

15장 어텐션 메커니즘 (Attention Mechanism)

어텐션 메커니즘과 어텐션과 바닷나우 어텐션에 대해 알아봅시다!

2023년도 동계인턴 스터디 4주차
박성호

15-1 Attention Mechanism

입력 시퀀스가 길어질 때 seq2seq 문제점 해결 기법

- RNN에 기반한 seq2seq는 크게 두 가지 문제가 있는데,
 1. 고정된 크기 벡터에 정보 압축에 따른 정보 손실
 2. 고질적인 RNN의 기울기 소실 문제→ 출력 시퀀스의 정확도가 떨어진다.
- 어텐션의 아이디어는 예측하는 매 시점 디코더에서 전체 입력 문장을 참고하되, 해당 시점 예측해야 할 단어와 연관이 있는 입력 단어부분을 집중해서 본다.
- $Attention(Q, K, V) = AttentionValue$

Attention

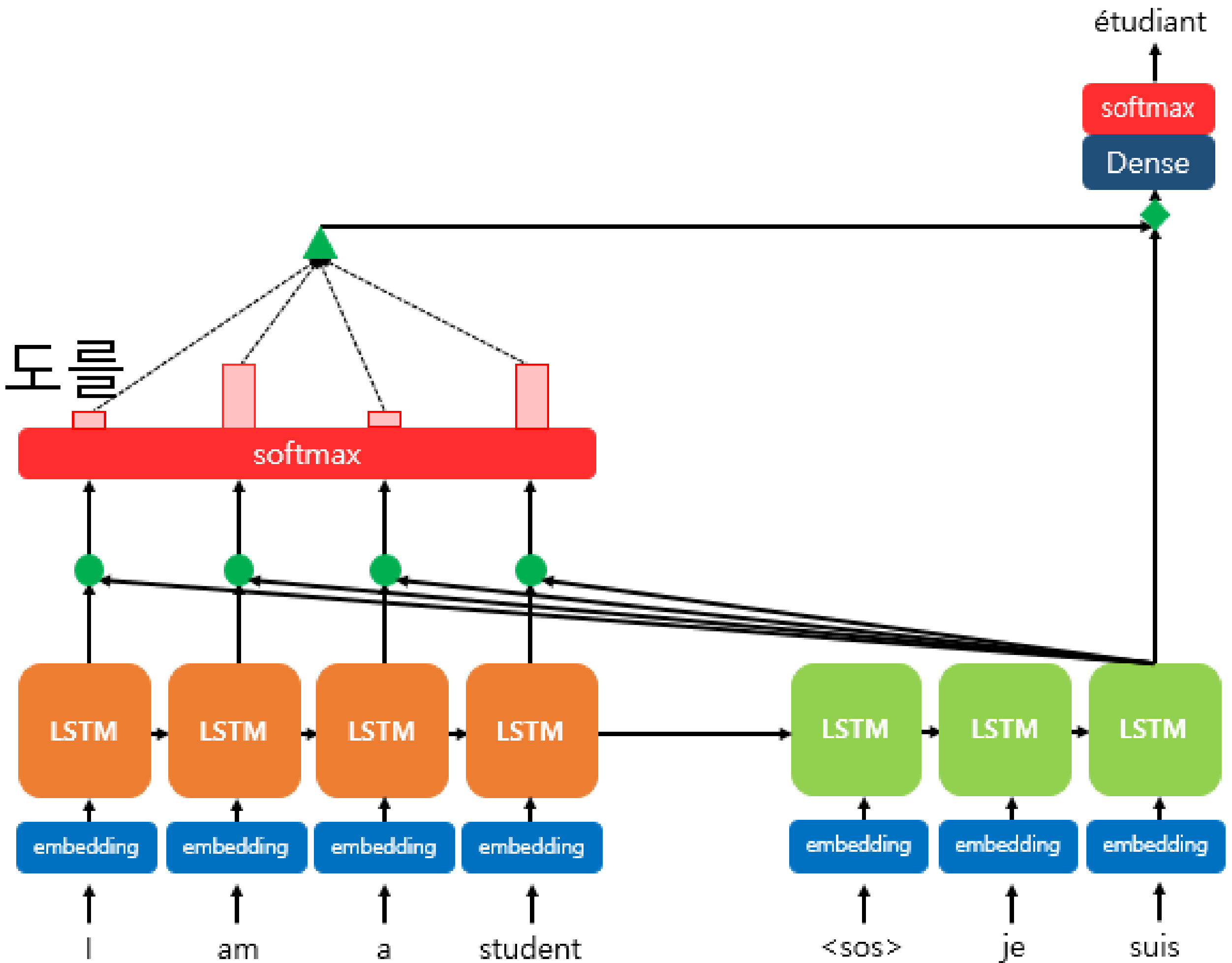
- 어텐션 함수는 주어진 Query에 대해서 모든 Key와의 유사도를 Key와 맵핑된 Value에 반영한다.

즉 어텐션은 Q 벡터, K 벡터, V 벡터를 입력으로 받아 Query와 Key-Value쌍을 출력에 맵핑 하는 것

- *“By letting the decoder have an attention mechanism, we relieve the encoder from **the burden of** having to encode all information in the source sentence into a **fixed length vector**. “*
(Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." arXiv preprint [arXiv:1409.0473](https://arxiv.org/abs/1409.0473) (2014).)

Dot-Product Attention

- 'je', 'suis'를 예측 한 상황에서 세 번째 LSTM 셀로 출력(étudiant)을 예상할 때, 각각의 입력 단어가 도움이 되는 정도를 소프트맥스 출력값으로 표현

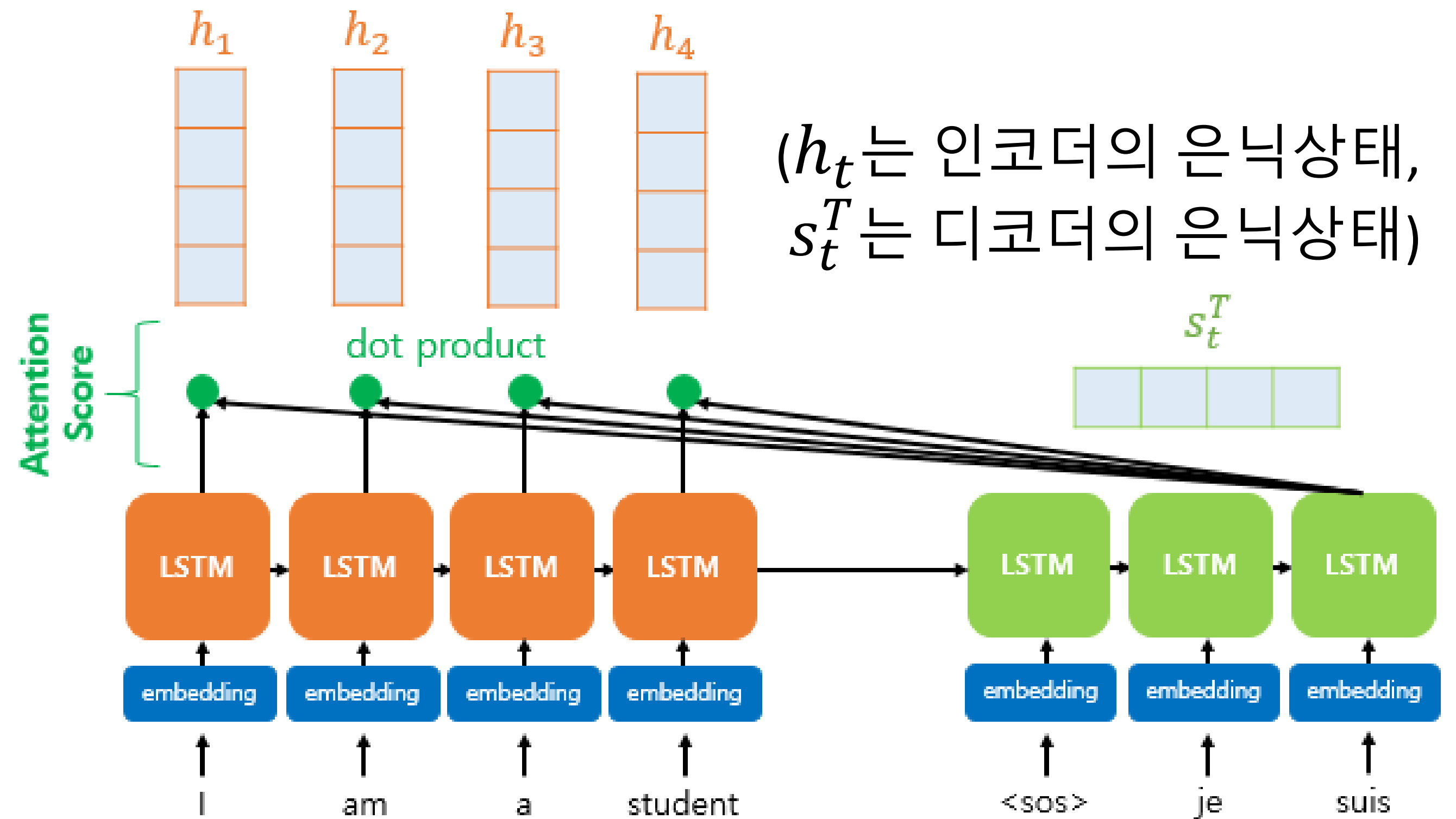


Dot-Product Attention 과정 (1)

- 1) 어텐션 스코어를 구한다.

SCO

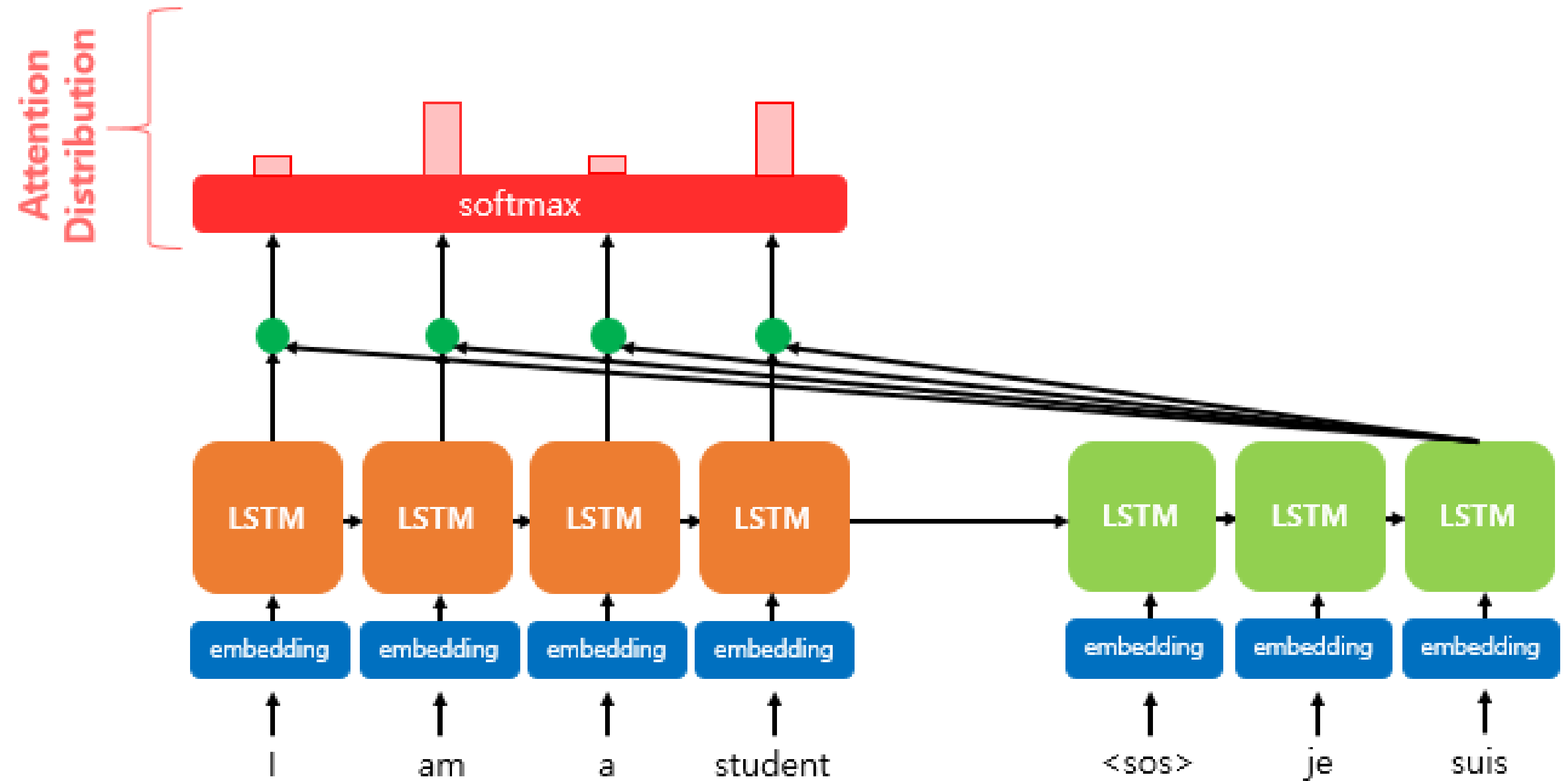
시점 t 의 모든 어텐션 스코어
모음값 e^t 는 $[s_t h_1, \dots, s_t^T h_N]$



Dot-Product Attention 과정 (2)

- 2) softmax 함수를 통해 어텐션 분포를 구한다.

시점 t의 어텐션 분포

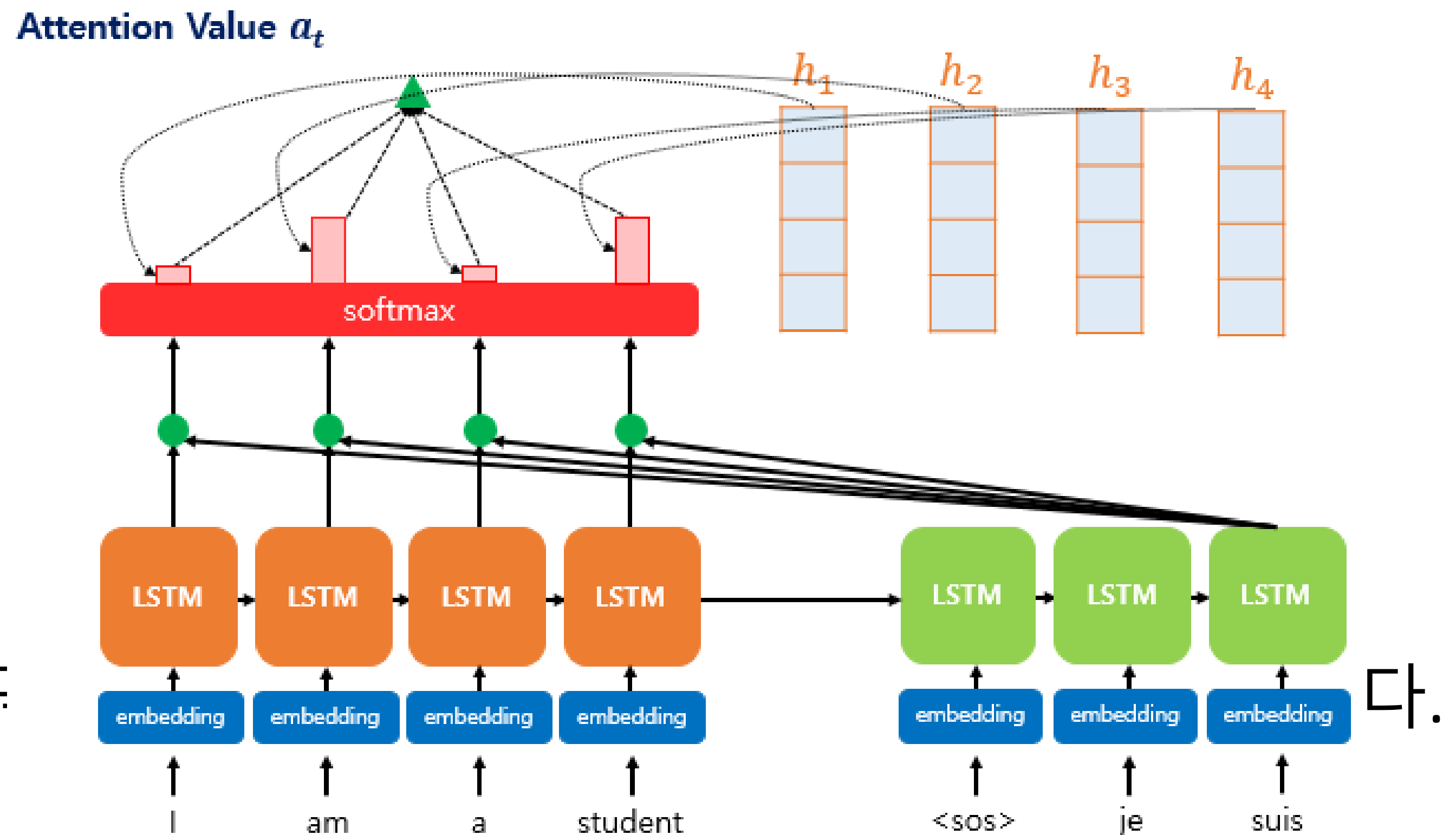


Dot-Product Attention 과정 (3)

- 3) 어텐션 가중치와
은닉 상태를 가중합하여
어텐션 값(value)을 구한다.

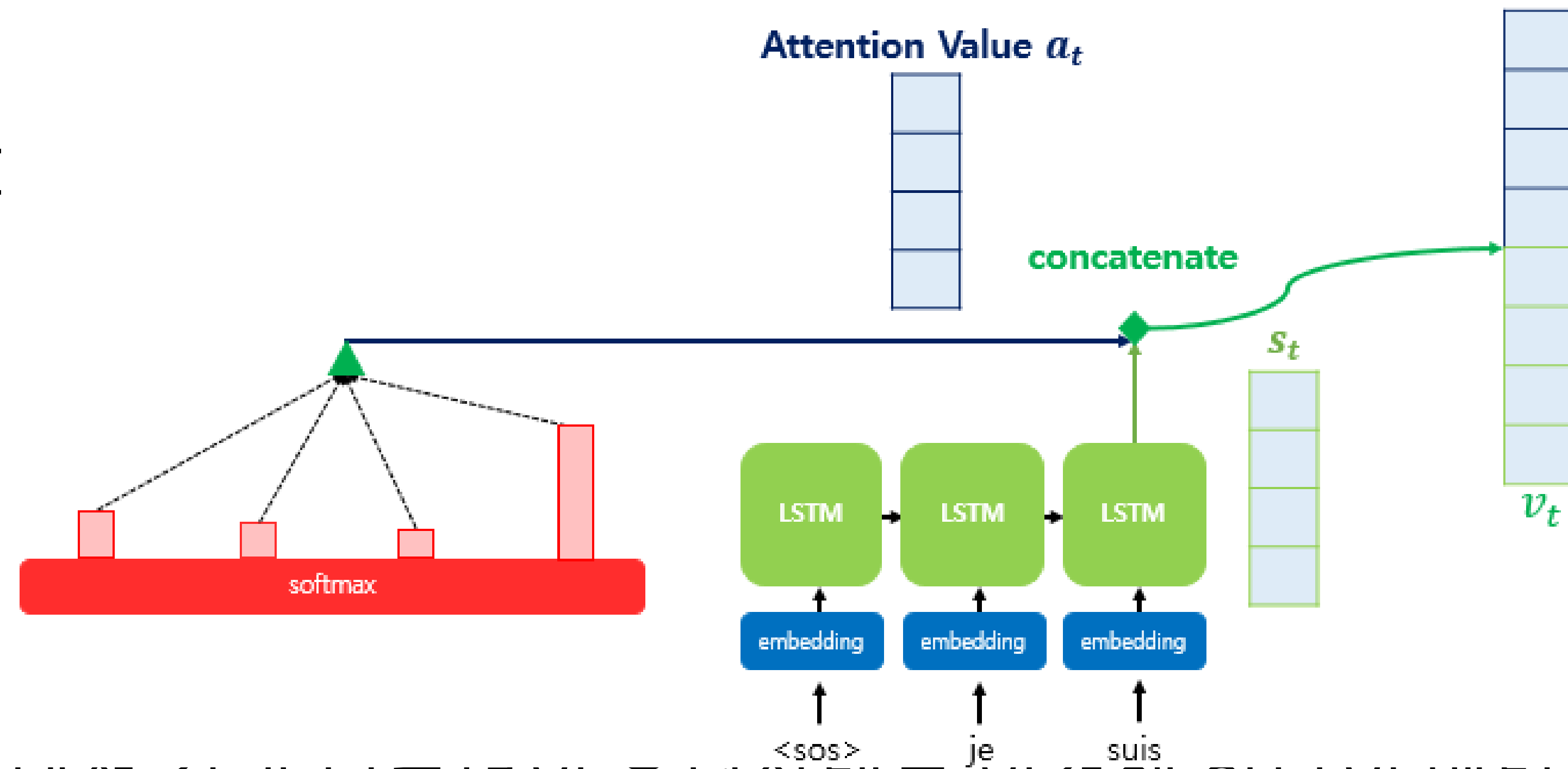
시점 t 의 어텐션 값

- 어텐션 값 a_t 는 인코더의 문



Dot-Product Attention 과정 (4)

- 4) 어텐션 값과 디코더의 t 시점의 은닉 상태를 연결한다



- 어텐션 메커니즘은 어텐션 값과 디코더의 은닉 상태를 연결해 하나의 벡터 v_t 를 만든다.

Dot-Product Attention 과정 (5)

- 5) \tilde{s}_t 를 구하고
출력층의 입력으로 사용

$$\tanh \left(\begin{array}{|c|c|c|c|c|c|c|c|} \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline \end{array} \times \begin{array}{|c|} \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \end{array} \right) = \begin{array}{|c|} \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \\ \hline \end{array}$$

W_c

v_t

\tilde{s}_t

- 논문에서는 v_t 를 바로 출력으로 사용
학습 가능한 가중치 행렬 W_c 과 편향 b_c 를 사용해
 $\tilde{s}_t = \tanh(W_c[a_t; s_t] + b_c)$ 를 구하고
예측 벡터 $\hat{y}_t = \text{Softmax}(W_y \tilde{s}_t + b_y)$ 를 얻는다.

다른 종류의 어텐션

- seq2seq + 어텐션 모델에 쓰일 수 있는 다양한 어텐션 종류가 있는데, 어텐션 스코어 함수를 구하는 것에서 차이가 있다.

- *dot*의 스코어함수는 $socre(s_t, h_i) = s_t^T h_i$,

*scaleddot*의 스코어 함수는 $socre(s_t, h_i) = \frac{s_t^T h_i}{\sqrt{n}}$,

*general*의 스코어 함수는 $socre(s_t, h_i) = s_t^T W_a h_i$,

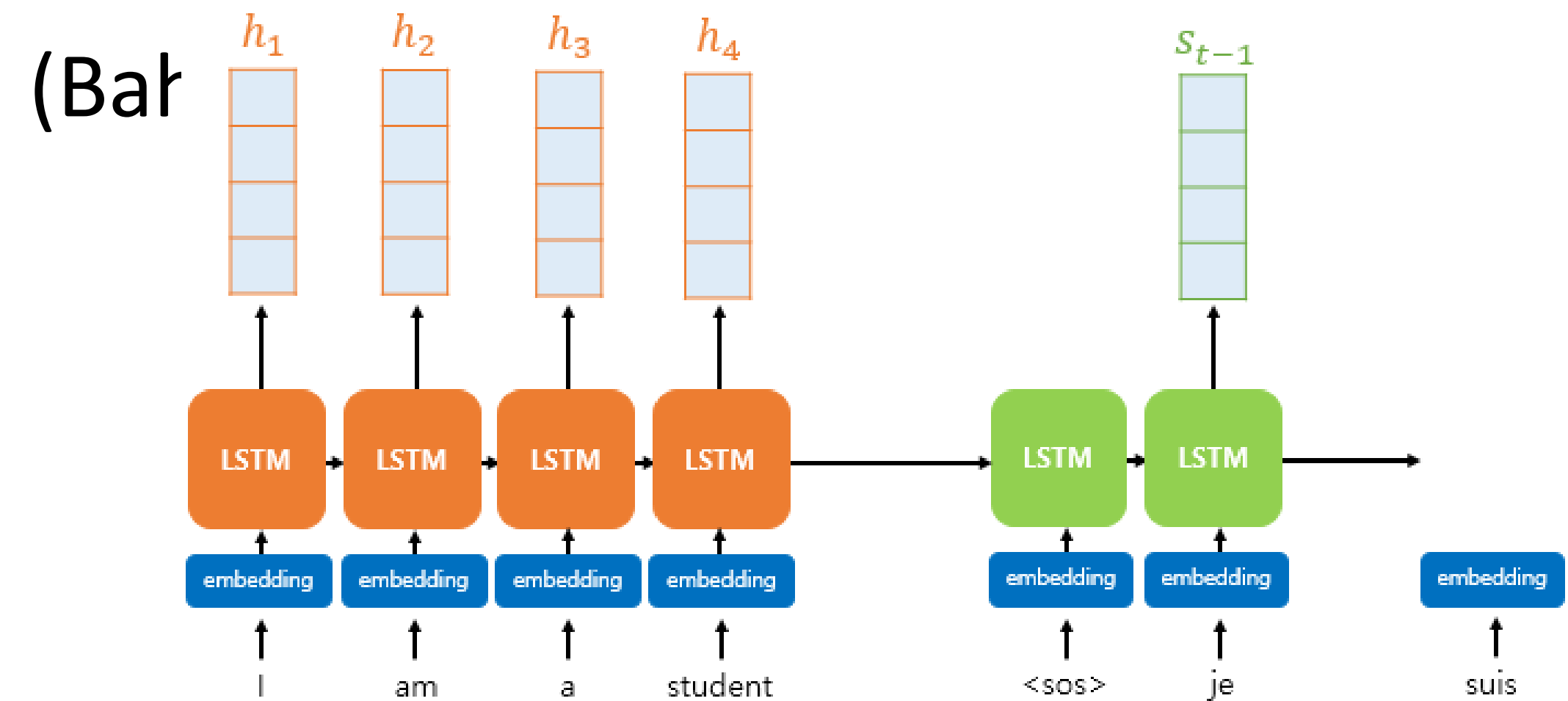
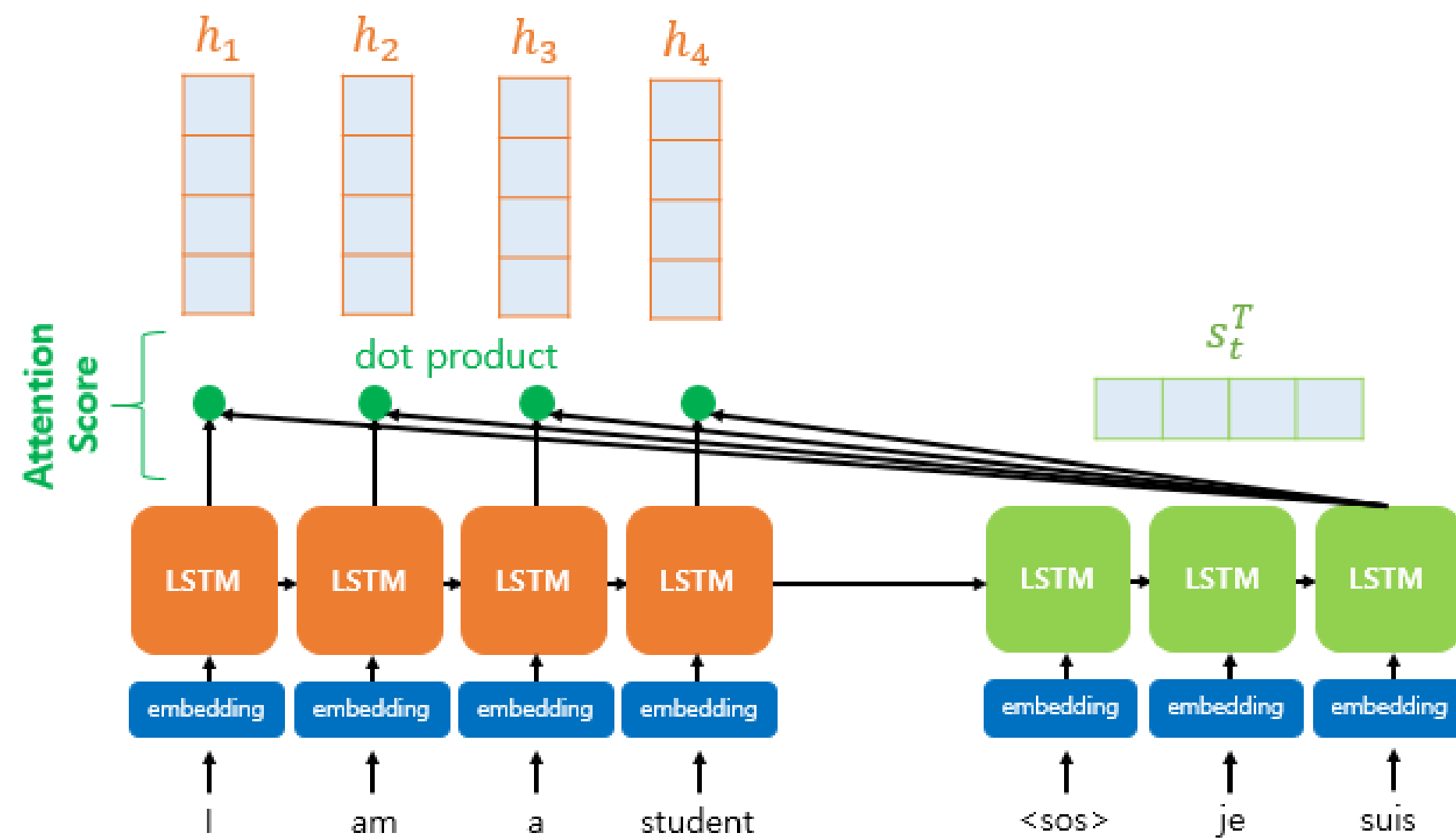
*concat*의 스코어함수는 $socre(s_t, h_i) = w_a^T \tanh(W_b [s_t; h_i])$,

*location - base*는 α_t 산출 시에 s_t 만 이용하는 기법으로, $\alpha_t = \text{Softmax}(W_a s_t)$

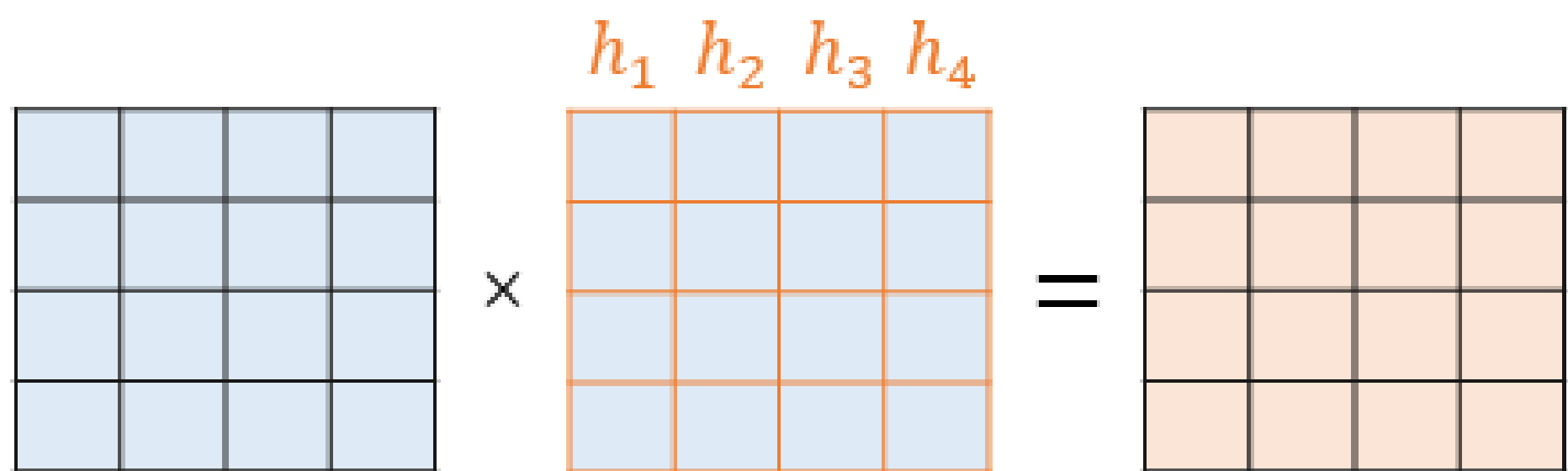
S_t 는 Query, h_i 는 Keys, W_a 와 W_b 는 학습 가능한 가중치 행렬

15-2 Bahdanau 어텐션 함수

Dot-Product 어텐션과 달리 Query가 $t-1$ 시점 은닉상태로 설계

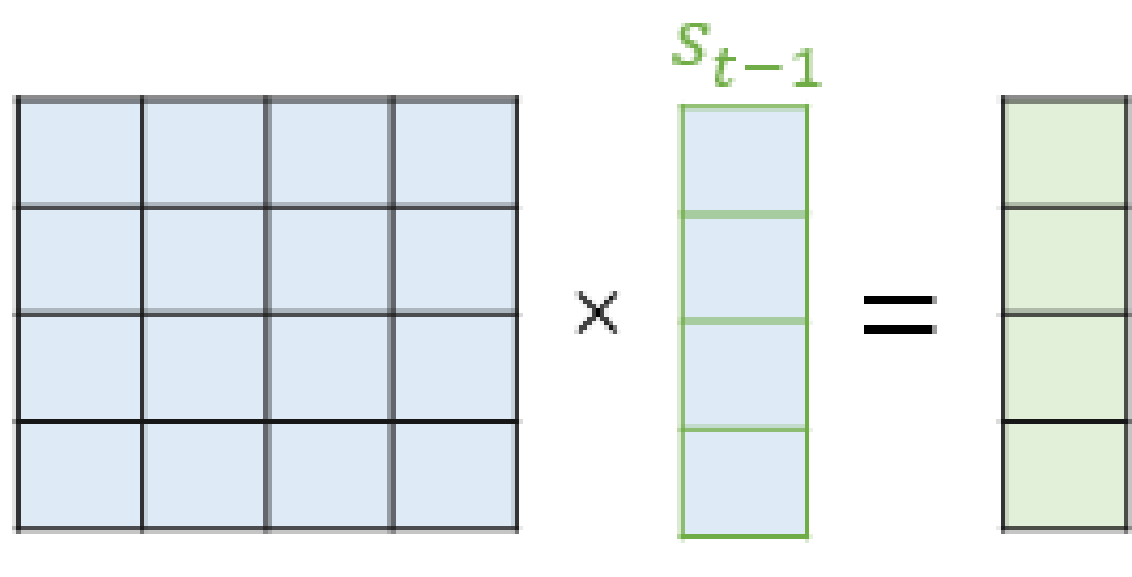


Bahdanau Attention 과정 (1)

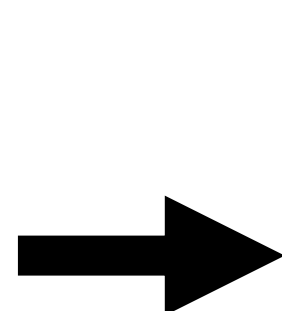


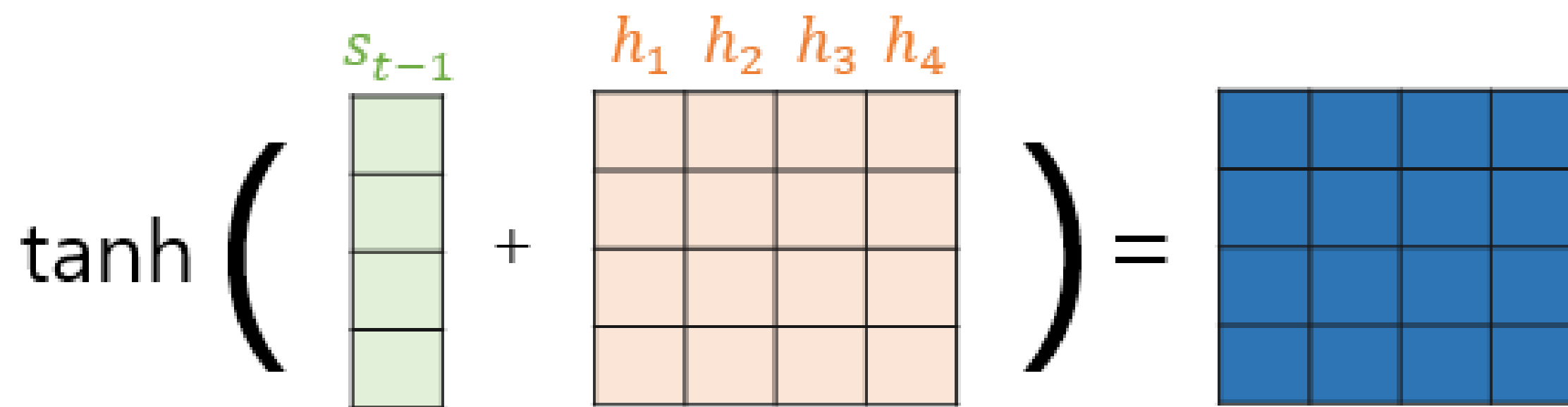
$$\begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \times \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ & & & \\ & & & \\ & & & \end{bmatrix} = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}$$

$(W_b s_{t-1} + W_c h_i)$ (단, W_a, W_b, W_c 는 학습 가능한 가

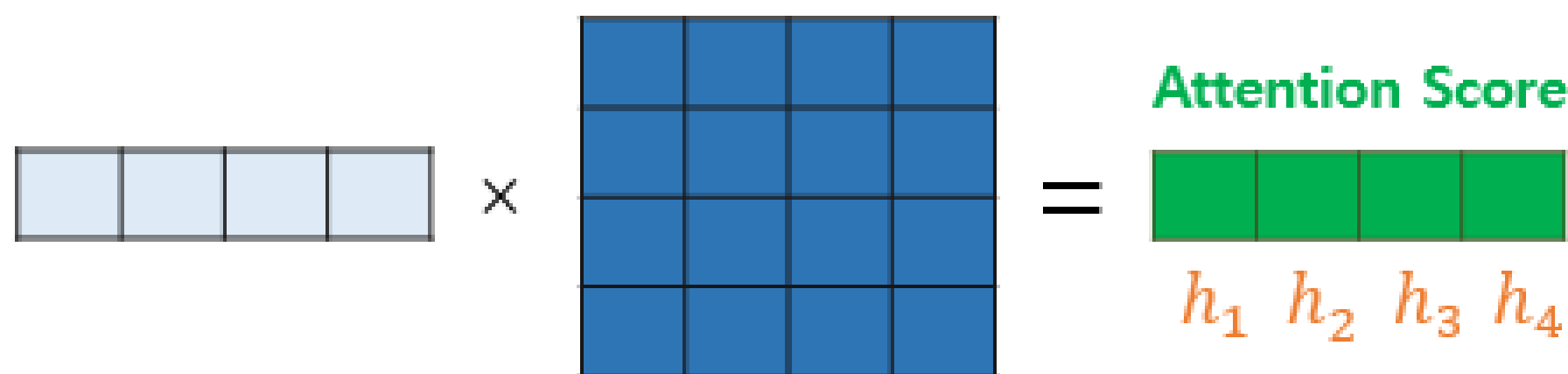


$$\begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \times \begin{bmatrix} s_{t-1} \\ \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$





$$\tanh \left(\begin{bmatrix} s_{t-1} \\ \\ \\ \end{bmatrix} + \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ & & & \\ & & & \\ & & & \end{bmatrix} \right) = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}$$



$$\begin{bmatrix} & & & \end{bmatrix} \times \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} = \begin{bmatrix} \text{Attention Score} \\ \\ \\ \end{bmatrix}$$

$h_1 \quad h_2 \quad h_3 \quad h_4$

Bahdanau Attention 과정 (2)

- 2) softmax 함수를 통해 어텐션 분포를 구한다.

$$\text{softmax} \left(\begin{array}{c} \text{Attention Score} \\ \text{[Green Box]} \\ h_1 \ h_2 \ h_3 \ h_4 \end{array} \right) = \begin{array}{c} \text{Attention} \\ \text{Distribution} \\ \text{[Red Box]} \end{array}$$

어텐션 분포 각각의 값은 어텐션 가중치라고 한다.

Bahdanau Attention 과정 (3)

을 구한다.

h_1 h_2 h_3 h_4

\times

$=$

Context Vector

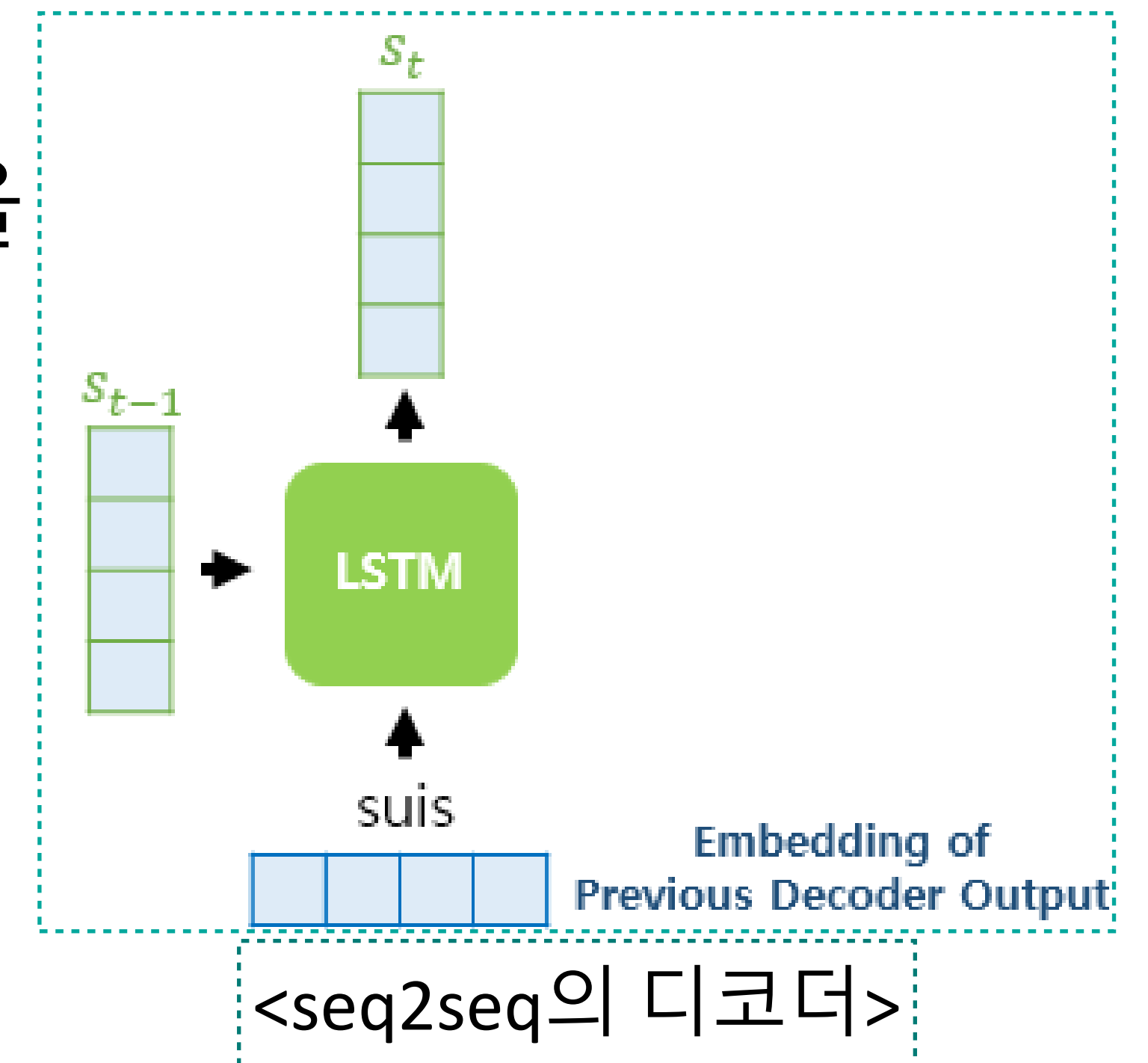
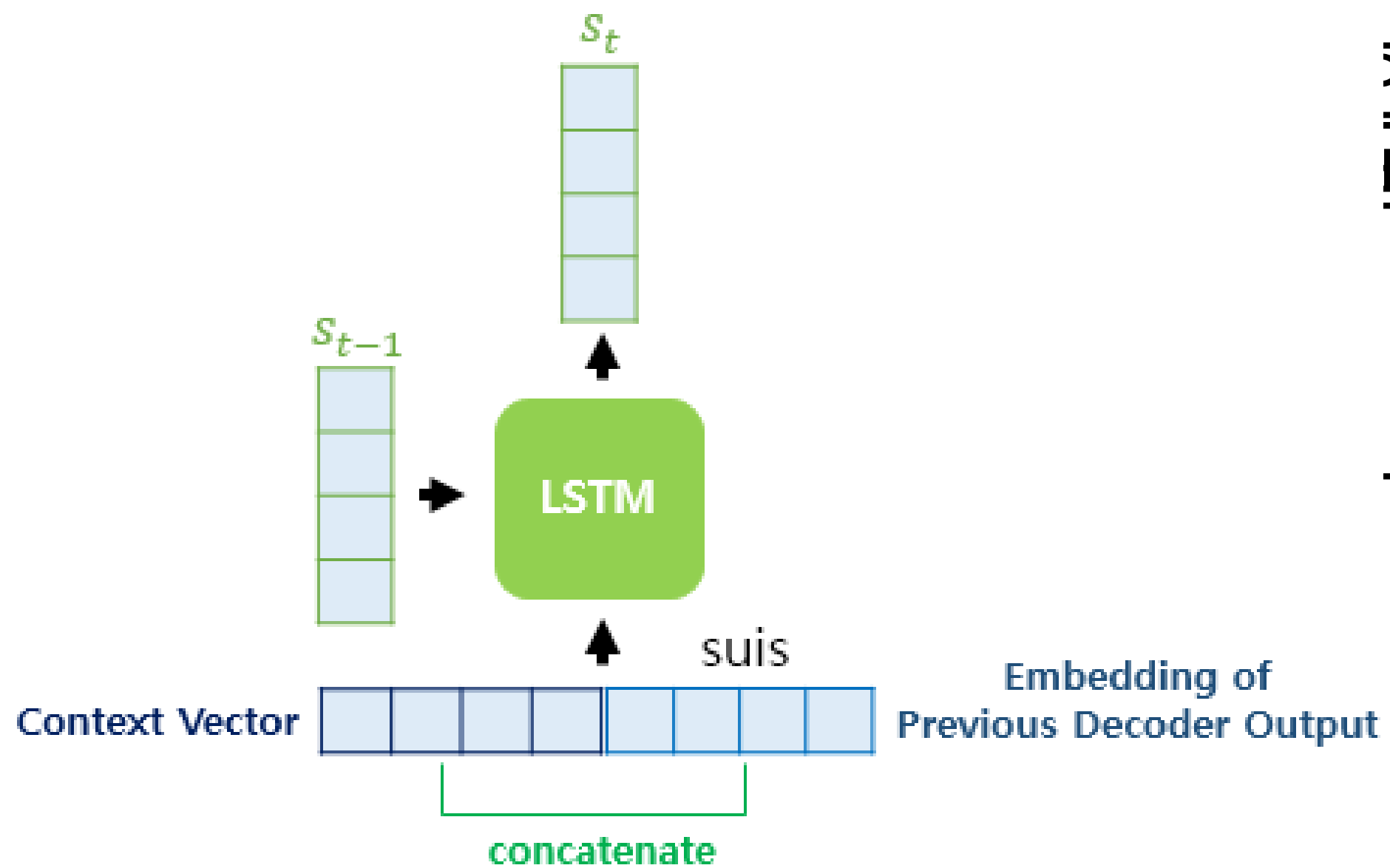
--	--	--	--

Bahdanau Attention 과정 (4)

한다.

벡터를 연결하여 입력으로 사용한다

현재 시점의 예측값을



끝

감사합니다.