**주식 예측 인공지능 만들기**

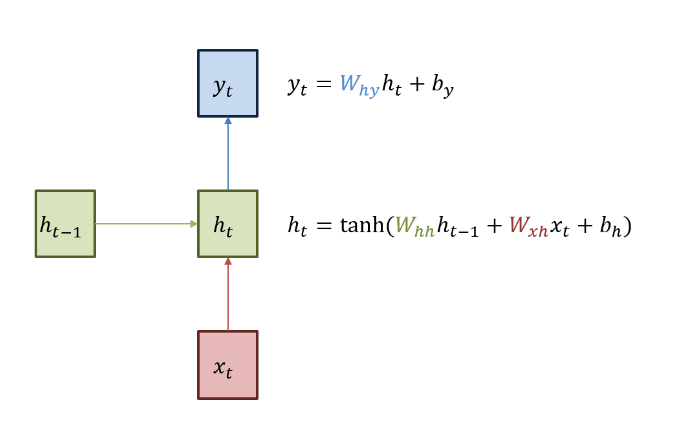
Team18- 김가륜(21900059), 송예지(22100396), 김민경(22100090), 이진주(22100579)

**Problem description**

* 과거 데이터를 이용하여 에코프로 주식 예측 인공지능 만들기
* 국내 기업 에코프로의 상장일(23-06-07)부터 작성시점(23-06-03)까지의 과거 주식 데이터를 수집하고, 이를 통해 학습시킨 AI모델을 활용하여 다음날의 주식을 예측한다.
* 수집하는 데이터는 시가(Open), 고가(High), 종가(Close), 거래량(Volume), 전일대비 데이터(Change)로 한다.
* 주식 데이터는 Sequential한 특징을 가지므로, AI 모델로 RNN/LSTM을 채택한다.

**Background and conventional approach**

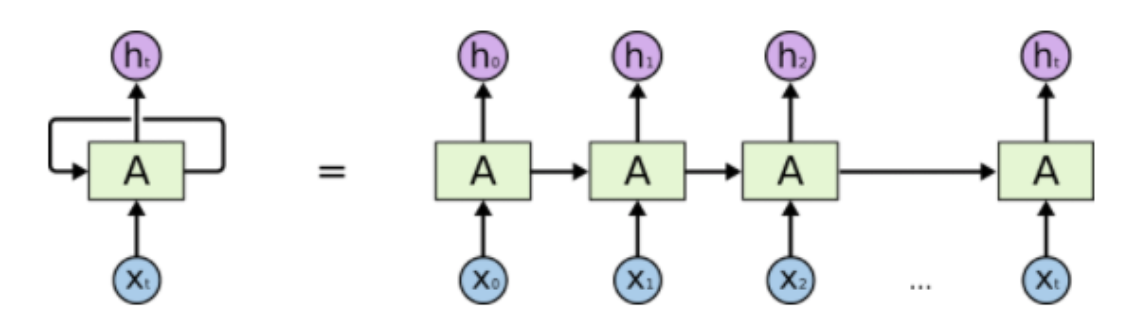
* RNN(Recurrent Neural Network)
* 히든 노드가 방향을 가진 엣지로 연결돼 순환 구조(directed cycle)를 이루는 인공 신경망의 한 종류.
* 음성, 문자 등 시간 개념과 관련된 순차적인 데이터 처리에 특화되었으며 데이터의 전체적인 흐름을 이해하도록 도와준다.
* 시퀀스 길이에 관계 없이 input과 output을 받아들일 수 있는 네트워크 구조.
* 기본 구조는 이하 그림과 같이 현재 상태의 히든 ht를 직전 시점의 히든 ht-1를 이용하여 갱신하는 것이다. 즉, 이전의 데이터를 통해 학습된 셀의 상태 정보가 다음 데이터의 학습에 다시 사용된다.



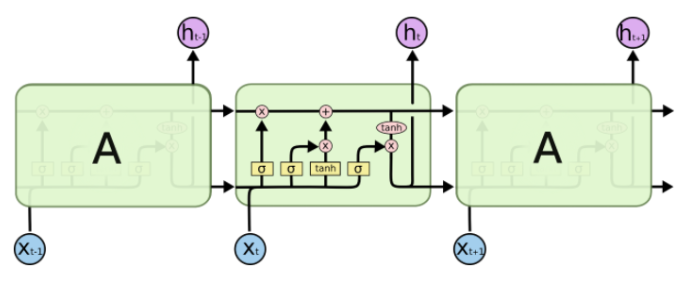
* 필요한 활성화 함수는 다음과 같으며 relu, tanh 등을 사용한다.

h𝘵 = 활성화함수(W(h𝘵-₁, x𝘵) + b)

h₁ = 활성화함수(W(h₀, x₁) + b)



* vanishing gradient problem: 관련 정보와 그 정보의 사용 지점 사이의 거리가 멀 경우 학습 능력이 크게 저하되는 RNN의 결점.
* LSTM(Long Short-Term Memory Models)
* RNN의 결점을 보완한 버전. RNN의 히든 state에 cell-state를 추가하여 직전 데이터뿐만 아니라 거시적인 과거 데이터를 함께 고려할 수 있게 만들었다.
* 내부적으로 다음과 같은 구조를 가지고 있다.

****

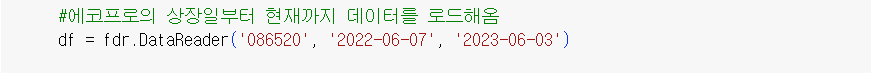
* cell state가 일종의 컨베이어 벨트 역할을 하여 state가 오래 경과하더라도 그래디언트의 원활한 전파를 보장한다. forget date, input gate를 사용한다.

**Proposed method**

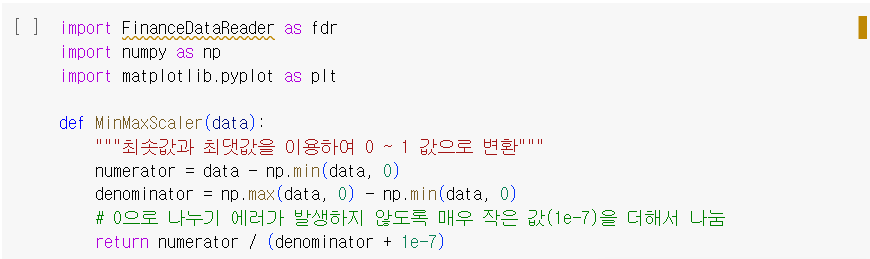
1. data collection and preparation

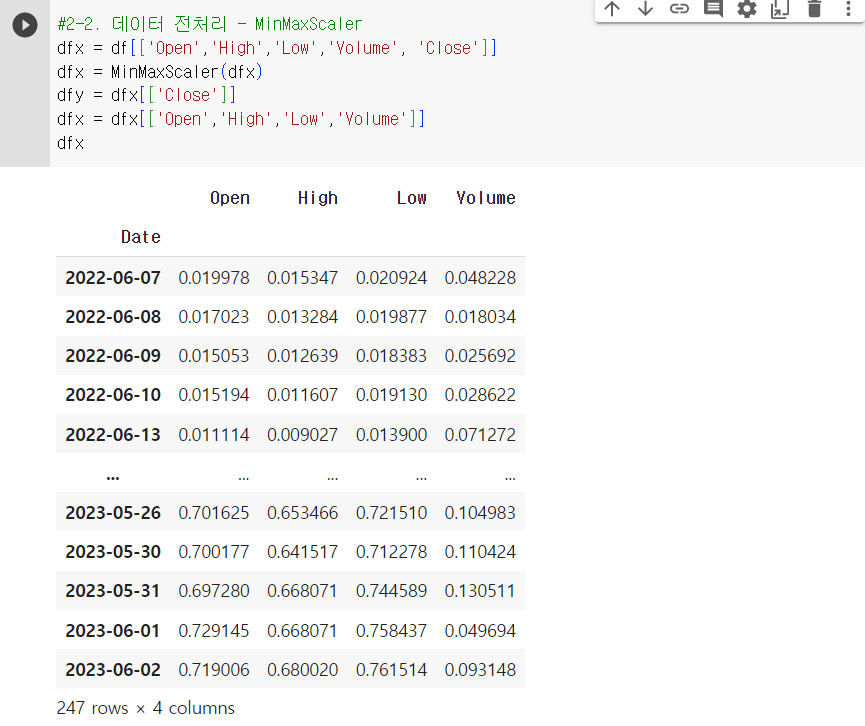
FinanceDataReader 모듈을 사용하여 에코프로 주식 데이터를 수집.

에코프로 종목코드를 파라미터로 주어 상장일인 2023.6.7부터 작성시점 금일인 2023.6.3까지의 데이터를 로드하였다.

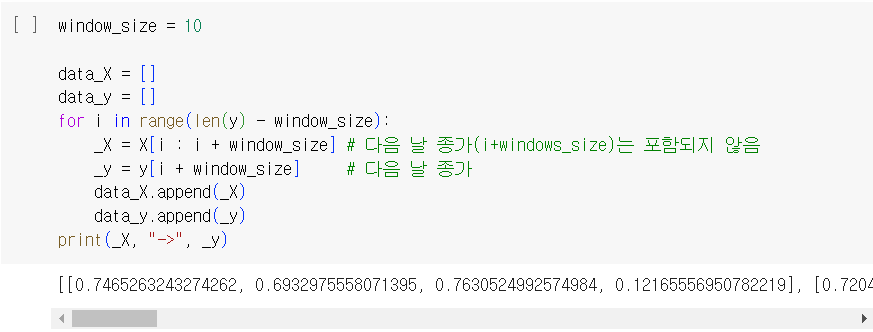


* MinMaxScaler 함수를 정의하여 데이터 전처리

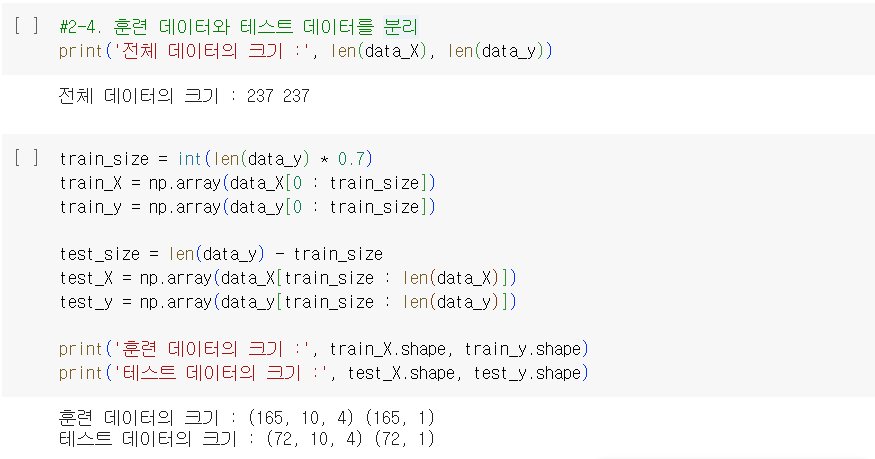




* 10일간의 OHLVC 데이터를 이용하여 다음날의 종가를 예측

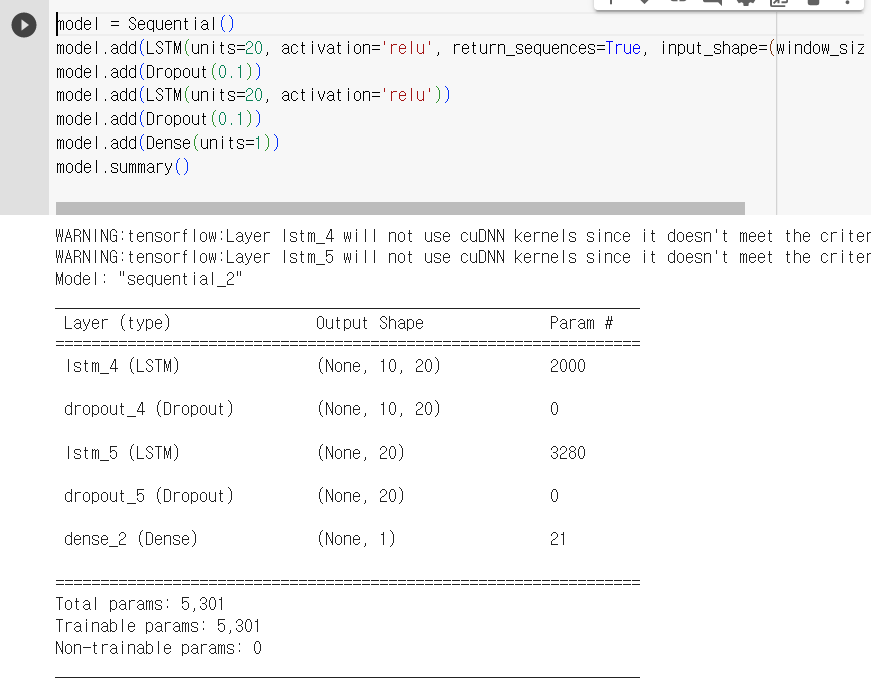


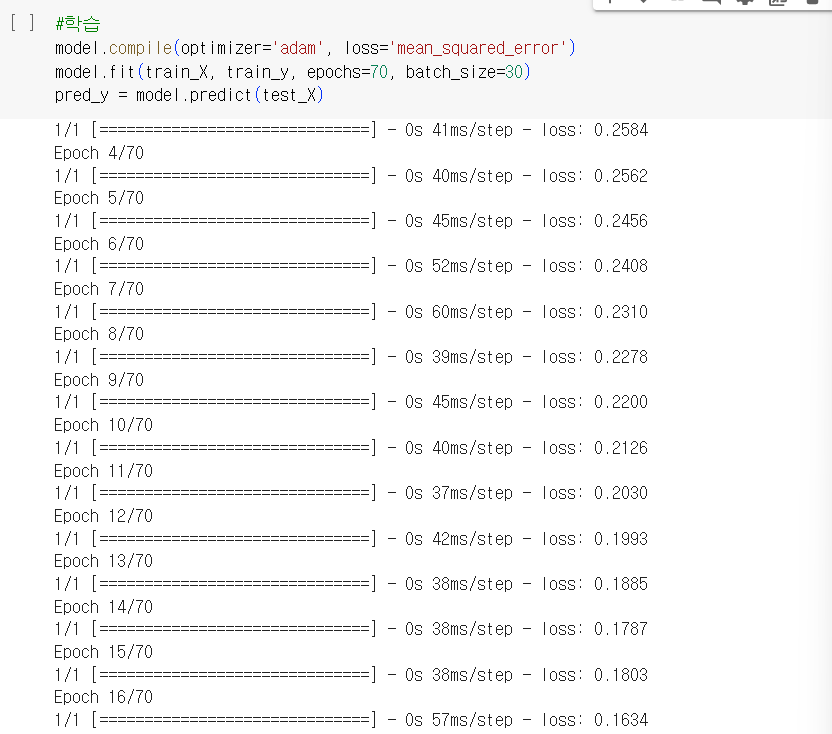
* 훈련용 데이터와 테스트 데이터를 7:3 비율로 분리하기



1. AI approach

* LSTM모델을 정의하여 학습시키기





1. Evaluation criteria:

data sampling 단위, 과거 몇개의 데이터를 볼지(window size) 등등을 바꿔가며 비교하여 최적의 방법을 찾는다.

* data sampling 단위에 따른 예측 정확도 차이

| data sampling | window size | result |
| --- | --- | --- |
| 9:1 | 10 |  |
| 7:3 | 10 |  |
| 5:5 | 10 |  |

* window size 단위에 따른 예측 정확도 차이 (위에서 찾은 최적 data sampling을 사용함)

| data sampling | window size | result |
| --- | --- | --- |
| 9:1 | 5 |  |
| 9:1 | 10 |  |
| 9:1 | 15 |  |

**Results**

* 인공지능이 데이터를 학습하여 패턴을 통하여 결론을 도출한다. 따라서 학습시키는 데이터양이 많을 수록 데이터에 정확하게 대처할 확률이 높다 따라서 data sampling 비율을 9:1로 하여서 학습시키는 데이터의 양을 늘렸다.
* window\_size는 중간 결과의 도출을 위해 최근 며칠까지의 데이터를 참고할 것인지 정해주는 변수로, 이 값이 작을수록 최근의 큰 변화에 민감하게 반응하고, 클 수록 최근의 큰 변화에 덜 민감하게 반응하게 된다. 이 특성은 주식 분석에 자주 사용되는 이동평균선과 유사한 성질을 가지는 것으로 보이는데, 차이점은 이동평균선은 단순히 데이터의 평균값만 계산하여 준다면, 이 인공지능 모델은 여러날의 평균을 고려할 뿐만 아니라 다음날에 올 수 있는 예측값도 제시할 수 있다는 점이다.
* 결론적으로 data sampling의 비율은 9:1로 결정했다. window\_size는 다양한 상황에 따라서 다르게 적용할 수 있을것이고 이에대한 내용은 discussion에 정리했다.

**Discussion**

* Why are your results better/worse than other methods?

단순히 순차적인 데이터의 처리라고 생각한다면 RNN을 사용할 수도 있지만, 단기 기억 뿐인 RNN을 보완하여 장기와 단기로 나누어서 정보를 전달하는 LSTM의 구조가 더욱 정교하다고 판단했기 때문에 LSTM 모델을 채택하여 사용하였다. 이로써 순차 데이터를 처리하는 주식 데이터를 처리할 때 입력의 길이가 길어지더라도 이전 정보를 더 오래 기억할 수 있었고, 역전파시 그래디언트의 손실을 최소화하여 더 효과적인 예측을 가능하게 했다.

* How to improve the result?

data sampling의 단위, window size의 값을 바꾸어가며 정확도를 비교하고, 데이터의 품질을 향상시킬 수 있는 방법을 고민하였다.

실제 결과를 비교해본 결과, data sampling의 비율은 학습 데이터를 가능한 한 많은 비율로 두는 것이 효율적이라 판단하였으며 window\_size는 어떤 주식을 분석하느냐, 혹은 투자전략에 따라서 달리 하는 것이 효율적이라 판단하였다. 예를 들어, 단기간에 큰 변화를 보이는(noise가 많은) 주식의 짧은 기간내 거래를 예측하고 싶다면 window\_size를 줄이고, noise를 무시하고 긴 기간을 두고 거래하고 싶다면 window\_size를 늘리는 방법과 같이 활용할 수 있을 것이다.

**Conclusion**

주식 데이터의 순차적인 특성을 고려하여, 순차적 데이터의 처리에 유용한 LSTM 모델을 사용하여 에코프로 주가를 예측하는 인공지능을 만들었다. 예측 정확도 측면에서 보았을 때, 많은 sampling data를 넣을수록 더 효율적인 예측이 가능했으며, window size는 상황에 따라 최적의 조건이 달라졌다.

더 설명하자면 작은 window\_size는 주식의 단기적인 움직임을 빠르게 반영하여 빠른 트렌드를 파악할 수 있고, 큰 window\_size를 사용하면 주가의 장기적인 추세를 파악할 수 있다. 그러므로 장기적인 투자 목표와 전략에 맞는 window\_size를 선택하여 사용하는 것이 좋을 것 같다.

무엇보다 주식 분석과 투자는 개인의 선호 및 전략에 따라 다를 수 있으므로, window\_size의 선택은 절대적이기보다 주관적인 특성이 강하다. 데이터의 양과 주가의 특성을 고려하여 적절한 window\_size를 선택하고, 시뮬레이션을 통해 해당 window\_size의 유효성을 평가하는 것이 도움이 될 것이다.

또한 이번에 사용된 데이터는 일별 단위의 데이터들이었기 때문에 train시킬 수 있는 데이터의 양이 비교적 충분치 않다는 문제가 생겼다. 또한 하루단위의 데이터의 특성상 단기간 투자보다 장기간 투자에 더 어울리는 데이터를 제공한다고 생각한다. 만일 보다 단기간에 초점을 맞춘 예측 인공지능으로서 보완하고자 한다면, 사용되는 데이터셋의 단위를 시간당 데이터와 같은 보다 세밀한 단위로 선택하는 방법이 있을 것이다.

**역할 분담**

* 김가륜(21900059): 모델 선정 토의, 보고서 정리 및 퇴고, 레퍼런스 분석, 코드 작성, 코드 분석, 발표
* 송예지(22100396): 모델 선정 토의, 배경지식 조사, 레퍼런스 탐색, 레퍼런스 분석, 코드 분석, 발표 자료 준비
* 김민경(22100090): 모델 선정 토의, 배경지식 조사, 보고서 정리 및 퇴고, 레퍼런스 분석, 발표 자료 준비
* 이진주(22100579): 모델 선정 토의, 보고서 정리 및 퇴고, 레퍼런스 탐색, 레퍼런스 분석, 코드 분석

**Discussion**

* 김가륜(21900059): 이전에 배웠던 인공지능 알고리즘을 이용해서 다른 주제에 대해서 결과물을 만들어내는 것이 매우 흥미로운 작업이었다. 하지만 결과가 예측한 만큼 잘 나오지 않아 아쉬움이 많이 남았다.
* 송예지(22100396): 수업 시간에 순환 신경망 모델을 악보를 예측하는데 사용한 것을 기반으로 알고리즘을 주식 예측에 적용해 보았다. 다음 날 주가를 예측하는데 있어서 흥미로운 작업이였고 기능을 더 향상시키면 좋을 것 같다.
* 김민경(22100090): LSTM 인공지능을 통해서 실생활에 적용할 수 있었다. 주가를 분석하고, 예측하는 경험을 할 수 있었다. 실제 에코프로 주식과 예측한 주식값의 차이가 있었는데, 이를 줄일 수 있는 방법을 탐구해서 더 좋은 예측 프로그램을 만들고 싶다.
* 이진주(22100579): 단순히 이론으로만 배우던 LSTM과 같은 인공지능을 실생활에서 접하던 주식이라는 토픽과 연관지어 활용할 수 있다는 점이 재미있고 흥미로웠다. 예측 정확도가 기대한 만큼 나오지는 않았지만, 여러 시도를 통해 개선해볼 수 있겠다는 기대를 가질 수 있었다.

**Reference**

<https://seong6496.tistory.com/169> #주식 데이터 가져오기

<https://www.ktb.co.kr/trading/popup/itemPop.jspx> #종목코드 조회

<http://tykimos.github.io/2017/04/09/RNN_Layer_Talk/> #LSTM 레퍼런스

<https://wikidocs.net/173005> #RNN 레퍼런스

<https://ratsgo.github.io/natural%20language%20processing/2017/03/09/rnnlstm/> #RNN 레퍼런스

<https://ardino.tistory.com/53> #RNN, LSTM 비교 레퍼런스