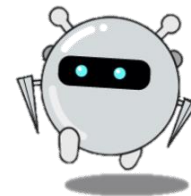


# Seq2Seq

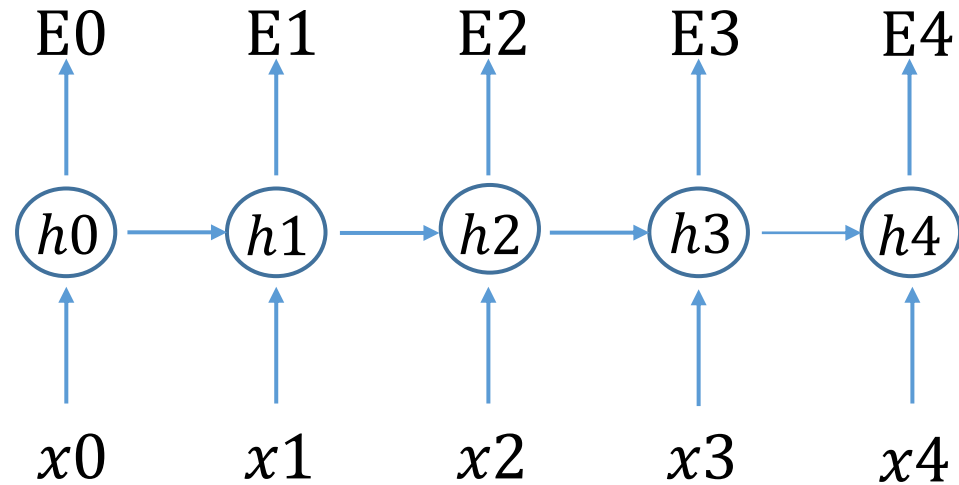
멘토 현시은



# Seq2Seq

- Seq2Seq

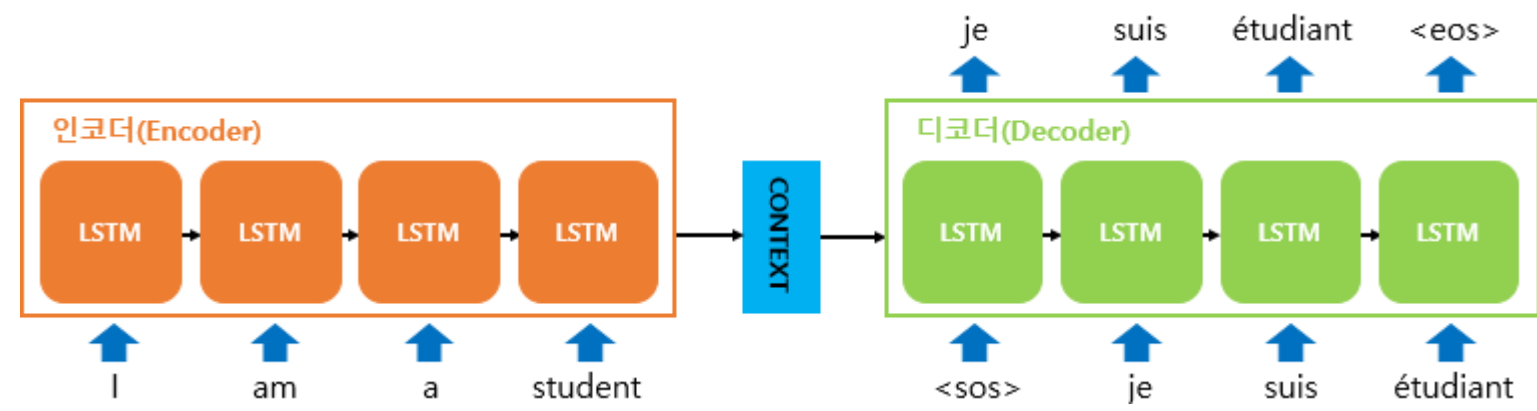
## RNN의 한계



- 과거의 정보 소실 문제
- 각 step에서 입력과 출력의 크기가 같지 않는 경우 사용이 까다롭다.
- Casuality(인과율)이 성립. 즉  $t$ 번째 step의 결과는  $t+1$ 의 결과를 참고할 수 없음.

- Seq2Seq

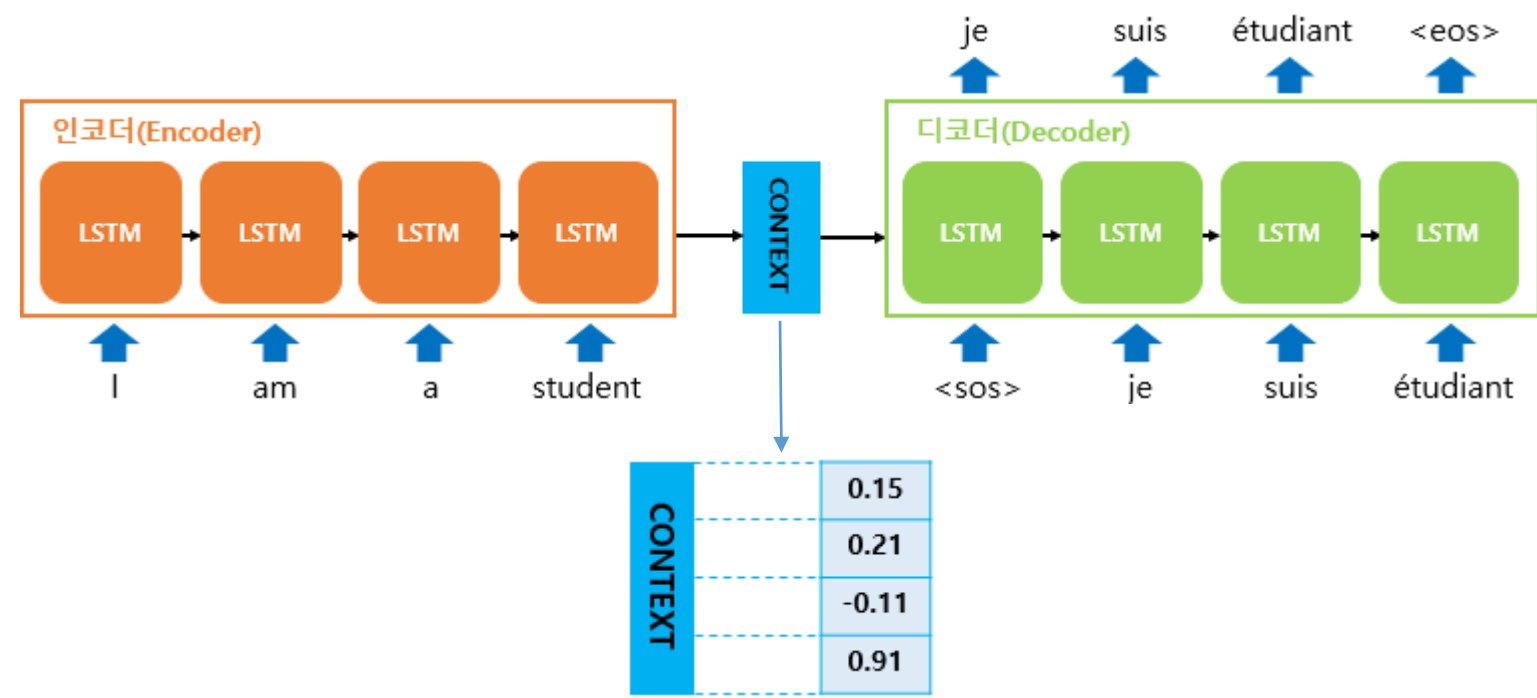
# Seq2Seq



- RNN을 어떻게 조립했느냐 따라서 Seq2Seq 구조가 만들어 진다.
- 두개의 RNN로 이루어져 있음.
- 모든 정보를 내포하고 있는 인코더 RNN 셀의 마지막 시점의 은닉 상태를 (Context vector) 디코더 RNN셀의 첫 은닉 상태로 넣어준다.

- Seq2Seq

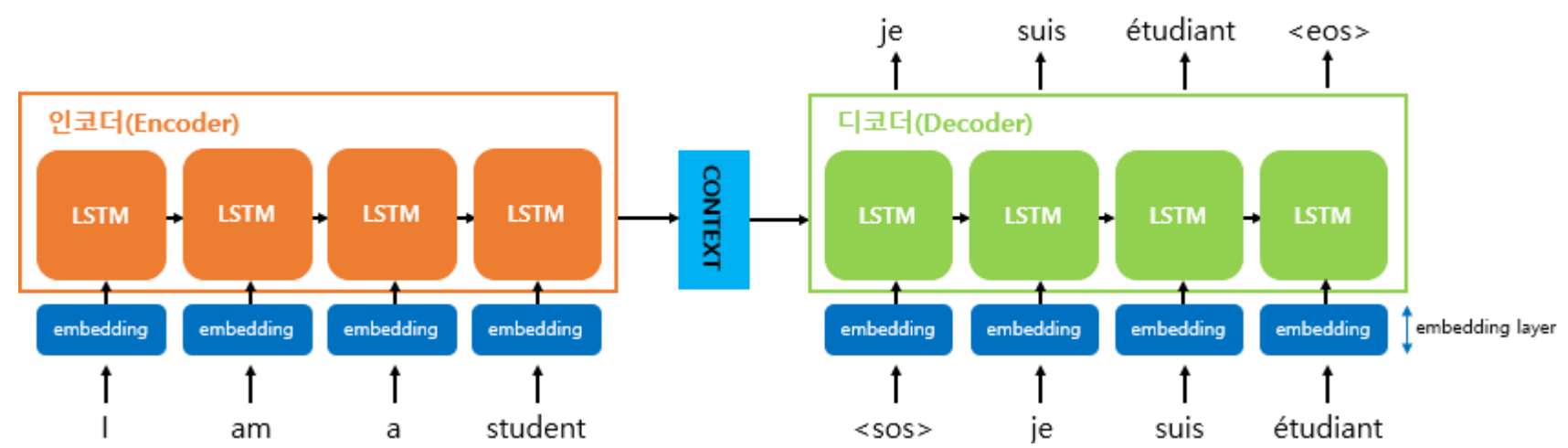
# Seq2Seq



- 실제 Context vector는 수백 이상의 차원을 갖고 있다.
- Context vector를 Decoder의 초기 은닉 상태 사용 할 수도, 디코더가 단어를 예측하는 매 시점마다 input으로 넣어 줄 수도 있다.

- Seq2Seq

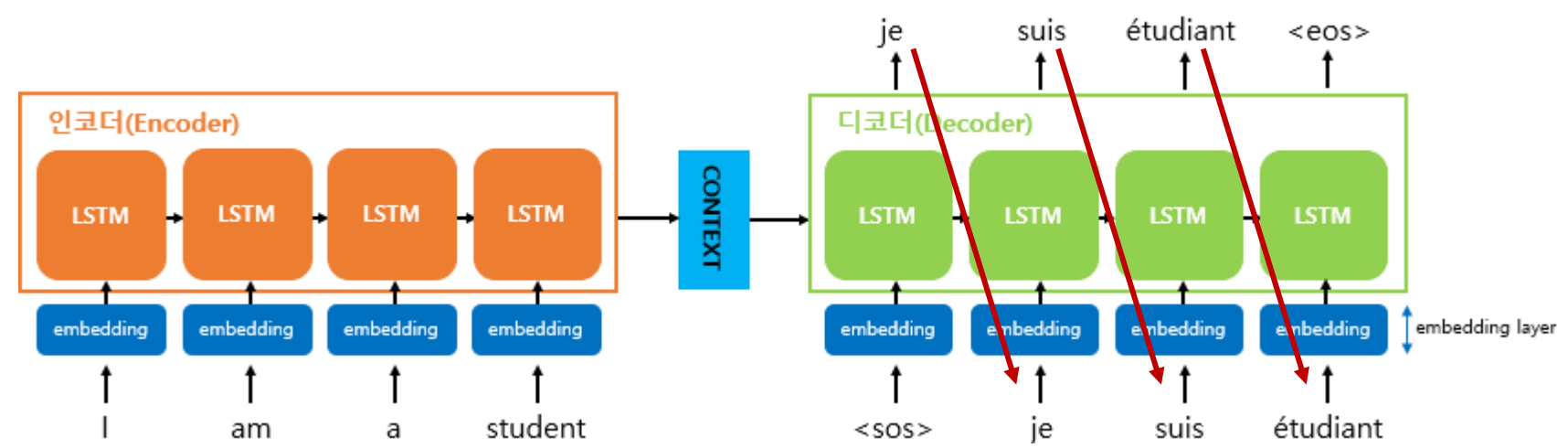
Seq2Seq



I		0.157	am		0.78	a		0.75	student		0.88
		-0.25			0.29			-0.81			-0.17
		0.478			-0.96			0.96			0.29
		-0.78			0.52			0.12			0.48

- Seq2Seq

# Seq2Seq - 학습과정

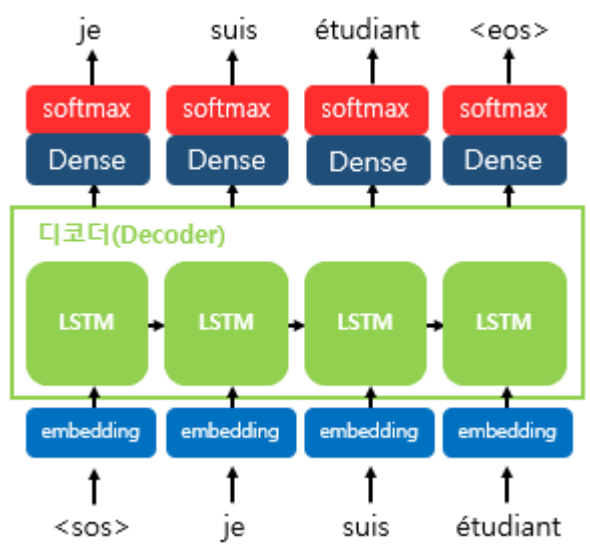


- Training : teaching force / Test(= Inference) : Greedy decoding

디코더의 input에 이전 시점의 output 넣어줌  
output 토큰 선정 기준 : 가장 높은 확률 값

- Seq2Seq

# Seq2Seq - 학습과정



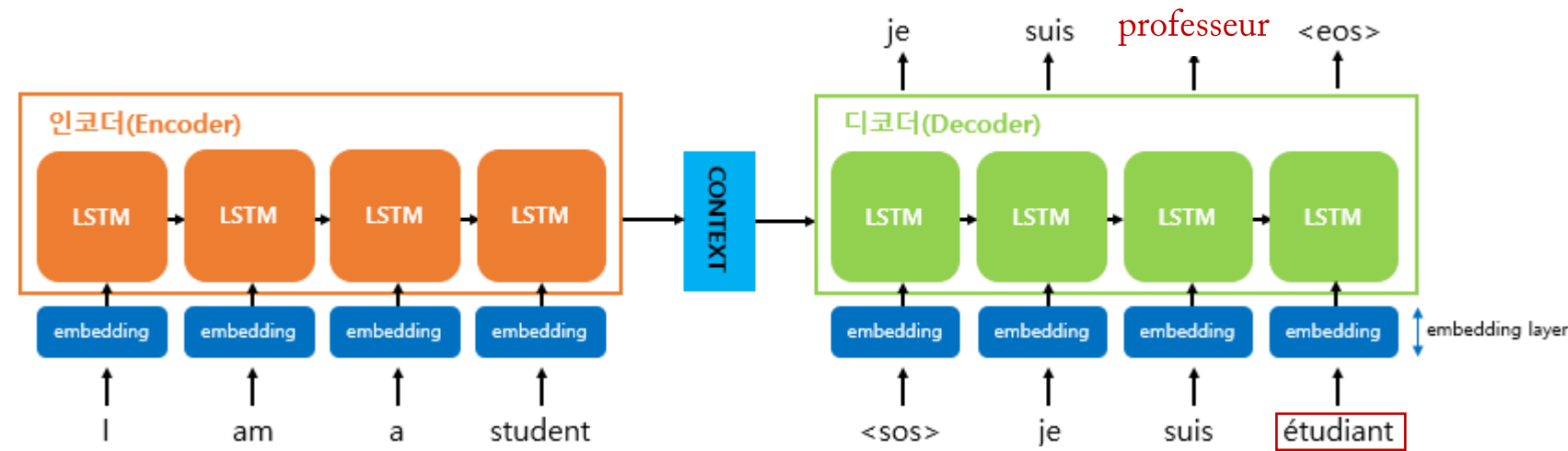
- Training : teaching force / Test(= Inference) : Greedy decoding

디코더의 input에 이전 시점의 output 넣어줌  
output 토큰 선정 기준 : 가장 높은 확률 값



- Seq2Seq

# Seq2Seq - 학습과정

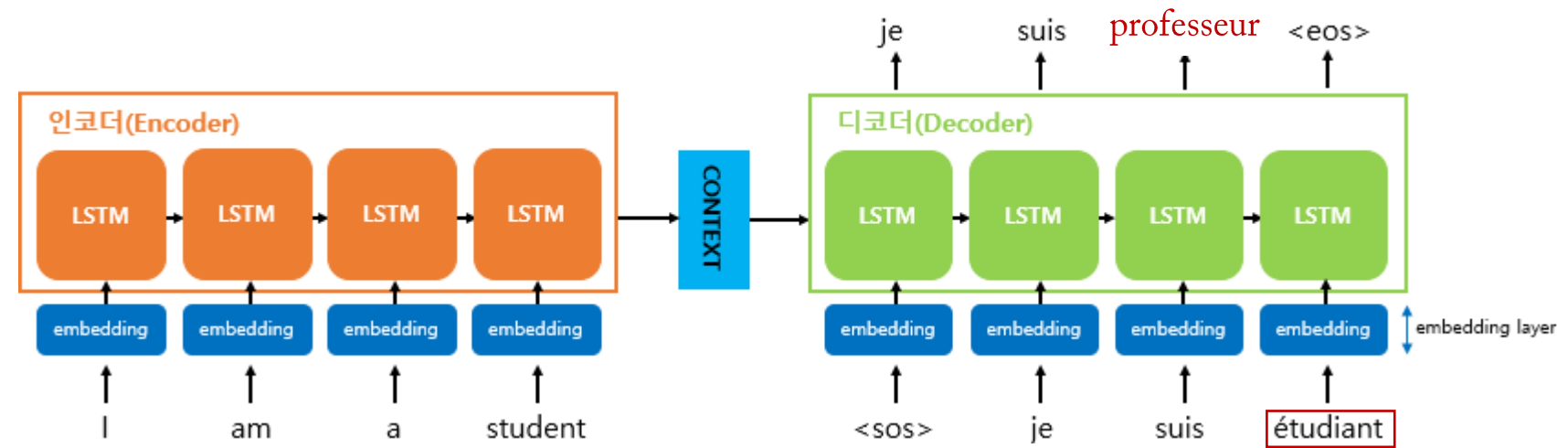


- Training : teacher force / Test(= Inference) : Greedy decoding

디코더의 input에 정답이 들어감.  
이전 step의 결과가 오답일 경우, 이후 학습에 영향 미치지 않도록 방지

- Seq2Seq

# Seq2Seq - 학습과정



- Training : teacher force / Test(= Inference) : Greedy decoding

Cross-Entropy Loss 사용  
각 step에 Cross-Entropy Loss를 모두 더하거나, 평균을 내서 loss로 사용



감사합니다

참고자료 : <https://wikidocs.net/24996>