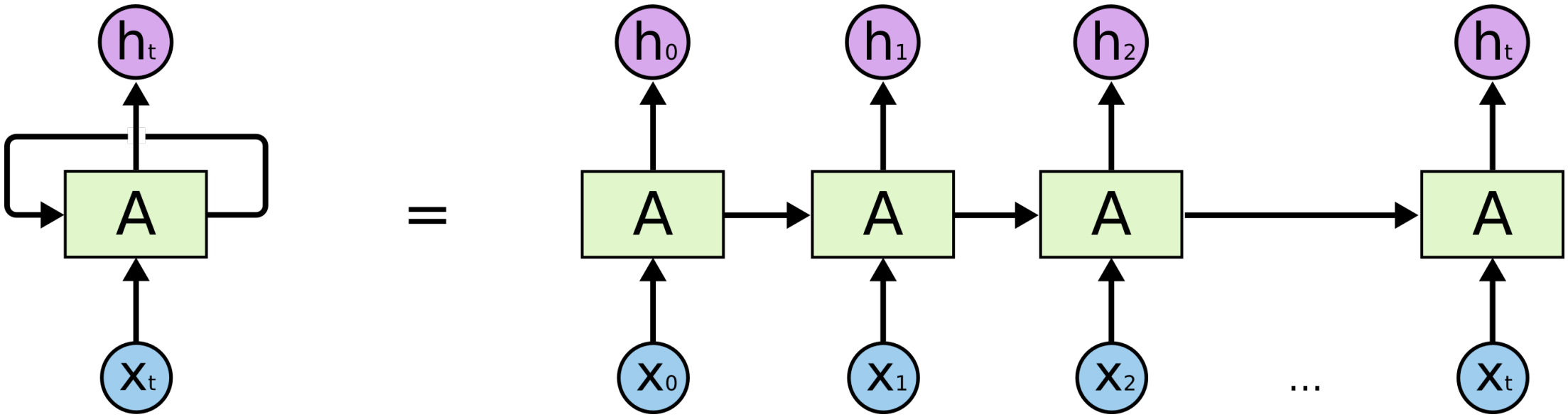
Attention is All You Need

한양대 HAI 지영채

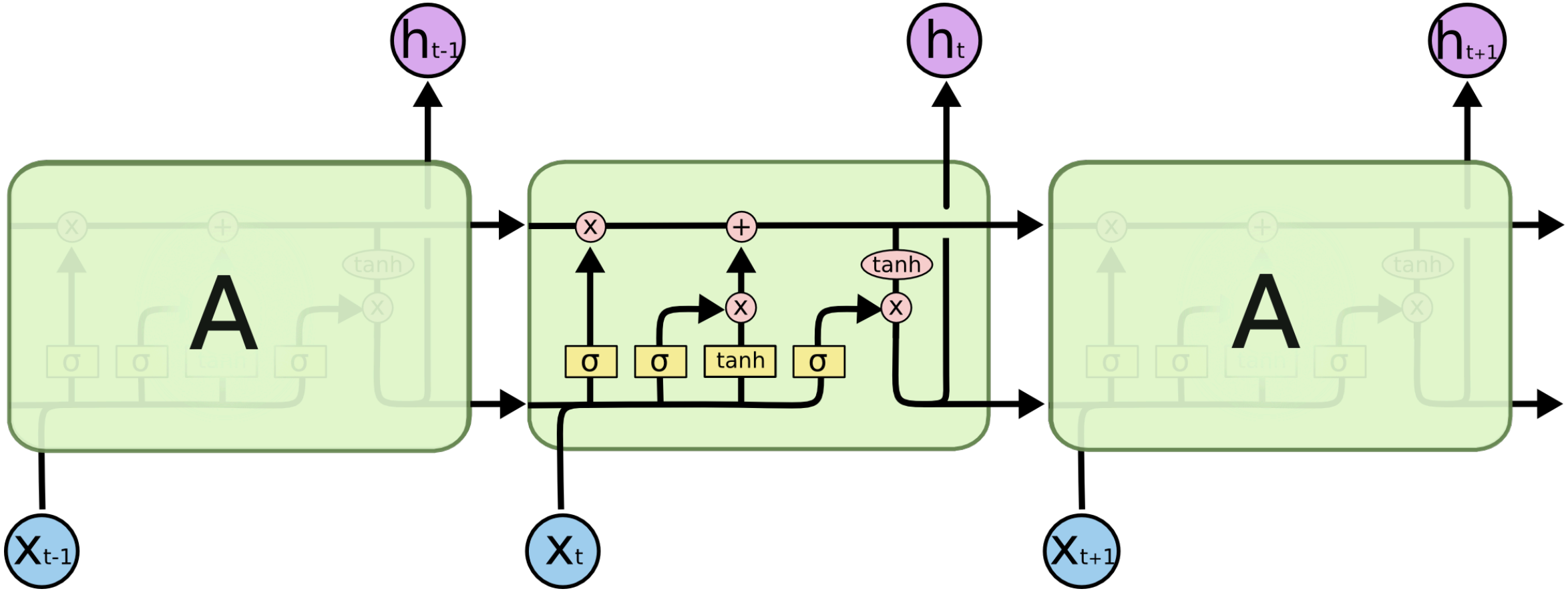
Claims

1. Superior in quality — 28.4 BLEU WMT English to German Translation
2. More Parallelizable
3. Less Train Time

RNN Review

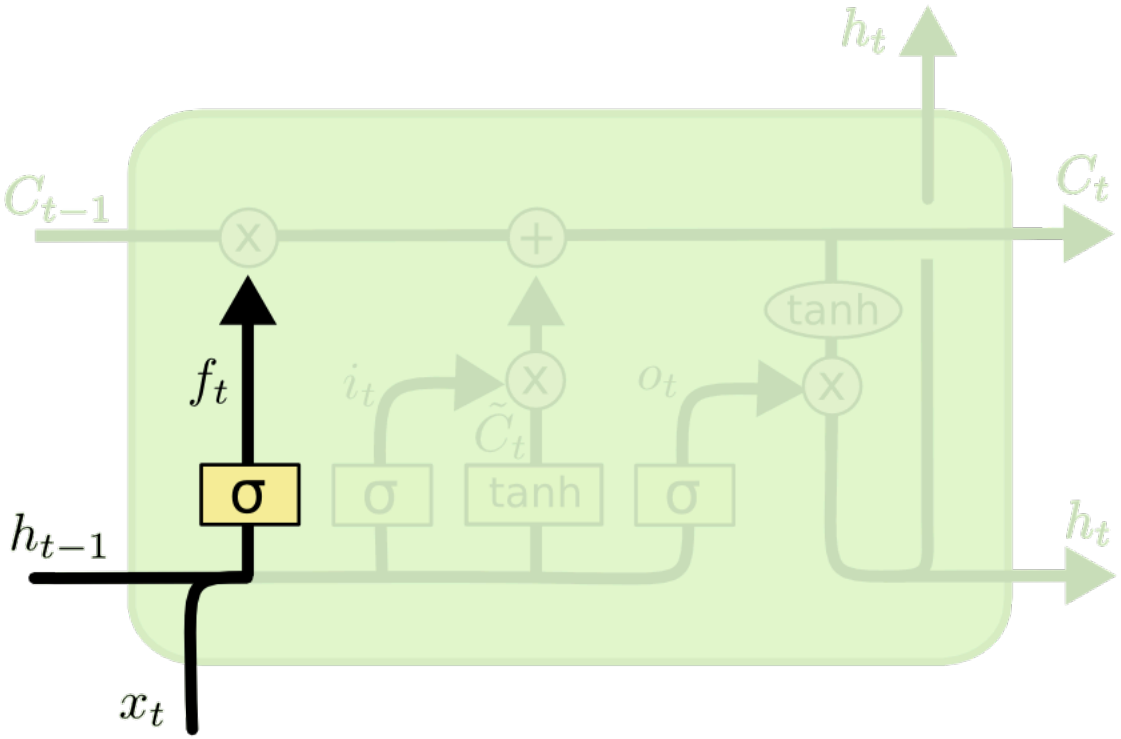


RNN Review – LSTM (Long and Short Term Memory)



RNN Review – LSTM (Long and Short Term Memory)

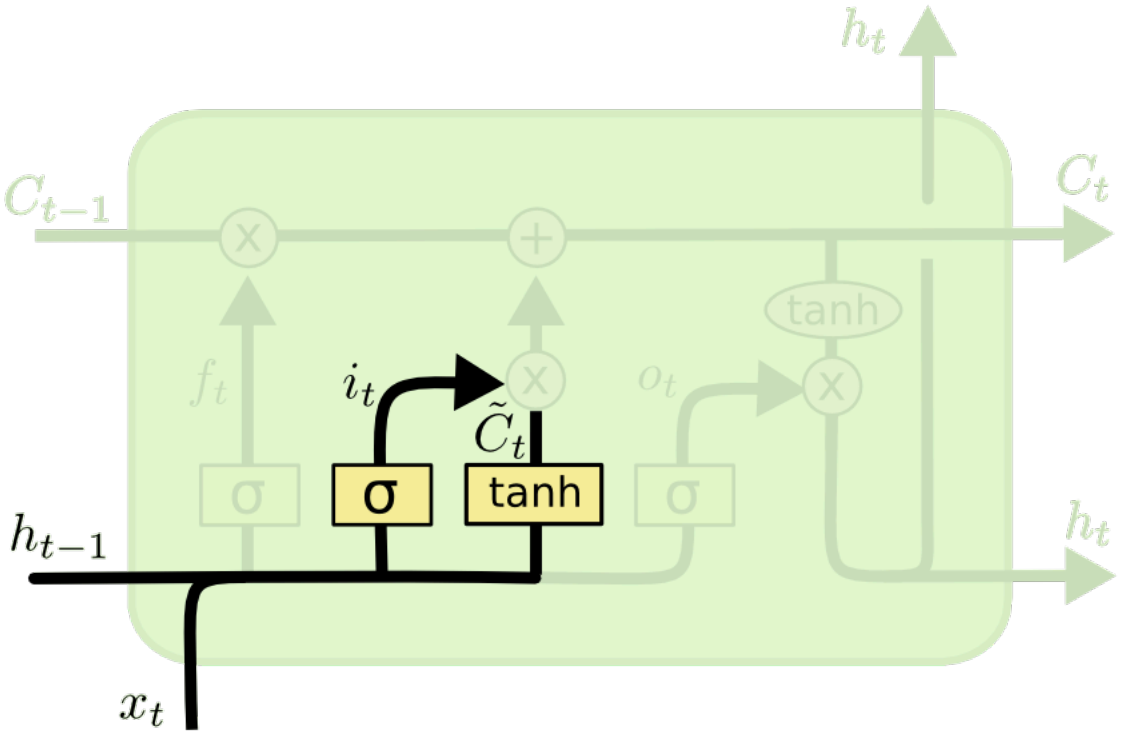
Forget Long Term Memory



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

RNN Review – LSTM (Long and Short Term Memory)

Add Long Term Memory

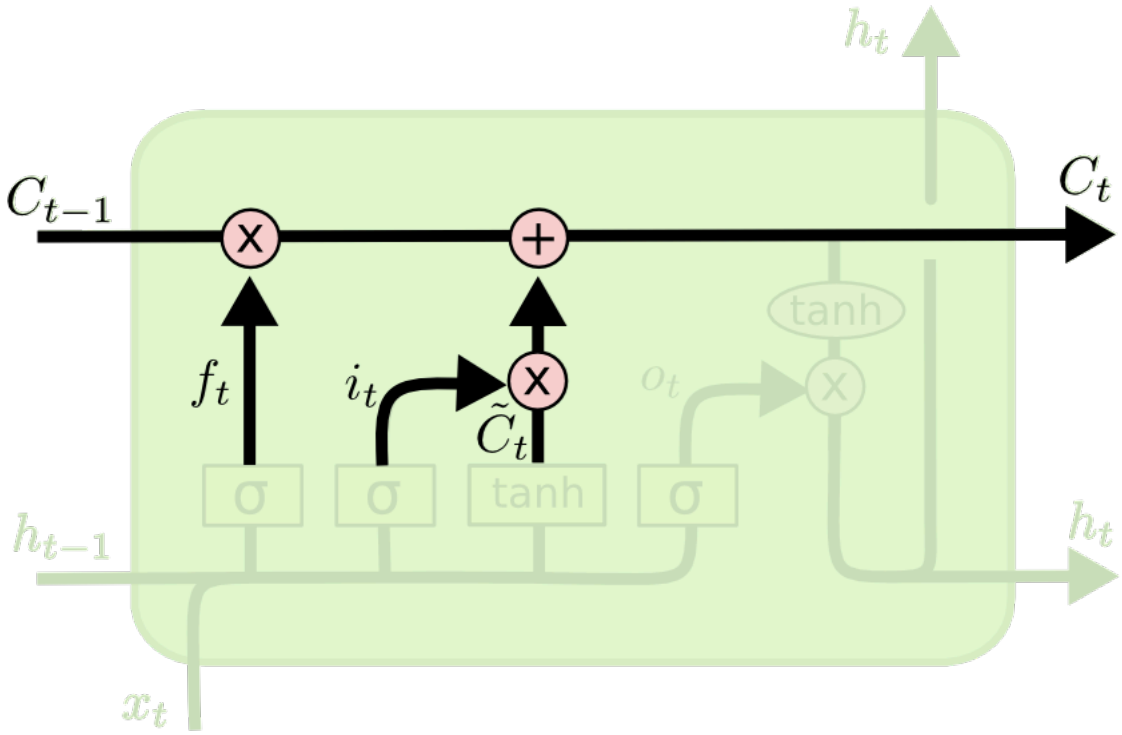


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

RNN Review – LSTM (Long and Short Term Memory)

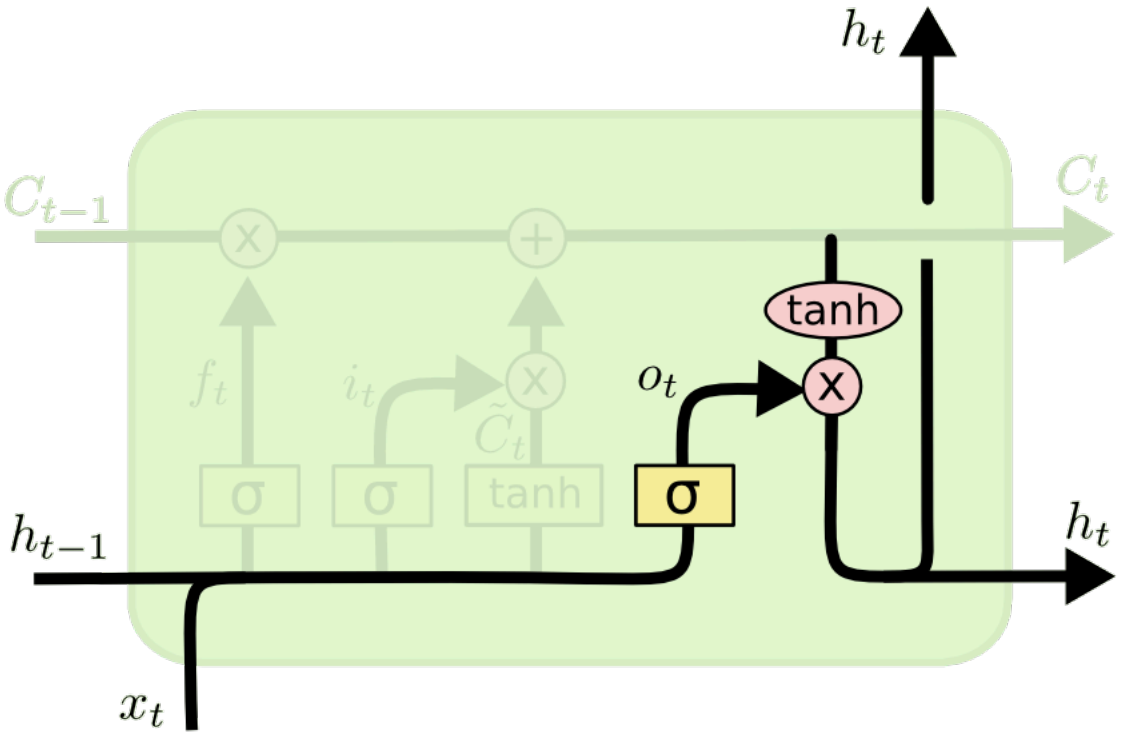
Update Long Term Memory



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

RNN Review – LSTM (Long and Short Term Memory)

Update Short Term Memory

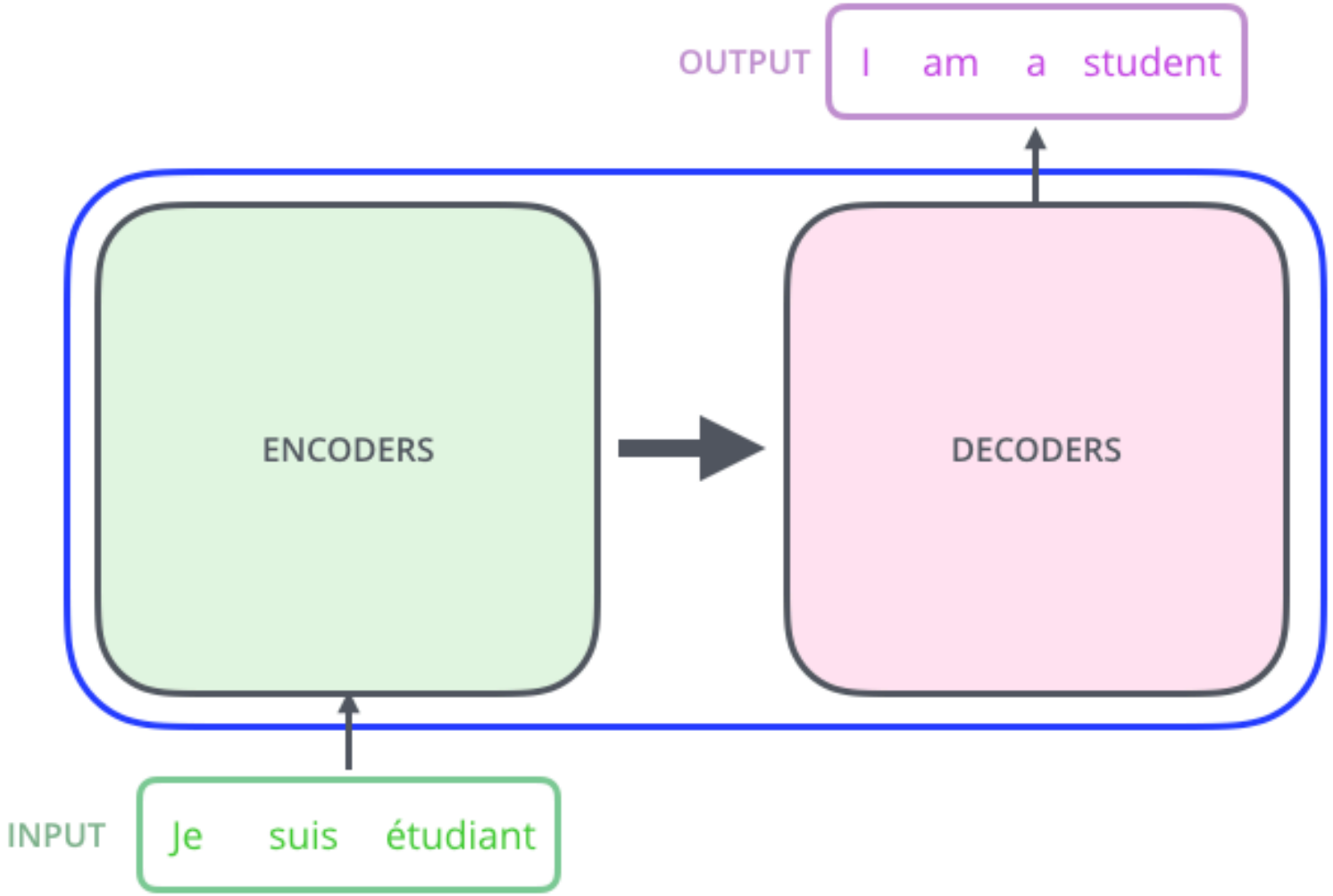


$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

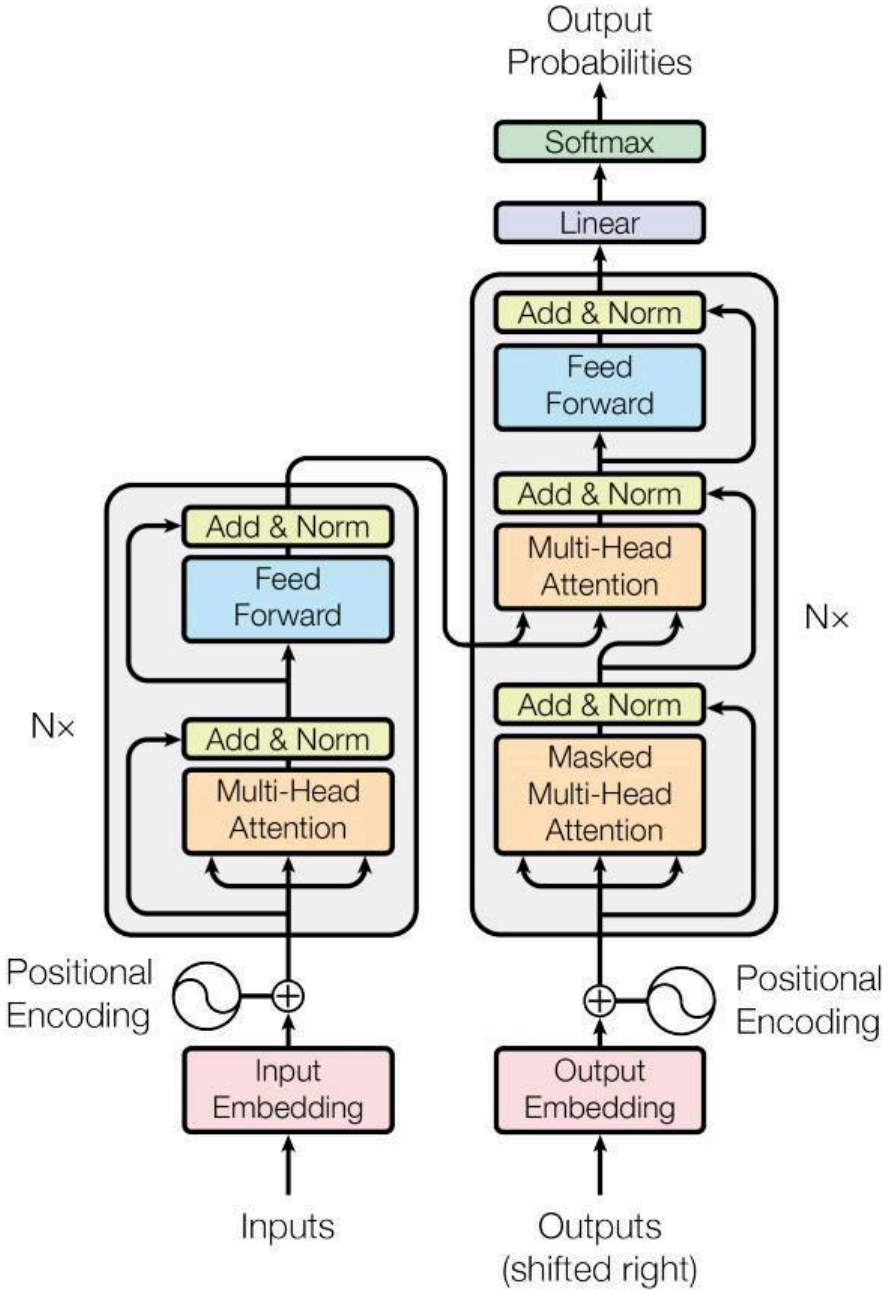
$$h_t = o_t * \tanh\left(C_t\right)$$

Transformer

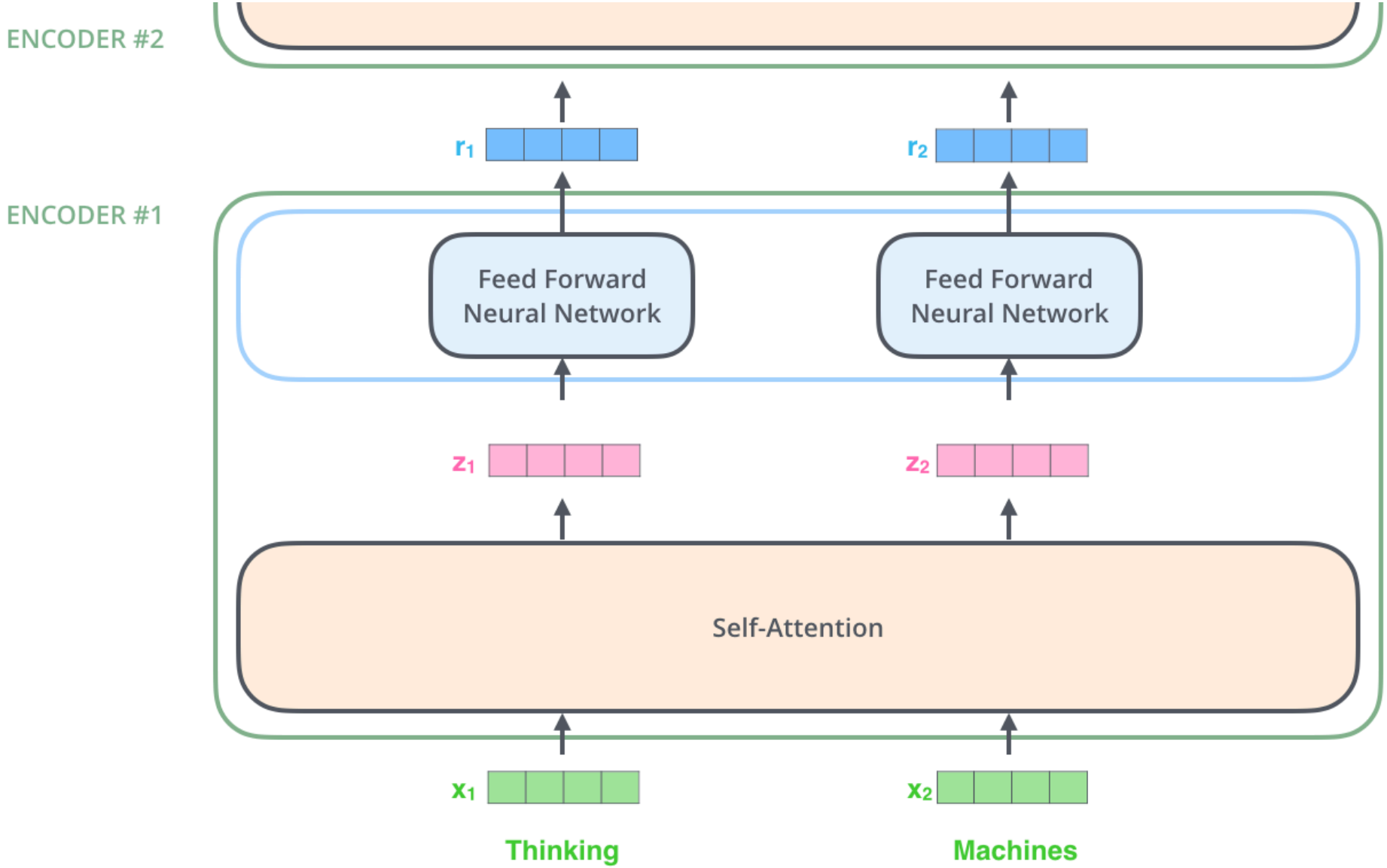
High-Level Look



High-Level Look



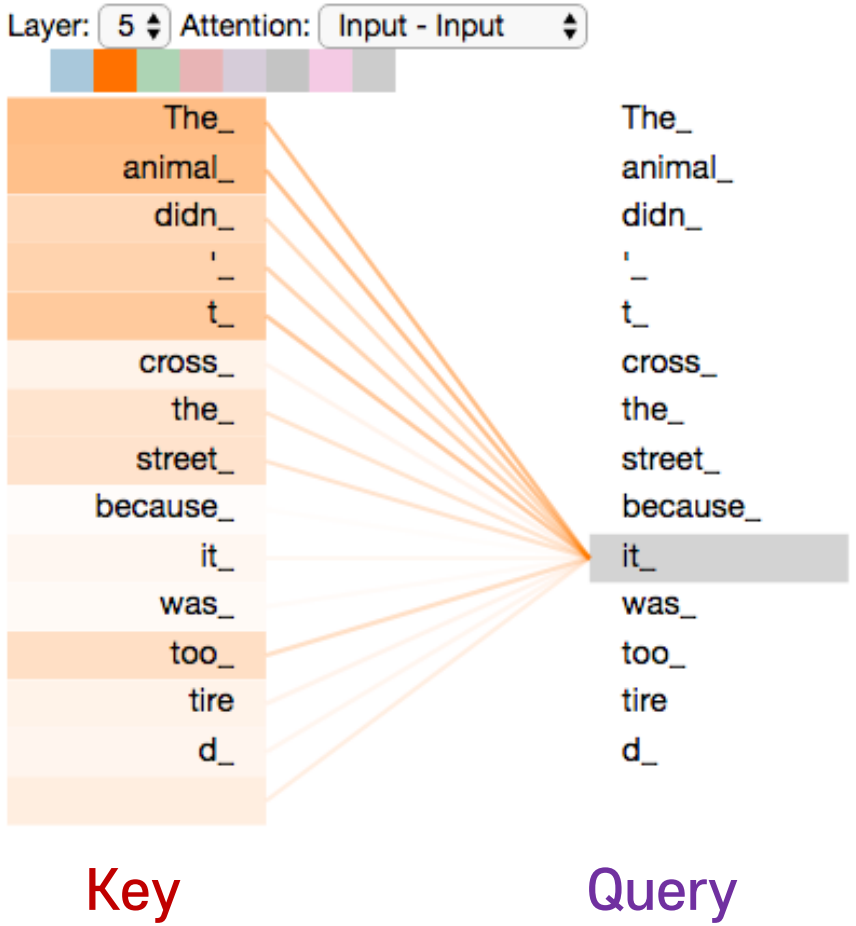
Encoder


ENCODER #2
ENCODER #1
r1
r2
Feed Forward Neural Network
Feed Forward Neural Network
z1
z2
Self-Attention
x1
x2
Thinking
Machines

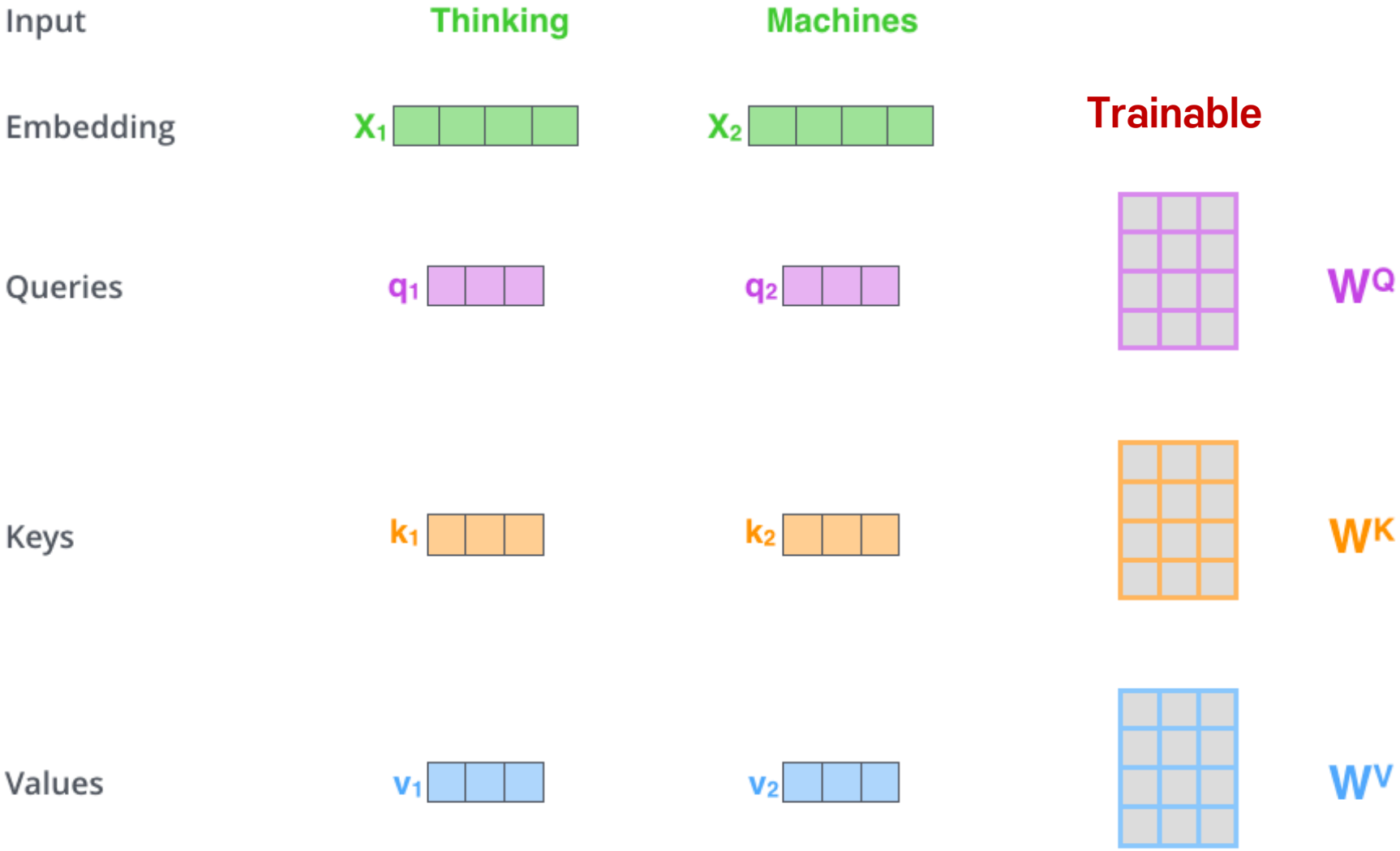
Attention

Attention

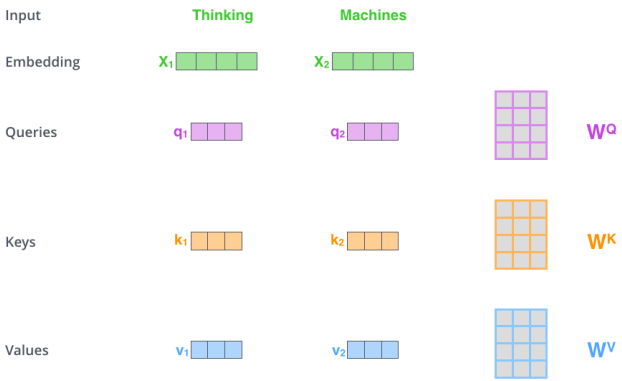
Query 가 들어왔을 때 어떤 key에 집중할 것인가?



Attention


Input
Thinking
Machines
Embedding
X_1
X_2
Trainable
Queries
q_1
q_2
W^Q
Keys
k_1
k_2
W^K
Values
v_1
v_2
W^V

Attention — Scaled Dot Product Attention



$X_{(query)}$ [query_length, query_dim]

$Y_{(context)}$ [context_length, context_dim]

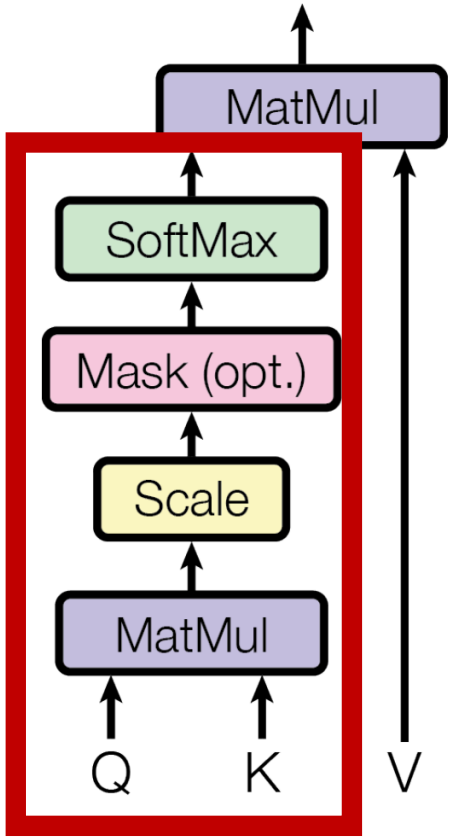
Q [query_length, hidden_dim] = $X * W_Q$ [query_dim, hidden_dim]

K [context_length, hidden_dim] = $Y * W_K$ [context_dim, hidden_dim]

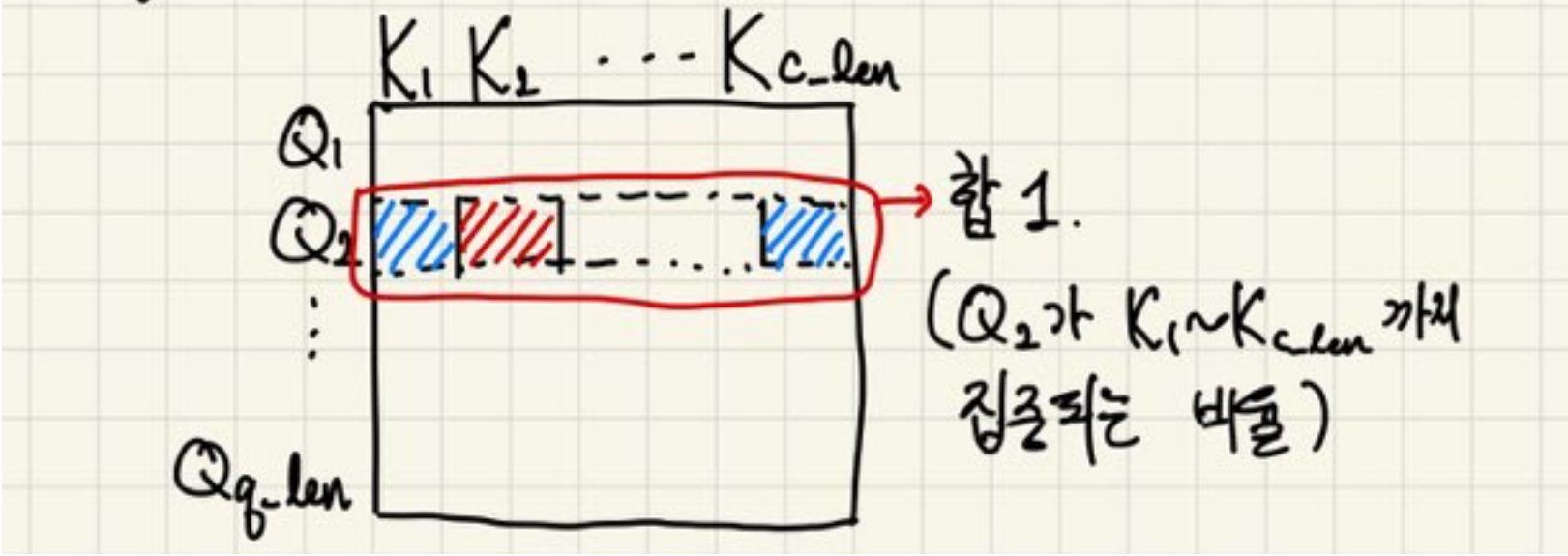
V [context_length, output_dim] = $Y * W_V$ [context_dim, output_dim]

Attention — Scaled Dot Product Attention

Scaled Dot-Product Attention

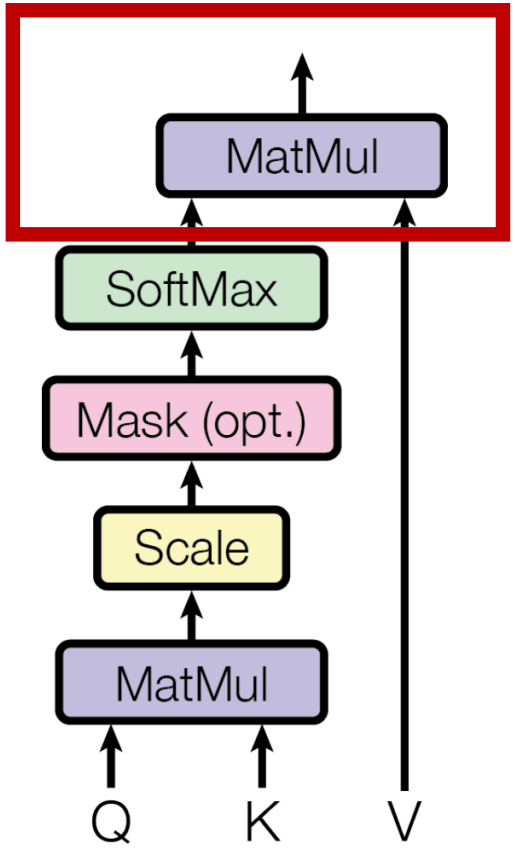


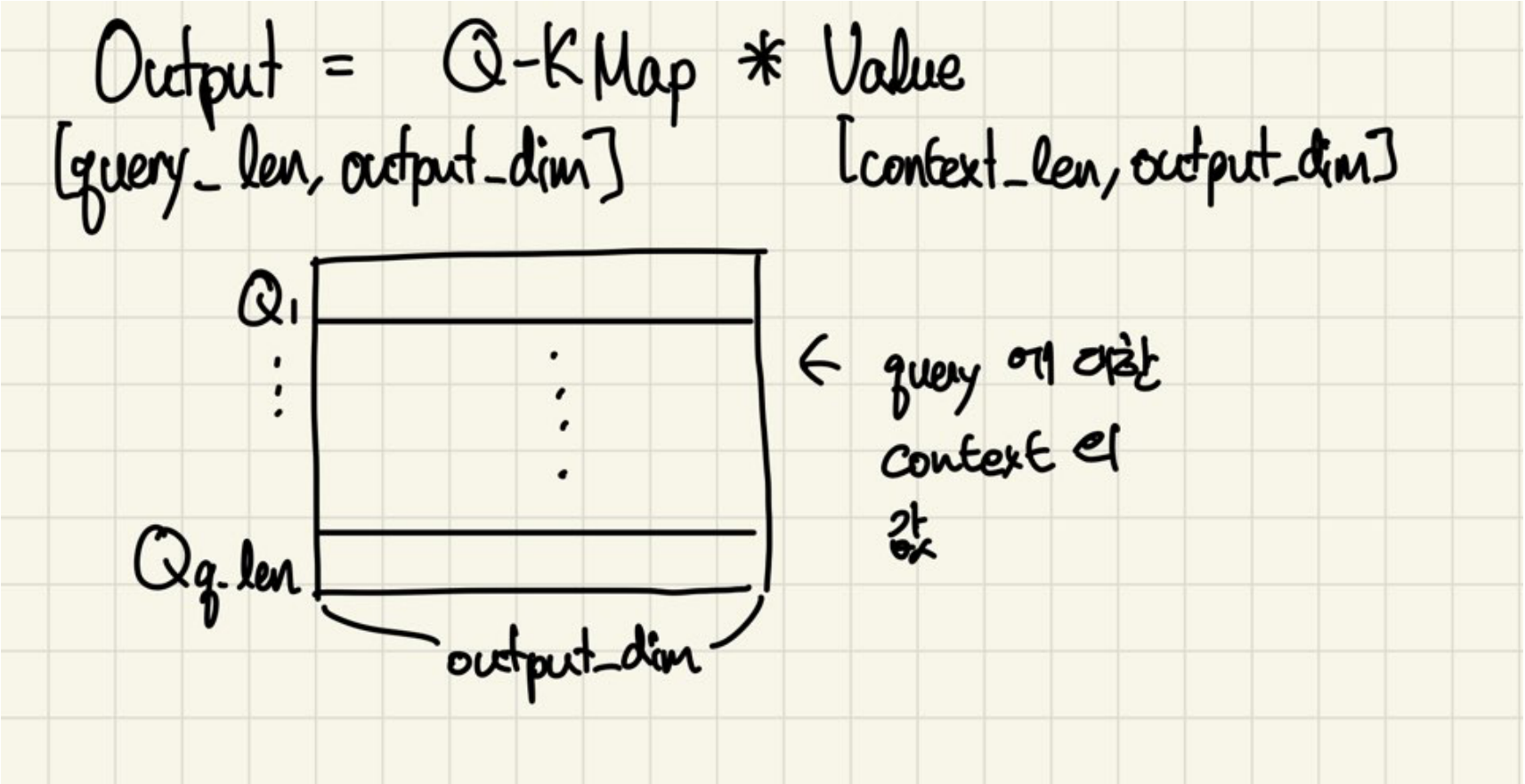
$$\underset{[q\_len,\ c\_len]}{Q\text{-}K\ Map} = \mathrm{softmax}\left(\frac{1}{\sqrt{h\_dim}}\, Q \cdot K^{T}\right)$$



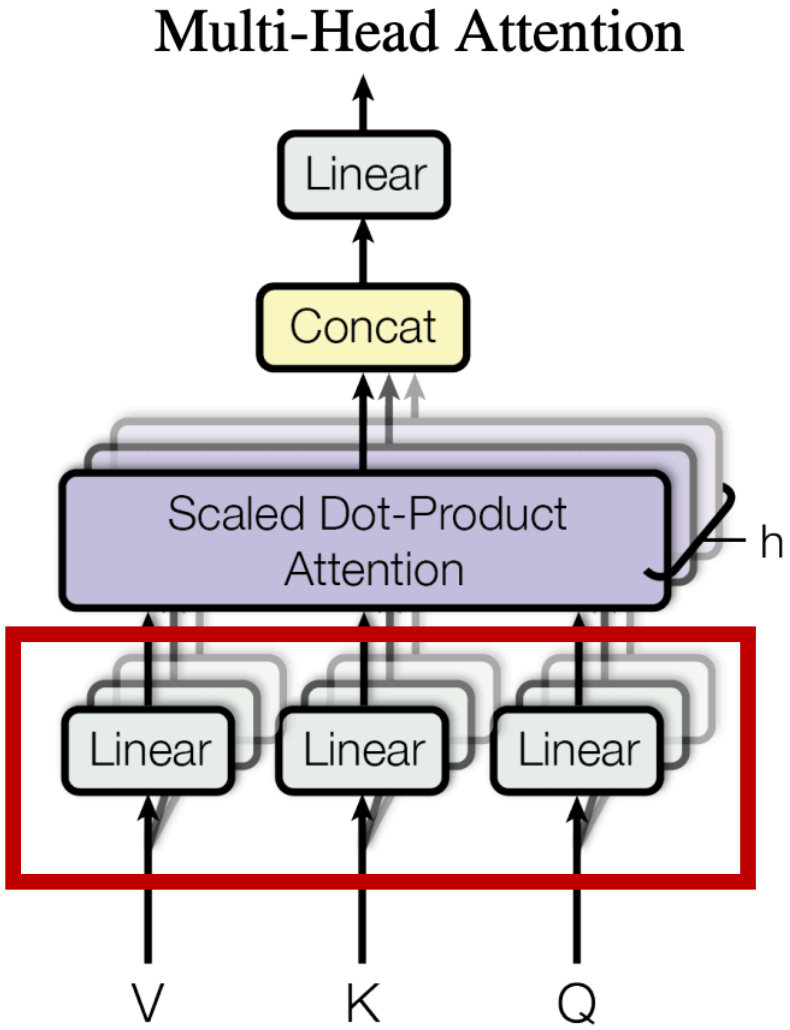
Attention — Scaled Dot Product Attention

Scaled Dot-Product Attention





Attention — Multi-Head Attention



$X_{(query)}$ [query_length, query_dim]

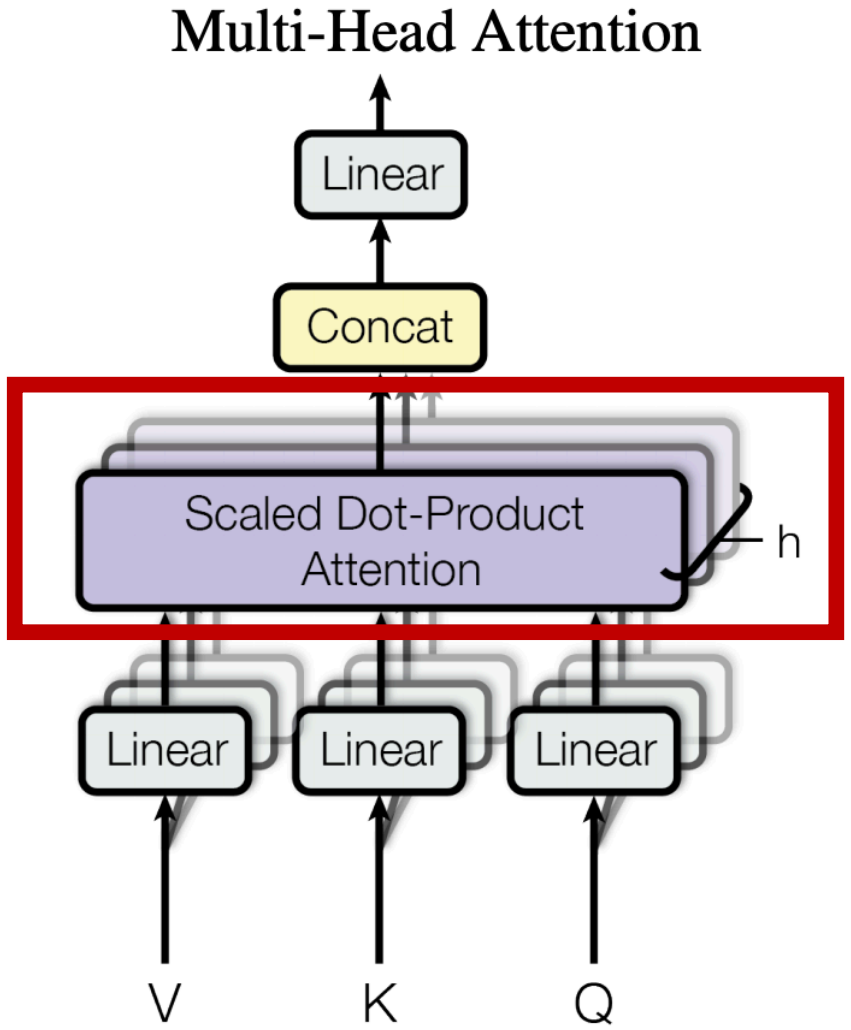
$Y_{(context)}$ [context_length, context_dim]

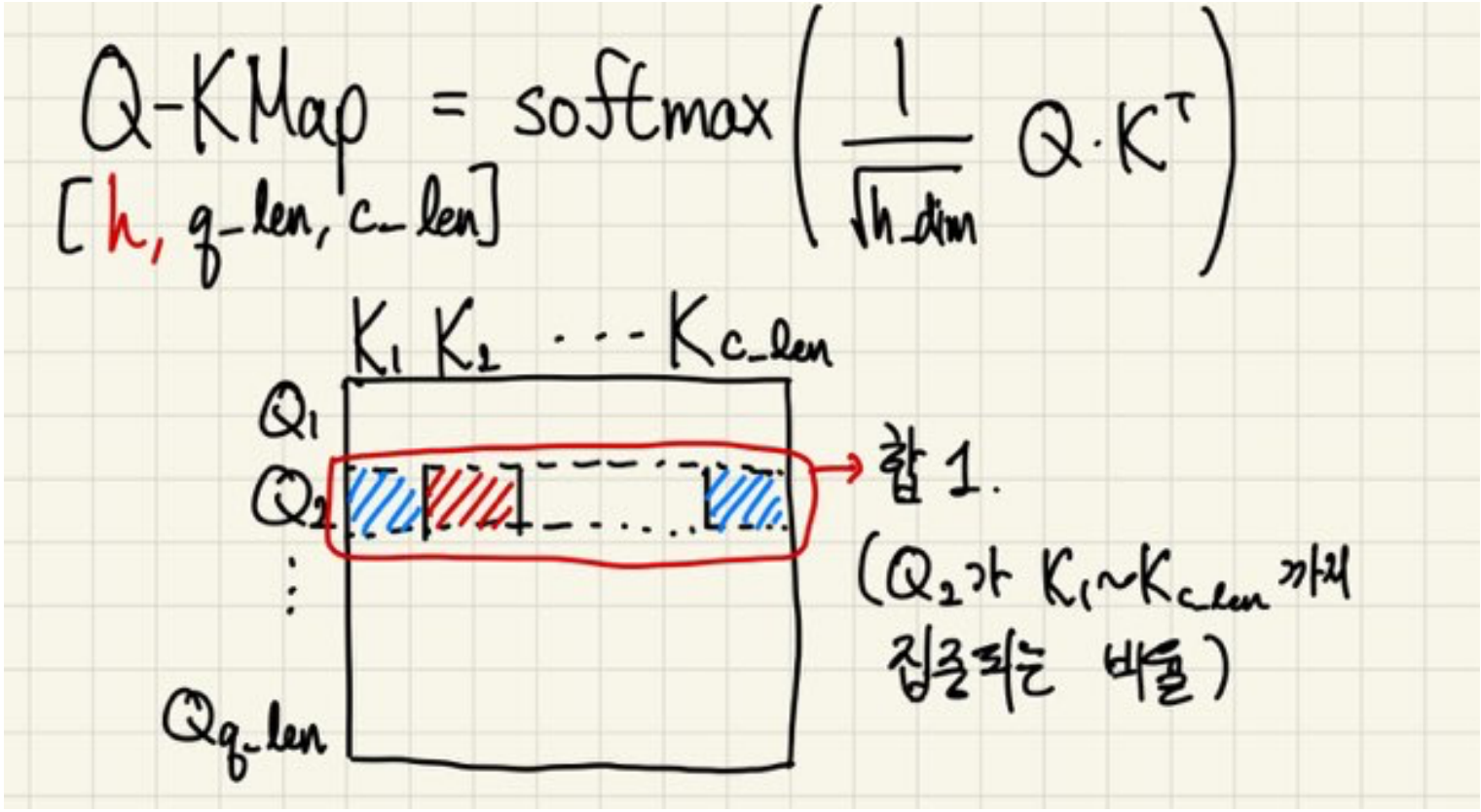
Q [query_length, hidden_dim $*h$] = $X * W_Q$ [query_dim, hidden_dim $*h$]

K [context_length, hidden_dim $*h$] = $Y * W_K$ [context_dim, hidden_dim $*h$]

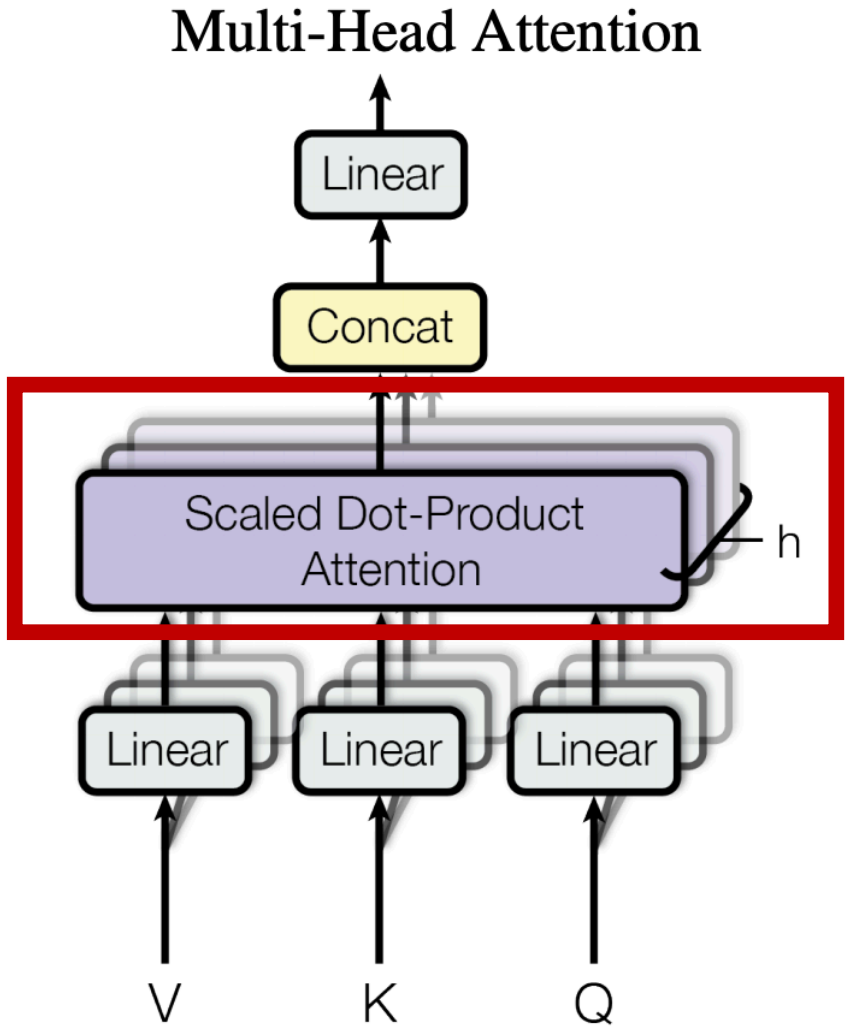
V [context_length, output_dim $*h$] = $Y * W_V$ [context_dim, output_dim $*h$]

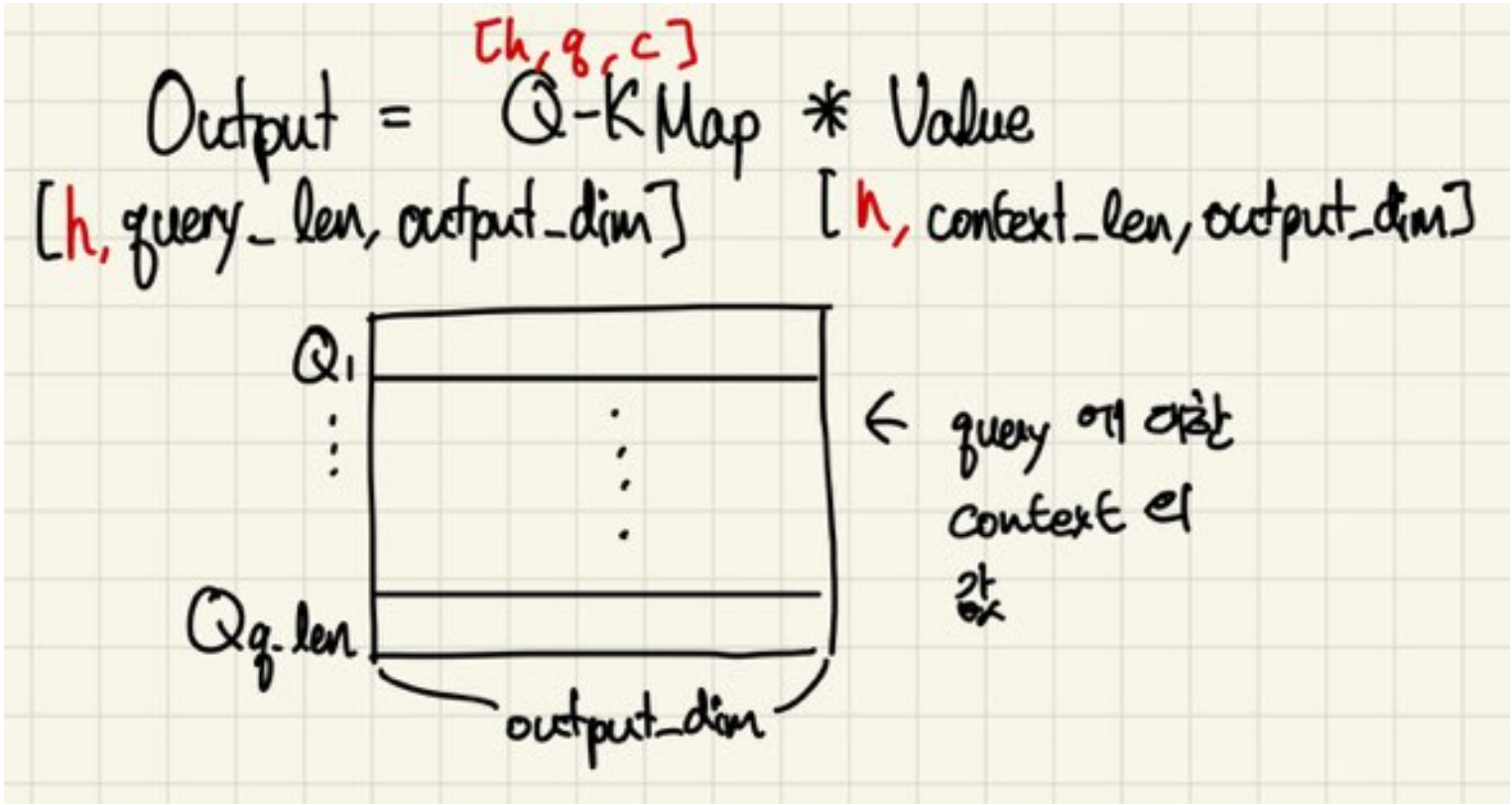
Attention — Multi-Head Attention



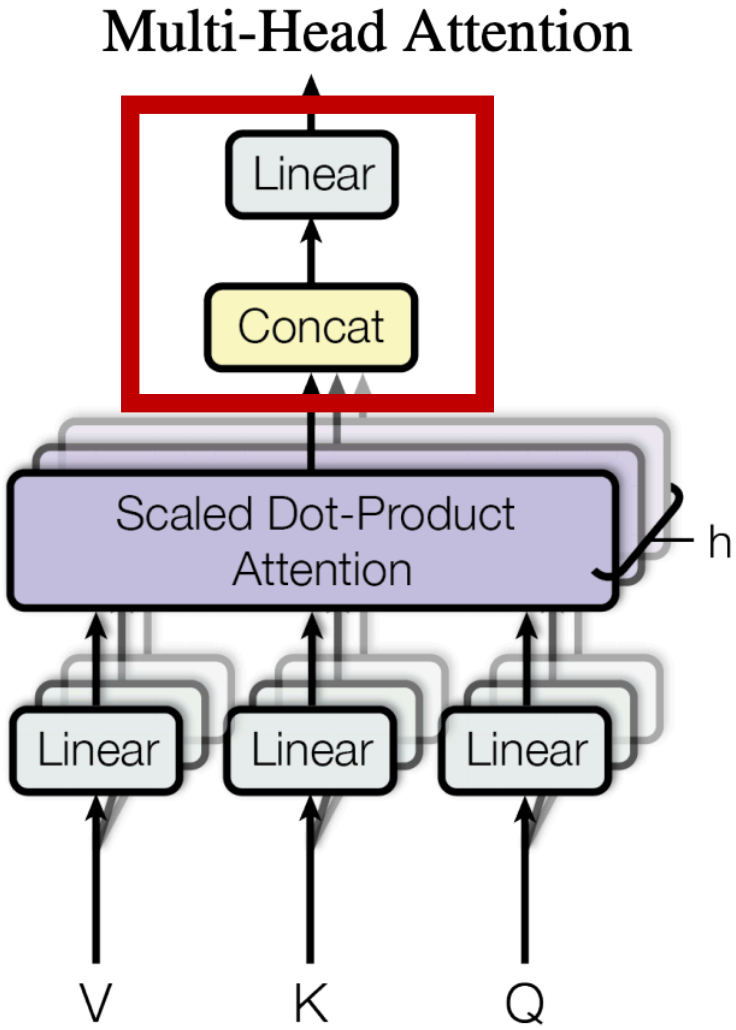


Attention — Multi-Head Attention





Attention — Multi-Head Attention



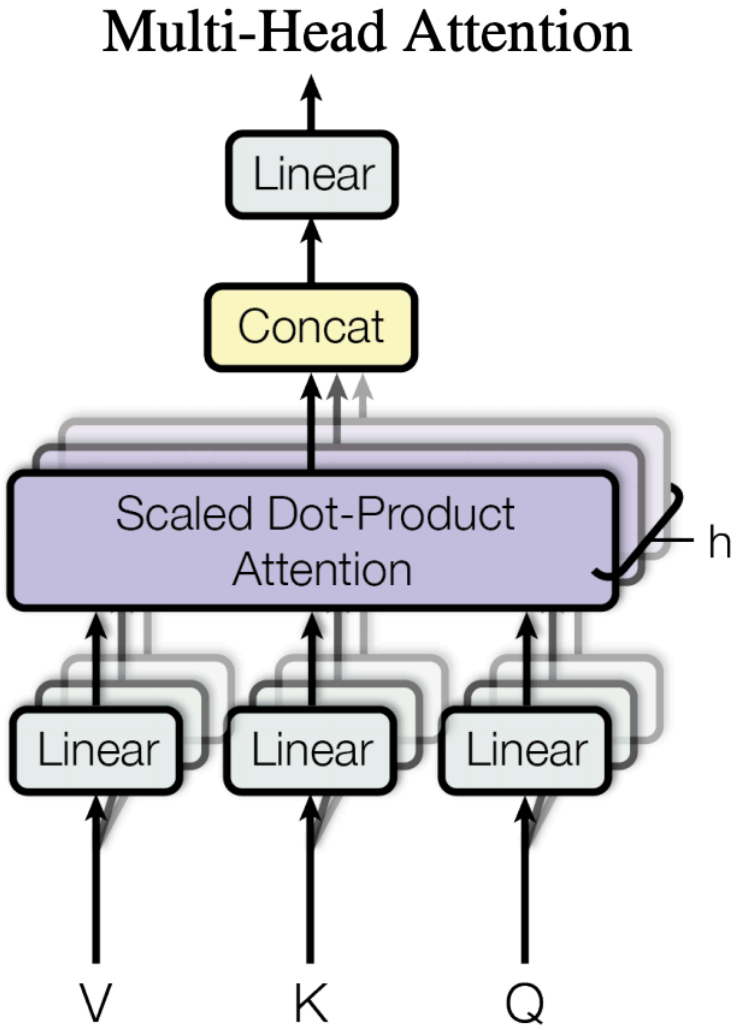
Output [query_len, output_dim * h]

↓ Linear (output_dim*h, real_output_dim)

[query_len, real_output_dim]

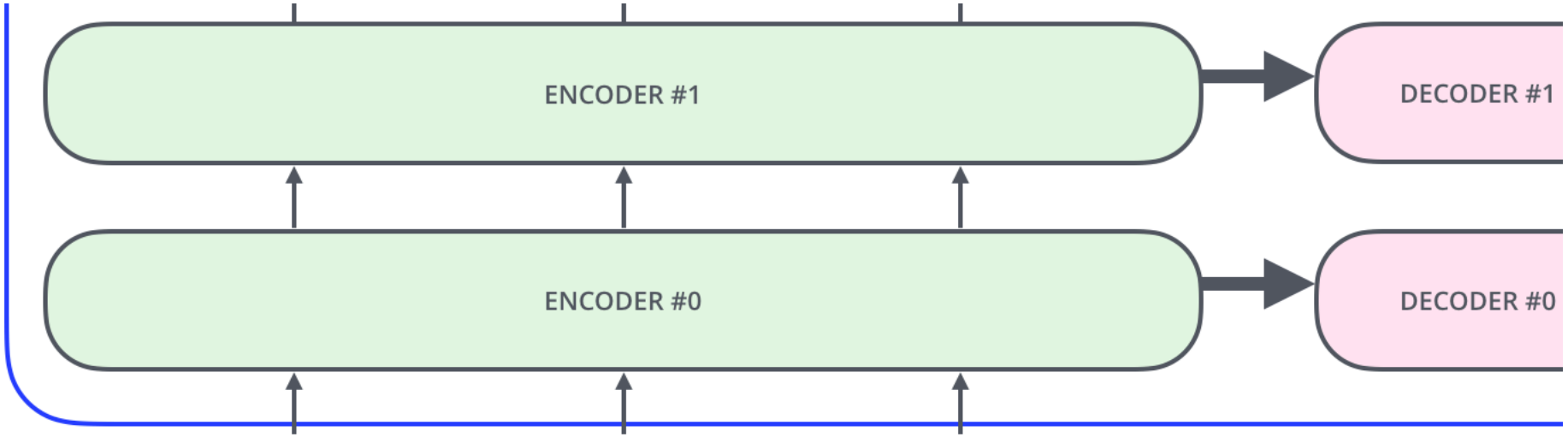
≃ query_dim

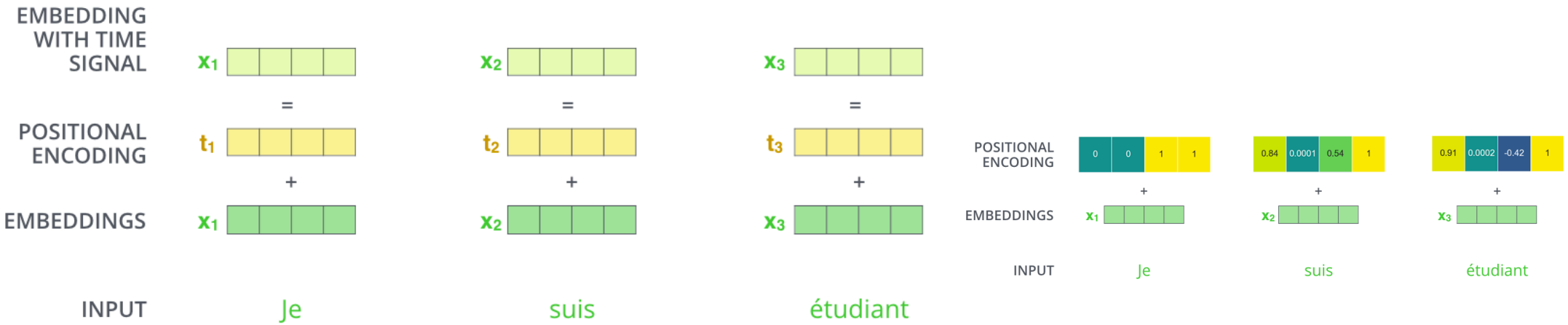
Attention — Multi-Head Attention



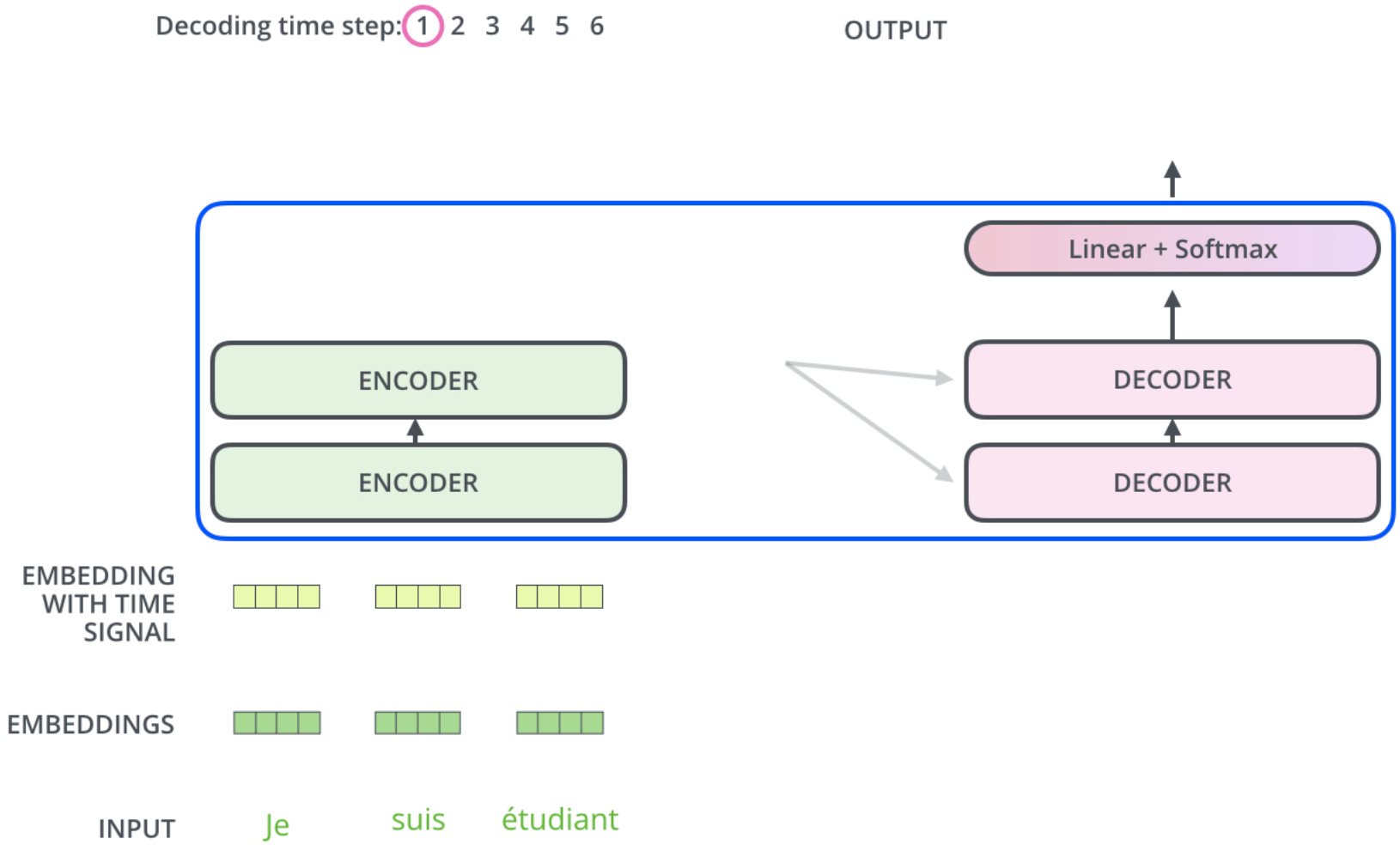
h = 8
d_model = 512

Positional Encoding





Decoder


Decoding time step: 1 2 3 4 5 6
OUTPUT
Linear + Softmax
ENCODER
ENCODER
DECODER
DECODER
EMBEDDING WITH TIME SIGNAL
EMBEDDINGS
INPUT
Je
suis
étudiant

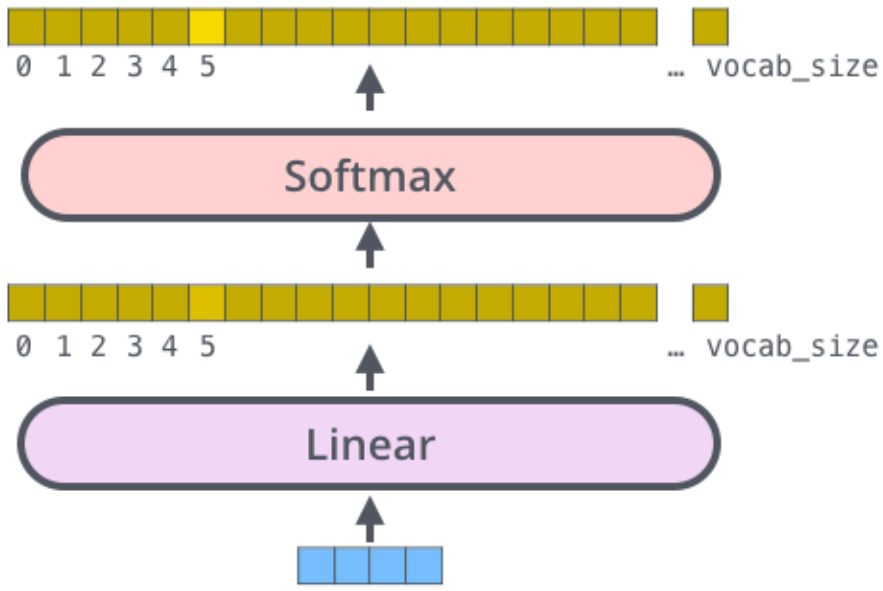
Decoder

Which word in our vocabulary is associated with this index?

am

Get the index of the cell with the highest value (`argmax`)

5



Decoder

Target Model Outputs

Trained Model Outputs

끝