Guresen, Erkam, Gulgun Kayakutlu, and Tugrul U. Daim. "Using artificial neural network models in stock market index prediction." Expert Systems with Applications 38.8 (2011): 10389-10397.

[13] S. P. Mahasagara, A. Alamsyah and B. Rikumahu, "Indonesia infrastructure and consumer stock portfolio prediction using artificial neural network backpropagation," 2017 5th International Conference on Information and Communication Technology (ICoIC7), Melaka, 2017, pp. 1-4.

[14] Moody, J. , & Saffell, M. (2001). Learning to trade via direct reinforcement. IEEE Transactions on Neural Networks, 12 (4), 875-889 .

[15] Du, Xin, Jinjian Zhai, and Koupin Lv. "Algorithm Trading using Q-Learning and Recurrent Reinforcement Learning." positions 1 (2016): 1.

[16] Wang, Y., Wang, D., Zhang, S., Feng, Y., Li, S., & Zhou, Q. Deep Q-trading.

[17] Moody, J. , Wu, L. , Liao, Y. , & Saffell, M. (1998). Performance functions and reinforcement learning for trading systems and portfolios. Science, 17 (February 1997), 441-470 .