

주가 예측 모델에서의 분할 예측을 통한 성능향상 탐구

여태건우¹ · 유도희¹ · 남정원¹ · 오하영^{2*}

Exploring performance improvement through split prediction in stock price prediction model

Tae Geon Woo Yeo¹ · Dohui Ryu¹ · Jungwon Nam¹ · Hayoung Oh^{2*}

¹Undergraduate Student, Web Programming, Korea Digital Media High School, Ansan, 15255 Korea

^{2*}Associate Professor, College of Computing & Informatics, Sungkyunkwan University, Seoul, 03063 Korea

요약

본 논문의 연구 취지는 예측하고자 하는 다음 날과 이전 날의 시가 사이 변동률을 예측값으로 두고 시가를 예측하는 기존 논문들과는 다르게 예측하고자 하는 다음날의 주가 순위를 일정한 간격으로 분할하여 생성된 각 구간마다의 시가 변동률을 예측값으로 하는 모델을 통하여 최종적인 다음날의 시가 변동률을 예측하는 새로운 시계열 데이터 예측 방식을 제안하고자 한다. 예측값의 세분화 정도와 입력 데이터의 종류에 따른 모델의 성능 변화를 분석했으며 연구 결과 예측값의 세분화 정도에 따른 모델의 예측값과 실제값의 차이가 예측값의 세분화 개수가 3일 때 큰 폭으로 감소한다는 사실도 도출해 낼 수 있었다.

ABSTRACT

The purpose of this study is to set the rate of change between the market price of the next day and the previous day to be predicted as the predicted value, and the market price for each section is generated by dividing the stock price ranking of the next day to be predicted at regular intervals, which is different from the previous papers that predict the market price. We would like to propose a new time series data prediction method that predicts the market price change rate of the final next day through a model using the rate of change as the predicted value. The change in the performance of the model according to the degree of subdivision of the predicted value and the type of input data was analyzed.

키워드 : 기계 학습, 오토인코더, DNN, 주식, 하이퍼파라미터

Keywords : Machine learning, AutoEncoder, DNN, Stock, HyperParameter

Received 24 February 2022, Revised 27 March 2022, Accepted 3 April 2022

* Corresponding Author Hayoung Oh (E-mail: hyoh79@gmail.com, Tel:+82-2-583-8585)

Associate Professor, College of Computing & Informatics, Sungkyunkwan University, Seoul, 03063 Korea

Open Access <http://doi.org/10.6109/jkiice.2022.26.4.503>

print ISSN: 2234-4772 online ISSN: 2288-4165

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

주식의 주가 예측은 오래된 연구 주제로서 그동안 수많은 연구가 이루어져 왔으며, 그중 딥러닝을 이용한 선행 연구 중 [1] 연구에 따르면 하나의 주식에 대해 4개의 특징을 뽑아 인자분석을 진행한 후, 가장 점수가 높은 5개 주식 종목을 LSTM에 전달하여 적용한 후 과거 데이터를 이용하여 투자전략을 시험해보는 백테스팅을 진행하여 주식의 주가를 예측한 연구가 존재한다. [2] 연구에서는 Sliding Window 방식을 사용하여 주식시장의 기간에 따라 비중을 다르게 두는 모델 학습 방법을 제시하였다. 이 연구에서는 LSTM, RNN, CNN 모델들을 학습하였으며 3가지의 모델의 성능을 모두 비교한 결과 CNN 모델의 성능이 가장 뛰어났다. [3] 연구에서는 AutoEncoder(AE) 모델을 활용하여 시장참여자들의 감정 상태를 고려한 어텐션 메커니즘(Attention Mechanism)과 변이형 오토인코더(Variational Autoencoder)를 결합한 새로운 분석 기법 ‘VATD’를 새로이 제안한 후, 이를 CNN이나 LSTM과 같은 기존 기법들과 성능을 비교하여 새로운 분석 기법의 우수성을 검증하였다. [4] 연구에서는 공적분(co-intergration)과 인공 신경망을 활용한 하이브리드 모델을 제안하여 주가 예측의 정확도를 향상시켰으며 Johansen의 공적분 검정 결과 KOSDAQ은 KOSPI보다 시장의 추세와 외부 변수의 영향을 더욱 많이 받는 것을 확인하였다. [5] 연구에서는 실제 주식시장의 주가 차트와 가상의 주가 차트를 구별할 수 있는 GAN 네트워크를 활용하여 주가예측의 성능을 향상시켰다. [6] 연구에서는 CNN 모델을 활용하여 종가와 거래량으로 이루어진 사진을 학습하여 나온 값을 Deep Q Network 강화학습 모델에 학습시켜 주식의 매도(Long), 공매도(Short), 매도와 공매도를 모두 취하는 Neutral Position 중에서 하나의 포지션을 결정하는 의사결정 모델을 만들어 수익률을 계산하는 방법을 제시했다. [7] 연구에서는 웨이블릿 변환(Wavelet Transformation)으로 노이즈 값을 제거한 주식 데이터를 Stacked 오토인코더로 값을 재생성한 후 LSTM(Long-Short Term Memory)로 학습하여 주식의 종가를 예측하여 다음 날의 종가를 예측하는 모델을 제시한다. 그리고 [8] 연구에서는 코스닥 상위 150종목의 차트 데이터를 불러와 캔들차트 이미지 데이터를 CNN 모델에 입력값으로 넣어 주가들의 패턴을 학습한 후 미래 주가의 등락을 예측하고자 한다.

이러한 연구들에서 나타나는 공통점은 모델의 예측 값을 예측하고자 하는 다음날과 이전 날 시가 사이의 변동률로 두고 모델을 설계하였다는 사실이며 최신 기준 연구 흐름은 기간 사이의 주가를 연속적으로 예측하고자 하는 시도가 정확도에 미치는 영향을 간과하고 있는 것으로 판단된다.

따라서 본 논문은 다음날의 시가 변동률을 예측하기 위해 예측하고자 하는 다음날의 주가 차트를 일정한 간격으로 분할하여 생성된 각 구간마다의 시가 변동률을 예측값으로 하는 모델을 통하여 최종적인 다음날의 시가 변동률을 예측하는 새로운 시계열 데이터 예측 방식을 제안하고자 한다.

$$R_1 = t + t^* \frac{P_1}{100} \tag{1}$$

$$R_n = R_{n-1} + R_{n-1}^* \frac{P_n}{100} \quad (n = 2, \dots, 7)$$

$$K = |M - C|$$

식(1)은 딥러닝 모델의 오차 값을 계산하는 데 활용된다. 모델은 다음날 상대적 종료 시점 가격인 R_n 과 상대적 시작 시점 가격인 t 사이 다음날 하루 동안의 기간을 일정한 간격으로 분할한 각 구간 s 마다의 주가 변동률인 P_n 을 예측한다. 모델의 s 개의 예측값인 각 구간의 주가 변동률은 P_n 으로서 표현되며 식 (1)을 이용하여 구해진 다음 날 하루 동안의 주가 변동률 M 값과 다음날 하루 동안의 실제 주가 변동률의 절댓값 차를 최종적인 오차 값 K 로 활용하게 된다.

즉, 본 논문에서 이러한 새로운 방식이 시계열 데이터를 예측하는 모델의 정확성에 이로온 효과를 가져다주는 것이라는 가정하에 위 방식에서 예측값의 세분화 정도와 입력 데이터의 종류가 학습에 주는 영향도의 변화 양상과, 그에 따른 모델의 성능 변화를 분석, 구현하고 결과를 도출한다.

II. 제안하는 기법

2.1. 오토인코더 모델 & 데이터 전처리

그림 1은 제안하는 기법을 위한 모델에 필요한 데이터를 전처리하는 과정을 나타낸 그림이다.

2.1.1. 주가 데이터 수집 & 캔들 차트 이미지 생성

그림 1의 과정 ①, ②에서는 3개의 오토인코더 모델의 학습에 필요한 일, 주, 월 기간의 주가 차트 이미지를 생성한다. 본 연구에서 주가의 위치적 특성들을 효율적으로 반영하기 위해, 시계열 형식의 데이터 대신 이미지 형식을 실험에서 사용하였다.

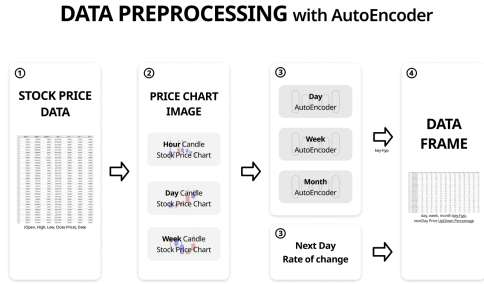


Fig. 1 Data Preprocessing of the proposed scheme

주가 차트 이미지를 만들기 위해 국내 유가증권시장(KOSPI)에 상장되어있는 ETN, ETF, 우선주를 제외한 825종목의 시가, 시가, 고가, 저가 데이터를 불러왔다. 표 1은 주가 데이터의 종류 및 크롤링 기간을 의미한다.

Table. 1 Type of stock price data and Crawling Date

Type of stock price data	Crawling Date
Hourly Candle	2021. 6. 8. → 2021. 12. 10.
Daily Candle	2019. 7. 10. → 2021. 12. 10.
Weekly Candle	2016. 3. 7. → 2021. 12. 10.

캔들 차트 데이터 형식으로 정의하기 위해 ‘mpl_finance’, 캔들 차트 이미지를 생성하고 저장하기 위해 ‘matplotlib’ 파이썬 라이브러리를 사용하였고, 이를 통해 주가 데이터를 기반으로 이미지를 생성했다.

그림 2은 생성한 캔들차트 이미지의 주가 데이터 기간 인포그래픽을 보여준다. 이미지에 표현되는 주가의 기간 설정은 위의 사진처럼 예측하고자 하는 날을 t라고 했을 때, day 기간은 t-1일 차 동안의 7개의 시간별 주가 데이터로 구성되고 week 기간은 t-5일 차부터 t-1 일차까지의 5개의 일봉 주가 데이터로 구성되며 month 기간은 t-25 일차부터 t-1까지의 5개의 주봉 주가 데이터로 구성된다. 다운받은 주가 데이터는 위의 규칙에 따라 정해진 day, week, month 기간 44x28 크기의 사진으로 생성된다. 오토인코더에서 각 기간을 대표하는 특성들을 중점적으로 학습하기 위하여 봉 개수를 5~6으로 설정

후 이미지의 복잡성을 적절히 조절하였다.

또한 다운받은 주가 데이터를 모두 활용하는 것이 아닌, 주가 캔들 데이터들이 봉 개수에 맞게 모두 온전히 존재할 때만 이미지를 생성하였다. 이러한 원리로 생성된 시간봉, 일봉, 주봉 캔들차트 사진의 개수는 82,346개로 모두 일정하다.

Candle Chart Date Term

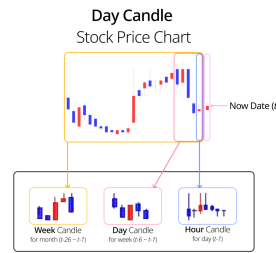


Fig. 2 Stock price data period infographic of the created candle chart image

2.1.2. AutoEncoder 모델

2.1.1 주가 데이터 수집 & 캔들 차트 이미지 생성에 이어 2.1.2에서는 AutoEncoder 모델을 설명한다. 그림 1의 과정 ②, ③에서는 Day, Week, Month 데이터의 특성을 추출하기 위해, 각각 Autoencoder 모델 3개를 만들어 주가 이미지를 학습시킨다. 각각 시간봉, 일봉, 주봉 주가 캔들차트 이미지를 학습시켜서 오토인코더 모델 3개를 제작하였다.

오토인코더 모델 3개의 공통적인 형식은 다음과 같다. 인코더의 입력 부분에 주가 차트 이미지의 크기인 1,012(44x23)만큼의 입력값을 받을 수 있는 층이 존재한다. 이 층을 통해 주가 차트 이미지를 입력한다. 입력값을 축소하는 노드는 순서대로 512, 256, 128, 64, 16으로 쌓았다. 데이터가 각 층을 지나면서 Feature의 크기가 감소한다. 우리의 특성을 확인할 수 있는 오토인코더의 가운데 층이자 인코더의 출력층에는 4개의 노드를 쌓았다. 즉 1,012개의 특성이 있는 이미지가 4개의 특성을 출력하는 결과를 볼 수 있다. 활성화 함수는 모든 은닉층에 leaky ReLU를 사용하였고, 최적화 알고리즘은 Adam(Adaptive Moment Estimation)을 사용하였다.

2.1.3. DataFrame 변환 과정

2.1.3에서는 그림 1 과정 ③, ④의 DataFrame 변환 과

정을 설명한다. 오토인코더의 인코더를 통해 출력된 모델 특성값을 key-hyp라고 정의하며, Day 오토인코더 모델일 경우에는 key-hyp-day, week 오토인코더 모델은 key-hyp-week, month 모델의 경우에는 key-hyp-month라고 정의한다. 본 연구 과정에서는 모든 주가 데이터의 key-hyp와 모델의 예측값을 구하여 DNN 학습데이터인 데이터프레임에 저장한다. 특성값은 다음과 같은 형식으로 Dataframe에 추가한다.

모델의 예측값은 하루 동안의 주가 캔들차트를 n개로 세분화한 각 기간들의 변동률로, 기간의 시가를 다음 방식을 통해 구한다.

$$\left(\frac{\text{opening} - \text{price}_n}{\text{opening} - \text{price}_{n-1}} - 1 \right) \times 100 \quad (2)$$

$$\left(\frac{\text{opening} - \text{price}_n}{\text{closing} - \text{price}_{n-1}} - 1 \right) \times 100 \quad (3)$$

식(2)는 한 기간의 시가와 이전 기간의 시가의 변화율을 구하는 함수로, 마지막을 제외한 나머지 기간에서의 변화율을 구할 때 사용된다. 본 과정을 통해 계산된 7종류의 변동률을 다음과 같은 형식으로 데이터프레임에 추가한다. 또한 식(3)은 마지막 구간에서의 변동률만을 구할 때 사용하는 적인 데이터프레임의 형식은 표 2와 같다. Stock_Index는 주식의 종류와 날짜로 인덱싱하는 것이다. Day Feature, Week Feature, Month Feature는 오토인코더에서 나온 4개의 Feature 원소로 구성된 벡터이고, 등락률은 1~7등분의 1~7개의 등락률 원소들로 이루어진 벡터이다.

Table. 2 DataFrame Column Table

Stock Index	{stockTicker}_YYYYMMDD
Day Feature	[dhk(day-hyp-key)1, dhk2, dhk3, dhk4]
Week Feature	[whk(week-hyp-key)1, whk2, whk3, whk4]
Month Feature	[mhk(month-hyp-key)1, mhk2, mhk3, mhk4]
Fluctuation Rate (S: 1)	[P1]
Fluctuation Rate (S: 2)	[P1, P2]
Fluctuation Rate (S: 3)	[P1 ... P3]
Fluctuation Rate (S: 4)	[P1 ... P4]
Fluctuation Rate (S: 5)	[P1 ... P5]
Fluctuation Rate (S: 6)	[P1 ... P6]
Fluctuation Rate (S: 7)	[P1 ... P7]

2.2. DNN 모델 및 실험 과정

그림 3은 DNN 모델 실험 개요를 보여준다.

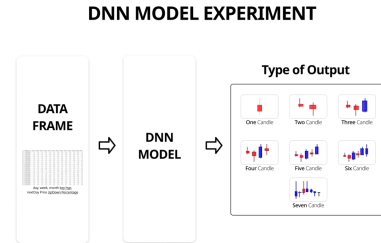


Fig. 3 Overview of proposed DNN model

데이터 전처리에 이어 3.2에서는 DNN 모델 및 실험 과정을 설명한다. 그림 4는 DNN 모델 예측값을 학습시키기 위한 시가 변화율 계산과정을 보여준다.

DNN 모델의 입력값 Input은 당일치를 포함하는 일, 주, 월 주식 차트의 key-hyp이고, 예측하는 Output 값은 다음 날 주식 차트의 각 세분화된 기간들의 시가변화율이다.

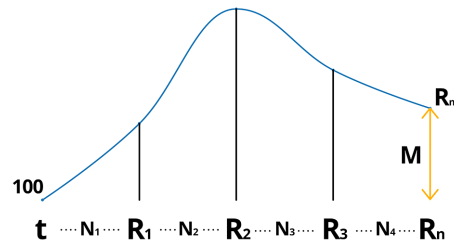


Fig. 4 Graph illustration for calculating the percentage change in stock price

$$R_1 = 100 + 100 \times \frac{P_1}{100} \quad (4)$$

$$R_n = R_{n-1} + R_{n-1} \times \frac{P_n}{100} \quad (n = 2, \dots, 7)$$

$$K = |M - C|$$

R_n = 각 구간의 변동률 비교값

P_n = 각 구간의 변동률 예측값

K = 오차값

C = 최종 변동률 실제값

M = 최종 변동률 예측값 ($R_n - 100$)

N_s = 분할된 구간 ($s = 1, \dots, 7$)

식(4)는 딥러닝 모델의 오차 값을 계산하는 데 활용된다. 모델은 다음날 상대적 종로 시점 가격인 R_n 과 상대적 시작 시점 가격인 t 사이 다음날 하루 동안의 기간을 일정한 간격으로 분할한 각 구간 s 마다의 주가 변동률

인 P_n 을 예측한다. 모델의 s 개의 예측값인 각 구간의 주가 변동률은 P_n 으로서 표현되며 식 (6)을 이용하여 구해진 다음 날 하루 동안의 주가 변동률 M 값과 다음날 하루 동안의 실제 주가 변동률의 절댓값 차를 최종적인 오차값 K 로 활용하게 된다. 또한 다음날의 초기 시가를 모르기 때문에 현시점의 시가 t 를 100으로 잡아 다음날 하루 동안의 주가 변동률 M 값과 다음날 하루 동안의 실제 주가 변동률 C 을 비교할 수 있도록 설계하였다.

이러한 연구에 사용되는 DNN 모델의 input은 표 3과 같다. 표 상에 표시되어있는 O는 세분화 정도에 어떠한 Feature 값이 포함되어있는지 나타낸다.

Table. 3 Input subdivision type Table

Element \ Form	1	2	3	4	5	6	7
key-hyp-day	O	O	O		O		
key-hyp-week	O	O		O		O	
key-hyp-month	O		O	O			O

본 연구 과정에서는 이러한 모델의 학습 과정에서 input 값의 형식은 표 3에 따라, Output 값의 형식은 표 2의 Fluctuation Rate Column에 따라 바꾸어가며 실험하고 비교하였고 실험 결과 예측된 마지막 예측값이자 예측 날 날짜, 실제 마지막 시가의 값을 MAE로 Loss 값을 구하여 평가하였다.

본 연구 과정은 모델의 세분화 예측 정도에 따라 모델이 받아들이는 정보의 비중이 어떤 식으로 달라지는지를 파악하기 위하여 존재한다. 모델의 모든 은닉층의 활성화 함수는 ReLU(Rectified Linear Unit)를 사용하였고, 최적화 알고리즘은 Adam을 사용하였다.

- batch size = 32
- Learning Rate = 0.0005
- train:test = 81182: 19035
- up:down = 10000+ α :10000- α (α 는 100 미만의 작은 변화로서, 복원추출 시마다 값이 달라짐)

먼저 각 모델에서 각 기간의 key-hyp 들이 각 모델에 계 주는 영향도를 분석하였다. 분석을 위해, 테스트 세트에서의 3개 key-hyp 중 영향도를 확인하고자 하는 기간의 key-hyp만 랜덤하게 섞어 기존 모델을 사용하여 모델의 성능을 나타내는 지표인 loss 값을 구한 후 기존 test set에서의 loss 값과의 차이 비교한다. 이러한 차이가 더욱 벌어질수록 그 기간의 key-hyp는 모델이 test set을 예측하는 데 더욱 중요한 역할을 하였다고 판단할 수

있다. 따라서 이 오차를 input 값이 모델의 성능에 끼치는 영향도를 나타내는 지표로 사용하였다. 결과는 Input 형식을 1,2,3,4로 4번 반복하여 결과 파트의 표 4과 같이 4가지 관점에서 그래프로 제시한다.

III. 성능평가

3.1. Output 값의 개수에 따른 모델 성능 탐구

본 실험 결과에서는 Input과 Output의 형식에 따라, AE-DNN 모델의 예측값과 실제 값을 차이 값인 Loss 값과 등락 여부의 정확도인 Accuracy 값이랑 비교하였다. 표 8은 그림 5-8 그래프의 X축과 Y축의 형식이다.

Table. 4 Presenting graphs from a total of 4 perspectives (Fig.5-8 Table of the x and y axes of the graph)

	Fig.5	Fig.6	Fig.7	Fig.8
x	Input	Output	Input	Output
y	Loss	Loss	Accuracy	Accuracy

표 4는 그림 6, 8를, 표 5는 그림 5, 7의 그래프 색깔 형식을 나타낸다.

Table. 5 Fig.6, 8 Graph Color Type Table

	Red	Green	Blue	Mint	Pink	Yellow	Black
key-hyp-day	O	O	O		O		
key-hyp-week	O	O		O		O	
key-hyp-month	O		O	O			O

Table. 6 Fig.5, 7 Graph Color Type Table

	Red	Green	Blue	Mint	Pink	Yellow	Black
Segmentation Num	1	2	3	4	5	6	7

그림 5-8은 비교한 결과를 시각화한 차트이다.

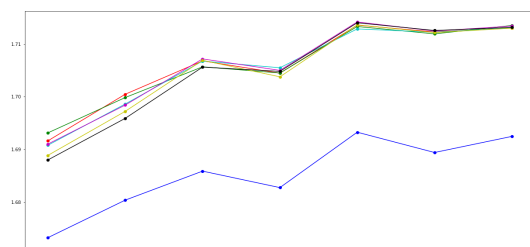


Fig. 5 Loss value graph of input value format diversification experiment

위 그래프 그림 5을 해석하면 7가지 Output 종류의 모델 전체가 input 개수가 점점 증가할수록 loss 값도 점점 증가하는 경향을 보인다. 특이한 것은, output 개수가 3개인 모델은 대체로 loss 값이 다른 모델과 비교하여 300 정도의 차이를 보이는 것으로 도출되었다. 이는 매우 이례적인 경우로서 다음날의 시가를 예측하는데 가장 적합한 Output 개수가 3이라는 것을 의미한다.

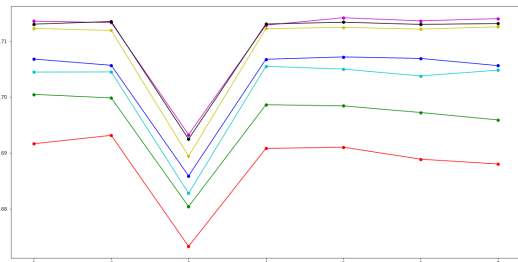


Fig. 6 Loss value graph of output value format diversification experiment

위 그래프 그림 6을 해석하면 7가지 Output 종류의 모델 전체가 output 개수가 점점 증가할수록 2를 제외한 loss 값은 약한 하강 경향을 보인다. 모델 중 가장 특이한 경향을 보이는 것은 output 개수가 3개인 모델인데 이 모델은 대체로 loss 값이 다른 모델과 비교하여 300 정도의 차이를 보이는 것으로 도출되었다. 이는 매우 이례적인 경우로서 다음날의 시가를 예측하는데 가장 적합한 output 세분화 정도가 3이라는 것을 의미한다.

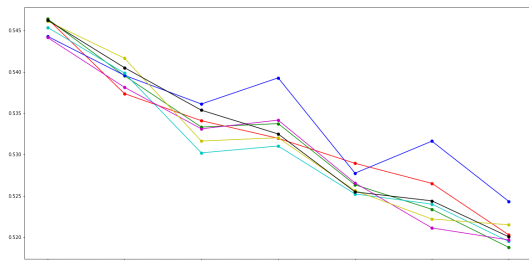


Fig. 7 Accuracy graph of input value format diversification experiment

위 그래프 그림 7에서 점점 Input의 형식이 단순화됨에 따라 정확도가 낮아지는 것을 볼 수 있으며, 대체로 Input 값의 형식을 다양하게 주는 것이 모델의 정확도를 높인다는 것을 알 수 있었다.

IV. 결론

본 연구에서 다음날의 시가 변동률을 계산할 때 다음날 최종적인 변동률을 모델의 예측 값으로 두고 모델을 설계하기보다, 최종 변동률을 예측하는데 필요한 이전 기간의 분할된 봉 사이의 변동률들을 모델의 예측값으로 두어 다음날의 변동률을 구하는 방식이 더욱 유리하다는 사실을 알 수 있었다. 아울러 그 과정에서 가장 유리한 세분화 정도는 3분할이라는 결과를 도출해냈으며 이러한 결과는 앞으로의 딥러닝을 이용한 다음날의 주가 예측에서 큰 도움을 줄 수 있을 것으로 예상된다[9].

REFERENCES

- [1] B. Li and S. Kim "LSTM artificial neural network prediction of stock prices in China," *Journal of Northeast Asian Studies*, vol. 32, no. 2, pp. 61-84, 2020.
- [2] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Udipi, pp. 1643-1647, 2017.
- [3] S. Bae and B. Choi, "Chart-based Stock Price Prediction by Combing Variation Autoencoder and Attention Mechanisms," *Information Systems Review*, vol. 23, no. 1, pp. 23-43, Feb. 2021.
- [4] Y. Oh and Y. Kim, "A Two-Phase Hybrid Stock Price Forecasting Model: Cointegration Tests and Artificial Neural Networks," *The KIPS Transactions: Part B*, vol. 14B, no. 7, pp. 531-540, Dec. 2007.
- [5] X. Zhou, Z. Pan, G. Hu, S. Tang, and C. Zhao, "Stock Market Prediction on High-Frequency Data Using Generative Adversarial Nets," *Mathematical Problems in Engineering*, pp. 1-11, Apr. 2018.
- [6] J. Lee, R. Kim, Y. Koh, and J. Kang, "Global Stock Market Prediction Based on Stock Chart Images Using Deep Q-Network," *IEEE Access*, vol. 7, pp. 167260-167277, Nov. 2019.
- [7] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PloS one*, vol. 12, no. 7, p. e0180944, Jul. 2017.
- [8] H. -J. Song and S. -J. Lee, "A Study on the Optimal Trading

Frequency Pattern and Forecasting Timing in Real Time Stock Trading Using Deep Learning: Focused on KOSDAQ,” *The Journal of Information Systems*, vol. 27, no. 3, pp. 123-140, Sep. 2018.

- [9] Github/YouTube of Author-https://youtu.be/y8CM_OsbpVg,
<https://github.com/doch2/AE-DNN-model-data>,
https://github.com/ytgw0/AE_dnn-and-DNN_experiment,
<https://youtu.be/fBZ8UDx8VZY>



여태건우(Tae Geon Woo Yeo)

한국디지털미디어고등학교 웹프로그래밍과
※ 관심분야 : 딥러닝(메타학습, 생성모델, Transformer), 뇌과학



유도희(Dohui Ryu)

한국디지털미디어고등학교 웹프로그래밍과
※ 관심분야 : 머신러닝, 추천시스템, 클라우드 컴퓨팅



남정원(Jungwon Nam)

한국디지털미디어고등학교 웹프로그래밍과
※ 관심분야 : 인공지능경망, 클라우드 컴퓨팅



오하영(Hayoung Oh)

성균관대학교 소프트웨어융합대학 글로벌융합학부 부교수
※ 관심분야 : 머신러닝, 딥러닝, 추천시스템, 데이터분석