

# 네이버 주가분석 5주차 발표자료 (논문 리뷰 중심)

2020.09.02

이문형

# 목차

- **Introduction**

- A Survey
- 금융 데이터로 딥러닝을 할 경우 문제점

- **Literature Review**

- Forecasting of Stock Prices Using Brownian Motion—Monte Carlo Simulation
- Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning

- **Case Study (LSTM + 강화학습 policy gradient)**

- 2019 국내 빅데이터 연합동아리 BOAZ BIGDATA CONFERENCE
- 2019 대한산업공학회 대학생 프로젝트 경진대회

# **1. Introduction**

Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020).  
**Deep learning for financial applications: A survey.**  
Applied Soft Computing, 106384.

✓ 연구 트렌드

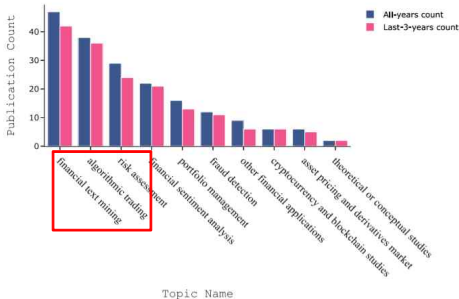


Fig. 8. The histogram of publication count in topics.

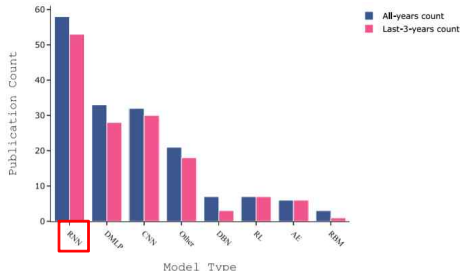


Fig. 9. The histogram of publication count in model types.

Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020).  
**Deep learning for financial applications: A survey.**  
 Applied Soft Computing, 106384.

✓ 연구 트렌드

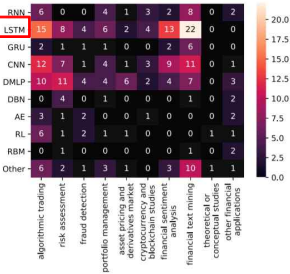


Fig. 10. Topic-model heatmap.

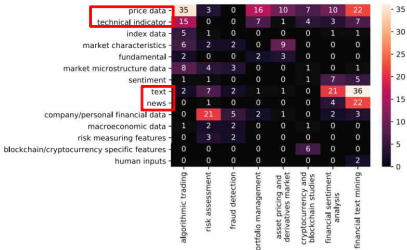


Fig. 11. Topic-feature heatmap.

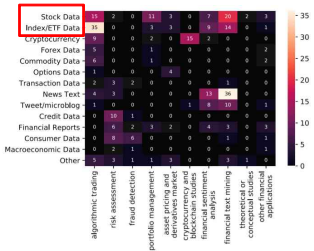


Fig. 12. Topic-dataset heatmap.

Jiang, W. (2020).

## Applications of deep learning in stock market prediction: recent progress.

arXiv preprint arXiv:2003.01859.

### ✓ 연구 트렌드

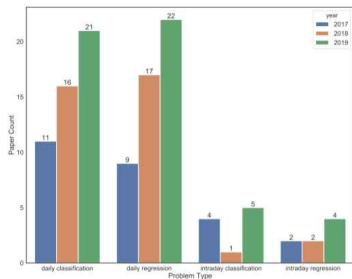


Figure 1: The paper count of different problem types.

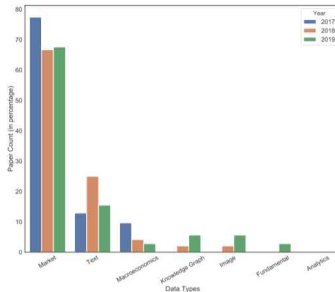


Figure 3: The usage of different raw data types.

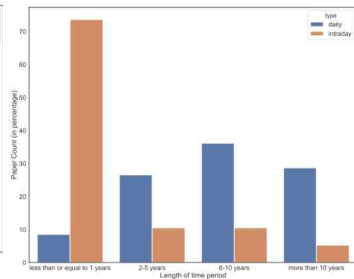


Figure 4: The distribution of data length.

Jiang, W. (2020).

## Applications of deep learning in stock market prediction: recent progress.

arXiv preprint arXiv:2003.01859.

### ✓ 연구 트렌드

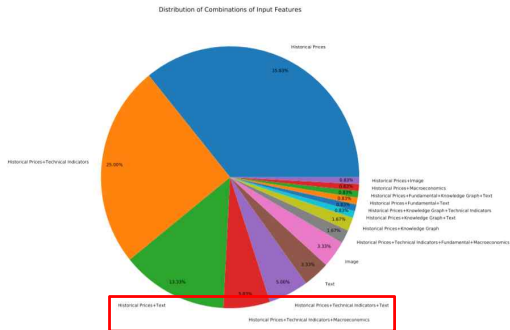


Figure 5: Distribution of combinations of input features.

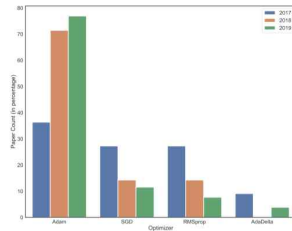


Figure 8: The usage of different optimizers.

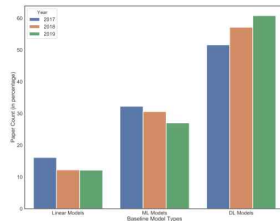


Figure 9: The usage of different baselines.

Jiang, W. (2020).

## Applications of deep learning in stock market prediction: recent progress.

arXiv preprint arXiv:2003.01859.

### ✓ 연구 트렌드

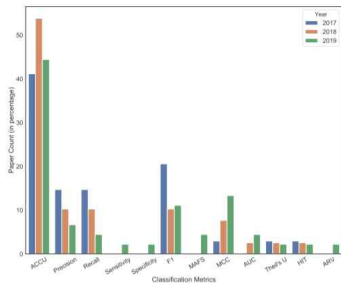


Figure 10: The usage of different classification metrics.

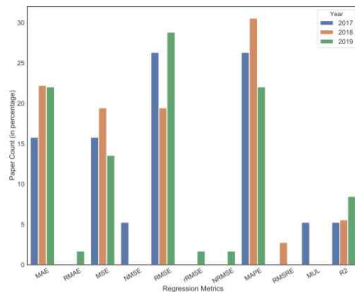


Figure 11: The usage of different regression metrics.



## 금융 데이터로 딥러닝을 할 경우 문제점 (문효준 크래프트테크놀로지스, Naver 테크톡)

### ✓ 문제점 1 : 시계열 Feature 자체의 Noise

- 딥러닝이 포착할 수 있는 정보량보다 노이즈가 더 큰 경우가 많다.
- 따라서, LSTM을 사용했을 때 단순 AR 모형처럼 오른쪽으로 lagging이 있는 형태로 모델링이 된다.

### ✓ 해결방안 : Time-series denoising

#### 1) Moving average, EMA

-> smoothing은 되나 여전히 lagging 발생

2) Bilateral filter : feature와 time에 kernel을 적용한 시계열 feature의 weighted sum, 딥러닝을 사용할 때에 batch 단위 (즉 sequence 단위)로 값이 주어지므로 사용할 수 있는 filtering 기법

-> smoothing parameter 조절의 문제

#### 3) CNN stacked autoencoder 기반 denoising module

Input - Autoencoder(CNN기반) - feature extraction network - output

## 금융 데이터로 딥러닝을 할 경우 문제점 (문효준 크래프트테크놀로지스, Naver 테크톡)

### ✓ 문제점 2 : 시계열 Feature 종류 대비 짧은 시계열 길이

- GAN을 통한 데이터 생성 -> 개별적인 생성은 가능하지만, 전체 시계열의 상관성을 고려한 생성은 어려움

### ✓ 해결방안 : 도메인 지식 기반 feature engineering & selection

- 기존 쿼트가 모델을 만드는 방식은 경제적 함의점을 도출하여 이를 모델 포트폴리오에 반영하는 형식

- 이를 차용하여 feature engineering을 통해 단순 선형 모델에서 딥러닝을 통해 최적의 함수 추정  
- 팩터 모델, 자산 배분 모델 등에서 매우 잘 작동하는 것을 확인함

ex) e.g. Factor investing : 주식의 Quality, Size, Value, Momentum, Low Risk 등의 factor 별 투자 방식

- 논문을 리서치해서 직관적 사고방식을 모사

- 시장 유동성이 높으면 모멘텀 수익률이 높아짐

- 경제 활동이 많을 때, 통화 확장정책 시기, 통화 공급이 많을 때에는 가치주 수익률이 높아짐

- 금리 인상시기, 경제 확장기에는 성장주 대비 가치주 수익률 높아짐

- 경제 침체기에는 부채비율이 낮거나 기업가치 좋은 주식 수익률이 높아짐

-> 이를 반영하여 factor에 weight를 조정하는 modeling 사용. 이 때 유동성을 캡처할 수 있는 하위 feature들로부터 extraction하는 network와 이를 기반으로 수익률을 학습하는 network를 따로 운용.

## 금융 데이터로 딥러닝을 할 경우 문제점 (문효준 크래프트테크놀로지스, Naver 테크톡)

✓ 문제점 3 : 문제점 1, 2로 인한 오버피팅 문제

✓ 해결방안 :

- 1) Asynchronous Multi Network Learning : 여러 개의 network를 만들어서 경쟁시키는 구조
- 2) Bayesian Inference : validation에서도 관찰이 불가능하다면, 오버피팅보다는 uncertainty quantification 필요

### 1. Monte Carlo Dropout

[Gal, Y., & Ghahramani, Z. (2016, June). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In international conference on machine learning (pp. 1050-1059)]

### 2. Monte Carlo Batch Normalization

[Teye, M., Azizpour, H., & Smith, K. (2018). Bayesian uncertainty estimation for batch normalized deep networks. *arXiv preprint arXiv:1802.06455*]

-> Tau, Dropout rate, Activation에 따른 영향도 높다는 단점

3. Deep learning regression + Gaussian Process regression이 성능이 가장 좋았음.

- 선지도학습 후 마지막 FC 전의 feature를 가지고 GPR 학습

-> 선형 독립적으로 노드들이 representation learning이 잘 됐다는 가정 하에 효과적인 GPR 학습이 가능

## **2. Literature Review**

Estember, R. D., & Maraña, M. J. R. (2016, March).

## 1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

### ✓ purpose

- Geometric Brownian Motion method를 사용해서 주가를 예측함

### ✓ data

- October 2014 to April 2015
- Philippine Stock Exchange에 속하는 6개 기업의 주가

### ✓ method

- Geometric Brownian Motion
- Artificial Neural Network (비교 모델)

Estember, R. D., & Maraña, M. J. R. (2016, March).

## 1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

### ✓ background

#### 랜덤워크와 브라운운동의 수학적 표현

- 동전의 앞면이 나오면 +1, 뒷면이 나오면 -1 을 계속 누적함. → 여러 가지 흥미로운 특성이 나타남. → 주가 모형의 기초

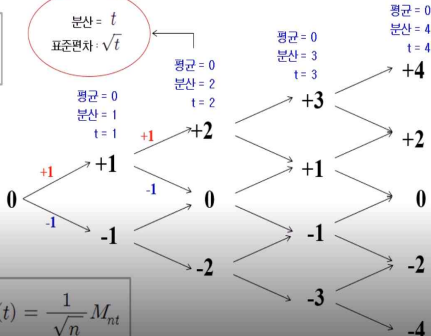
#### 랜덤워크의 수학적 정의

$$X_j = \begin{cases} +1, & \text{if } H \\ -1, & \text{if } T \end{cases}$$

$$M_k = \sum_{j=1}^k X_j,$$

$$k = 1, 2, 3, \dots, M_0 = 0$$

$$\begin{aligned} \text{분산} &= t \\ \text{표준편차} &= \sqrt{t} \end{aligned}$$



$$2) E(M_k) = 0$$

$$Var(X_j) = 1$$

$$k_0 < k_1 < k_2 \dots < k_m \text{ 일 때}$$

$$M_{k_{i+1}} - M_{k_i} = \sum_{j=k_i+1}^{k_{i+1}} X_j$$

$$3) Var(M_{k_{i+1}} - M_{k_i}) = \sum_{j=k_i+1}^{k_{i+1}} Var(X_j) = \sum_{j=k_i+1}^{k_{i+1}} 1 = k_{i+1} - k_i$$

$$6) E(W^{(n)}(t_j) - W^{(n)}(t_i)) = 0$$

$$7) Var(W^{(n)}(t_j) - W^{(n)}(t_i)) = t_j - t_i = \Delta t$$

$$8) \Delta W^{(n)}(t) \sim N(0, (\sqrt{\Delta t})^2)$$

$$9) \Delta W^{(n)}(t) \sim N(0, 1^2 (\sqrt{\Delta t})^2)$$

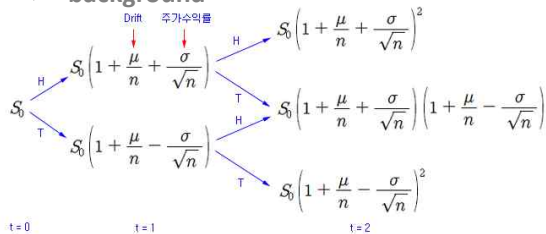
#### 브라운운동의 수학적 정의

$$W(t) = M_{nt} \xrightarrow{\text{크기 조정}} W(t) = \frac{1}{\sqrt{n}} M_{nt}$$

## 1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

### ✓ background



$$1) \quad S_t = S_0 \left(1 + \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}}\right)^H \left(1 + \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}}\right)^T$$

↑ 앞면에 나온 횟수    ↑ 뒷면에 나온 횟수

$$2) \quad nt = H + T \quad \leftarrow \text{총 시행 횟수}$$

$$3) \quad M_{nt} = H - T \quad \leftarrow \text{랜덤워크} \quad M_{nt} = M \quad \text{로 표기}$$

$$4) \quad H = \frac{1}{2}(nt + M) \quad T = \frac{1}{2}(nt - M)$$

$$5) \quad S_t = S_0 \left(1 + \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}}\right)^{\frac{1}{2}(nt + M)} \left(1 + \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}}\right)^{\frac{1}{2}(nt - M)}$$

- Drift = t기간 까지의 무위험 수익률, 인플레이션 등 표준편차 = Volatility (Drift와 수익률은 모두 1보다 작은 수)
- Drift와 Volatility(주가 수익률)를 전체 기간의 단위로 정의 (연간 단위)
- 매 기마다 적용되는 Drift와 표준편차는  $1/n$ ,  $1/\sqrt{n}$ 을 적용 (일간 단위)
- 매 기에 적용되는 주가 수익률 (표준편차)은 모두 동일한 것으로 가정 (불합리하지만, 문제를 단순화 하기 위한 가정임)
- 식 4) = 식 2)와 식 3)의 연립방정식
- 식 5) = 식 4)를 식 1)에 대입

Estember, R. D., & Maraña, M. J. R. (2016, March).

## 1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

$$6) \ln S_t = \ln S_0 + \frac{1}{2}(nt + M) \ln \left( 1 + \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}} \right) + \frac{1}{2}(nt - M) \ln \left( 1 + \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}} \right)$$

$$9) \ln S_t = \ln S_0 + \mu t - \frac{\sigma^2}{2}t + \frac{M}{\sqrt{n}}\sigma - \frac{\mu^2 t}{2n} - \frac{\mu\sigma}{n} \frac{M}{\sqrt{n}}$$

$$7) \ln \left( 1 + \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}} \right) \approx \left( \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}} \right) - \frac{1}{2} \left( \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}} \right)^2$$

$$n \rightarrow \infty : \frac{M}{\sqrt{n}} = W_t, \quad \frac{\mu^2 t}{2n} = 0, \quad \frac{\mu\sigma}{n} = 0 \quad 12) \quad \frac{S_t}{S_0} = e^{\sigma W_t + \left( \mu - \frac{1}{2}\sigma^2 \right)t}$$

$$= \frac{\mu}{n} + \frac{\sigma}{\sqrt{n}} - \frac{\mu^2}{2n^2} - \frac{\mu\sigma}{n\sqrt{n}} - \frac{\sigma^2}{2n}$$

$$10) \ln S_t = \ln S_0 + \mu t - \frac{\sigma^2}{2}t + \sigma W_t$$

$$13) \ln \left( \frac{S_t}{S_0} \right) = \sigma W_t + \left( \mu - \frac{1}{2}\sigma^2 \right)t$$

$$8) \ln \left( 1 + \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}} \right) \approx \left( \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}} \right) - \frac{1}{2} \left( \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}} \right)^2$$

$$= \frac{\mu}{n} - \frac{\sigma}{\sqrt{n}} - \frac{\mu^2}{2n^2} + \frac{\mu\sigma}{n\sqrt{n}} - \frac{\sigma^2}{2n}$$

$$11) S_t = S_0 e^{\sigma W_t + \left( \mu - \frac{\sigma^2}{2} \right)t}$$

← 가하브라운운동식 (GBM)

$$14) E \left[ \ln \left( \frac{S_t}{S_0} \right) \right] = \left( \mu - \frac{1}{2}\sigma^2 \right)t \quad \leftarrow \sigma E(W_t) = 0$$

- 식 6) = 식 5)에 자연로그를 취함, 식 7), 식 8) = 식 6)의 로그 항에 대한 테일러 급수식 적용(2차항 까지)
- 식 9) = 식 7)과 식 8)을 다시 식 6)에 대입하고, 전개후 정리
- 식 10) = 식 9)에서  $n \rightarrow$  무한대 (브라운 운동), 식 13) = 주가의 로그 수익률(+/- 가능), 식 14) = 식 13)의 평균
- 식 11)의 GBM 주가 모형은 불확실성인 위너과정을 ( $W_t$ ) 포함하고 있기 때문에, 결정론적인 식이 아니라 확률론적인 식
- 식 11)은 매 기마다 표준편차 (변동성)가 일정하게 적용되었다는 문제점
- 하지만, 단순한 연속적인 주가 모형으로, 이용하기 편리하기 때문에 금융 분야에서 널리 사용
- (ex, 이 식을 이용하여 몬테카를로 주가 시뮬레이션을 만들어서 금융 상품을 평가(설계) 하는 등에 활용)



Estember, R. D., & Maraña, M. J. R. (2016, March).

## 1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

### ✓ method & result

$$\text{Daily Rate of Return} = \frac{\text{Annual Rate of Return}}{\text{No. of trading days in a year}}$$

$$\text{Daily Volatility} = \frac{\text{Annual Volatility}}{\sqrt{\text{No. of trading days in a year}}} \quad \text{MAPE} = \left( \frac{1}{n} \sum \frac{|\text{Actual} - \text{Forecast}|}{|\text{Actual}|} \right) \times 100\%$$

$$\text{Average Drift} = \text{Daily Rate of Return} - 0.5 \times \text{Daily Volatility}^2$$

#### D. Statistical Test

The t- test is used to determine significance between two means. The formula is shown below:

$$t_0 = \frac{\bar{y} - \mu_0}{s/\sqrt{n}} \quad (6)$$

- Table 2 : draft, volatility를 상수가 아닌 변수로 사용 (10-Day Moving Variable을 사용)

- Tabl2 3 : 단기 예측 - GBM의 예측오차가 ANN보다 낮음

TABLE II. GEOMETRIC BROWNIAN MOTION IMPROVEMENT STATISTICS

| Number of Days Volatility and Drift Moved | Average Percentage Error | Standard Deviation of the Error |
|---|--------------------------|---------------------------------|
| Normal (Constant Volatility and Drift)    | 10.70                    | 0.05935                         |
| 5-Day Moving Variables                    | 10.51                    | 0.44532                         |
| 10 -Day Moving Variables                  | 8.37                     | 0.05173                         |
| 15-Day Moving Variables                   | 9.48                     | 0.05435                         |
| 20-Day Moving Variables                   | 11.50                    | 0.06965                         |
| 25-Day Moving Variables                   | 15.78                    | 0.07328                         |

TABLE III. SAMPLE ACTUAL AND FORECAST STOCK PRICES OF AEV USING GBM AND ANN METHODS

| Date                                       | Actual Stock Price (PhP) | Forecast Stock Price (PhP) |            | Percentage Error |            |
|--|--------------------------|----------------------------|------------|------------------|------------|
|  |                          | GBM Method                 | ANN Method | GBM Method       | ANN Method |
| Oct 29, 2013                               | 601.50                   | 604.3172                   | 607.8357   | 0.47             | 1.05       |
| Oct 30, 2013                               | 601.50                   | 608.2000                   | 620.6611   | 1.11             | 3.19       |
| Oct 31, 2013                               | 603.00                   | 579.9872                   | 624.7586   | 3.82             | 3.61       |
| Nov 4, 2013                                | 599.5                    | 590.6060                   | 623.0708   | 1.48             | 3.93       |
| Nov 5, 2013                                | 597.00                   | 585.9534                   | 622.3937   | 1.85             | 4.25       |
| Nov 6, 2013                                | 589.50                   | 590.0566                   | 628.5628   | 0.09             | 6.63       |
| Nov 7, 2013                                | 578.00                   | 591.5019                   | 633.1355   | 2.34             | 9.54       |
| Nov 8, 2013                                | 592.50                   | 601.3716                   | 633.7067   | 1.50             | 6.95       |
| Average Percentage Error                   |                          |                            |            | 1.58             | 4.89       |
| Standard Deviation of the Percentage Error |                          |                            |            | 12.22            | 34.53      |

1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia (pp. 704-713).

✓ result

- Table 4 : 장기간 예측에서, GBM이 더 예측오차가 낮고, 표준편차는 큰 차이가 나지 않는다(p-value=0.9363)
- Table 5 : GBM이 효과적, 정확도가 높다(93.79%>91.17%, p-value=0.0204)

| Company                    | Standard Deviation of the Percentage Error |            |
|----------------------------|--|------------|
|                            | GBM Method                                 | ANN Method |
| 1. MPI                     | 0.34                                       | 0.12       |
| 2. DMC                     | 3.19                                       | 3.07       |
| 3. JGS                     | 5.83                                       | 7.89       |
| 4. AEV                     | 3.74                                       | 3.44       |
| 5. AC                      | 0.83                                       | 1.72       |
| 6. GTCAP                   | 7.89                                       | 5.26       |
| Average Standard Deviation | 3.64                                       | 3.58       |
| p - value (2 – tail)       | 0.9363                                     |            |

Table 4.

TABLE V. AVERAGE THREE-YEAR FORECAST AVERAGE PERCENTAGE ERROR OF GBM AND ANN METHODS

| Forecast Year        | Average Percentage Error |            |
|----------------------|--------------------------|------------|
|                      | GBM Method               | ANN Method |
| 1 <sup>st</sup> Year | 6.42                     | 7.97       |
| 2 <sup>nd</sup> Year | 7.54                     | 9.23       |
| 3 <sup>rd</sup> Year | 6.21                     | 9.29       |
| Average              | 6.72                     | 8.83       |
| p - value ( 2-tail)  | 0.0204                   |            |

1. Forecasting of Stock Prices Using Brownian Motion–Monte Carlo Simulation.

In Proceedings of the 2016 International Conference on Industrial Engineering and Operations

Management Kuala Lumpur, Malaysia (pp. 704-713).

TABLE VI. STATISTICAL TEST ON THE FORECAST OF STOCK PRICES OF SAMPLE HOLDING COMPANIES USING GBM AND ANN METHODS

| Company | Year 1           |                 | Year 2           |                |
|---------|------------------|-----------------|------------------|----------------|
|         | p-value (2-tail) | Interpretation  | p-value (2-tail) | Interpretation |
| AC      | 0.5384           | Not Significant | 5.93E-52         | Significant    |
| AEV     | 2.89E-09         | Significant     | 5.52E-07         | Significant    |
| DMC     | 6.82E-69         | Significant     | 1.52E-11         | Significant    |
| JGS     | 2.16E-09         | Significant     | 0.01169          | Significant    |
| MPI     | 0.16176          | Not Significant | 0.02608          | Significant    |
| GTCAP   | 3.34E-58         | Significant     | 0.00054          | Significant    |

✓ result

- Table 6 : 2년 기간 예측에서, GBM은 ANN과 상당한 차이를 보임(p-value)  
GBM은 ANN과 크게 다르며 ANN보다 더 예측을 잘함

- Table 7 : 실제 주가 6개월앞을 예측해봄 (2개 기업은 변동이 큼)

- Table 8 : AC 기업은 기대수익률, 일간수익률, 변동성 측면에서 투자 권장  
DMC기업은 변동이심하고 하락 추세라 투자 X

TABLE VII. FORECAST OF STOCK PRICES OF THE SAMPLE HOLDING COMPANIES USING GBM METHOD

| Date                | MPI   | DMC    | JGS    | AEV    | AC      | GTCAP    |
|---------------------|-------|--------|--------|--------|---------|----------|
| October 2014        | 4.574 | 16.010 | 58.286 | 56.954 | 692.218 | 1018.964 |
| November 2014       | 4.482 | 12.189 | 56.812 | 62.008 | 711.486 | 1062.860 |
| December 2014       | 4.539 | 9.597  | 54.488 | 66.725 | 724.518 | 1098.356 |
| January 2015        | 4.350 | 7.630  | 57.415 | 68.197 | 778.954 | 1105.013 |
| February 2015       | 4.582 | 4.723  | 60.745 | 66.569 | 799.549 | 1108.346 |
| March 2015          | 4.924 | 6.065  | 66.912 | 62.472 | 873.824 | 1068.749 |
| April 2015          | 5.339 | 9.289  | 70.893 | 59.305 | 969.654 | 1032.668 |
| Average Stock Price | 4.686 | 8.531  | 61.034 | 63.938 | 802.688 | 1077.201 |
| Standard Deviation  | 0.343 | 3.194  | 5.831  | 3.742  | 89.420  | 40.518   |

✓ conclusion

- 다른 모델과 성능 비교해볼 필요

- drift, volatility 구하는 방법이 핵심

TABLE VIII. FINANCIAL FORECAST SUMMARY USING GBM METHOD

| Category                      | MPI    | DMC      | JGS   | AEV   | AC    | GTCAP  |
|-------------------------------|--------|----------|-------|-------|-------|--------|
| Max Return on Investment (%)  | 22.95  | -79.23   | 30.59 | 20.47 | 33.23 | 17.20  |
| Ave. Daily Rate of Return (%) | 0.0008 | -0.00005 | 0.02  | 0.03  | 0.03  | 0.0007 |
| Volatility                    | 1.38   | 10.60    | 1.52  | 1.46  | 1.14  | 1.27   |

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

### ✓ **purpose**

- 1) Deep Q-Network를 통해 거래 할 주식의 수를 예측할 수있는 거래 시스템을 제안함
- 2) Q-value를 활용한 다양한 행동 전략을 연구함  
(혼란스러운 시장 환경 하에서 수익을 내도록)
- 3) 전이 학습 접근법을 제안함 (불충분한 재무 데이터로 인한 과적합을 방지)

### ✓ **data**

- S&P500, KOSPI, HSI, and EuroStoxx50 주가 지수

### **method**

- Deep Q-Learning
- Transfer Learning

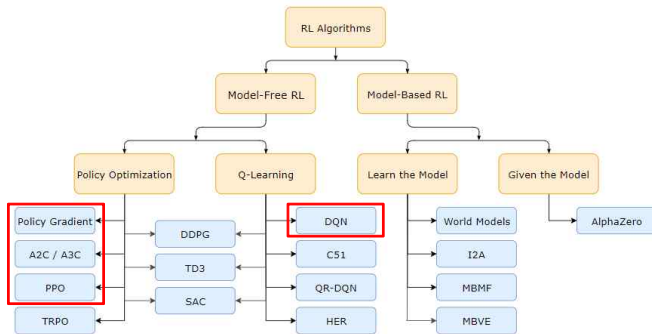
Jeong, G., & Kim, H. Y. (2019).

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ background

### A Taxonomy of RL Algorithms



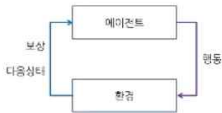
A non-exhaustive, but useful taxonomy of algorithms in modern RL. [Citations below.](#)

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

### ✓ background

1. 에이전트가 환경에서 자신의 상태를 관찰
2. 그 상태에서 어떠한 기준에 따라 행동을 선택
3. 선택한 행동을 환경에서 실행
4. 환경으로부터 다음 상태와 보상을 받을
5. 보상을 통해 에이전트가 가진 정보를 수정함

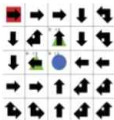


$s_0, a_0, r_1, s_1, a_1, r_2, \dots, s_T$

- Agent : 상태를 관찰, 행동을 선택, 목표지향
- Environment : 에이전트를 제외한 나머지
- State : 현재 상황을 나타내는 정보
- Action
- Reward : 행동의 좋고 나쁨을 알려주는 정보

상태  $s$ 에서 행동  $a$ 를 선택할 확률

$$\text{정책 } \pi(a|s) = P[A_t = a | S_t = s]$$



$$\text{가치함수 } v_{\pi}(s) = E_{\pi}[R_{t+1} + \gamma R_{t+2} + \dots | S_t = s]$$

$$\text{큐함수 } q_{\pi}(s, a) = E_{\pi}[R_{t+1} + \gamma R_{t+2} + \dots | S_t = s, A_t = a]$$

- Policy : 상태  $s$ 에서, 행동  $a$ 를 선택할 확률
- Value function : 행동 선택의 기준
- Q function : 상태  $s$ 에서 행동  $a$ 를 선택했을 때, 미래에 받을 것이라 기대하는 보상의 합

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

### ✓ background

너무 먼 미래에 대해서 기대를 품기보다는 가까운 미래에 대해서 구체적인 기대를 품기로 했다

$$q_{\pi}(s, a) = E_{\pi}[R_{t+1} + \gamma(R_{t+2} + \dots) | S_t = s, A_t = a]$$

$$q_{\pi}(s, a) = E_{\pi}[R_{t+1} + \gamma q_{\pi}(S_{t+1}, A_{t+1}) | S_t = s, A_t = a]$$

### 벨만 기대 방정식(Bellman expectation equation)

미래에 대해서 기대만 하기보다는 실제로 부딪혀보면서 학습하기로 했다

현재 큐함수  $\leftarrow$  보상 + 감가율  $\times$  다음 큐함수

$$q(s, a) \leftarrow r + \gamma q_{\pi}(s', a')$$

점진적인 큐함수의 업데이트

$$q(s, a) = q(s, a) + \alpha(r + \gamma q(s', a') - q(s, a))$$

탐욕정책  $\pi'(s) = \operatorname{argmax}_a q_{\pi}(s, a)$

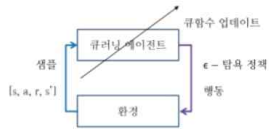
$\epsilon$ -탐욕정책  $\pi(s) = \begin{cases} a^* = \operatorname{argmax}_a q(s, a), & 1 - \epsilon \\ a \neq a^*, & \epsilon \end{cases}$

• 기왕 기억을 활용하는 김에 좋은 기억을 활용해보자

→ 다음 큐함수 중에서 가장 값이 큰 큐함수를 이용해서 현재 큐함수를 업데이트

(Q-Learning)

$$q(s, a) = q(s, a) + \alpha(r + \gamma \max_{a'} q(s', a') - q(s, a))$$



- Bellman expectation equation : Q 함수를 업데이트, 이를 위해 SARSA (s,a,r,s',a')이 필요함

- Qfunction : 상태 s에서 행동 a를 선택했을 때, 미래에 받을 것이라 기대하는 보상의 합

- Greedy Policy : 지금 할 수 있는 행동중에서 Q 함수가 가장 높은 행동을 선택

- 엡실론-Greedy Policy : 일정 확률로 랜덤하게 선택 (탐험)

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ **method** 1) Q value를 통한 주식 수 결정  
(상수 L=10으로 제한)

$$S = \{s_1, s_2, \dots, s_T\},$$

$$A = \{1, 0, -1\}.$$

$$r_t = \left(1 + a_t \times \frac{p_t - p_{t-1}}{p_{t-1}}\right) \frac{p_{t-1}}{p_{t-n}},$$

$$profit_t = a_t \times \frac{p_t - p_{t-1}}{p_{t-1}}$$

$$Total\ profit = \sum_t profit_t$$

$$r_t = num_t \times \left(1 + a_t \times \frac{p_t - p_{t-1}}{p_{t-1}}\right) \frac{p_{t-1}}{p_{t-n}},$$

$$profit_t = num_t \times a_t \times \frac{p_t - p_{t-1}}{p_{t-1}}.$$

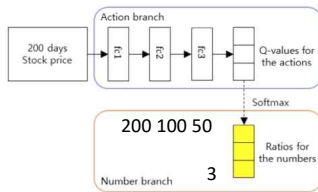


Fig. 2. Architecture of the NumQ method.

$$fc1 = Relu(W_1 I_t + b_1) \quad (11)$$

$$fc2 = Relu(W_2 \times fc1 + b_2) \quad (12)$$

$$fc3 = W_3 \times fc2 + b_3 \quad (13)$$

$$Q_{action}(s_t, a_t) = W_4 \times Relu(fc3) + b_4 \quad (14)$$

$$R_{num}(s_t, a_t) = Softmax(W_4 \times Sigmoid(fc3) + b_4) \quad (15)$$

$$num_t = R_{num}(s_t, a_t^*) \times L \text{ where } a_t^* = \operatorname{argmax} Q_{action}(s_t, a_t) \quad (16)$$

- state : t 시점 종가 - t-1 시점 종가  
(from t-199 to t)

- action : {1: 매도, 0: 관망, -1: 매수}



## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ **method** 1) action-dependent DNN regressor를 통한 주식 수 결정

- Fig 4 : 3단계 훈련

- action-independent 의 경우는 마지막 출력이 1개

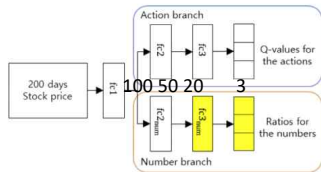


Fig. 3. Architecture of the NumDReg – AD method.

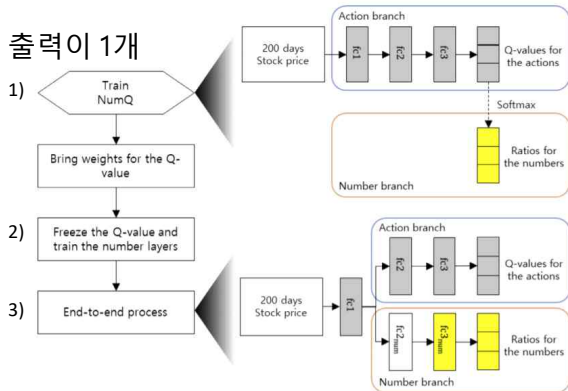


Fig. 4. Flow chart of the NumDReg – AD with three – step training training process.

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

### ✓ method 2) Action strategies in a confused market

- 확률의 차이가 사전에 결정된 임계값보다 낮으면 전략에 따라 행동함.

ex) 선택한 Q- 값이 다른 값에 비해 현저하게 큰 비율을 갖지 않으면 주식에 명확한 추세가 없다고 가정한 다음 특정 조치를 취함

$$\frac{|Q(s_t, a_{\text{BUY}}) - Q(s_t, a_{\text{SELL}})|}{\sum |Q(s_t, a)|} < \text{threshold}.$$

---

#### Algorithm 1 Q learning including the action strategy.

---

Initialize the Q network and load the pretrained network.

Total profit = 0

1: **for** each episode **do**:

2: Set state  $s_t$ ;

3: Obtain the Q-values for the action and the number in accordance with the Q-network.

4: **if**  $\frac{|Q(s_t, a_{\text{BUY}}) - Q(s_t, a_{\text{SELL}})|}{\sum |Q(s_t, a)|} < \text{threshold}$ , **then**:

5: The action is given by the strategy;

6: **else**:

7: Action  $a_t = \text{argmax} \{Q(s_t, a)\}$ ;

8: Set next state  $s_{t+1}$ ;

9: Calculate  $r_t$  and  $\text{profit}_t$ ;

10: Store memory  $(s_t, a_t, r_t, s_{t+1})$  in buffer.

11: **for** each mini-batch sample **do**:

12:  $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \text{learning rate} * (r_t + \gamma Q(s_{t+1}, a') - Q(s_t, a_t))$ ,

13: Total profit  $\leftarrow$  Total profit +  $\text{profit}_t$ .

14: **end for**

15: **end for**

---

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ **method 3) Transfer learning from the index component stocks**

- correlation and the neural network을 바탕으로 하위 종목을 선택함

- 6그룹으로 나눔

- 상관관계에서 (높은 관계 / 높은 관계 반 낮은 관계 반 / 낮은 관계)

- 신경망에서 (낮은 MSE / 낮은 MSE 반 높은 MSE 반 / 높은 MSE)

- 4개 지수에 포함된 구성 종목 수에 따라 종목 수를 다르게 선택함

- 각 그룹에서 모든 주식은 별도로 사전 훈련됨, 그러나 사전 훈련 된 모든 주식은 가중치를 공유함

- 총 수입이 가장 높은 그룹을 선택함

**Algorithm 2** The entire process, including transfer learning, determining the share trading number method, and using the action strategy.

- 1: Load the index data and the index component stocks;
- 2: Calculate the correlation between the index and the index components stocks or calculate the MSE between them with the neural network;
- 3: Create the six groups with the highest or lowest relationship or both using correlation and the MSE;
- 4: Select the best group among the six groups;
- 5: Pretrain the model including NumQ or NumDReg – AD with the chosen group;
- 6: **if** NumQ, **then**:
- 7:   Load NumQ weights pretrained by transfer learning;
- 8:   Train the model, including NumQ with the action strategy;
- 9: **else if** NumDReg – AD, **then**:
- 10:   Load NumDReg – AD weights pretrained by transfer learning;
- 11:   Train the model, including NumDReg – AD with the action strategy;
- 12: **else if** NumDReg – AD with three – step training, **then**:
- 13:   Load NumQ weights pretrained by transfer learning;
- 14:   Freeze NumQ weights and only train the number branch based on the NumDReg – AD algorithm;
- 15:   Complete an end-to-end process with the action strategy;
- 16: **else if** NumDReg – ID, **then**:
- 17:   Load NumQ weights pretrained by transfer learning;
- 18:   Freeze NumQ weights and only train the number branch based on the NumDReg – ID algorithm;
- 19:   Complete an end-to-end process with the action strategy;

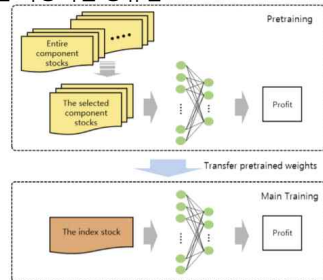


Fig. 6. Training process of transfer learning.

Jeong, G., & Kim, H. Y. (2019).

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ result

- hyper parameter

Network : learning rate = 0.0001, gamma = 0.85, batch size = 64

end-to-end process : learning rate = 0.0000001 for NumDReg - AD, 0.00001 for NumDReg - ID



Fig. 8. Comparison of the transfer learning models according to the relationship between S&P500 and its component stocks.

**Table 1**  
Data Descriptions.

| Main training                   | Training period           |                           | Test period               |
|---------------------------------|---------------------------|---------------------------|---------------------------|
| Pretraining                     | Training                  | Validation                | x                         |
| S&P500 index (SP500)            | Jan 1, 1987–Nov 4, 2002   | Nov 5, 2002–Aug 10, 2006  | Aug 11, 2006–Dec 31, 2017 |
| Hang Seng Index (HSI)           | Jan 2, 2001–May 2, 2008   | May 3, 2008–Jul 21, 2009  | Jul 22, 2009–Dec 29, 2017 |
| EuroStoxx50 index               | Apr 05, 1991–Feb 19, 2003 | Feb 20, 2003–Jul 13, 2005 | Feb 21, 2003–Dec 29, 2017 |
| Korea Stock Price Index (KOSPI) | Jul 1, 1997–Jul 11, 2006  | Jul 12, 2006–May 05, 2008 | Mar 06, 2008–Dec 28, 2017 |

**Table 2**  
The abbreviations of models.

| Abbreviation | Pretrained data measure | Main training data | Pretrained Number method | Number method | Step-wise training | Action strategy |
|--------------|-------------------------|--------------------|--------------------------|---------------|--------------------|-----------------|
| MKT          | N/A                     | N/A                | N/A                      | N/A           | N/A                | N/A             |
| RL           | N/A                     | Index              | N/A                      | N/A           | N/A                | N/A             |
| IDX          | Index                   | Index              | N/A                      | N/A           | N/A                | N/A             |
| CR           | Correlation             | Index              | N/A                      | N/A           | N/A                | N/A             |
| NE           | Neural Network          | Index              | N/A                      | N/A           | N/A                | N/A             |
| NQ           | NE                      | Index              | N/A                      | NumQ          | N/A                | N/A             |
| NENQ         | NE                      | Index              | NumQ                     | NumQ          | N/A                | N/A             |
| NDA          | NE                      | Index              | N/A                      | NumDReg – AD  | N/A                | N/A             |
| NENDA        | NE                      | Index              | NumDReg – AD             | NumDReg – AD  | N/A                | N/A             |
| NENDA3       | NE                      | Index              | NumQ                     | NumDReg – AD  | Yes                | N/A             |
| NENDE3       | NE                      | Index              | NumQ                     | NumDReg – ID  | Yes                | N/A             |
| NENDE3-BUY   | NE                      | Index              | NumQ                     | NumDReg – ID  | Yes                | Buy             |
| NENDE3-HOLD  | NE                      | Index              | NumQ                     | NumDReg – ID  | Yes                | Hold            |
| NENDE3-SELL  | NE                      | Index              | NumQ                     | NumDReg – ID  | Yes                | Sell            |

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

✓ result

- Table 4. : 사전 훈련된 모델이 성능이 좋음  
신경망이 주식과 지수 관계 모델링에 더 좋음  
종목 수가 많은 경우는 상관 관계가 낮은 모델이 사전 훈련에 좋지만,  
반대의 경우는 상관 관계가 높은 모델이 좋음
- Table 6. : 임계치 0.2에서 hold 전략이 대체로 높은 수익을 얻음

Table 4  
Results of the pretrained model.

| Model |        | S&P500       |        |        |              | KOSPI       |  |
|-------|--------|--------------|--------|--------|--------------|-------------|--|
| MKT   |        | 0.9654       |        |        |              | 0.6176      |  |
| RL    |        | 0.8647       |        |        |              | 0.5511      |  |
| IDX   |        | 1.3621       |        |        |              | 0.6501      |  |
| CR    | High   | High and Low | Low    | High   | High and Low | Low         |  |
|       | 0.9254 | 1.0392       | 1.4310 | 1.0097 | 1.0761       | 1.4988      |  |
| NE    | High   | High and Low | Low    | High   | High and Low | Low         |  |
|       | 1.2920 | 1.0761       | 1.595  | 1.0515 | 1.6083       | 2.0641      |  |
| Model |        | HSI          |        |        |              | EuroStoxx50 |  |
| MKT   |        | 0.2852       |        |        |              | -0.3359     |  |
| RL    |        | 0.3841       |        |        |              | 0.5096      |  |
| IDX   |        | 0.5932       |        |        |              | 0.8675      |  |
| CR    | High   | High and Low | Low    | High   | High and Low | Low         |  |
|       | 0.6949 | 0.6559       | 0.6235 | 0.9877 | 0.6129       | 0.3791      |  |
| NE    | High   | High and Low | Low    | High   | High and Low | Low         |  |
|       | 0.7422 | 0.7242       | 0.5422 | 1.2901 | 0.8179       | 0.4752      |  |

Table 5  
The model's results from trading different numbers of shares using the NQ and ND models, and their transfer learning applications with step-wise training.

| Model  | S&P500 | KOSPI   | HSI     | EuroStoxx50 |
|--------|--------|---------|---------|-------------|
| NQ     | 3.8912 | 3.2163  | 4.7626  | 3.3882      |
| NDA    | 2.9759 | 2.5334  | 4.3639  | 2.6322      |
| NENQ   | 7.7169 | 3.9665  | 7.2330  | 4.0376      |
| NENDA  | 4.4279 | 4.5590  | 6.1747  | 3.1082      |
| NENDA3 | 8.6873 | 5.8359  | 9.1167  | 4.3487      |
| NENDI3 | 9.6789 | 10.5704 | 10.4296 | 6.1502      |

Table 6  
Results of the NENDI3 model and the models applying action strategies.

| Model       | S&P500  | KOSPI   | HSI     | EuroStoxx50 |
|-------------|---------|---------|---------|-------------|
| NENDI3      | 9.6789  | 10.5704 | 10.4296 | 6.1502      |
| NENDI3-BUY  | 9.1976  | 7.8825  | 11.5862 | 6.9175      |
| NENDI3-HOLD | 11.4188 | 13.1416 | 9.3136  | 9.3237      |
| NENDI3-SELL | 4.5838  | 10.7684 | 11.2926 | -9.5000     |

Jeong, G., & Kim, H. Y. (2019).

## 2. Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning.

Expert Systems with Applications, 117, 125-138.

### ✓ contribution & application

- 거래 주식 수, 액션 전략, 보상 방법 등이 중요
- > feature를 줄이고 예측 정확도 높이는 핵심
- 전이 학습 대상에 대해 고민 -> 상관 관계 고려해 선정
- 액션 전략에 NLP 응용 가능
- 학습 여부에 따라 보상을 변경 (ex. 가치의 차이 만큼 보상)
- LSTN + DQN 알고리즘 적용 검토
- DQN 말고, policy optimization 알고리즘 적용 검토



Fig. 9. Comparison of the NQ and ND models, and their transfer learning applications using S&P500.



Fig. 10. Comparison between NENDA, NENDA3, and NENDE3 models using S&P500.

### **3. Case study**

BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ **purpose**

- 주가 예측에 다양한 방법을 시도함 (보조지표, 뉴스, 분류모델, LSTM, RL 등)

### ✓ **data**

- 2010.08.06 ~ 2018.09.12
- KOSPI 시가총액 300위 해당 종목(date, open, high, low, close, volume, code, name, etc)
- KOSPI 지수(일자, 체결가, 거래량, 거래 대금, 등락, etc)

### **method**

- LSTM + Policy Gradient RL
- 폭락 방지 알고리즘 (뉴스)



BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ **method**

- 기존 주가 예측 방법으로 접근 (주가 등락 classification)

### ✓ **preprocessing**

- 시계열 변수와 주가 관련 보조지표 계산  
(RSI, MACD, Stochastic, 이동평균(MA, EMA, WMA, EVWMA, VWMA 등), DMI, BB, etc)

### ✓ **test**

- DNN, SVM, Decision Tree, RandomForest 등 10개 모델 K-FOLD 적용 시 최대 60% 초반대 ACC  
-> 많이 부족하다고 생각해 뉴스 데이터 추가 사용 (약 10만개, 제목, 내용, 날짜, 전일비차트)

BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ test

- 형태소 분석 및 Word2Vec (전처리, 조사 숫자 제외, Konlpy-okt(사전 기반), Soynlp(통계 기반), Khaiii(딥러닝 기반), skip-gram model 이용(Word2Vec))

- TF-IDF, RNN, LSTM, GRU, CNN-RNN 형태소 분석기와 모델을 조합하여 등락 예측 -> 정확도 50% 전후  
-> 사용 불가능

-> **주식 종목별로 강화학습 모델을 생성함**

### ✓ data

Ebest 증권사의 Xing API

### ✓ preprocessing

- 거래 정지 이유 별 데이터 수정(액면분할, 액면병합, 차등감자, 기업분할, 기업병합, 인적분할 등)  
- 종목 별로 변수를 다르게 추가함. 종목별로 결과가 잘 나오는 변수 조합이 존재함.

ex) 메리츠화재 : 코스피지수 + BB 등, KCC: ADX, DMI, BOX 등

BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ component

- \* 에이전트 - 투자자,      \* 환경 - 데이터,
- \* 액션 - 매수, 매도, 관망,      \* 보상 - 이익, 손해,
- \* 보상 규칙 - 이익률 3% 달성 시 이익
- \* 거래수수료와 거래세 반영 (일반적 case로)

### ✓ model

- 정책 신경망 클래스에서 사용함, 5 은닉층의 LSTM(3개층 256차원, dropout 50%)

입력층 : 학습 데이터 차원 + 에이전트 상태 차원 (주식 보유 비율, 포트폴리오 가치 비율)

출력층 : 매수, 매도, 관망

- policy gradient 사용

이유 : Q러닝은 기대 손익을 예측하나, 정책 경사는 어떤 행동이 현재 상태에서 가장 좋을 지를 확률적으로 판단하기 때문에 주식 예측에 더 효과적이라고 알려져 있음.

- 확률적 경사 하강법(SGD), Learning rate는 0.01, 0.001 등 종목별로 다르게 설정함

BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ result 백테스트

- 아모레퍼시픽 : 6개월 +10.86%, 동원산업 : 6개월 6% + 하락 방어

### ✓ 폭락 방지 알고리즘

1) 익일 주가 등락 여부에 따른 공부정 강도 생성

- 뉴스 파싱 -> 주가 등락 심했을 때, 이전의 뉴스를 직접 모음

- 뉴스마다 긍정 / 부정 라벨링 -> 단어마다 긍정 / 부정 뉴스 등장 빈도수 측정

$Diff = (\text{긍정 뉴스 등장 수} - \text{부정 뉴스 등장 수}) / \text{총 뉴스 등장 수}$

$Strength1 = Diff$ 의 합,  $Strength2 = Diff$ 의 합 / 뉴스 총 단어 수

$Pos\_count1 = Diff > 0$  단어 수,  $Pos\_count2 = Diff > 0.2$  단어 수

$Pos\_count3 = Diff > 0.2$  이고, 빈도수가 100 이상인 단어 수

$Neg\_count1 = Diff < 0$  단어 수,  $Neg\_count2 = Diff < -0.2$  단어 수

$Neg\_count3 = Diff < -0.2$  이고, 빈도수가 100 이상인 단어 수

2) 시간대별 뉴스 처리

- 20분 마다 동작하도록 배치, 시간대별 종목에 대한 공부정 점수 계산 -> 부정 점수 급등 시점 포착

- 부정점수와 MACD 보조지표의 매도 신호 모두 1일 경우 전량 매도

BOAZ 10기 구진모 사공용협 전소정.

## 1. 강화학습을 이용한 주식 트레이딩 알고리즘.

제 9회 BOAZ BIGDATA CONFERENCE

### ✓ 실전 트레이딩 epoch 1000 (16개 종목)

3일차 수익률 1.08

### ✓ conclusion & limitation

- 등락 예측이 목표인 기존 방법보다 강화학습의 성능이 좋음
- 수익도 중요하지만 폭락 방지 알고리즘도 중요함 (안정적인 수익이 우선)
- 빠른 폭락 감지 보완 필요, ex) 아모레퍼시픽 폭락임에도 신호가 없었음
- **종목마다 다른 feature selection 기준** : 사람이 직접 수작업으로 테스트함

정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019).

## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ purpose

- 기업선정(가치투자)모델 (재무비율 및 수정주가 데이터),
- 모멘텀 투자모델 (주가 및 거래지표 데이터)의 앙상블을 통한 안정적인 수익률 달성

### ✓ data

| 데이터 구성                           |               |   |                                  |                                       |  |      |                                  |                  |                                  |                      |                                  |                                    |                  |      |                  |
|----------------------------------|---------------|---|----------------------------------|---------------------------------------|--|------|----------------------------------|------------------|----------------------------------|----------------------|----------------------------------|------------------------------------|------------------|------|------------------|
| 재무비율                             |               |   | Features(Annual)                 |                                       | 재무비율   |      |                                  | Features(Annual) |                                  | 수정주가                 | Features (Daily)                 | 주가                                 | Features (Daily) | 거래지표 | Features (Daily) |
| KOSPI<br>KOSDAQ<br>상장기업<br>2138개 | 안정성<br>변수 21개 | 타인자본비율(%)<br>---<br>유보액/총자산(%)<br>---<br>영업수익증가율<br>(전년동기)(%) | KOSPI<br>KOSDAQ<br>상장기업<br>2138개 | 주당지표<br>변수 6개                         | EPS(원)<br>---<br>수정EBITDAPS(원)<br>---<br>PER(배)<br>---<br>PEG(배) | 수정주가 | KOSPI<br>KOSDAQ<br>상장기업<br>2138개 | 수정주가             | KOSPI<br>KOSDAQ<br>상장기업<br>2138개 | 시가<br>종가<br>고가<br>저가 | KOSPI<br>KOSDAQ<br>상장기업<br>2138개 | 거래량<br>외국연순매수량<br>기관순매수량<br>개연순매수량 |                  |      |                  |
|                                  | 성장성<br>변수 27개 | DPS증가율(보통주,<br>전년동기)(%)                                       |                                  | DPS(보통주, 현금<br>+주식)(원)<br>---<br>배당성향 |  |      |                                  |                  |                                  |                      |                                  |                                    |                  |      |                  |
|                                  | 수익성<br>변수 20개 | 매출총이익률(%)<br>---<br>ROE(%)                                    |                                  |                                       |  |      |                                  |                  |                                  |                      |                                  |                                    |                  |      |                  |
|                                  | 활동성<br>변수 3개  | 총자산회전율(회)<br>---<br>총자본회전율(회)                                 |                                  | 매출비중1(결산)<br>---<br>코스닥 대형            |  |      |                                  |                  |                                  |                      |                                  |                                    |                  |      |                  |

### ✓ method

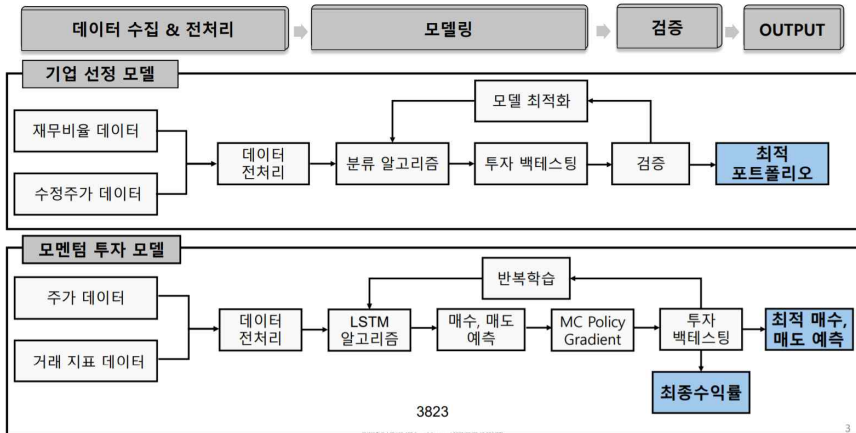
- 기업선정 모델 : 분류 모델(Randomforest)
- 모멘텀 투자모델 : LSTM + 강화학습(Policy Gradient)

정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019).

## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ project flow

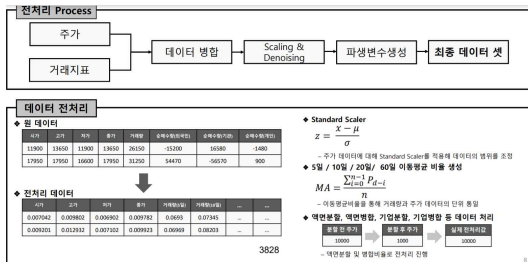
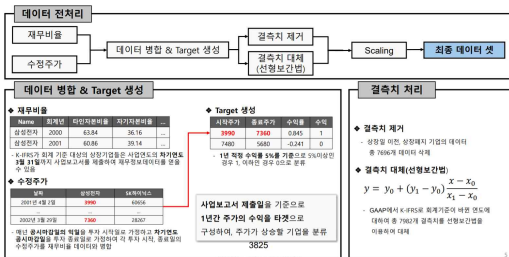


정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019).

## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ preprocessing



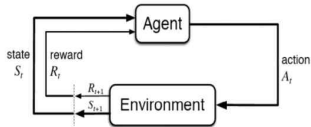


## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ method (모멘텀 투자모델)

#### 강화학습 설정

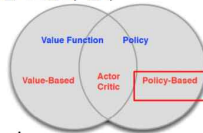


#### Parameter setting

| Action      | 매수, 매도, 관망                 |
|-------------|----------------------------|
| Environment | 주가데이터, 거래지표                |
| state       | 주식 보유개수, 계좌 잔액, 현재 주식가격... |
| Agent       | 행동을 수행하는 투자자               |
| Reward      | 긍정적 학습 : +1, 부정적 학습 : -1   |
| Reward 규칙   | 손익률 3% 기준                  |

#### Policy-based

##### ◆ 강화학습 모델의 범주



##### ◆ Value based

- 미래의 보상을 최대화하기 위한 알고리즘

특정 행동을 취함으로써 얻어지는 보상에 대해 계산

$$v_{\pi}(s) = \mathbb{E}_{\pi}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | S_t = s]$$

##### ◆ Policy based

- 특정 상태에서 Agent가 어떤 행동을 해야 하는가에 대한 알고리즘
- 현재 상태를 바탕으로 최적화된 행동을 Stochastic 한 방법으로 판단

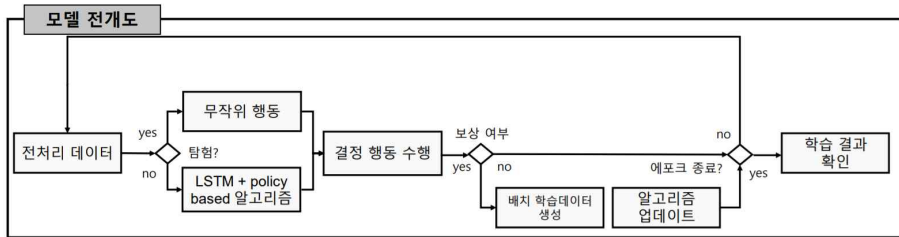
$$\pi(a|s) = \mathbb{P}[A_t = a | S_t = s]$$

정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019).

## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ method (모멘텀 투자모델)



### 탐험 Algorithm

◆ Decaying  $\epsilon$  - greedy algorithm    ◆ 목적식 :  $\text{MIN}(\text{regret function}) = \text{MAX}(\text{누적 reward})$

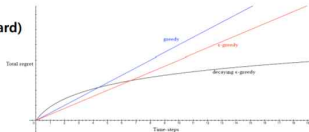
-  $\epsilon$ 의 확률로 임의의 행동 결정

- LSTM+Policy based 알고리즘만으로 모델을 구성하면 최적 값이 아닌 값에 수렴할 수 있으므로 일정 비율로 랜덤 행동을 수행

Regret function

$$L_t = \mathbb{E} \left( \sum_{\tau=1}^t V^* - Q(a_\tau) \right)$$

$V^*$  = 최적 value  
 $Q(a_\tau)$  = 행동에 대한 평균 보상값



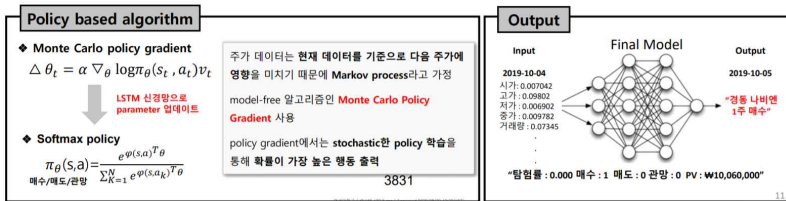
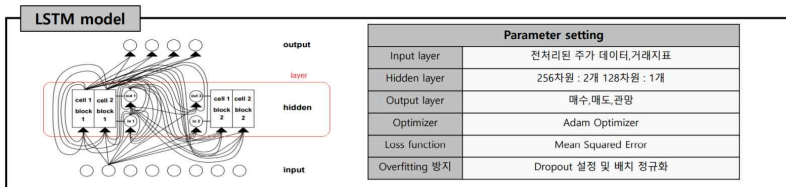
-  $\epsilon$ -greedy, greedy 함수에 비해 Decaying  $\epsilon$ -greedy 함수가 total regret을 최소화

정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019).

## 2. 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링.

대한산업공학회 추계학술대회 논문집, (), 3821-3835.

### ✓ method (모멘텀 투자모델)



### ✓ 백테스팅

- 투자 기간 : 2017년 3월 31일 ~ 2018년 4월 1일

- 포트폴리오(5종목) 수익률 : -2%, 코스피 수익률 : 13%, 모멘텀(5종목) 수익률 : 7.52%

# 느낀점 정리

- ✓ 프로젝트 도메인 결정이 필요함 (범위가 너무 넓음)
- ✓ 종목(포트폴리오) 선택도 중요해보임
- ✓ 주식 트레이딩 모델은 분류/회귀보다 강화학습이 도전할만함 (제 생각)
- ✓ 매매 횟수나 매매 패턴에도 제약을 두는 것을 고려
- ✓ 학습 시간에 따라 (가중치 학습에) 유전알고리즘 사용 고려
- ✓ 거래량과 같은 시장 미시 구조 feature 조사
- ✓ 프로토타입 -> 기존 연구의 파라미터 바꾸기(종목, 기간, 보상 규칙, feature, 하이퍼파라미터 튜닝 등)
- ✓ contribution을 위한 알고리즘 공부(feature engineering, 시계열, NLP, LSTM, RL 등) 개념 공부 + 논문리서치 병행

# Reference

- **Git-hub, NAVER branch에 관련 코드 및 논문 수집**
- Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020). Deep learning for financial applications: A survey. *Applied Soft Computing*, 106384.
- Jiang, W. (2020). Applications of deep learning in stock market prediction: recent progress. *arXiv preprint arXiv:2003.01859*.
- <https://www.youtube.com/watch?v=dB8cpsnZ5FA&t=733s>
- Estembar, R. D., & Marañón, M. J. R. (2016, March). Forecasting of Stock Prices Using Brownian Motion-Monte Carlo Simulation. In *Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia* (pp. 704-713).
- Jeong, G., & Kim, H. Y. (2019). Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning. *Expert Systems with Applications*, 117, 125-138.
- <https://www.youtube.com/watch?v=5x6nbN-6tFU&t=1054s>
- 정해성, 김용현, 임한준, 정기백, 정진태, 최원화. (2019). 재무비율과 기술적 분석을 통한 AI 주식 트레이딩 알고리즘 모델링. *대한산업공학회 추계 학술대회 논문집*, (), 3821-3835.
- <https://blog.naver.com/chunjein/100188340479>
- <https://www.slideshare.net/WoongwonLee/ss-78783597>

감사합니다.

Q&A