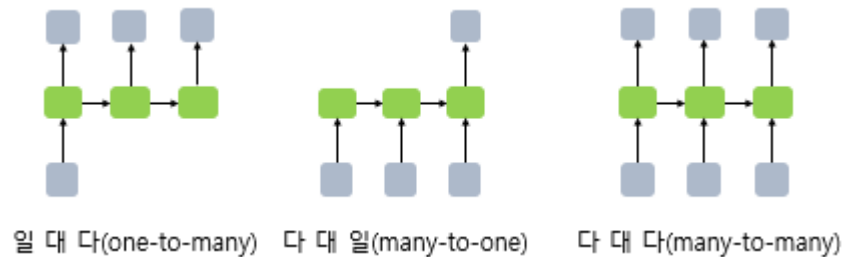


시퀀스 투 시퀀스 (Sequence-to-Sequence, seq2seq)



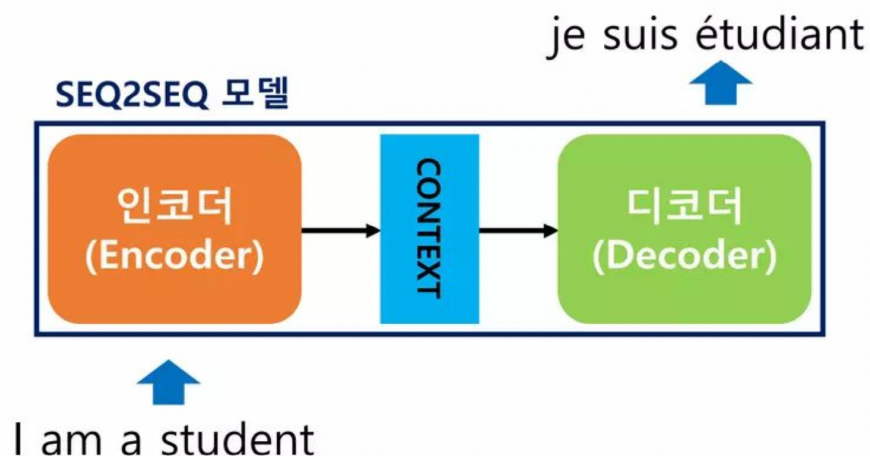
RNN을 이용하면 다 대 일 구조로 텍스트를 분류할 수 있고, 다 대 다 구조로는 객체명 인식이나 품사 태깅과 같은 문제를 풀 수 있다.

자연어 처리에서 번역기나 텍스트 요약과 같은 경우 입력 문장과 출력 문장의 길이가 다를 수 있는데 이때 시퀀스 투 시퀀스 모델을 사용할 수 있다.

시퀀스-투-시퀀스(Sequence-to-Sequence, seq2seq)

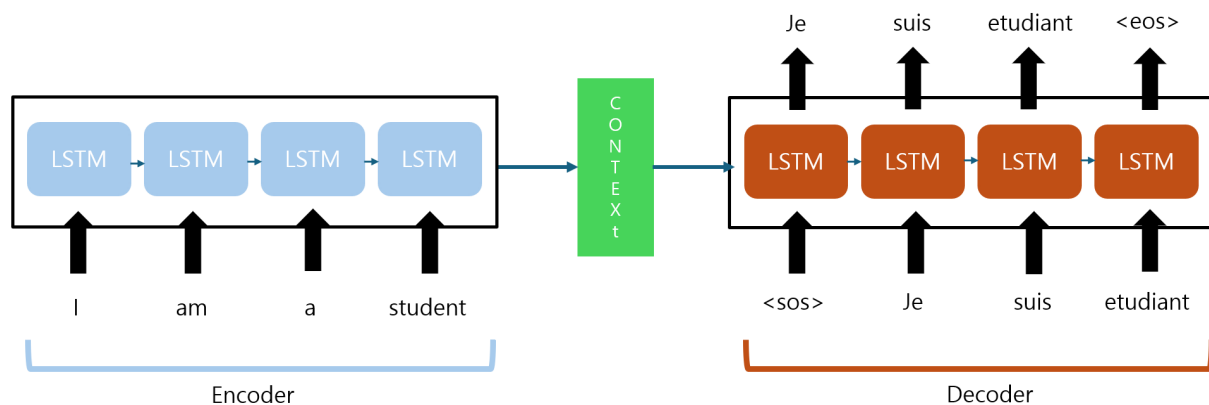
입력된 시퀀스로부터 다른 도메인의 시퀀스를 출력하는 Encoder-Decoder 모델

seq2seq 는 번역기에서 대표적으로 사용되는 모델로, 기본적으로 **RNN**을 어떻게 조립했느냐에 따라서 **seq2seq**라는 구조가 만들어진다.



seq2seq 는 크게 인코더(Encoder) 와 디코더(Decoder)라는 두 개의 모듈로 구성 된다.

- 인코더: 소스 시퀀스를 받아 고정 길이의 벡터로 변환합니다. 이 벡터는 소스 시퀀스의 문맥 정보를 담고 있으며, 디코더에게 전달되어 출력 시퀀스의 생성 기반으로 사용됩니다.
- 디코더: 인코더로부터 받은 정보를 사용하여 타겟 시퀀스를 단계별로 생성합니다. 각 단계에서 디코더는 이전 단계의 출력을 참고하여 다음 단어를 예측합니다.



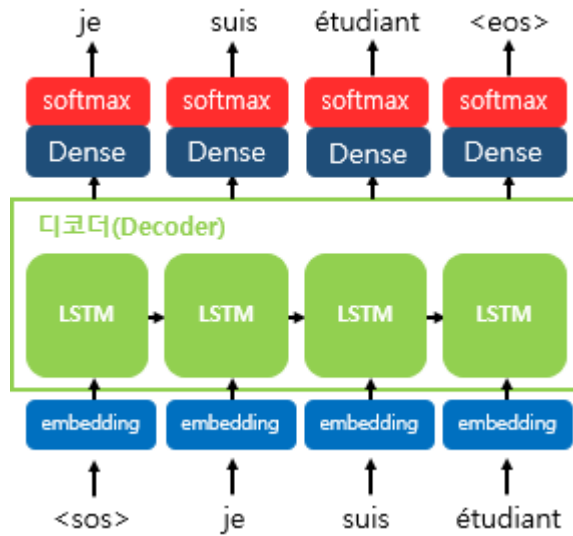
보통 RNN보다 LSTM, GRU와 같은 RNN 알고리즘을 변형한 모델이 셀로 들어감 -> 기존의 RNN은 기울기 소실 또는 기울기 폭발 문제 발생

인코더 설명

- 인코더를 살펴보면 입력 문장이 단어 토큰화를 통해 단어 단위로 쪼개지고 단어 토큰 각각은 RNN셀(LSTM, GRU)의 각 시점의 입력이된다.
- 인코더의 RNN 셀은 모든 단어를 입력 받은 뒤 인코더 RNN 셀의 마지막 시점의 은닉 상태를 디코더 RNN 셀로 넘겨주는데 이를 **컨텍스트 벡터**라고 한다.
- 컨텍스트 벡터는 **디코더 RNN 셀의 첫 번째 은닉 상태**로 사용된다.

디코더 설명

- 디코더는 문장의 시작을의미하는 <sos>심볼이 들어간다.
심볼이 입력되면 다음에 등장할 확률이 높은 단어를 예측한다. 이렇게 다음에 올 단어를 예측하고 그 예측한 단어를 다음 시점의 RNN 셀의 입력으로 넣는 행위를 반복한. 문장의 끝을 의미하는 심볼인 <eos>가 다음 단어로 예측될 때까지 반복된다.
- 디코더에서 각 시점(time step)의 RNN 셀에서 출력 벡터가 나오면, 해당 벡터는 소프트맥스 함수를 통해 출력 시퀀스의 각 단어별 확률값을 반환하고, 디코더는 출력 단어를 결정한다.



seq2seq는 훈련 과정과 테스트 과정의 작동 방식이 조금 다르다.

훈련 과정

디코더에게 인코더가 보낸 컨텍스트 벡터와 실제 정답인 상황인 je suis étudiant를 입력 받았을 때, je suis étudiant 가 나와야 된다고 정답을 알려주면서 훈련한다.

테스트 과정

앞서 설명한 과정과 같이 디코더는 오직 컨텍스트 벡터와 만을 입력으로 받은 후에 다음에 올 단어를 예측하고, 그 단어를 다음 시점의 RNN 셀의 입력으로 넣는 행위를 반복한다.