

<LSTM 주가 예측 모델의 보조 지표로의 활용 가능성 검증>

[SR 계량팀]

22기 김민성 22기 이현민 22기 장규범 23기 최인재 23기 한승민

Abstract

본 연구는 딥러닝 기술을 주식시장의 가격 예측에 활용하려는 시도가 늘어남에 따라, 과거 가격 데이터를 통해 익일의 주가의 등락을 예측하는 딥러닝 주가 예측 모형의 투자 보조 지표로서의 활용 가능성을 검증하였다.

구체적으로는 딥러닝 중에서도 RNN(순환 신경망)을 시계열 데이터 예측에 더 적합하게 보완한 LSTM(장단기 메모리) 모형을 이용하여, 2년치 데이터를 훈련용 데이터셋과 검증용 데이터셋으로 분리하고 100회 학습시킨 후 익일 주가의 향방을 예측하였다. 이때, 실제 주가의 향방과 일치할 경우 win, 다를 경우 lose로 판단하여 승률을 계산하였다.

모델에 사용한 매개변수는 LSTM 주가예측 모델의 성능 검증 선행 연구에서 가장 효율적인 성능을 보였던 활성화 함수와 최적화 함수의 조합인 쌍곡 탄젠트 함수(Hyperbolic Tangent), 하드 시그모이드 함수(Hard sigmoid), Adam 함수(Adaptive Moment Estimation)를 사용하였다.

예측 결과, 총 153개의 종목 중 84개의 종목과 익일 주가 향방이 일치했고, 69개 종목에서는 예측한 향방과 실제 향방이 달랐다. 이에 따라, 54.902%의 승률을 도출하였다.

목차

I . Introduction

II. Literature review

III. Method

- 3. 1. Artificial Neural Network (ANN)
- 3. 2. Recurrent Neural Network(RNN)
- 3. 3. Long Short Term Memory(LSTM)
- 3. 4. 활성화함수(Activation function)
- 3. 5. 최적화함수(Optimization function)
- 3. 6. 손실함수(Loss function)

IV.Data Description

- 4. 1. 데이터의 출처
- 4. 2. 데이터 전처리 방법
- 4. 3. 훈련용 데이터셋과 검증용 데이터셋의 분리
- 4. 4. LSTM 주가 예측 모형

V. main result

VII. Conclusion

VIII. APPENDIX

I. Introduction

인공 지능 기술의 발전으로 컴퓨터에게 데이터를 학습시켜 여러가지 데이터를 예측하려는 시도가 과거부터 이어져 오고 있다. 여기서 학습이란, 훈련 데이터로부터 가중치 매개변수의 최적값을 자동으로 획득하는 것을 뜻한다. 이러한 인공 지능 기술 중 딥러닝 기술을 주식 시장에 활용하려는 시도가 활발히 이루어지고 있다.

LSTM(Long-Short-Term-Memory-models)은 딥러닝의 일종인 순환신경망(RNN, Recurrent Neural Network)를 발전시킨 모델이다. LSTM은 기존 순환 신경망 모델이 관련 데이터가 있는 지점과 그 데이터를 사용하는 지점 사이의 거리가 멀 경우 학습 능력이 크게 저하되는 현상을 개선하여, 장기 시계열 데이터에도 적용할 수 있게 되었다. 따라서 LSTM을 사용하여 과거의 주식 가격의 패턴을 학습하고, 미래 주가 흐름을 예측하는 데에도 유용하게 여겨지고 있다.

하지만 LSTM을 이용한 주가 예측 모델만을 실제 주식 투자 전략으로 활용하기에는 아직 어려움이 있어 보인다. 그 이유로는 첫째, 주가가 패턴을 보이지 않는 비추세가 지속된 구간의 경우, LSTM 모델에서 패턴을 발견하기 힘들다. 종목에 따라 추세를 지속하지 않는 구간이 길어질 경우, 의미 없는 값을 반환하는 경우가 발생한다.

둘째, 과거 데이터만을 이용해 예측한 주가 데이터는 한계가 있다. 주가는 과거 주가 외에도 주식의 내재적 가치, 시장상황 등 수많은 요소들의 영향을 받기 때문에 특정 사건이 발생한 시점에 주가가 급등락 하는 경우가 많기 때문에 과거 주가 패턴 학습만으로는 해당 변화를 효율적으로 감지할 수 없다.

셋째, 학습 데이터의 양이 적을 경우 예측 성능이 굉장히 낮아진다. 상장일이 비교적 최근인 종목의 경우 학습 횟수를 늘리더라도 훈련 데이터의 양이 적어, 모델에서 패턴을 발견하기 힘들기 때문에, 일부 종목에 대해서는 모델을 적용하기 힘들다.

따라서, 본 연구에서는 먼저 선행 연구¹⁾에서 가장 높은 예측 성능을 보인 매개변수를 활용하여 주가 예측 모델을 만들고, 예측 가격을 직접 활용하는 것이 아니라 등락에 대한 예측만을 이용하여 투자자의 매수/매도 결정에 도움을 줄 수 있을지 보조 지표로의 활용 가능성을 판단해 보고자 한다.

등락 예측 성공 여부는, 예측 가격을 통한 전일 대비 상승/하락 여부가 실제 주가의 전일 대비 상승/하락 여부와 들어맞는지 비교해 승률을 측정해 확인하는 방식으로 진행하였다. 이때, 주식 종목은 KO SPI200으로 한정했으며, 상장일이 2년 이하인 종목은 제외하였다.

II. Literature review

정종진, 김지연(2020)은 LSTM 주가예측 모델의 매개변수별 성능을 측정하였다. 훈련에는 애플, 어도비, 아마존, 컴캐스트, 코스트코, CISCO, 인텔, 마이크로소프트, 엔비디아, 펍시 등 10개, 예측에는 구글,

페이스북, 페이팔, 브로드컴, 퀄컴 등 5개 종목이 사용되었다. 전체 데이터셋은 시간순으로 3:3:4의 비율로 분할하였고, 최종 성능지표는 셋의 평균값으로 하였다. LSTM의 매개변수는 초기화(Initialization) 방법, 정규화(Regularization) 방법, 활성화 함수(Activation Function)의 세 가지로 구분하였고, RMSE(Root Mean Square Error) 손실함수를 사용한 성능평가를 통해 최적의 매개변수를 도출하였다. 그 결과 초기화 방법으로는 kernel initializer에서 Xavier normal(glorot normal), recurrent initializer에서 random normal, bias initializer에서는 he uniform 방법을 적용한 조합이 최적이었다. 정규화 방법으로는 kernel regularizer와 activity regularizer를 조합했을 때의 성능지표가 가장 좋았다. 최적화 함수는 기본 활성화함수로 tanh, 순환 활성화 함수로 hard sigmoid를 사용했을 때 가장 성능이 좋았으며 매우 근소한 차이로 (elu, hard sigmoid)조합이 두 번째로 성능이 좋았다. 기본 활성화함수가 tanh일 때는 최적화함수가 Adam일 때, 기본 활성화함수가 elu일 때는 최적화함수가 Adamax일 때 최적의 성능을 보였다.

신동하, 최광호, 김창복(2017)은 딥러닝을 활용한 주가예측 모델에서 보조지표의 효용성을 분석하고 DNN, LSTM 모델의 예측성능을 비교했다. 그 결과 대부분의 보조지표는 추가 시 예측 성공률을 저하하는 현상을 보였고, DNN 모델보다 시계열 데이터 예측에 적합한 LSTM 모델의 예측율이 15% 정도 높았다. 성능 측정 시 학습 데이터는 시가, 고가, 저가, 종가, 거래량, Stochastic, CCI, 등락구분 5일, 등락구분 20일, 환율, 환율 이동평균 5일, 전산업생산지수 등을 사용하였다.

이낙영, 오경주(2019)는 디노이징 필터와 LSTM을 조합한 주가예측 모델을 제시했다. 데이터는 2017년 1월 2일부터 2018년 12월 28일까지의 KOSPI200 선물지수 중 시가, 고가, 저가, 종가, 거래량을 사용하였고, 이 데이터의 노이즈를 MA(Moving Average), 사비츠키-골레이(savitzky-golay) 필터를 통해 제거했다. 그 후 Adam 최적화기법을 적용한 LSTM으로 종가 지수를 예측했다. 그 결과 디노이징 필터를 적용한 LSTM 모델이 디노이징 필터를 적용하지 않은 모델보다 예측 성능이 좋았다.

김수현(2020)은 한국 주식시장에서 LSTM 기반 모형의 예측성을 분석했다. 데이터는 Data Guide Pro에서 추출한 2010년 1월 기준 시가총액 상위 100개 주식의 일간 통합 수익률을 사용했다. LSTM은 12개월 동안의 데이터로 학습하였고, 손실함수는 MSE(Mean Square Error)를 사용하였다. 이 모델을 기반으로 100개 주식 전체를 하나로 통합한 1개월, 3개월, 6개월 매수-공매도(Long-Short) 포트폴리오를 구성하여 테스트하였다. 그 결과 1개월, 3개월 포트폴리오는 유의미한 수익성을 보였고, 6개월 포트폴리오는 그렇지 못했다. 학습기간이 적을수록 유의미한 수익성을 보이는 기간은 짧았다. 따라서 LSTM을 활용한 주가예측은 장기적 예측보다 단기적 예측에 적합하다는 점이 시사되었다.

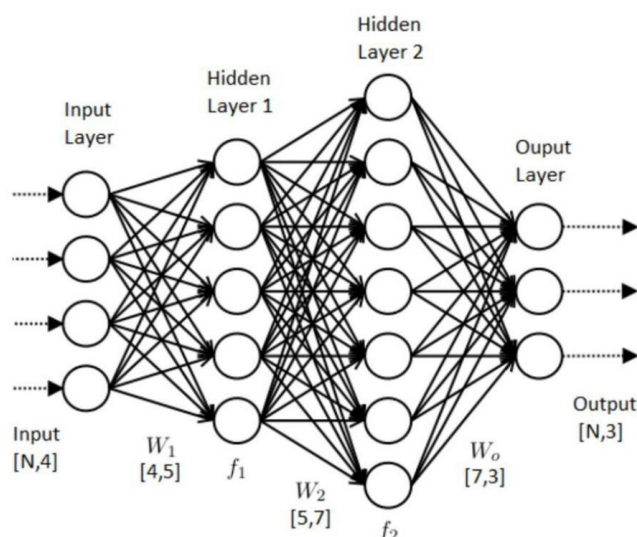
최훈(2021)은 주요 신경망 모델인 ANN, DNN, k-NN, CNN, RNN, LSTM의 주가 예측 성공률을 비교하였다. 그 결과 주가 속성 중 시가, 고가, 저가, 종가, 거래량을 사용한 LSTM 모델이 84%로 예측 성공률이 가장 높았다.

김하얀, 주귀화, 김석찬(2021)은 LSTM을 활용하여 삼성전자의 이전 4일의 종가를 바탕으로 다음 날의 종가를 예측하는 모델을 구현하였다. 학습 데이터는 전체의 80%, 검증 데이터는 전체의 20%로 구성하였고, 활성화함수는 tanh, 손실함수는 huber, 최적화함수는 adam을 활용하였다. MSE(Mean Squared Error)로 모델의 정확도를 측정한 결과 0.0004의 값이 나와 실제 값의 경향성을 높은 정확도로 따라갔음이 확인되었다.

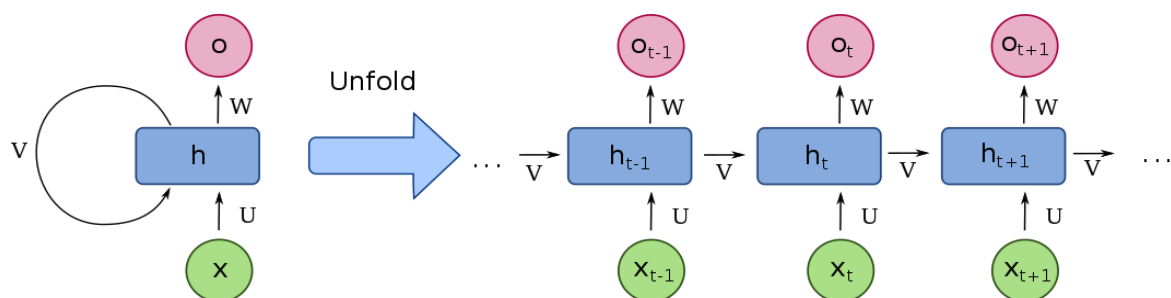
III. Method

3. 1. Artificial Neural Network (ANN)

인공신경망(ANN)은 시냅스의 결합으로 네트워크를 형성한 인공 뉴런(노드)이 학습을 통해 시냅스의 결합 세기를 변화시켜, 문제 해결 능력을 가지는 모델 전반을 가리킨다. 인공신경망은 입력계층, 은닉계층, 출력계층으로 구성된다. 데이터 셋의 특징(feature)의 수는 차원이나(dimensionality) 입력계층에 있는 노드의 수를 결정한다. 은닉계층에서 노드는 입력값을 출력값이나 예측값으로 변환시키기 위해 활성화함수를 사용한다. 출력계층은 다양한 출력값의 확률 벡터들을 생성하고, 비용(cost) 혹은 오차(error)가 가장 적은 벡터를 선택한다.



3. 2. Recurrent Neural Network(RNN)



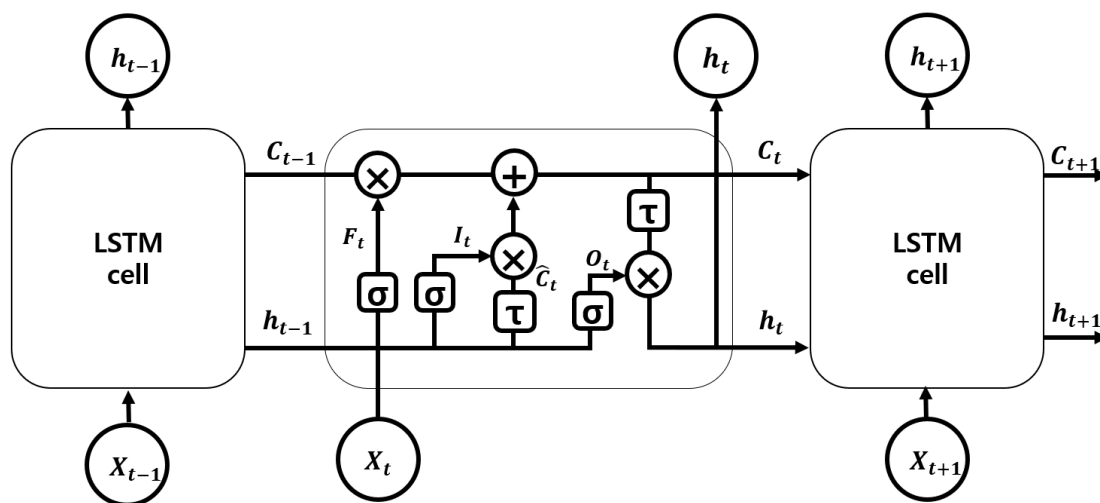
순환신경망(RNN)은 인공 신경망의 한 종류로, 순서(sequence)가 있는 데이터에서 맥락을 파악하여 다음 데이터를 예측하는 것을 목적으로 고안되었다. RNN에서는 데이터가 하나의 셀 안에서 순환하게 되는데, 이러한 이유로 RNN은 순서가 정해진 여러 데이터를 계속해서 받아들일 수 있어 시계열 예측에 적합하다.

그러나 기본 RNN 모델에서는 이와 같은 특성 때문에 가중치가 중복 조정되어 생기는 문제가 있다. 가중치가 1보다 작은 경우 입력 데이터의 양이 많아질수록 학습 과정에서 가중치의 크기가 작아지게 되는데, 이때 많은 양의 데이터를 처리한다면 가중치가 0에 가깝게 되어 학습이 제대로 되지 않는 '기울기 소실(vanishing gradient)' 문제가 발생한다. 반대로 가중치가 1보다 큰 경우, 많은 양의 데이터 처리 시 가중치가 발산하여 마찬가지로 학습이 제대로 되지 않는 '기울기 증폭(exploding gradient)' 문제가 발생한다. 이 문제를 해결하기 위해 LSTM 모델이 고안되었다.

3. 3. Long Short Term Memory(LSTM)

LSTM은 셀 스테이트(Cell State, C_t)를 통해 과거 학습 결과를 큰 변함없이 전달하는 구조로 기울기 소실 문제를 해결한 RNN의 한 종류이다. LSTM unit은 세 가지의 gate(input gate, output gate, forget gate)와 cell로 구성되어 있다. LSTM은 첫 번째 단계로 sigmoid 함수를 이용해 삭제할 정보를 결정한다. 두 번째 단계에서 또 다른 sigmoid 함수와 tanh 함수를 이용해 새로운 정보가 셀 스테이트에 저장될지 결정한다. 세 번째 단계에서는 셀 스테이트를 업데이트하고 마지막 sigmoid 함수와 셀 스테이트에서 나온 출력을 통과시킨 마지막 tanh 함수로 어떤 값을 출력할지를 결정한다.

LSTM의 작동 과정을 그림과 수식으로 표현하면 다음과 같다.



$$h_t = O_t \otimes \tau(C_t)$$

$$O_t = \sigma(h_{t-1}W_o + x_oU_o + b_o)$$

$$F_t = \sigma(h_{t-1}W_f + x_tU_f + b_f)$$

$$\hat{C}_t = \tau(h_{t-1}W_c + x_tU_c + b_c)$$

$$I_t = \sigma(h_{t-1}W_i + x_iU_i + b_i)$$

$$C_t = F_t \otimes C_{t-1} \oplus I_t \otimes \hat{C}_t$$

F_t : forget gate

U_f : forget gate에 대한 input weight vector

I_t : input gate

U_i : input gate에 대한 input weight vector

O_t : output gate

U_o : output gate에 대한 input weight vector

σ : sigmoid function

U_o : memory cell에 대한 input weight vector

τ : Tanh function

W_f : forget gate에 대한 recurrent weight vector

X_t : t에서의 input value

W_i : input gate에 대한 recurrent weight vector

h_t : t에서의 hidden state output

W_o : output gate에 대한 recurrent weight vector

C_t : t에서의 cell state

W_c : memory cell에 대한 recurrent weight vector

X_{t-1} : t-1에서의 input value

b_f : forget gate에 대한 input bias vector

h_{t-1} : t-1에서의 hidden state output

b_i : input gate에 대한 input bias vector

C_{t-1} : t-1에서의 cell state

b_o : output gate에 대한 input bias vector

\otimes : pointwise 곱

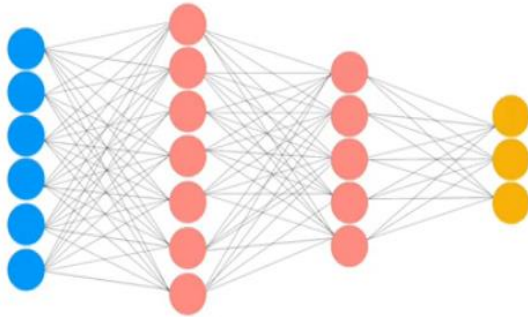
b_c : memory cell에 대한 input bias vector

\oplus : pointwise 합

여기서 F_t, I_t, O_t 는 각각 forget gate, input gate, output gate이다. σ 는 sigmoid 함수로, input value를 0과 1 사이에 있는 값으로 변환시키는 활성화함수(activation function)이다. τ 는 Tanh 함수로 output value를 -1에서 1 사이의 값으로 변환한다. x_t, h_t, C_t 는 각각 t에서의 input value, hidden state output, 셀 스테이트를 나타낸다. h_{t-1}, C_{t-1} 은 각각 t-1에서의 hidden state output, 셀 스테이트를 나타낸다. 연산자 \otimes, \oplus 은 각각 pointwise 곱과 합이다. U_f, U_i, U_o, U_c 는 각각 forget gate, input gate, output gate, memory cell에 대한 input weight vector다. W_f, W_i, W_o, W_c 는 각각 forget gate, input gate, output gate, memory cell에 대한 recurrent weight vector이다. b_f, b_i, b_o, b_c 는 각각 forget gate, input gate, output gate, memory cell에 대한 input bias vector이다.

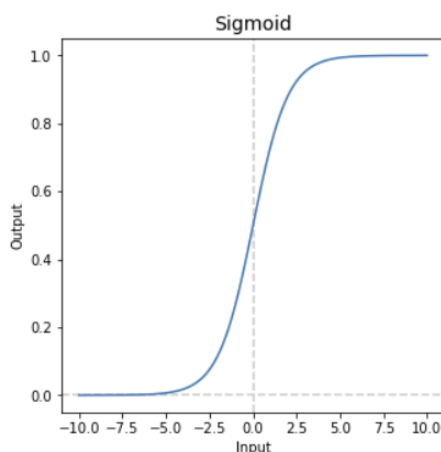
3. 4. 활성화함수(Activation function)

뉴런과 뉴런 사이에 정보가 전달될 때, 입력 받은 정보는 그대로 전달되지 않고 재해석되어 다음층으로 출력된다. 이때 활성화 함수가 입력 받은 신호의 총합을 바꾸어 출력하는 신경세포의 세포체 역할을 한다. 우리가 만든 인공 신경망은 '선형 결합' 형태로 이루어져 있다. 각 node에서 Input값들을 받아, 가중치를 곱해서 더하여 다음 레이어의 노드를 구성하는 형태이다.



만일 선형 결합 상태에서 선형 함수를 사용한다면, 은닉층(hidden layer)의 존재와 관계없이 결과가 선형 형태로 나오기 때문에 적용하는 의미가 적어진다. 따라서, 선형적이고 단순한 예측에서 벗어나 좀더 복잡한 예측을 하기 위해서 비선형의 활성화 함수가 필요하다. 활성화 함수는 종류에 따라 다음층으로 출력 혹은 무시되는 정보의 범위와 특징을 결정한다. 그렇기에 목적과 상황에 따라 적절한 활성화 함수를 설정하는 것이 필요하다. 활성화 함수의 종류로는 sigmoid, tanh, relu, hard sigmoid 함수 등이 있다.

3. 4. 1. sigmoid

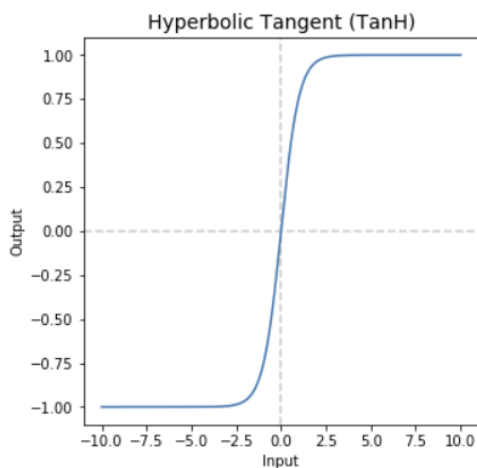


$$\text{Sigmoid}(x) = \sigma(x) = \frac{1}{1 + \exp(-x)}$$

sigmoid 함수는 입력값이 큰 값을 가질수록 1에, 작은 값을 가질수록 0에 근사하는 특징이 있다. 출력값은 0~1 사이의 연속적인 실수이며, 그래프는 점 (0, 0.5)에 대칭인 매끄러운 S자 형태를 띈다. 그렇기에 Sigmoid는 모든 점에서 미분이 가능하며 기울기가 급격하게 변할 때 발생하는 기울기 폭주 현상이 발생하지 않는다는 장점이 있다.

하지만 sigmoid 함수의 미분값은 0~0.25 범위이며, 가운데에서 벗어난 값들은 기울기가 0에 근사한 값이 나오게 된다. 그렇기 때문에 층의 개수가 많아지면 0.25 이하의 값이 여러 번 곱해지면서 학습 성능이 저하되는 '기울기 소실 문제'가 발생한다. 또한 출력값의 중앙값이 0이 아니기 때문에 항상 양수를 출력해서 입력 가중치 합보다 출력 가중치 합이 커질 수 있는 편향 이동이 존재한다.

3. 4. 2. tanh

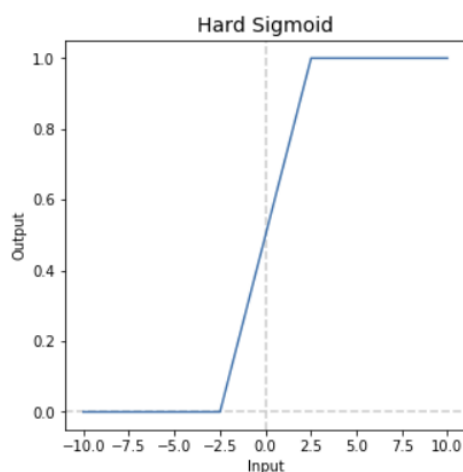


$$\text{Tanh}(x) = \tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}$$

tanh함수는 쌍곡선 함수 중 하나로 출력값은 -1~1 사이의 연속적인 실수이며 원점에 대해 대칭이다. sigmoid와 같이 매끄러운 S자 형태를 띄고 있지만 sigmoid의 한계점을 보완했다. sigmoid와 달리 중앙값이 0이기 때문에 편향이동 문제가 발생하지 않는다. 또한 미분값의 범위가 0~0.25인 sigmoid 함수와 달리 미분값의 범위가 0~1이기에, 기울기 소실 현상이 적게 나타난다. 따라서 은닉층에서 레이어를 쌓을 때 sigmoid함수보다는 tanh함수를 사용하는 것이 효율적이다.

하지만 sigmoid함수보다 범위가 넓은 뿐 그래프의 모양은 sigmoid함수와 동일하기에 기울기 소실 문제를 완전히 해결하지는 못한다.

3. 4. 3. hard sigmoid



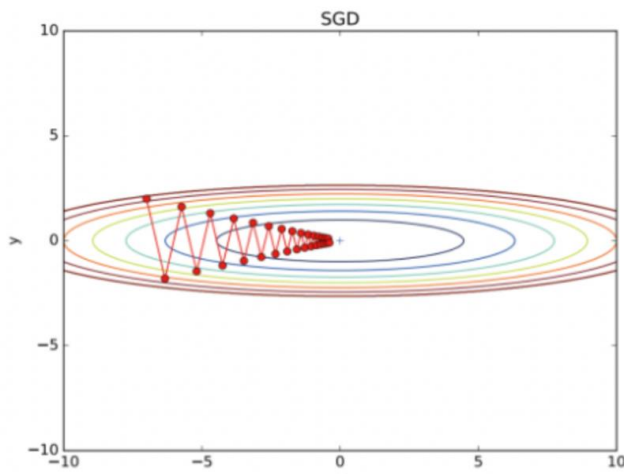
$$f(x) = \begin{cases} 1 & (x > 2.5) \\ 0.2x + 0.5 & (-2.5 \leq x \leq 2.5) \\ 0 & (x < -2.5) \end{cases}$$

Hard sigmoid 함수는 sigmoid 함수를 단편적으로(piece-wise) 직선화 시킨 형태를 갖는다. -2.5보다 작은 값에는 0을, [-2.5, +2.5]에는 1을 출력값으로 가지며 2.5보다 큰 값에는 0~1로 선형 증가한다. Hard sigmoid 함수는 지수함수를 계산할 필요가 없고 미분 함수가 단순하기 때문에 sigmoid 함수보다 계산속도가 빠르다는 장점이 있다.

하지만 sigmoid 함수보다 단순하기 때문에 오차가 더 높게 나타난다.

3. 5. 최적화함수(Optimization function)

딥러닝은 예측값을 실제값과의 차이 크기를 나타내는 손실함수를 통해 예측값이 얼마나 좋은지 판단한다. 이 손실함수값을 최소화하는 방향으로 학습을 진행하는데 그 과정을 최적화라고 한다. 최적화 함수는 데이터에 최적화된 파라미터인 가중치(weight)와 편향(bias)을 탐색한다. 최적화의 가장 기본적인 방법은 경사 하강법(SGD, Stochastic Gradient Descent)이다. 경사하강법은 1차 미분계수를 이용해 기울기가 0인 지점(손실된 값의 최저점)의 가중치를 찾도록 파라미터를 변경하는 방법이다. 하지만 움직임이 비효율적인 지그재그 형태라는 점과, 극소점이 여러개일 때 엉뚱한 최적화를 진행할 수도 있는 등 여러가지 문제가 있다. 이러한 문제들을 개선하기 위해 다양한 최적화 함수들이 연구되어 왔다. 그 중 adam은 딥러닝에서 가장 일반적으로 사용되는 최적화 알고리즘이다.

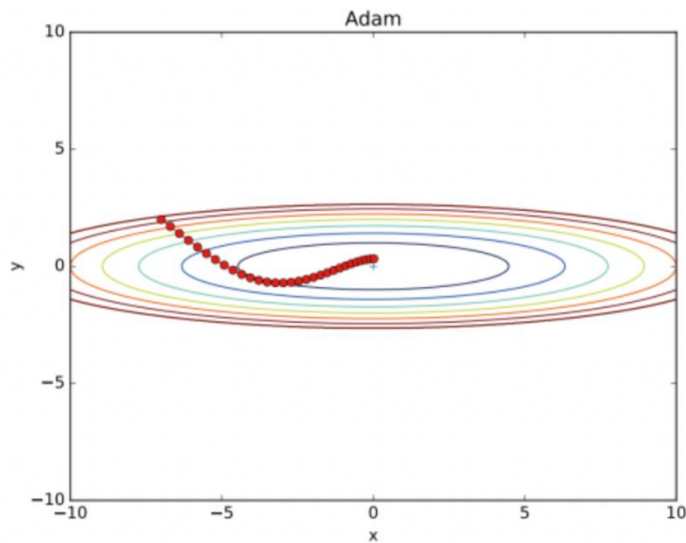


3. 5. 1. Adam (Adaptive Moment Estimation)

Adam은 파라미터 학습 방향을 개선한 Momentum의 장점과 파라미터 갱신 크기를 개선한 AdaGrad 장점이 결합된 알고리즘이다.

Momentum은 경사하강법에 관성을 이용한 기법으로 이전에 이동했던 방향을 기억해 더 효율적으로 움직인다. 또한 극솟값이 여러 개일 때, 엉뚱한 최저점(local minimum)에 빠져도 관성에 의해 추가로 이동하여 그 문제를 해결한다. AdaGrad는 파라미터를 조정할 때 목적지점에 가깝다고 보여지는 정도에 따라 step size(learning rate)를 크게, 작게하여 갱신정도에 차이를 두고 이동한다는 특징이 있다.

두 최적화 알고리즘이 합쳐진 Adam은 학습의 방향과 stepsize 모두 개선해, momentum과 같이 공이 구르는 듯한 움직임을 보이며 AdaGrad와 같이 파라미터마다 적응적으로 갱신 정도를 조정한다. Adam은 구현하기 쉽고 계산이 효율적이며 메모리 요구사항이 거의 없다는 장점이 있다. 따라서 데이터와 파라미터가 많이 필요하고 기울기에 노이즈가 많거나 sparse한 경우에 적합하다.



3. 6. 손실함수(Loss function)

손실함수는 모델이 예측한 값과 실제 값의 차이를 도출하는 함수로, 모델의 정확도를 평가하는 지표로 사용된다. 본 모델에서 사용하는 손실함수는 MSE(Mean Square Error)로, 그 정의는 다음 수식과 같다.

$$MSE = \frac{1}{T} \sum_{t=1}^T (y_t - y_t')^2$$

T : 총 예측기간

y_t : t기 주식의 실제 가격

y_t' : t기 주식의 예측 가격

IV. Data Description

4. 1. 데이터의 출처

본 연구에서는 2020년 5월 3일부터 2022년 5월 3일까지의 2년치 KOSPI200의 주가 데이터를 이용하여 주가 예측을 진행하였다. 야후 Finance API는 국내외 데이터를 편리하게 가져올 수 있다는 장

점을 지니고 있으나, 국내 데이터를 취급할 때 종가 및 수정 종가가 부정확하거나 빈 데이터가 존재하는 등 문제가 발생할 위험이 있었다. 따라서 정확한 분석을 위해, 네이버 금융의 주가 데이터를 크롤링하여 사용하였다.

대량의 데이터를 저장하고 이용하기 위해서, 크롤링한 데이터는 오픈소스 데이터베이스 관리 시스템인 MariaDB에서 저장하여 관리하였다. 아래 이미지는 MariaDB에서 제공하는 GUI기반 HeidiSQL에 접속하여 종목명 데이터와 OHLVC데이터를 불러온 것이다.

[이미지1 : HeidiSQL]

호스트: 127.0.0.1	데이터베이스: investor	테이블: company_info	데이터	쿼리			
이름 ^	행	크기	생성됨	업데이트	엔진	코멘트	유형
company_info	2,511	144.0 KiB	2022-03-13 23:05:...		InnoDB		Table
daily_price	1,184,087	163.8 MiB	2022-03-13 23:05:...		InnoDB		Table

[이미지2 : HeidiSQL 내 OHLVC 및 종목명]

investor.daily_price: 1,184,087 행 (중) (대략적), 제한 수: 1,000

A Z ↓ code	Z A ↓ date	open	high	low	close	diff	volume
000020	2022-05-04	12,650	12,750	12,400	12,500	200	96,221
000040	2022-05-04	861	865	852	853	6	203,201
000050	2022-05-04	15,900	15,900	15,450	15,650	250	13,625
000060	2022-05-04	43,750	44,500	43,500	44,250	300	113,909
000070	2022-05-04	85,300	85,700	84,900	84,900	100	4,044
000080	2022-05-04	37,700	37,750	36,650	37,100	700	239,745
000100	2022-05-04	60,200	60,700	59,000	59,700	800	223,913
000120	2022-05-04	121,000	122,000	120,500	122,000	1,000	23,070
000140	2022-05-04	14,150	14,300	13,850	13,950	200	19,784
000150	2022-05-04	90,200	92,800	88,200	90,200	600	114,883
000180	2022-05-04	2,945	2,980	2,920	2,940	0	667,646
000210	2022-05-04	65,400	65,700	63,400	63,600	1,800	98,376
000220	2022-05-04	7,480	7,530	7,400	7,400	100	57,052
000230	2022-05-04	42,350	43,800	40,000	43,400	1,050	658,200
000240	2022-05-04	14,300	14,400	14,100	14,200	200	57,991
000250	2022-05-04	43,250	44,500	42,950	43,500	450	136,163
000270	2022-05-04	82,800	83,600	82,800	83,500	0	860,580
000300	2022-05-04	1,230	1,240	1,220	1,220	15	417,063
000320	2022-05-04	12,300	12,300	11,900	12,000	300	39,531
000370	2022-05-04	4,695	4,850	4,655	4,805	110	451,701
000390	2022-05-04	9,590	9,630	9,510	9,600	10	27,347
000400	2022-05-04	2,000	2,035	1,970	2,035	50	488,448
000430	2022-05-04	3,450	3,650	3,395	3,455	45	772,397
000440	2022-05-04	36,100	36,150	34,050	34,500	2,050	303,484
000480	2022-05-04	83,200	83,200	82,100	82,500	200	273
000490	2022-05-04	17,900	18,000	16,550	16,650	1,500	2,387,686
000500	2022-05-04	26,550	28,150	26,500	26,850	350	127,921
000520	2022-05-04	8,730	8,760	8,460	8,560	230	126,754
000540	2022-05-04	4,050	4,050	3,985	4,000	50	61,014

각 종목 코드 및 회사명은 한국거래소 정보데이터시스템에서 스크레이핑하여 MariaDB내에 저장하여 사용하였다.

[이미지3: HeidiSQL 기업명 및 종목코드]

investar.company_info: 2,511 행 (총) (대략적), 제한 수: 1,000

 code	company	last_update
000020	동화약품	2022-05-05
000040	KR모터스	2022-05-05
000050	경방	2022-05-05
000060	메리츠화재	2022-05-05
000070	삼양홀딩스	2022-05-05
000080	하이트진로	2022-05-05
000100	유한양행	2022-05-05
000120	CJ대한통운	2022-05-05
000140	하이트진로홀딩스	2022-05-05
000150	두산	2022-05-05
000180	성장기업지주	2022-05-05
000210	DL	2022-05-05
000220	유유제약	2022-05-05
000230	일동홀딩스	2022-05-05
000240	한국엔컴퍼니	2022-05-05
000250	삼천당제약	2022-05-05
000270	기아	2022-05-05
000300	대유플러스	2022-05-05
000320	노루홀딩스	2022-05-05
000370	한화손해보험	2022-05-05
000390	삼화페인트공업	2022-05-05
000400	롯데손해보험	2022-05-05
000430	대원강업	2022-05-05
000440	중앙에너지스	2022-05-05
000480	조선내화	2022-05-05
000490	대동	2022-05-05
000500	가온전선	2022-05-05
000520	삼일제약	2022-05-05
000540	흥국화재	2022-05-05

4. 2. 데이터 전처리 방법

처리할 데이터의 양이 많고, 숫자가 클수록 연산 속도는 굉장히 느려진다. 따라서, 모든 주가 데이터를 0과 1 사이의 값으로 만드는 다음과 같은 MinMax Scaling 과정을 거쳤다.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

모든 데이터에서 최솟값을 빼서 분자에 넣고, 분모에는 데이터의 최대값에서 최솟값을 빼서 나누면, 모든 데이터를 0과 1 사이의 값으로 표준화할 수 있다. 이때, 0으로 나누는 에러가 발생하지 않도록 매

우 작은 값(10^{-7})을 분모에 더하여 계산하였다.이하는 MinMax Scaling된 삼성전자의 OHLVC 데이터 프레임이다. 연구에는 종가 데이터인 close 만을 사용하였다.

[이미지 4 : 삼성전자 OHLVC Scaling]

	open	high	low	volume	close
date					
2020-05-04	0.038328	0.020534	0.030733	0.216563	0.015064
2020-05-06	0.040650	0.022587	0.030733	0.118807	0.031286
2020-05-07	0.045296	0.024641	0.035461	0.067745	0.022016
2020-05-08	0.042973	0.025667	0.037825	0.085254	0.022016
2020-05-11	0.038328	0.023614	0.026005	0.097917	0.012746
...
2022-04-27	0.421603	0.357290	0.418440	0.119440	0.397451
2022-04-28	0.421603	0.357290	0.408983	0.104477	0.392816
2022-04-29	0.414634	0.400411	0.420804	0.217864	0.453071
2022-05-02	0.449477	0.400411	0.456265	0.070451	0.450753
2022-05-03	0.468060	0.416838	0.475177	0.071216	0.455388

496 rows × 5 columns

표에는 표현되지 않았으나 소수점 아래 17째자리까지 계산하여 분석에 활용하였다.

4. 3. 훈련용 데이터셋과 검증용 데이터셋의 분리

훈련용 데이터 셋과 검증용 데이터셋을 분리하여 사용하는 이유는, 훈련용 데이터셋만을 예측에 활용할 경우 훈련용 데이터에만 최적화되어 새로운 데이터를 올바르게 추정하지 못하는 '과적합(Overfitting)'현상을 피하기 위해서이다. 과적합은 모델의 성능을 떨어뜨리는 주된 원인이기 때문에 반드시 검증용 데이터 셋을 훈련용 데이터 셋과 분리해야만 한다. 학습 과정에서 사용되지 않은 검증용 데이터 셋을 사용함으로써, 학습이 잘 이루어 졌는지 새로운 데이터를 통해 점검할 수 있다.

이전 5일을 학습하여 다음 1일을 예측하는 모형을 만들 것이기 때문에, 총 491개의 데이터 중 70%인 343개의 데이터는 훈련용 데이터 셋으로 사용하였고, 나머지 148개의 데이터는 검증용 데이터 셋으로 사용하였다. 즉, 7:3으로 분리하였다.

4. 4. LSTM 주가 예측 모형

LSTM모형의 구축은 구글 Tensorflow에서 지원하는 Keras의 고수준 API를 사용하여 구현하였고, 분석에 사용되는 hyper parameter는 선행연구에서 보았던 가장 성능이 좋았던 매개변수를 사용하기 위해 다음 표와 같이 일관되게 고정하였다.

[표 1 : 매개변수 값]

매개 변수	값	의미
window_size	5	Time step(몇 일의 데이터를 이용하여 예측할 것인지 결정)
Batch_size	32	한번에 제공되는 학습 데이터의 양
Epoch	100	학습 횟수
Unit	5	LSTM 셀
Activation	Tanh : hard_sigmoid	활성화 함수
optimizer	Adam	최적화 함수
Loss	MSE	평균 제곱 오차(손실함수)

Window size를 5로 설정하여, 과거 5일의 데이터를 이용해 내일의 데이터를 예측하는 (5,5)의 input 형태를 가지는 LSTM층을 활용하였다. 기본 활성화 함수는 쌍곡 탄젠트 함수(Tanh)를 사용하였고, 순환 활성화 함수는 hard sigmoid를 사용하였다. 또한, 최적화 함수는 Adam을 사용하였다. 손실함수는 평균 제곱 오차(MSE, Mean Squared Error)을 이용하여 계산하였다. 시각화는 Matplotlib의 메소드를 이용하여 실제 주식 가격을 red line으로, 예측 가격을 blue line으로 plot하였다.

매개변수를 고정시킨 후, 2020년 5월 3일부터 2022년 5월 3일 까지의 데이터를 학습시켜 5월 4일의 예측 가격을 도출하였다.

스케일링 된 상태로 5월 4일의 예측 가격이 나오기 때문에, 예측 주가로 변환하기 위해서는 다음과 같은 관계식을 통해 변환하는 과정을 거쳐야 한다.

$$\text{raw_df.close}[-1] : \text{dfy.close}[-1] = x : \text{pred_y}[-1]$$

$\text{raw_df.close}[-1]$: 실제 주가 데이터(원 단위)

$\text{dfy.close}[-1]$: 0과 1 사이로 scaling된 실제 주가 데이터

x : 예측 주가(원 단위)

$\text{pred_y}[-1]$ = 0과 1 사이로 scaling된 예측 주가 데이터

따라서 원단위 예측주가 x 는 $x = \text{raw_df.close}[-1] * \text{pred_y}[-1] / \text{dfy.close}[-1]$ 가 될 것이다.

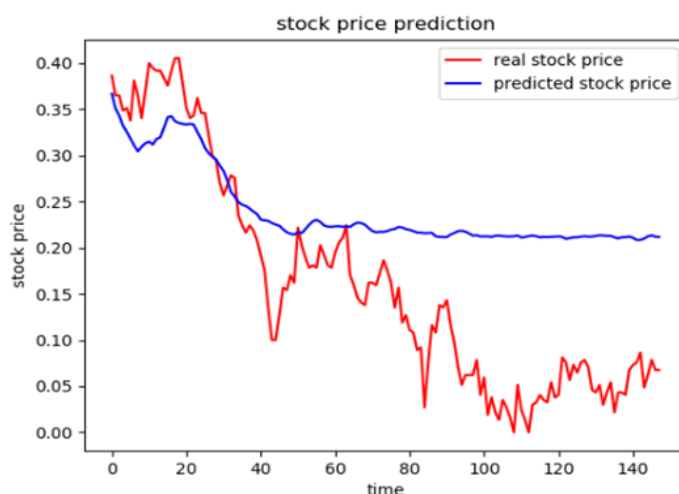
예측 그래프의 시각화는 Matplotlib의 메소드를 사용하여 구현하였다.

V. Main Result

본 연구에서는 LSTM 모형에 2020년 5월 3일부터 2022년 5월 3일까지의 KOSPI 200 기업 주가 데이터를 학습시켜 다음 날 종가를 예측하였다. 이를 바탕으로, 전일 대비 상승하였다면 내일의 주가 향방이 상승할 것으로, 전일 대비 하락하였다면 내일의 주가 향방이 하락할 것으로 예측하는 보조 지표로 활용하고자 하였다.

단, 상장일이 2020년 5월 3일 이후인 경우에는 배제했으며, 예측 그래프가 다음과 같이 실제 주가를 지나치게 벗어날 경우에도 제외하였다. 투자자가 보조 지표로서 본 모델을 활용할 경우, 예측에 실패했다고 판단한다면 투자 결정에서 배제할 것이기 때문이다. 해당 기준에 따라 총 153개 종목을 예측에 포함할 수 있었다.

[이미지 5,6 : 예측 실패 그래프 - KG스틸,현대홈쇼핑]

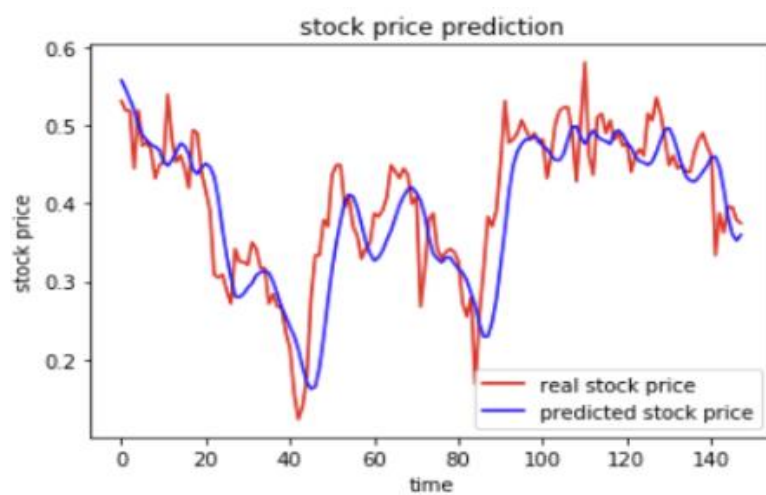
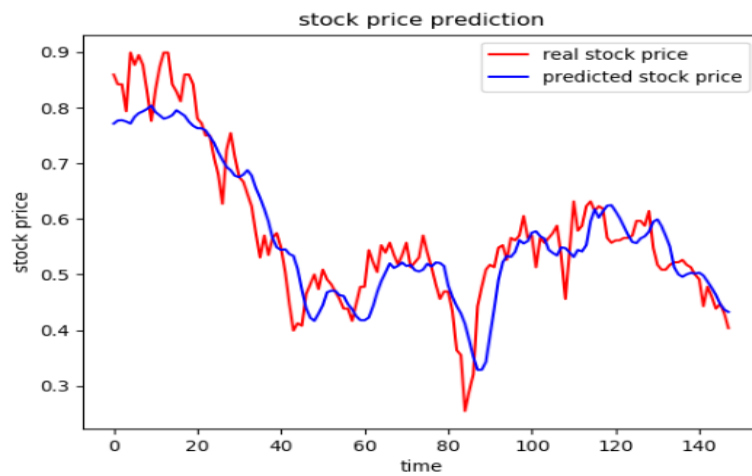




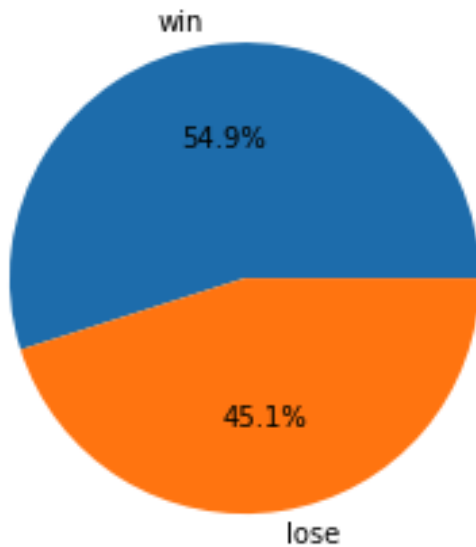
위와 같은 경우, 활성화 함수와 최적화 함수의 조합을 바꿔가며 시도하면 더 나은 예측 성능을 보이기도 했으나, 일관된 매개변수를 사용하기 위해 결과에서 배제하였다.

예측이 비교적 잘 된 그래프의 경우 다음과 같았다.

[이미지 7,8 : 롯데 관광개발, 신세계]

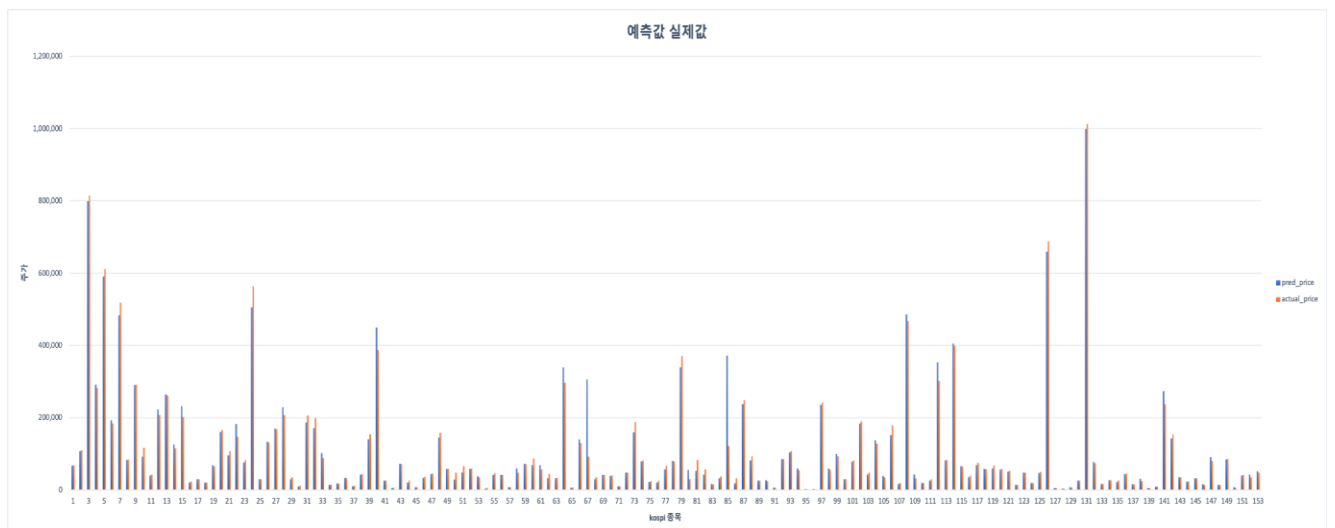


[이미지 9 : 보조지표 승률 파이차트]



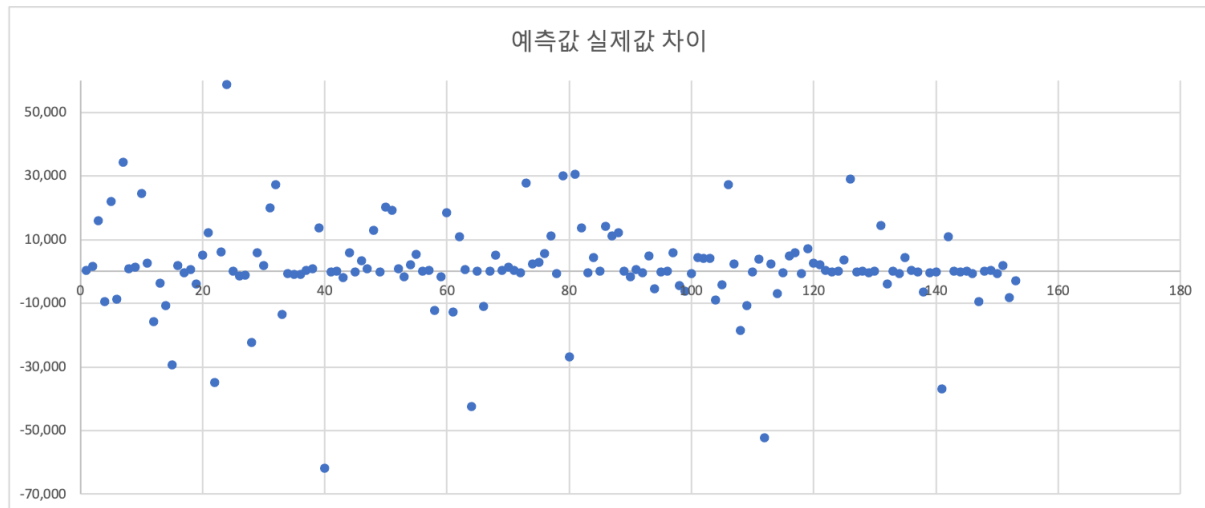
LSTM 주가예측 모델의 보조지표 활용 결과, 상승으로 예측된 종목은 63개, 하락으로 예측된 종목은 90개이며 실제로 상승한 종목은 66개, 하락한 종목은 87개였다. 전체 153 종목 중 84 종목은 주가 향방이 일치했고, 69개 종목에서 예측한 주가 향방과 실제 향방이 달랐다. 이에 따라, 54.902%의 승률을 얻었다.

[이미지 10 : 예측 가격과 실제 가격 그래프]



그래프에서 파란 선은 예측된 익일의 주가이고, 빨간 선은 실제 익일의 주가이다. LSTM 주가예측 모델의 153개 종목의 익일에 대한 예측 주가와 실제 주가를 비교해보면 큰 차이를 보이는 값은 적었다. 그러나, 이는 과거 주가 데이터에 의존하는 LSTM 모형의 특성이므로, 이 그래프만으로 모델의 예측 성능이 뛰어나다고 해석하기는 힘들다.

[이미지 11: 예측값과 실제 값 차이]



이미지11에서는 종목별 편차를 더 자세히 관찰할 수 있다.

VII. Conclusion

54.9%의 승률을 나타낸 LSTM 주가 예측 모델을 이용한 보조 지표는 50%를 근소하게 넘긴 수치로, 해당 지표를 따라 등락을 추종하며 매매하기에는 적합하지 않다. 그러나, 예측값과 실제 값과의 편차가 작기 때문에, 그 편차의 크기에 따라 분할 매수, 매도를 결정하거나 다른 지표와 함께 사용한다면 여러 매수 혹은 매도의 근거 중 하나로 사용할 수 있을 것으로 생각된다.

본 연구에서는 모델이 더 효율적인 예측을 하지 못한 이유를 다음과 같이 생각했다. 첫째, 주가 데이터만을 input 데이터로 사용했기 때문이다. 자유도가 높은 실수 데이터만을 input변수로 사용한 것은 LSTM모형으로 하여금 패턴을 예측하기 힘들게 만들 수 있다. 주가의 패턴을 모두 학습하였다고 하더라도, 시장에 존재하는 여러 변수들이 급등락을 만들 경우, 시계열 실수 데이터만을 학습한다면 현실적으로 예측 데이터가 이를 따라가기는 힘들다.

둘째, 종목마다 적합한 매개변수가 달랐기 때문이다. 일정 매개변수의 조합으로 예측에 실패하더라도, 다른 매개변수의 조합을 적용하면 비교적 효율적인 예측을 내놓을 수 있었다. 그러나, 투자자가 투자하고자 하는 모든 종목에 수많은 매개변수의 조합을 시도하는 것은 굉장히 비효율적이므로, 모형 설계 단계에서부터 더 효율적인 설계를 해 나가는 것이 더 빠른 길일 것이다.

본 연구는 모델에서 input 변수로 주가만을 사용하였다는 점, 익일의 주가만을 예측하여 다양한 타임 프레임의 투자 전략에 적용하기 어렵다는 점에서 한계를 지닌다. 자유도가 높은 시계열 데이터 만

을 사용하는 것이 아닌 다른 데이터를 LSTM의 input 변수로 활용하면 더 좋은 보조지표를 만들 수 있을 것으로 기대된다.

LSTM 모형은 현재 강 하류의 유량을 예측하거나, 기업의 신용점수를 예측하는 등 다양한 분야에서 활용되고 있다. 주식시장에서도 본 모형이 기술적 분석만에 의존하는 모델이 아닌 실제 투자를 전략적으로 보조할 수 있는 장치로서 작용하기 위해서는, 개별 기업이 지닌 펀더멘털을 효과적으로 설명할 수 있는 방향으로 발전해 나가는 것이 바람직하다고 생각한다.

VIII. Appendix

Ⅰ. 개발 환경

본 연구의 개발 환경은 다음과 같다.

* 운영체제 : Window 10 (64bit)

* MariaDB : 10.5.1 (64bit)

* Python : 3.8.1 (64bit)

* Pandas : 1.0.1

* BeautifulSoup : 4.8.2

* Matplotlib : 3.1.2

* Tensorflow : 2.8.0

* Keras : 2.8.0

예측 가격과 실제 증가, 방향 예측표

1	종목코드	종목명	5월 3일 증가	5월 4일 예측가격	방향 예상	5월 4일 실제증가	실제 변동 방향	win/lose
2	005930	삼성전자	67,500	67,588	상승	67,900	상승	win
3	000660	SK하이닉스	110,000	108,002	하락	109,500	하락	win
4	207940	삼성바이오로직스	832,000	799,059	하락	815,000	하락	win
5	035420	NAVER	282,000	291,388	상승	282,000	하락	lose
6	006400	삼성SDI	624,000	590,092	하락	612,000	하락	win
7	005380	현대차	183,500	192,705	상승	184,000	상승	win
8	051910	LG화학	517,000	483,550	하락	518,000	상승	lose
9	000270	기아	83,500	82,638	하락	83,500	하락	win
10	005490	POSCO홀딩스	289,000	290,120	상승	291,500	상승	win
11	028260	삼성물산	115,500	91,579	하락	116,000	상승	lose
12	055550	신한지주	41,650	39,937	하락	42,500	상승	lose
13	012330	현대모비스	206,500	222,862	상승	207,000	상승	win
14	034730	SK	261,000	264,079	상승	260,500	하락	lose
15	066570	LG전자	115,500	126,223	상승	115,500	하락	lose
16	096770	SK에노비이션	202,500	231,937	상승	202,500	하락	lose
17	015760	한국전력	22,150	21,110	하락	22,950	상승	lose
18	011200	HMM	28,900	29,651	상승	29,200	상승	win
19	034020	두산에너지리티	20,800	20,360	하락	21,000	상승	lose
20	032830	삼성생명	64,300	68,554	상승	64,700	상승	win
21	009150	삼성전기	166,500	160,931	하락	166,000	하락	win
22	010950	S-Oil	106,000	95,314	하락	107,500	상승	lose
23	018260	삼성에스디에스	147,500	182,784	상승	148,000	상승	win
24	033780	KT&G	82,500	76,695	하락	82,900	상승	lose
25	010130	고려아연	568,000	505,324	하락	564,000	하락	win
26	003490	대한항공	30,000	29,722	하락	29,850	하락	win
27	003670	포스코케미칼	134,000	133,796	하락	132,500	하락	win
28	090430	아모레퍼시픽	172,500	169,730	하락	168,500	하락	win
29	000810	삼성화재	204,000	229,783	상승	207,500	상승	win
30	030200	KT	35,200	29,416	하락	35,350	상승	lose
31	024110	기업은행	11,350	9,573	하락	11,450	상승	lose
32	086280	현대글로비스	205,500	186,014	하락	206,000	상승	lose
33	011170	롯데케미칼	197,000	171,184	하락	198,500	상승	lose
34	009540	한국조선해양	90,200	101,956	상승	88,400	하락	lose
35	032640	LG유플러스	13,950	14,686	상승	13,950	하락	lose
36	034220	LG디스플레이	17,300	18,219	상승	17,300	하락	lose
37	009830	한화솔루션	31,850	33,395	상승	32,400	상승	win
38	018880	한온시스템	11,300	10,904	하락	11,250	하락	win
39	004020	현대제철	43,500	42,691	하락	43,500	하락	win
40	011790	SKC	153,500	140,310	하락	154,000	상승	lose
41	097950	CJ제일제당	388,000	448,881	상승	387,000	하락	lose
42	035250	강원랜드	26,450	26,329	하락	26,300	하락	win
43	010140	삼성중공업	5,910	5,654	하락	5,850	하락	win
44	021240	코웨이	69,500	72,877	상승	71,000	상승	win
45	028050	삼성엔지니어링	26,050	20,512	하락	26,450	상승	lose
46	006800	미래에셋증권	7,970	8,061	상승	7,940	하락	lose
47	138040	메리츠금융지주	37,050	33,040	하락	36,500	하락	win
48	000720	현대건설	44,650	43,809	하락	44,700	상승	lose
49	011780	금호석유	154,500	145,082	하락	158,000	상승	lose
50	267250	HD현대	58,300	58,676.797	상승	58,500	상승	win
51	047810	한국항공우주	47,200	27,871	하락	48,150	상승	lose
52	005830	DB손해보험	65,400	47,348	하락	66,500	상승	lose
53	000100	유한양행	60,500	58,726	하락	59,700	하락	win
54	161390	한국타이어앤테크닉스	34,900	37,281	상승	35,500	상승	win

1	종목코드	종목명	5월 3일 증가	5월 4일 예측가격	방향 예상	5월 4일 실제증가	실제 변동 방향	win/lose
55	008560	메리츠증권	6,620	4,285	하락	6,520	하락	win
56	078930	GS	45,100	41,406	하락	46,700	상승	lose
57	241560	두산밥캣	41,200	41,418	상승	41,500	상승	win
58	003410	쌍용C&E	8,190	7,852	하락	8,150	하락	win
59	002790	아모레G	49,300	59,876	상승	47,600	하락	lose
60	071050	한국금융지주	71,000	72,676	상승	71,100	상승	win
61	020150	일진머티리얼즈	88,000	68,144	하락	86,700	하락	win
62	180640	한진칼	57,000	69,107	상승	56,400	하락	lose
63	036460	한국가스공사	41,000	32,341	하락	43,350	상승	lose
64	029780	삼성카드	33,250	32,779	하락	33,450	상승	lose
65	128940	한미약품	302,500	338,989	상승	296,500	하락	lose
66	028670	팬오션	6,820	6,616	하락	6,840	상승	lose
67	139480	이마트	129,500	139,958	상승	129,000	하락	lose
68	271560	오리온	91,100	305,431	상승	92,100	상승	win
69	004990	롯데지주	34,300	29,199	하락	34,350	상승	lose
70	006360	GS건설	41,900	41,453	하락	41,900	하락	win
71	016360	삼성증권	39,600	38,292	하락	39,600	하락	win
72	005940	NH투자증권	10,650	10,281	하락	10,600	하락	win
73	008930	한미사이언스	49,050	48,372	하락	47,950	하락	win
74	282330	BGF리테일	187,000	159,824	하락	187,500	상승	lose
75	010620	현대미포조선	81,800	78,720	하락	81,000	하락	win
76	047050	포스코엔터내셔널	24,500	21,436	하락	24,300	하락	win
77	030000	제일기획	26,250	20,326	하락	25,950	하락	win
78	000990	DB하이텍	67,700	56,639	하락	67,900	상승	lose
79	008770	호텔신라	80,500	80,194	하락	79,500	하락	win
80	002380	KCC	355,000	339,964	하락	370,000	상승	lose
81	007070	GS리테일	29,400	56,181	상승	29,350	하락	lose
82	052690	한전기술	75,800	52,422	하락	83,000	상승	lose
83	012450	한화에너지	56,000	43,197	하락	57,000	상승	lose
84	272210	한화시스템	15,850	16,331	상승	15,950	상승	win
85	000080	하이트진로	37,800	32,745	하락	37,100	하락	win
86	000120	CJ대한통운	121,000	371,249	상승	122,000	상승	win
87	001450	현대해상	31,350	17,712	하락	31,950	상승	lose
88	014680	한솔케미칼	244,000	237,442	하락	248,500	상승	lose
89	023530	롯데쇼핑	94,300	81,239	하락	93,500	하락	win
90	026960	동서	26,750	26,155	하락	26,200	하락	win
91	042660	대우조선해양	25,300	26,697	상승	25,050	하락	lose
92	047040	대우건설	6,750	6,149	하락	6,750	하락	win
93	001040	CJ	85,100	85,489	상승	85,100	하락	lose
94	010060	OCI	110,000	103,144	하락	108,000	하락	win
95	204320	만도	54,200	59,826	상승	54,500	상승	win
96	088350	한화생명	2,875	3,040	상승	2,855	하락	lose
97	001440	대한전선	1,865	1,912	상승	1,950	상승	win
98	004170	신세계	246,000	236,145	하락	242,000	하락	win
99	112610	씨에스윈드	57,500	59,644	상승	55,300	하락	lose
100	039490	키움증권	93,300	99,186	상승	93,100	하락	lose
101	000880	한화	29,500	29,955	상승	29,400	하락	lose
102	004000	롯데정밀화학	81,300	77,153	하락	81,600	상승	lose
103	006280	녹십자	187,000	184,406	하락	188,500	상승	lose
104	111770	영원무역	47,450	43,232	하락	47,450	하락	win
105	285130	SK케미칼	128,000	137,509	상승	128,500	상승	win
106	336260	두산퓨얼셀	34,550	39,099	상승	34,800	상승	win
107	069620	대웅제약	182,500	151,142	하락	178,500	하락	win

	A	B	C	D	E	F	G	H
1	종목코드	종목명	5월 3일 종가	5월 4일 예측가격	방향 예상	5월 4일 실제종가	실제 변동 방향	win/lose
108	064350	현대로템	19,050	16,753	하락	19,050	하락	win
109	298050	효성첨단소재	466,500	485,443	상승	467,000	상승	win
110	081660	힐라홀딩스	32,050	42,868	상승	32,250	상승	win
111	001230	동국제강	19,050	18,959	하락	18,900	하락	win
112	003090	대웅	29,950	25,905	하락	29,750	하락	win
113	004370	농심	300,000	353,688	상승	301,500	상승	win
114	004800	효성	83,400	81,111	하락	83,400	하락	win
115	298020	효성티앤씨	405,000	405,932	상승	399,000	하락	lose
116	011210	현대위아	64,800	65,459	상승	65,100	상승	win
117	051600	한전KPS	38,700	34,604	하락	39,400	상승	lose
118	069960	현대백화점	75,300	69,156	하락	75,000	하락	win
119	006260	LS	56,600	57,776	상승	57,200	상승	win
120	120110	코오롱인더	62,600	59,993	하락	67,200	상승	lose
121	014820	동원시스템즈	57,400	55,450	하락	58,000	상승	lose
122	010780	아이에스동서	53,600	50,934	하락	53,100	하락	win
123	000240	한국연컴퍼니	14,400	13,796	하락	14,200	하락	win
124	178920	PI첨단소재	47,950	47,565	하락	47,500	하락	win
125	020560	아시아나항공	19,950	19,489	하락	19,600	하락	win
126	010120	LS ELECTRIC	50,000	46,888	하락	50,600	상승	lose
127	000670	영풍	684,000	658,990	하락	688,000	상승	lose
128	001740	SK네트웍스	4,820	4,949	상승	4,805	하락	lose
129	073240	금호타이어	4,215	4,100	하락	4,170	하락	win
130	042670	현대두산인프라코어	6,760	7,058	상승	6,670	하락	lose
131	005850	에스엘	26,750	25,966	하락	26,050	하락	win
132	003240	태광산업	1,015,000	998,473	하락	1,013,000	하락	win
133	004490	세방전지	73,800	77,415	상승	73,400	하락	lose
134	241590	화승엔터프라이즈	16,550	16,606	상승	16,700	상승	win
135	079160	CJ CGV	27,300	27,405	상승	26,850	하락	lose
136	105630	한세실업	26,550	22,011	하락	26,300	하락	win
137	161890	한국콜마	45,850	44,472	하락	44,750	하락	win
138	032350	롯데관광개발	16,000	16,029	상승	16,000	하락	lose
139	001680	대상	24,700	30,927	상승	24,400	하락	lose
140	003520	영진약품	4,850	5,186	상승	4,775	하락	lose
141	005440	현대그린푸드	9,100	9,056	하락	9,020	하락	win
142	006040	동원산업	239,000	273,499	상승	236,500	하락	lose
143	006650	대한유화	153,500	142,087	하락	153,000	하락	win
144	020000	한성	35,400	35,444	상승	35,500	상승	win
145	069260	TKG휴켄스	22,100	23,183	상승	23,100	상승	win
146	103140	풍산	32,100	31,774	하락	31,850	하락	win
147	114090	GKL	15,050	15,376	상승	14,800	하락	lose
148	192820	코스맥스	82,400	90,121	상승	80,600	하락	lose
149	001800	오리온홀딩스	14,700	14,355	하락	14,550	하락	win
150	000070	삼양홀딩스	85,000	84,518	하락	84,900	하락	win
151	002350	넥센타이어	6,680	7,464	상승	6,690	상승	win
152	006120	SK디스커버리	42,200	40,136	하락	41,900	하락	win
153	284740	쿠쿠홈시스	34,700	42,867	상승	34,600	하락	lose
154	064960	SNT모티브	48,600	51,463	상승	48,500	하락	lose

출처

정종진, 김지연.(2020).LSTM을 이용한 주가예측 모델의 학습방법에 따른 성능분석.디지털융복합연구, 18(11),259-266.

신동하, 최광호, 김창복.(2017).RNN과 LSTM을 이용한 주가 예측을 향상을 위한 딥러닝 모델.한국정보기술학회논문지,15(10),9-16.

이낙영, 오경주.(2019).디노이징 필터와 LSTM을 활용한 KOSPI200 선물지수 예측.한국데이터정보과학회지,30(3),645-654.

김수현.(2020).LSTM 기반 모형의 주식시장 예측성 분석.Journal of The Korean Data Analysis Society, 22(5),1989-2000.

최훈.(2021).빅데이터를 활용한 인공지능 주식 예측 분석.한국정보통신학회논문지,25(10),1435-1440.

김하얀, 주귀화, 김석찬. (2021). LSTM을 이용한 주가 예측 알고리즘. 한국통신학회 학술대회논문집, (), 1019-1020.

Yang, C.-H.; Chang, P.-Y. Forecasting the Demand for Container Throughput Using a Mixed-Precision Neural Architecture Based on CNN-LSTM. *Mathematics* **2020**, *8*, 1784. <https://doi.org/10.3390/math8101784>

S. Siامي-Namini, N. Tavakoli and A. Siامي Namin, "A Comparison of ARIMA and LSTM in Forecasting Time Series," *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018, pp. 1394-1401, doi: 10.1109/ICMLA.2018.00227.

<https://towardsdatascience.com/deep-study-of-a-not-very-deep-neural-network-part-2-activation-functions-fd9bd8d406fc> - 활성화 함수

<https://velog.io/@kyj93790/%EB%B0%91%EB%B0%94%EB%8B%A5%EB%B6%80%ED%84%B0-%EC%8B%9C%EC%9E%91%ED%95%98%EB%8A%94-%EB%94%A5%EB%9F%AC%EB%8B%9D-6.-%ED%95%99%EC%8A%B5-%EA%B4%80%EB%A0%A8-%EA%B8%B0%EC%88%A0%EB%93%A4-part1-%EB%A7%A4%EA%B0%9C%EB%B3%80%EC%88%98-%EA%B0%B1%EC%8B%A0#sqd%EC%9D%98-%EB%8B%A8%EC%A0%90>

Cornell University. Diederik P. Kingma, Jimmy Ba - <https://arxiv.org/abs/1412.6980> - 최적화함수