2025 09 18
발표 자료

광운대학교 로봇학과
FAIR Lab

김한서

KWANGWOON UNIVERSITY

2D CNN 모델

FAIR Lab

KWANGWOON UNIVERSITY

이번 주 진행사항

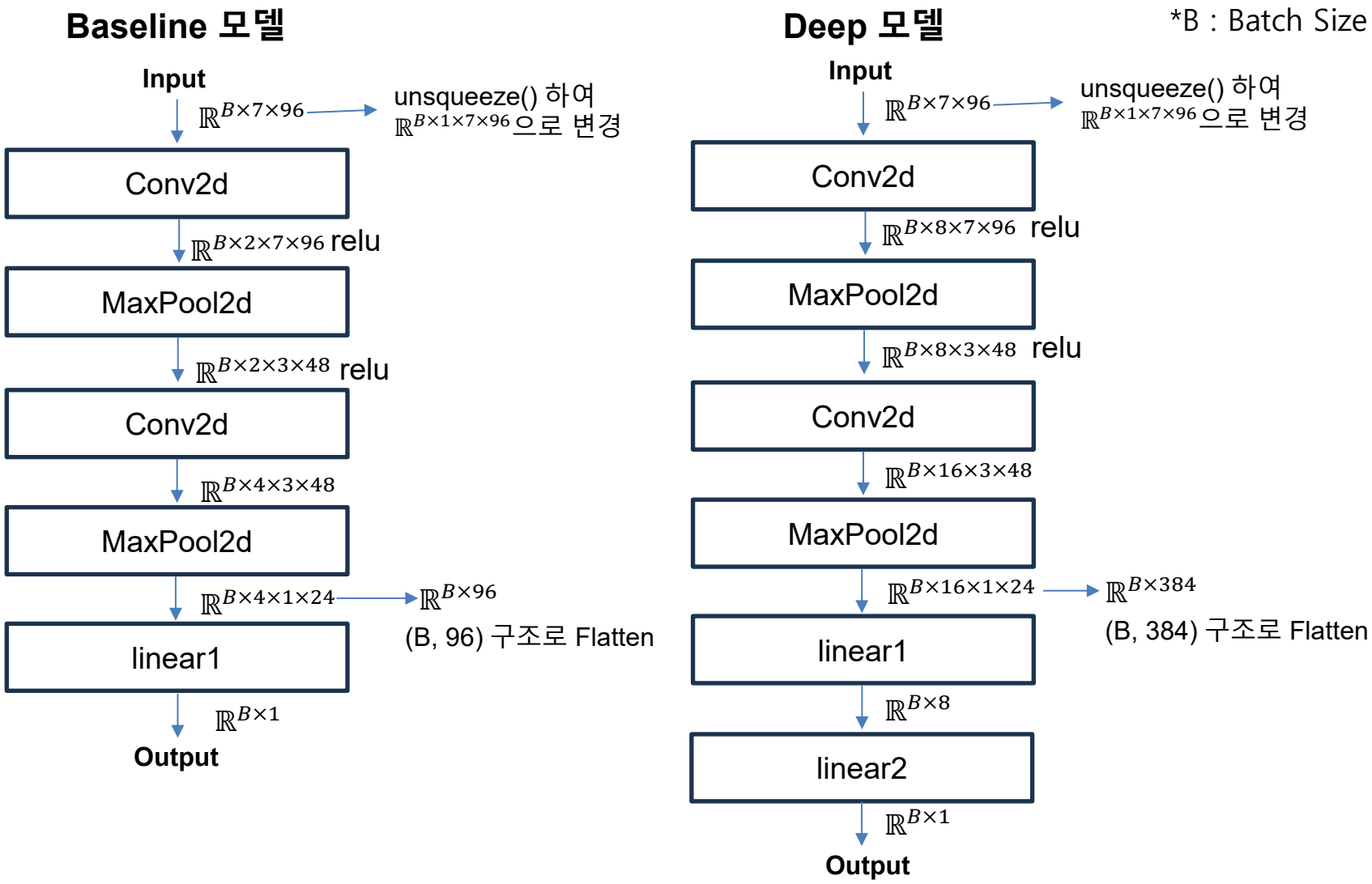
- 2D CNN 모델 학습
  - Baseline 모델(채널 2-4)과 Deep 모델(채널 8-16) 비교
  - Label scale 적용
  - 시각화
- Attention 리뷰

모델 구조

FAIR Lab

KWANGWOON UNIVERSITY

*B : Batch Size



데이터셋 및 실험 세팅

- 사용한 종목: 514개 종목
- 데이터 기간 : 2009-12-31 ~ 2023-12-31
  데이터 분할: Train, Valid, Test 6:2:2
- 전처리: 결측치 제거 및 np.inf 삭제
- 정규화: StandardScaler
- Input feature → Open, Close, High, Low, Volume, Vwap, Ticker
  Label feature → 20_day_return_rate

Learning rate	0.00001 → **0.000001**
Epoch	200
Batch size	64
Loss function	MSE Loss
Sequence Length	96
input_feature	7
Output_window	1

제안 모델

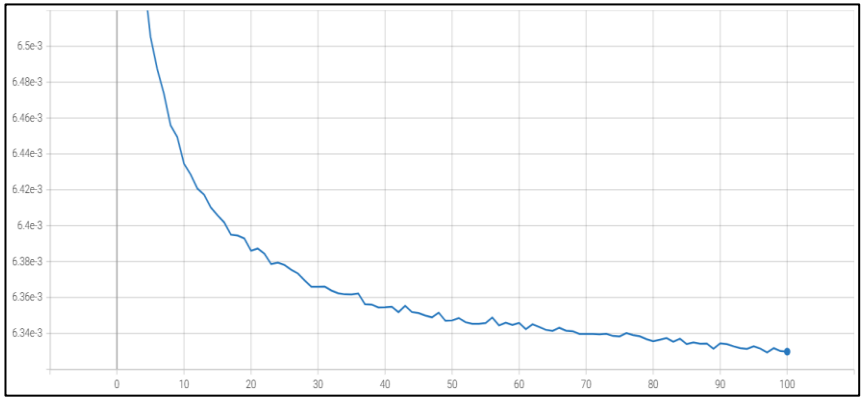
FAIR Lab

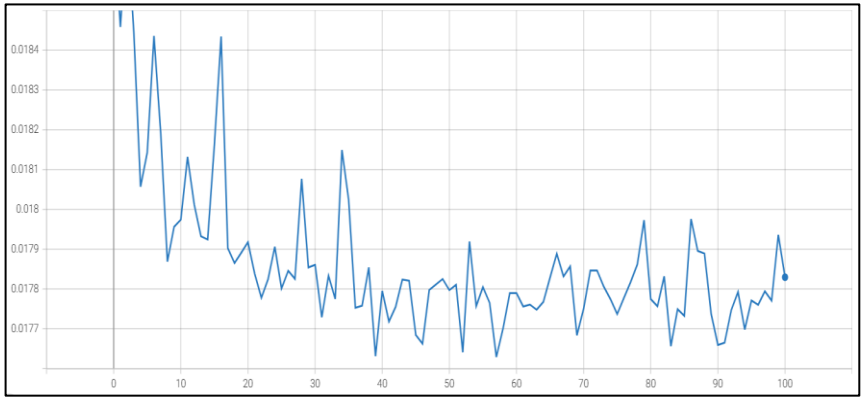
KWANGWOON UNIVERSITY

Baseline, Deep 모델 2D CNN 실험 결과

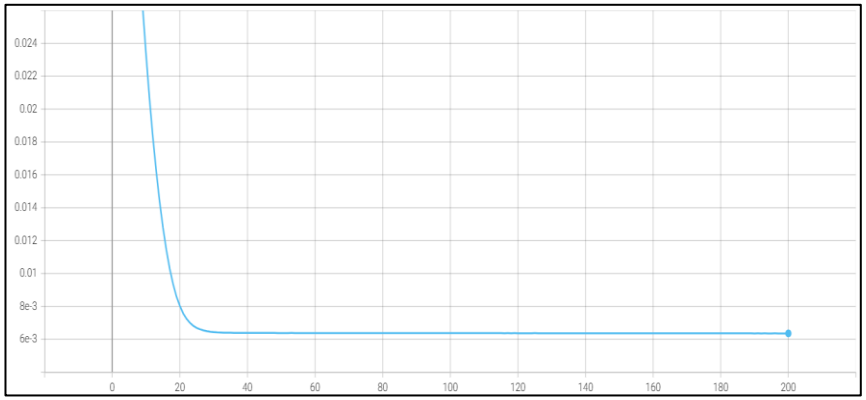
train loss **validation loss**

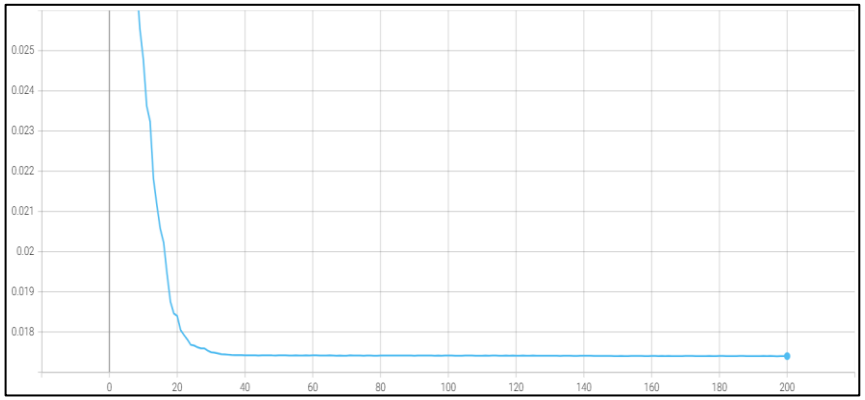
①





②





Test MSE ①	Test MSE ②
0.435675	0.435130

x: epoch y: loss

① → Baseline 모델 2D CNN 실험 결과
② → Deep 모델 2D CNN 실험 결과

Label scale → 1.0
Label → 20_day_return_rate

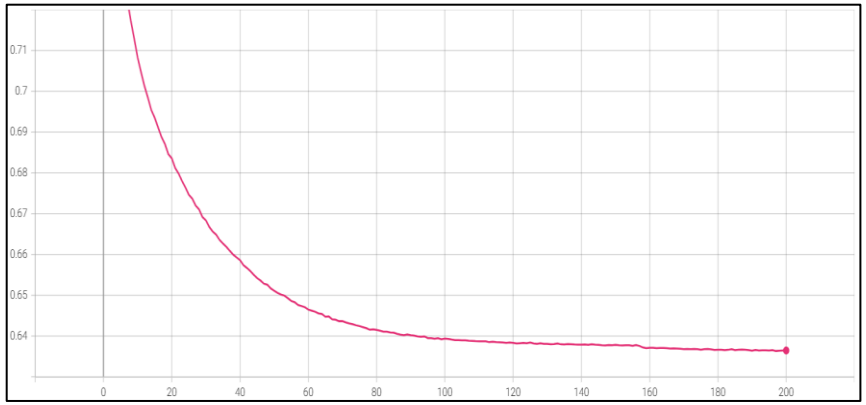
제안 모델

FAIR Lab

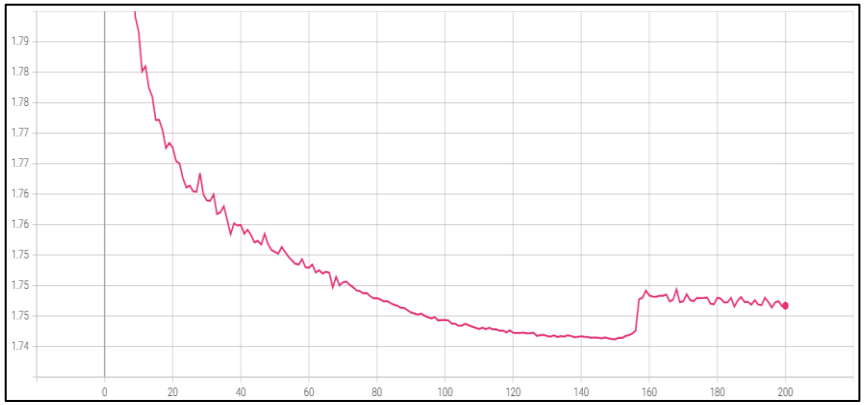
KWANGWOON UNIVERSITY

Deep 모델 2D CNN + Label scale 10.0

train loss



validation loss



x: epoch y: loss

Label scale → 10.0
Label → 20_day_return_rate

result

Test MSE
0.435182

Best Epoch	Min Valid Loss
143	1.741

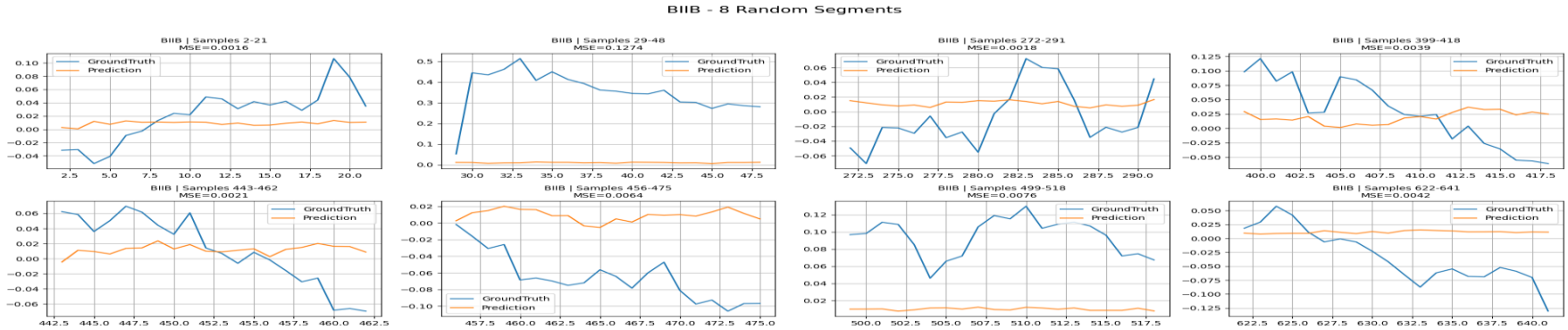
모델 시각화 비교

FAIR Lab

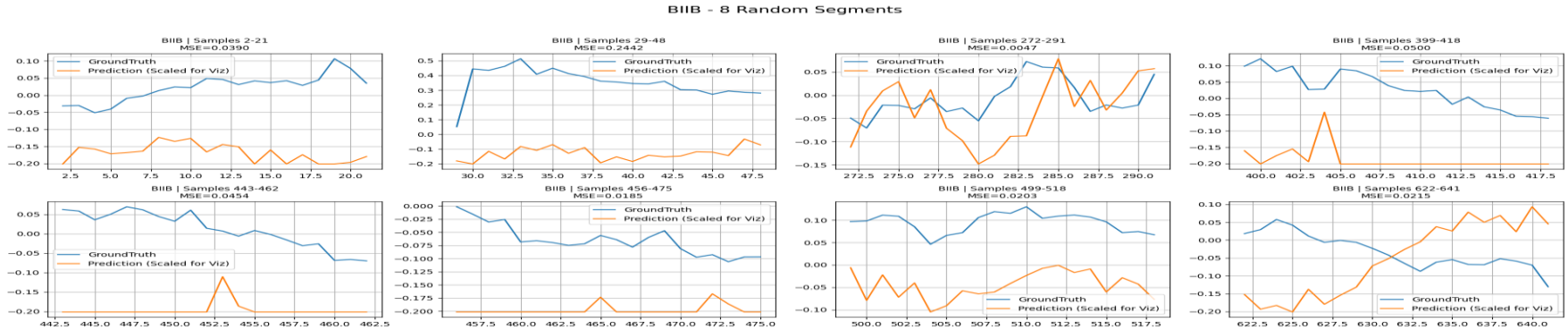
KWANGWOON UNIVERSITY

BIIB.csv 종목

①



②



x: 각 샘플의 index y: 해당 샘플의 값

Label scale → 1.0

① → Baseline 모델 BIIB.csv

② → Deep 모델 BIIB.csv

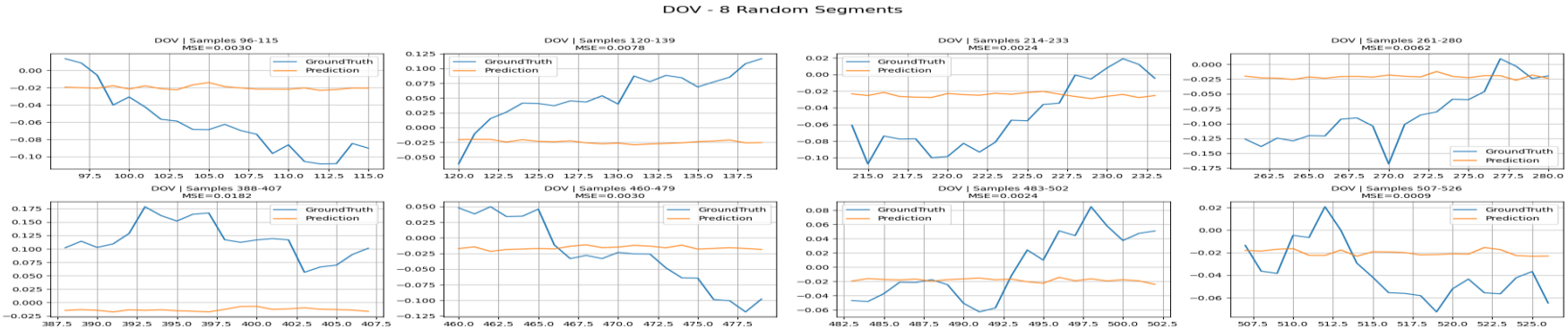
모델 시각화 비교

FAIR Lab

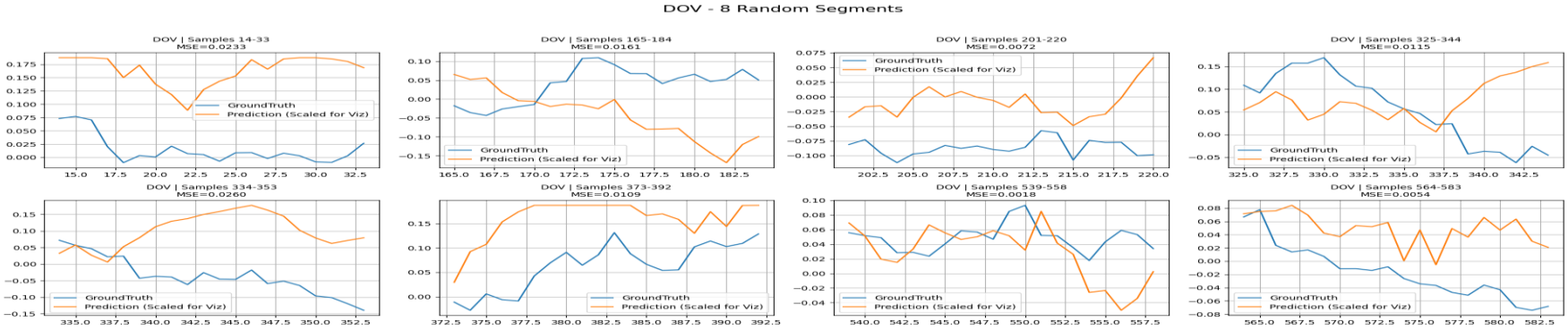
KWANGWOON UNIVERSITY

DOV.csv 종목

①



②



x: 각 샘플의 index y: 해당 샘플의 값

Label scale → 1.0

① → Baseline 모델 DOV.csv

② → Deep 모델 DOV.csv

정리

실험 결과 정리

- 2D CNN 결과 비교
  - Baseline, Deep 모델 Label scale 1.0의 경우, Baseline 모델과는 달리
    Deep 모델의 Loss curve가 요동치지 않고 빠르게 안정화되는 모습을 확인하였습니다.
- 샘플 시각화 결과
  - Baseline 모델은 예측값이 대부분 flat하게 나와 정답값을 따라가지 못했지만,
    Deep 모델은 어느정도 정답값을 따라가려는 모습을 보이고 있습니다.

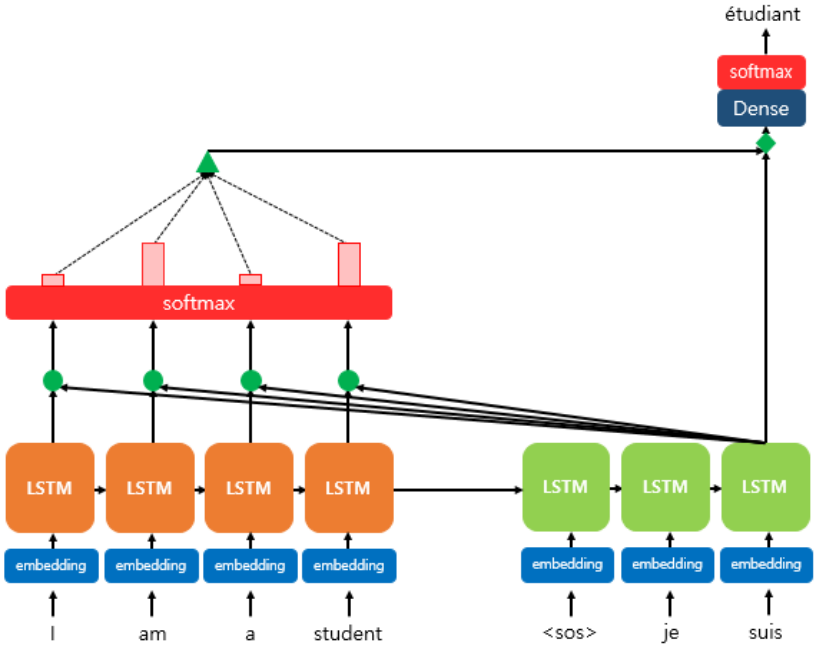
모델 채널	Label scale	Test MSE	학습 소요 시간
2 → 4	1.0	0.435675	6시간 45분 30초
8 → 16	1.0	0.435130	4시간 6분 44초
8 → 16	10.0	0.435182	5시간 59분 37초

이후 계획

- 모델 변경 또는 하이퍼파라미터 변경 후 실험 진행

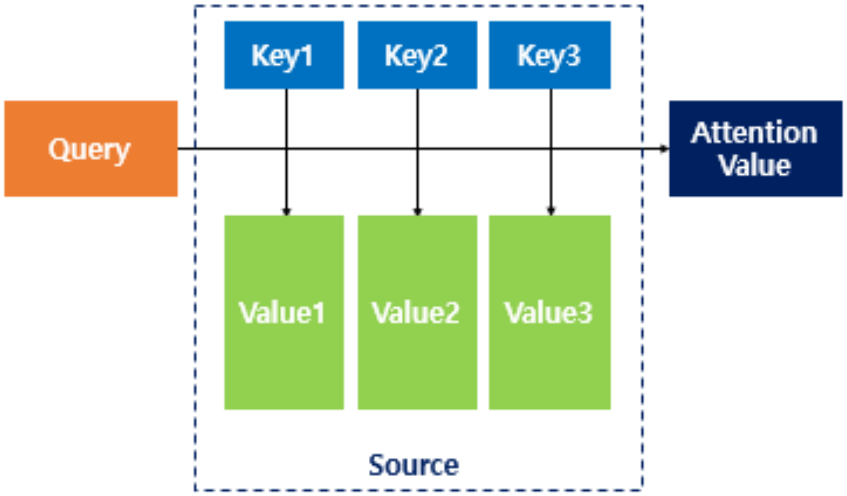
  → LSTM 또는 GRU 모델로 변경

Attention 리뷰



- Attention의 기본 아이디어는 디코더에서 출력 단어를 예측하는 매 시점마다 인코더에서의 원문 문장을 다시 한 번 참고하는데, 원문 문장을 전부 다 동일한 비율로 참고하는 것이 아닌 해당 시점에서 예측해야 할 단어와 연관이 있는 입력 단어 부분을 좀 더 집중(attention)해서 보게 된다.
  → 이러한 Attention 메커니즘을 통해 각 단어의 중요도를 동적으로 파악하여 문맥을 더 잘 이해하게 된다.

Attention 리뷰



- Attention 함수 표현 → **Attention(Q, K, V) = Attention Value**

  Attention 함수는 주어진 Query에 대해서 모든 Key와의 유사도를 각각 구한 뒤, 구해낸 유사도를 키와 맵핑되어 있는 각각의 Value에 반영해주고 유사도가 반영된 Value를 모두 더해 리턴한다.

  이를 attention 값이라고 한다.

- seq2seq + Attention 모델의 Q, K, V →

  Q : t 시점의 디코더 셀에서의 은닉 상태

  K : 모든 시점의 인코더 셀의 은닉 상태들

  V : 모든 시점의 인코더 셀의 은닉 상태들