Attention

- '중요한 부분만 집중(attention)하게 만들자'가 핵심 아이디어
- 특정 벡터에 주목하게 만들어 모델의 성능을 높이는 기법

Seq2Seq의 문제점을 보완하고자 사용하기 시작함
Seq2Seq의 문제점이란?
하나의 고정된 크기의 벡터(Context vector, 후에 어텐션에서 사용하는 Context vector와는 다름)에 모든 정보를 압축하려고 하니까 정보 손실이 발생

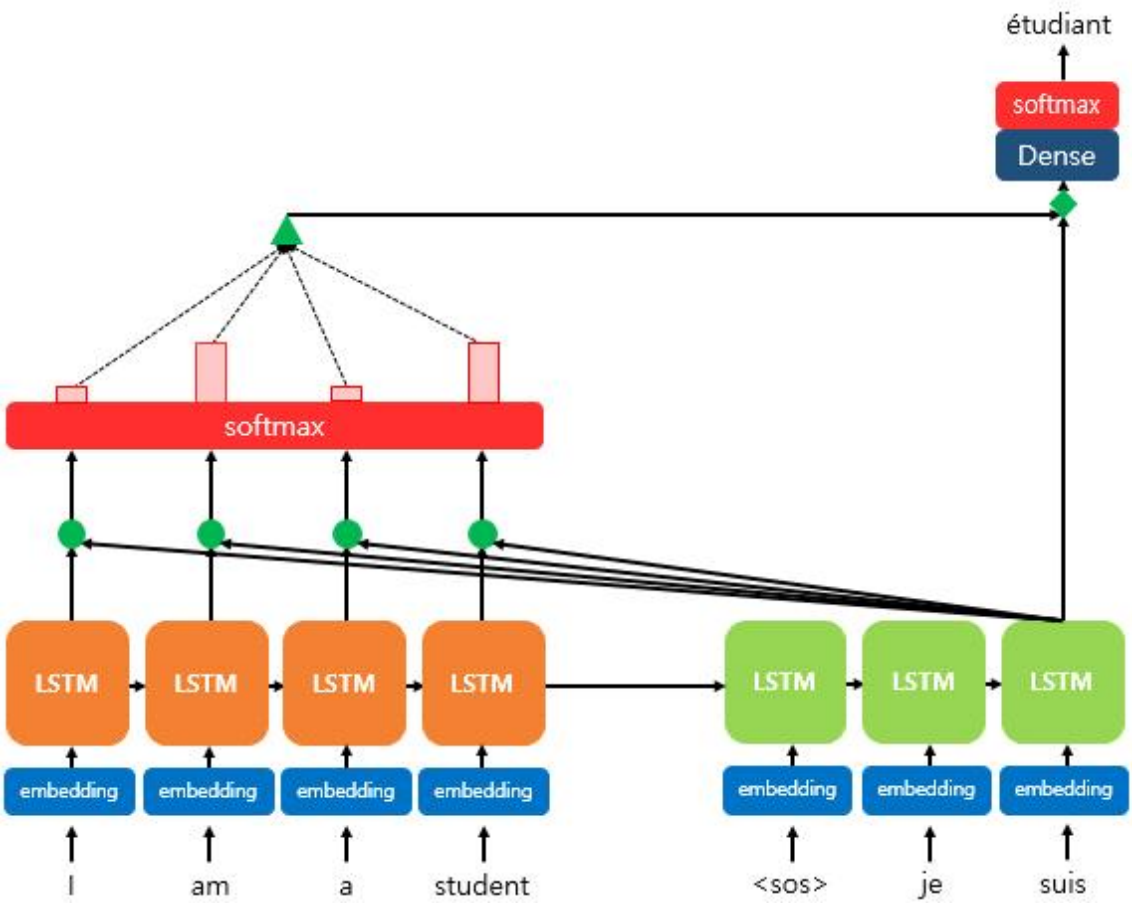
어텐션 종류 (**스코어 함수**가 다름, 작동원리는 같음)
- Dot Attention
- Scaled dot Attention
- General Attention
- Concat Attention

동작원리

1. 닷-프로덕트 어텐션 (Dot-Product Attention)

순서

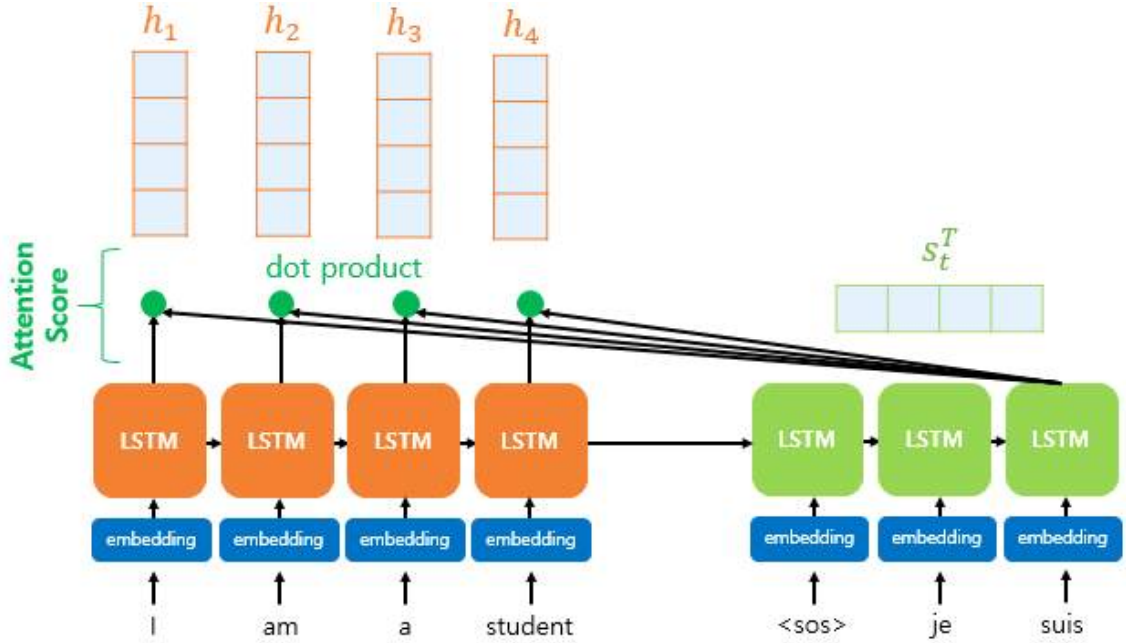
1) 어텐션 스코어(Attention Score)를 구함
2) 소프트맥스(softmax) 함수를 통해 어텐션 분포(Attention Distribution)를 구함
3) 각 인코더의 어텐션 가중치와 은닉 상태를 가중합하여 어텐션 값(Attention Value)을 구함
4) 어텐션 값과 디코더의 t 시점의 은닉 상태를 연결 (Concatenate)
5) 출력층 연산의 입력이 되는 St를 계산
6) St를 출력층의 입력으로 사용



현재 상황: 디코더의 3번 째 LSTM 셀에서 출력 단어를 예측하는 그림

- softmax 함수를 통해 각각의 입력 단어 중 어떤 단어를 중점적으로 봐야할지를 수치화하게 됨
- 위의 그림에서는 am, student를 중심으로 봄
- 수치화한 정보를 하나의 정보로 담아 디코더로 전송(초록색 삼각형)

1) 어텐션 스코어(**Attention Score**)를 구함

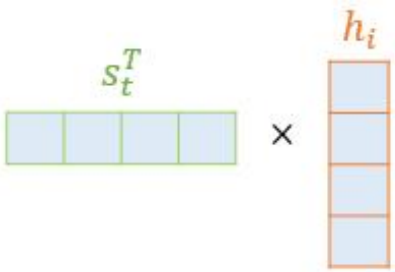


h1 ~ hN : 인코더의 hidden state
St : 디코더의 hidden state

디코더의 셀은 3개의 입력값을 필요로 함
- t-1의 hidden state
- t-1의 출력 단어
- Attention Value

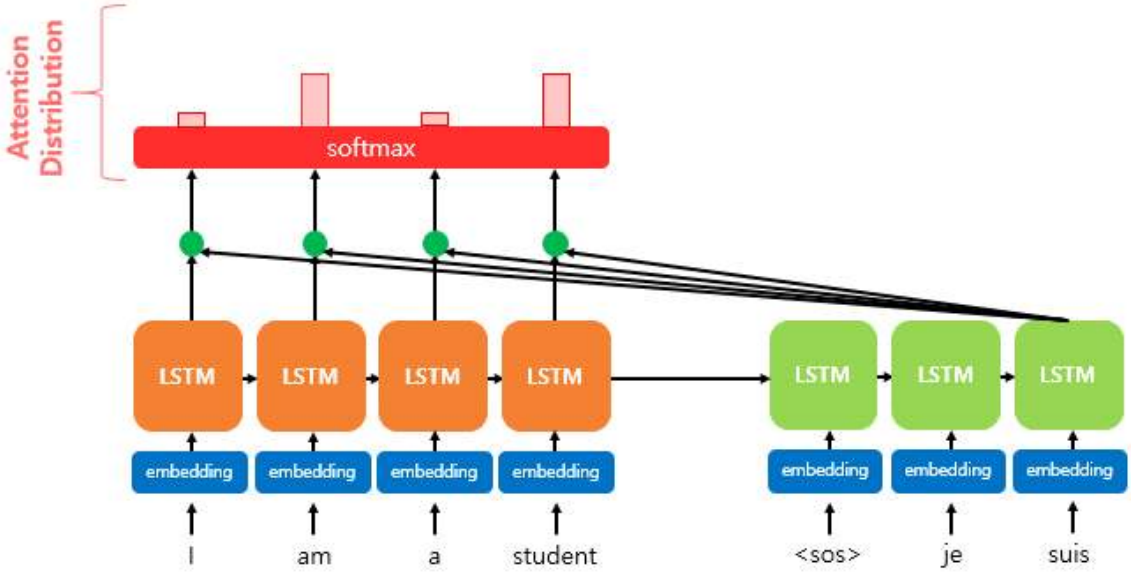
Attention Score란? (논문에서는 **Energy**라고 함)
현재 디코더의 시점 t에서 단어를 예측하기 위해, 인코더의 모든 hidden state 각각이 디코더의 현 시점의 hidden state(St)와 얼마나 유사한지 판단하는 점수값

계산 방법 = dot product => attention이름이 dot-product attention인 이유



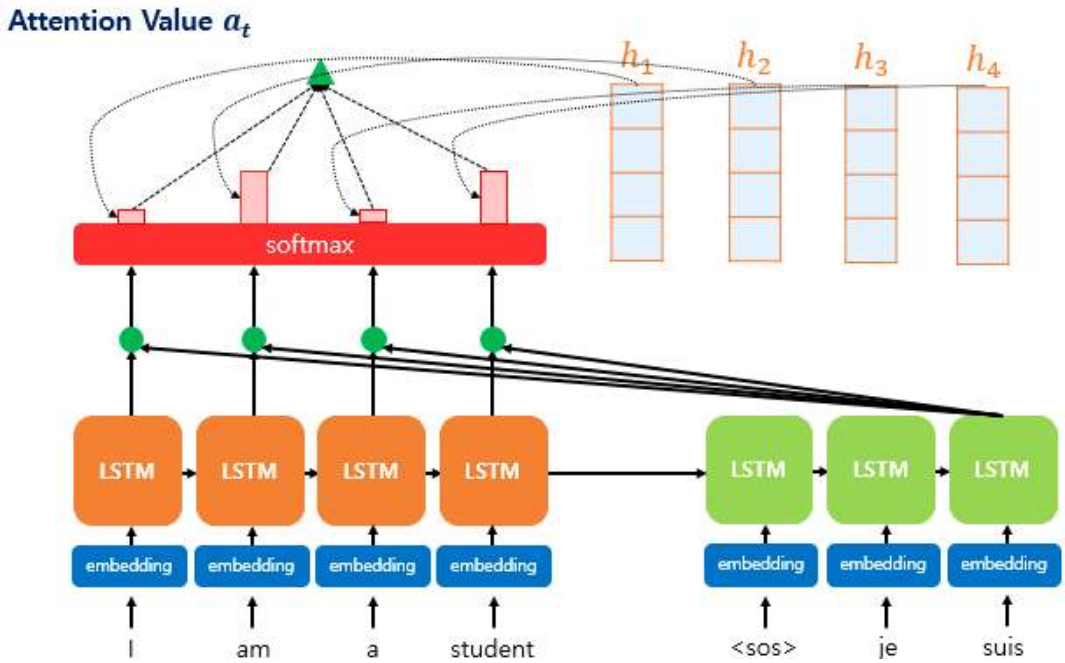
계산 한 값들을 i부터 N까지 모은 값 = e^t
= 소스 문장에서 나왔던 모든 출력 값들 중에서 어떤 값과 가장 연관성이 있는지 나타내줌

2) 소프트맥스(softmax) 함수를 통해 어텐션 분포(**Attention Distribution**)를 구함



어텐션 가중치를 모두 합하면 1이 됨, 논문에서는 **Weight**라고 함

3) 각 인코더의 어텐션 가중치와 은닉 상태를 가중합하여 **어텐션 값(Attention Value)**을 구함

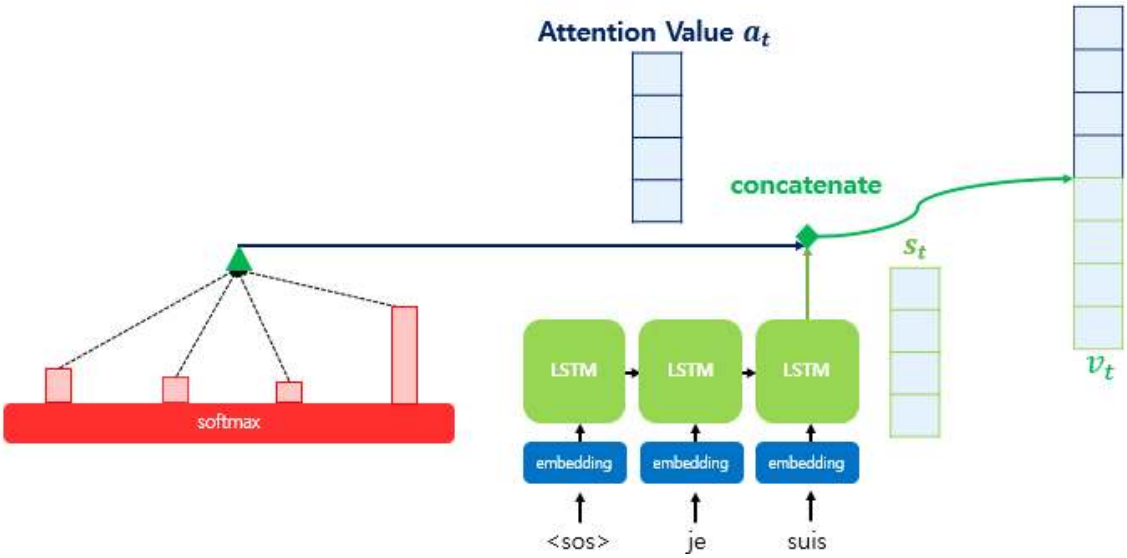


하나의 정보로 합침

각 인코더의 hidden state와 attention distribution(가중치)을 곱하고 곱한 값들을 모두 더함

$a_t = \sum_{i=1}^{N} \alpha_i^t h_i$, 이를 Context vector라 함(seq2seq의 context vector와 의미가 조금 다름)

4) 어텐션 값과 디코더의 t 시점의 은닉 상태를 연결 (Concatenate)



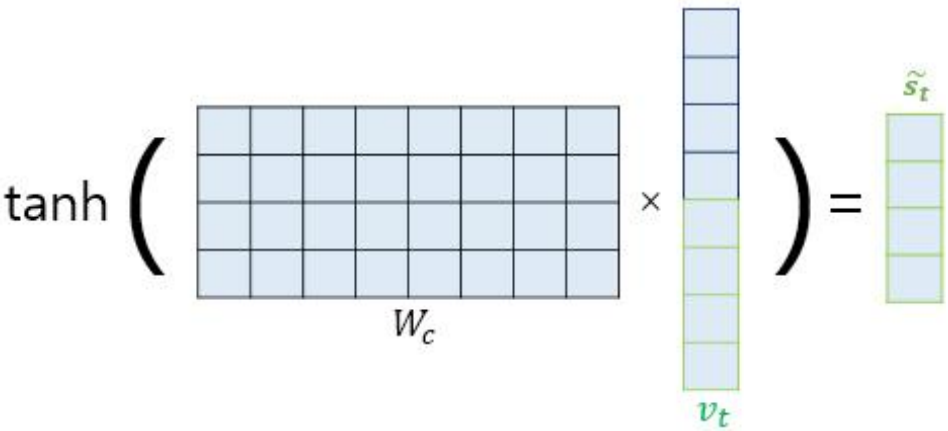
a_t : 어텐션 값

s_t : 디코더의 hidden state

위 2개의 값을 결합(concatenate)하여 하나의 벡터로 만듬 = v_t

위 v_t값을 연산의 입력으로 사용

5) 출력층 연산의 입력이 되는 $\widetilde{s_t}$를 계산



6) $\widetilde{s_t}$를 출력층의 입력으로 사용

$$\hat{y} = Softmax(W_y\widetilde{s_t} + b_y)$$

Summary

Attention 등장 배경

- Seq2Seq의 문제점을 보완하고자 사용하기 시작
- 특정 벡터에 주목하게 만들어 모델의 성능을 높임

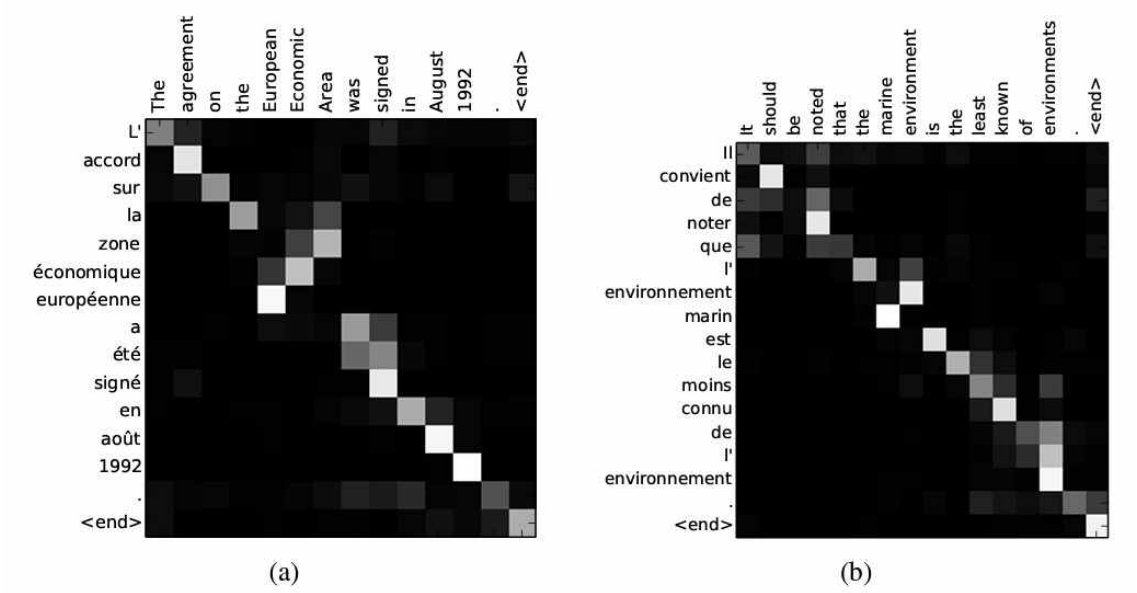
Attention의 주요 특징

- Attention Score (Energy)
- Attention Distribution (Weight)
- 위 값들을 이용해 Attention Value를 구함

Attention의 장점

- Seq2Seq의 단점(고정된 context vector의 사이즈)을 해결
- 즉, 한 개의 context vector가 아닌 여러 개의 context vector를 계속해서 계산해서 사용
- 인코더의 모든 부분 중 집중하고 싶은 부분을 구분할 수 있는 매커니즘

논문에서 나온 영어-불어 단어 유사도 그림



Reference

어텐션을 소개한 논문
- **Neural Machine Translation by Jointly Learning to Align and Translate (ICLR 2015)**
Dzmitry Bahdanau, Kyunghyun Cho(조경현), Yoshua Bengio
(https://arxiv.org/pdf/1409.0473.pdf)

참고 사이트
- https://ratsgo.github.io/from%20frequency%20to%20semantics/2017/10/06/attention/
- https://wikidocs.net/22893