**DL-HW #11**

2015004693\_양상헌

*실행환경: MAC OS TERMIINAL( MACBOOK PRO 2015 RETINA , MOJAVE 10.14.6 ) , ANACONDA*

1. **Source Code:**

* *Assignment11 폴더 참조*

*(DL\_HW\_11week.py)*

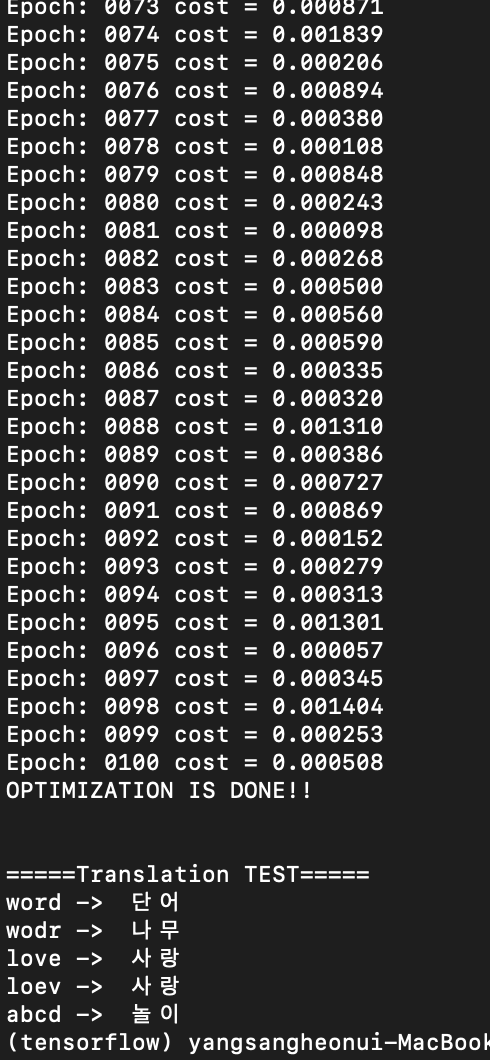
스크린샷이(가) 표시된 사진

자동 생성된 설명

11주차 실습 자료에 나온 SEQ2SEQ 모델에 간략하게 Attention을 추가해 주었다.

1. **Result:**

대형이(가) 표시된 사진

자동 생성된 설명 

1. **Disscussion:**

**< 코드 설명 >**

스크린샷이(가) 표시된 사진

자동 생성된 설명

소스코드의 전체적인 구조는 11주차 실습 자료에 나온 seq2seq 모델의 구조를 변경하지 않고 Attention기법만 추가한 구조이고, 실제로 매우 짧은 코드만 추가하여 구현하였다. 이 과제에서 사용된 Attention 매커니즘은 가장 단순한 방법 중 하나인 백터 내적을 이용 하는 것이다.

모든 인코더의 은닉 상태들의 정보를 가져와서 현재 디코더의 은닉 상태와 백터 내적을 진행하고, 이를 Softmax 활성화 함수를 사용하여 알파 값으로 만들어주었다. 이후에 각각의 인코더의 은닉 상태들 마다 생성된 알파 값을 곱한 다음 모두 더해주었다. 이렇게 만들어진 벡터가 바로 Attention 벡터이고 이를 디코더의 아웃풋에 tf.concat을 이용하여 붙여주었다. 기존의 디코더 아웃풋에는 Attention에 관련된 정보를 추가해 주었고, 이 Attention정보가 추가된 디코더의 아웃풋을 이용하여 모델을 동작하도록 하였다.

사용된 Attention 메커니즘 자체의 수학식은 복잡해 보일 수 있지만 위의 3줄의 Matrix 연산으로 이를 표현하는 것이 가능하다. 백터 내적을 행렬 곱으로 표현하는데 매우 큰 어려움이 있었고, Matrix 곱셈에서 차원을 맞춰주는 부분도 많은 시간이 들었다. (추가된 코드는 매우 짧지만, 지금까지 했던 과제들 중 가장 어려웠고, 가장 오랜시간이 걸린 과제이다.)

**< 결과 분석 >**

많은 횟수를 실행해 보았을 때, 평균적으로 기존의 seq2seq보다 더 높은 성능을 내는 것을 확인 할 수 있었다. 기존의 Seq2seq 모델은 거리가 멀어질 수 록 결과값에 영향을 미치는 부분이 약해지지만, Attention 값을 추가해줌으로써 거리에 상관없이 더 중요한 단어를 선택하여 결과값을 만들어 주게 됨으로 더 좋은 결과가 나오는 것을 확인할 수 있다.