**HW\_7**

**1.APPLE stock prediction**

*(Apple주식 예측은 튜토리얼에 나온 모델 하나로 진행하였습니다.)*

**스크린샷이(가) 표시된 사진

자동 생성된 설명**

Apple의 주식data를 load한 후 standard normalization을 하였습니다.

Model의 구조는 튜토리얼에 명시되어있는 구조에 맞게 디자인 하였습니다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

개체, 시계이(가) 표시된 사진

자동 생성된 설명

Epoch을 10으로 작게 설정하고, validation set을 train set에서 split하였다. 그럼에도 불구하고 정확도가 높게 도출되었다. Apple의 주식 data를 plt해 보았을 때 대체적으로 상향그래프였기 때문에 모델이 복잡하지 않게 쉽게 모델링 된것으로 추측할 수 있다.

**2. SAMSUNG stock prediction**

스크린샷이(가) 표시된 사진

자동 생성된 설명

Samsung의 주식 데이터를 load하여 standard normalization을 진행하였다.

Apple 주식과 다르게 train data와 test data의 형태가 다르기 때문에 Apple주식 예측 모델보다 복잡할것으로 예상된다.

Test data set의 error를 줄이는 것이 목표이므로, validation을 X\_test,Y\_test로 설정하였다.

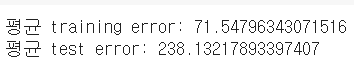
학습시간을 줄이고 정확도를 높이기 위해 Epoch=500으로 설정하고, ReduceLROnPlateau, ModelCheckpoint, EarlyStopping을 사용하였다.

Optimizer는 Rmsprop을 사용하였습니다.

**Case 1**

스크린샷이(가) 표시된 사진

자동 생성된 설명



LSTM(50)-LSTM(100)-LSTM(50)-Dense

**Case 2**

**스크린샷이(가) 표시된 사진

자동 생성된 설명**

**개체이(가) 표시된 사진

자동 생성된 설명**

GRU(50)-GRU(100)-GRU(50)-Dense

**Case 3**

**스크린샷이(가) 표시된 사진

자동 생성된 설명개체이(가) 표시된 사진

자동 생성된 설명**

GRU(100)-GRU(200)-Dense (+BatchNorm)

**Case 4**

**스크린샷이(가) 표시된 사진

자동 생성된 설명개체, 시계이(가) 표시된 사진

자동 생성된 설명**

GRU(50)-GRU(100)-Dense (+BatchNorm)

**Case 5**

**스크린샷이(가) 표시된 사진

자동 생성된 설명**

**개체이(가) 표시된 사진

자동 생성된 설명**

GRU(200)-GRU(300)-Dense (+BatchNorm)

**Case 6**

**스크린샷이(가) 표시된 사진

자동 생성된 설명**

**개체이(가) 표시된 사진

자동 생성된 설명**

GRU(100)-GRU(200)-Dense (+BatchNorm)

변경 : Optimizer = Adam

**결론**

LSTM과 GRU를 비교하였을 때 GRU가 성능이 더 좋았고,(Case1 VS Case2)

Time step은 100-200이 가장 성능이 높았다.(Case2 VS Case3 VS Case4)

또한 batchnormalization을 하였을때 error가 100이상 줄어들었고,(Case2 VS Case3)

이 경우 optimizer는 Adam보다 rmsprop가 더 적절하였다.(Case3 VS Case6)

결론적으로 test error는 case3 일 때 **135**로 가장 낮았다.