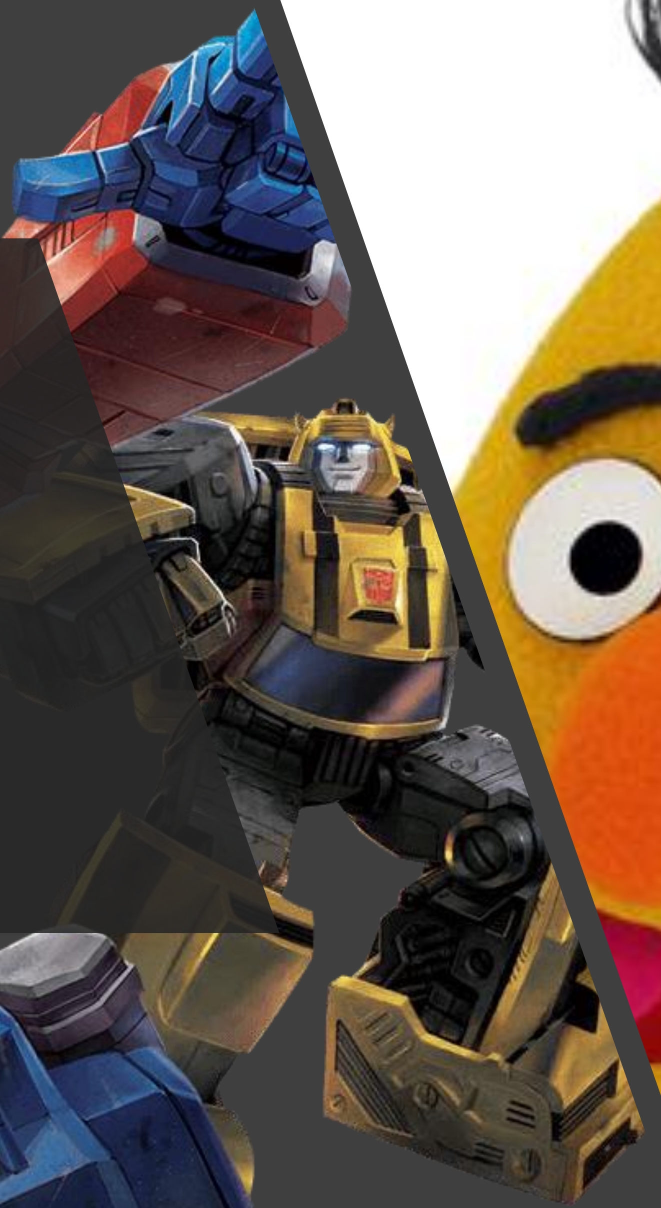
Transformer

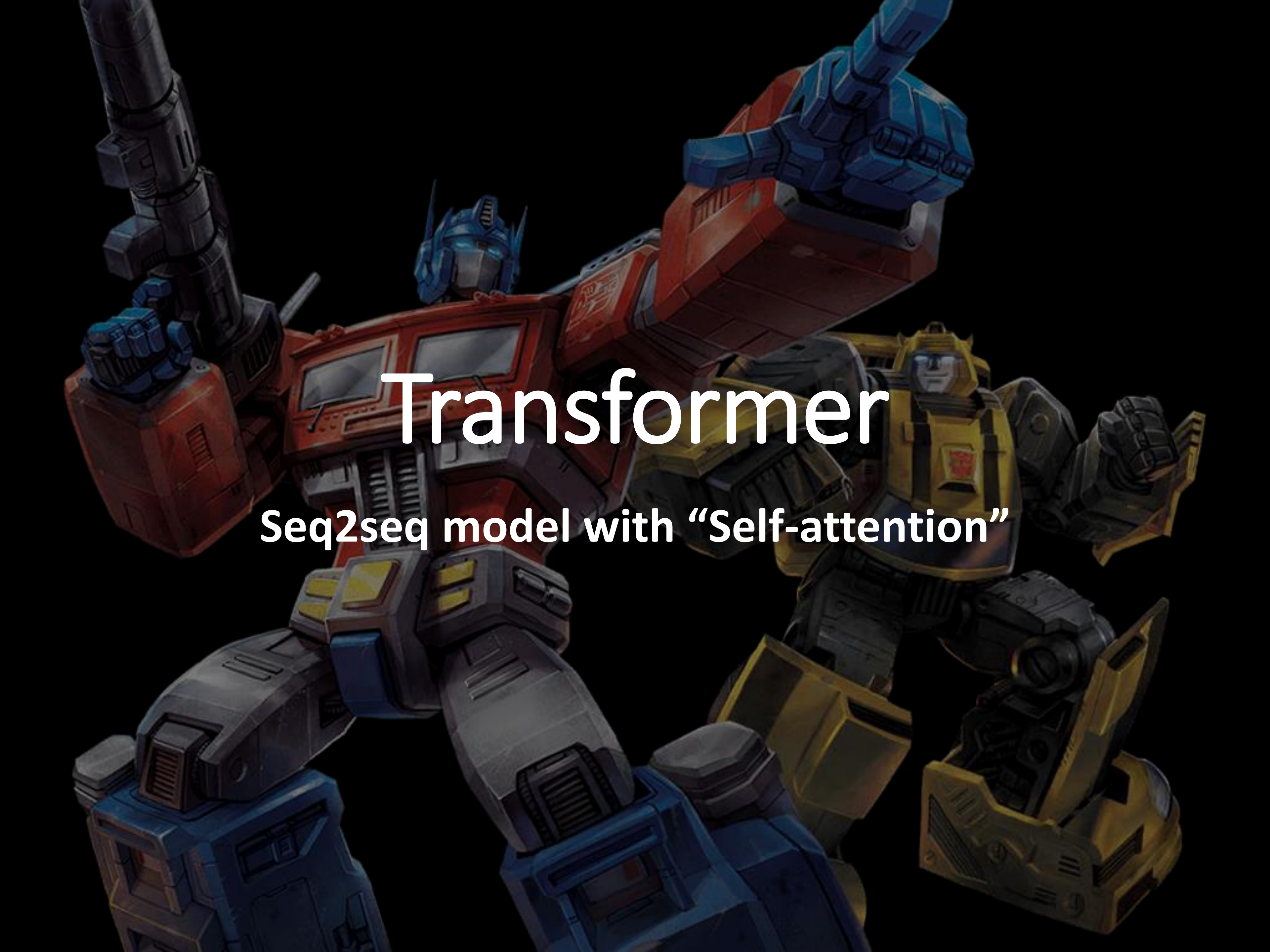
李宏毅

Hung-yi Lee

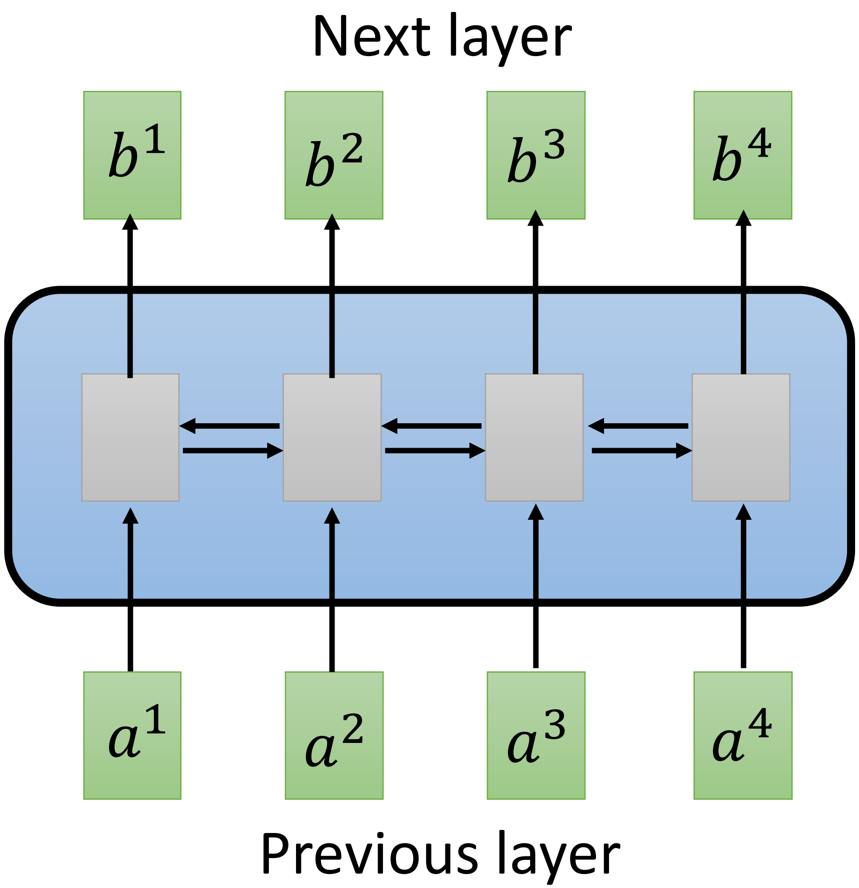


Transformer

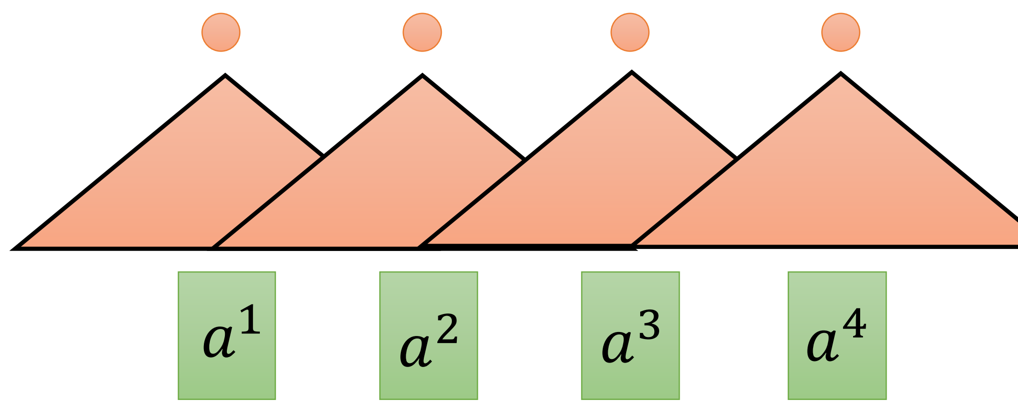
Seq2seq model with “Self-attention”



Sequence

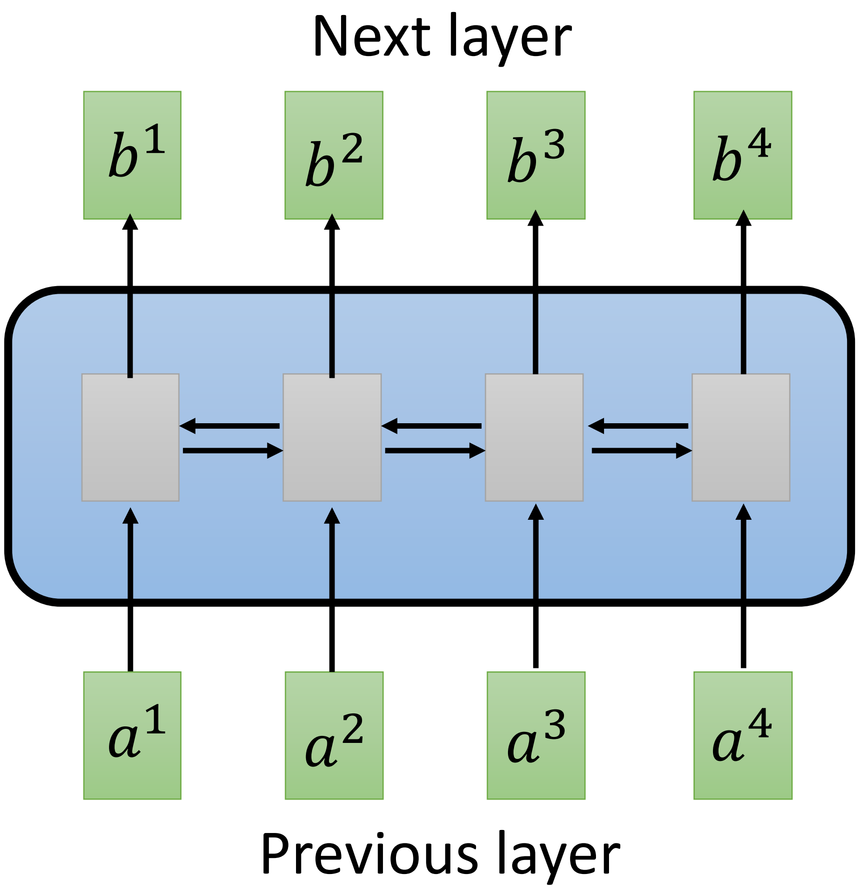


Hard to parallel !



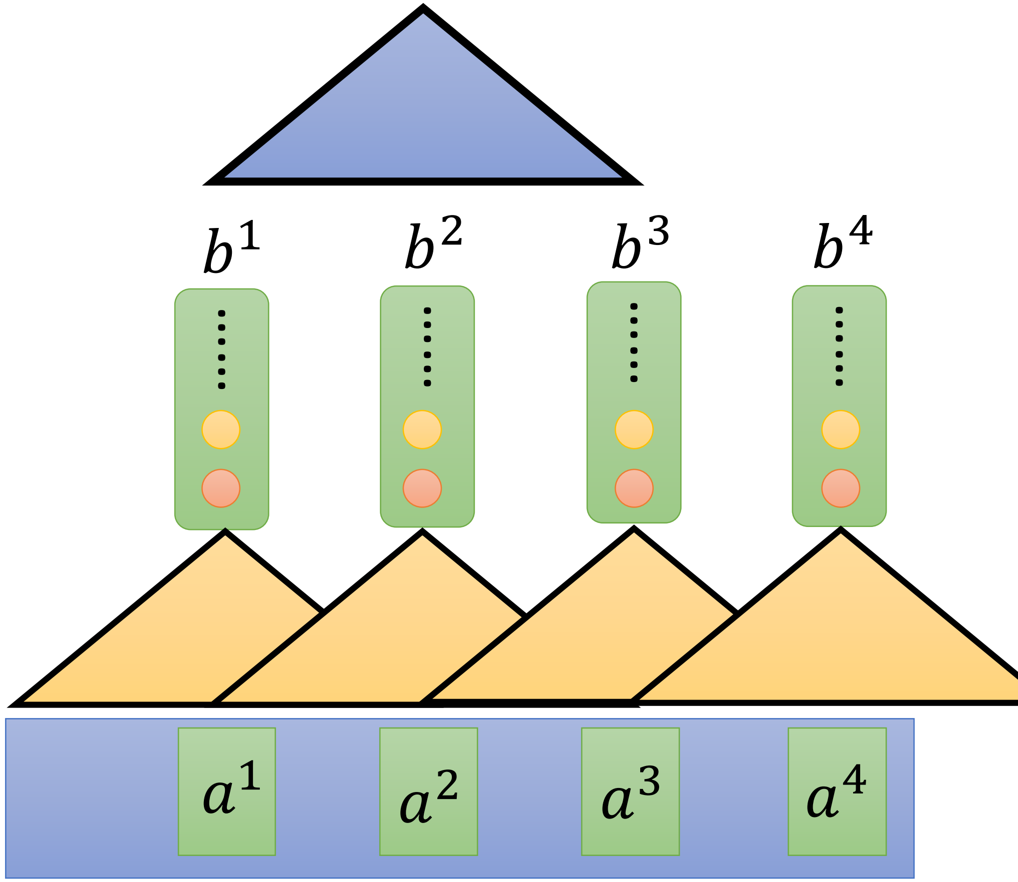
Using CNN to replace RNN

Sequence



Hard to parallel

Filters in higher layer can consider longer sequence

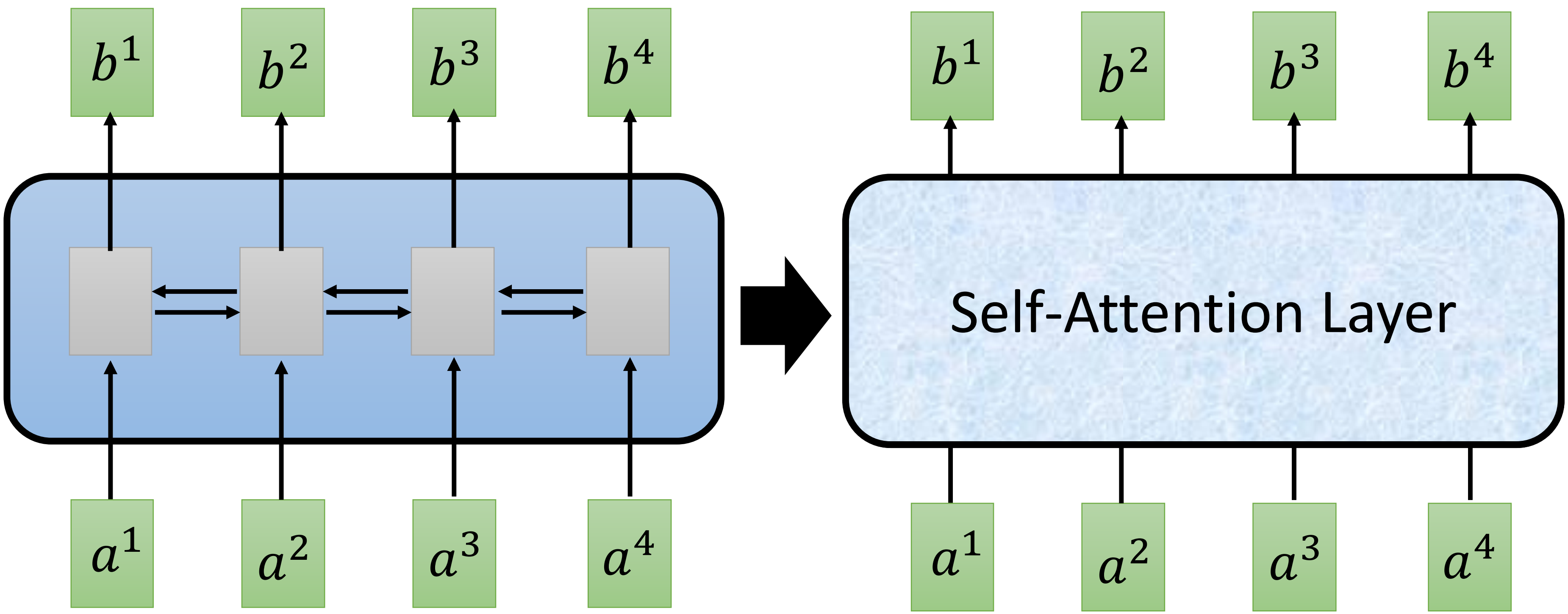


Using CNN to replace RNN
(CNN can parallel)

Self-Attention

b^i is obtained based on the whole input sequence.

b^1, b^2, b^3, b^4 can be parallelly computed.



You can try to replace any thing that has been done by RNN with self-attention.

Self-attention

https://arxiv.org/abs/1706.03762

q: query (to match others)

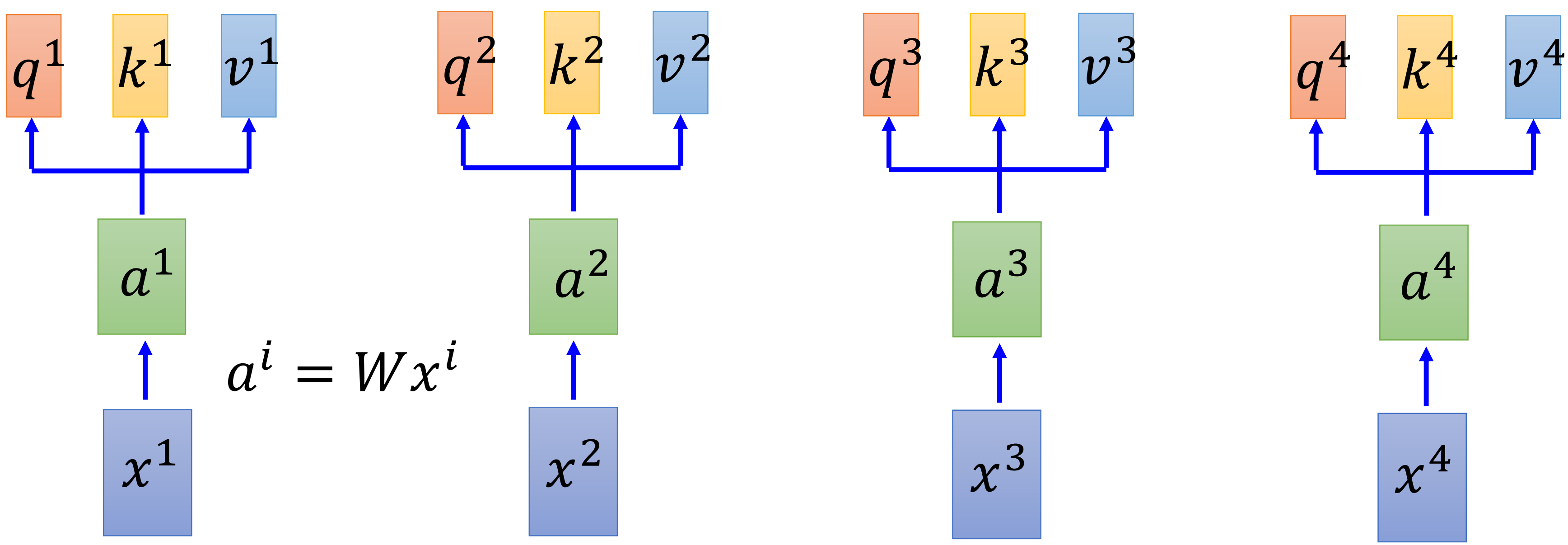
$$q^i = W^q a^i$$

k: key (to be matched)

$$k^i = W^k a^i$$

v: information to be extracted

$$v^i = W^v a^i$$

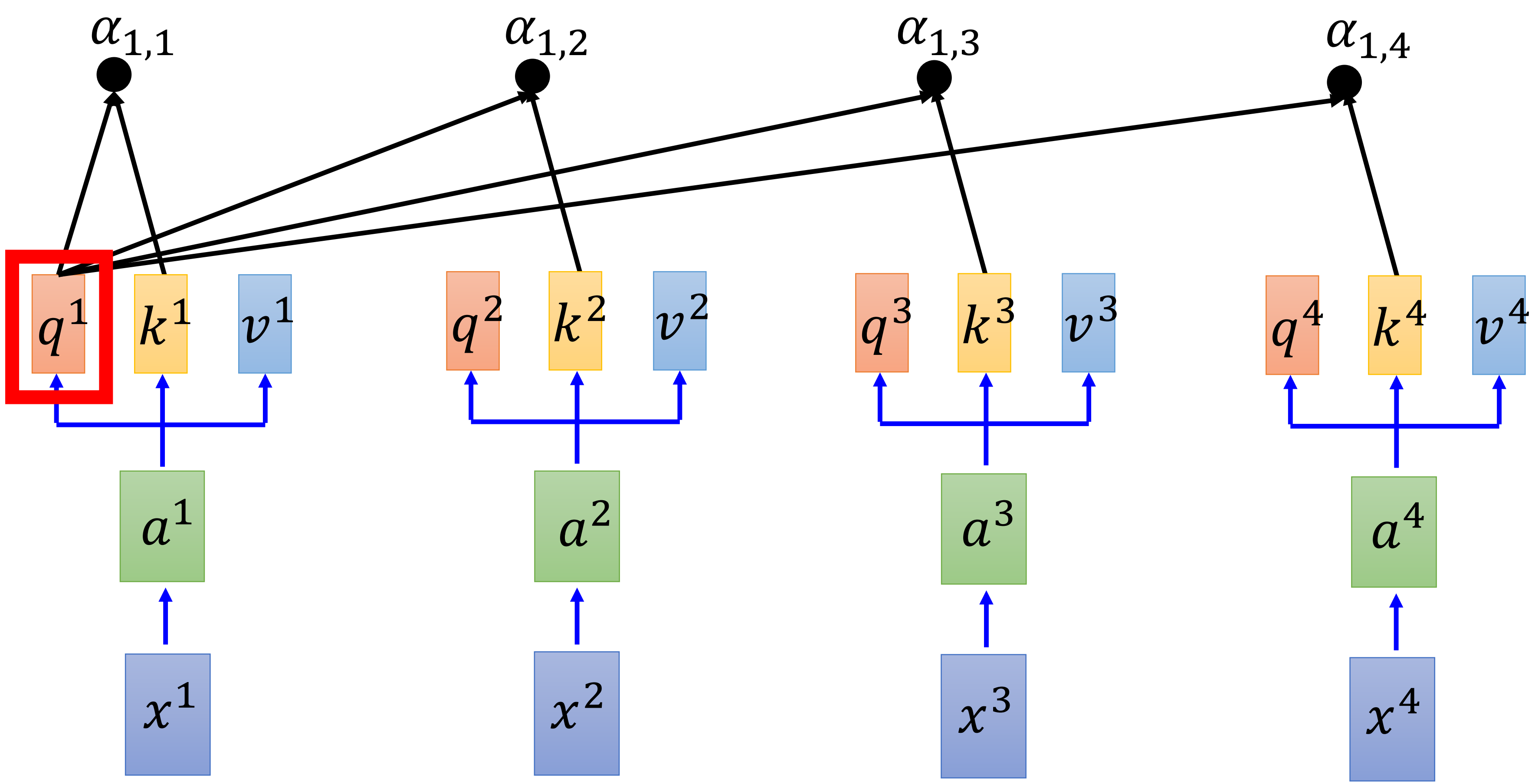


Self-attention

拿每個 query q 去對每個 key k 做 attention

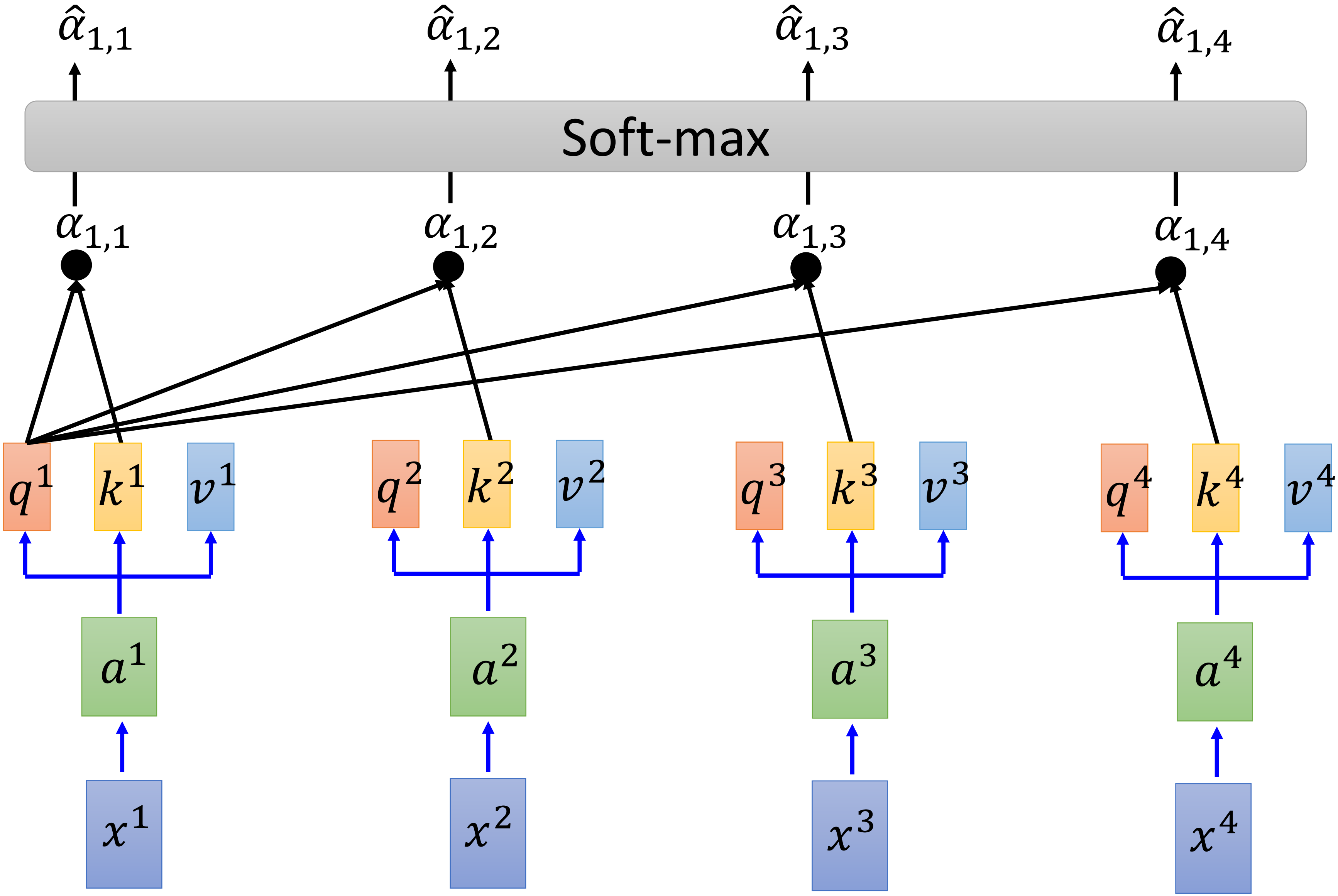
d is the dim of q and k

Scaled Dot-Product Attention: $\alpha_{1,i} = \underbrace{q^1 \cdot k^i}_{\text{dot product}} / \sqrt{d}$



Self-attention

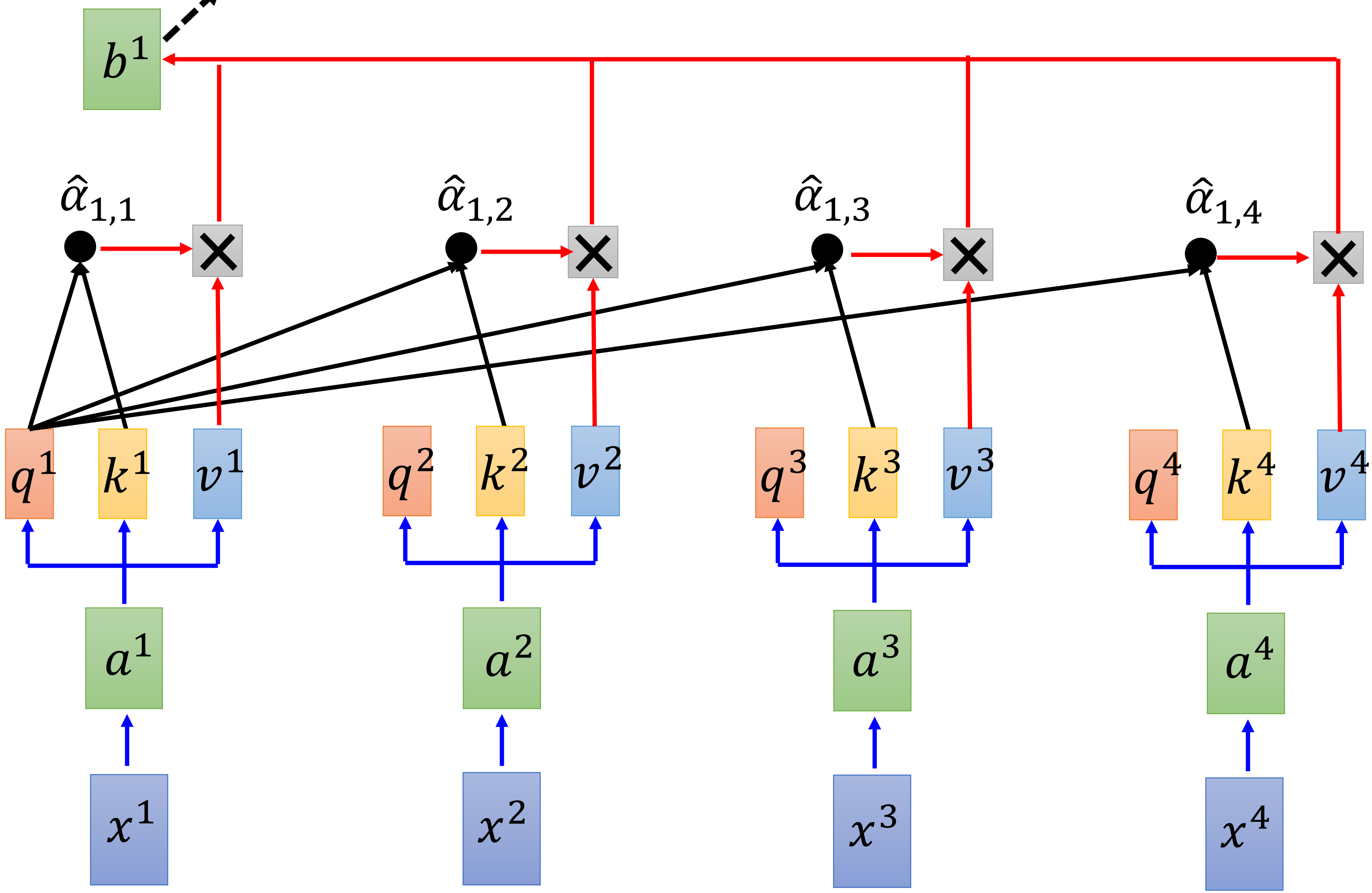
$$\hat{\alpha}_{1,i} = exp(\alpha_{1,i})/\sum_j exp(\alpha_{1,j})$$



Self-attention

Considering the whole sequence

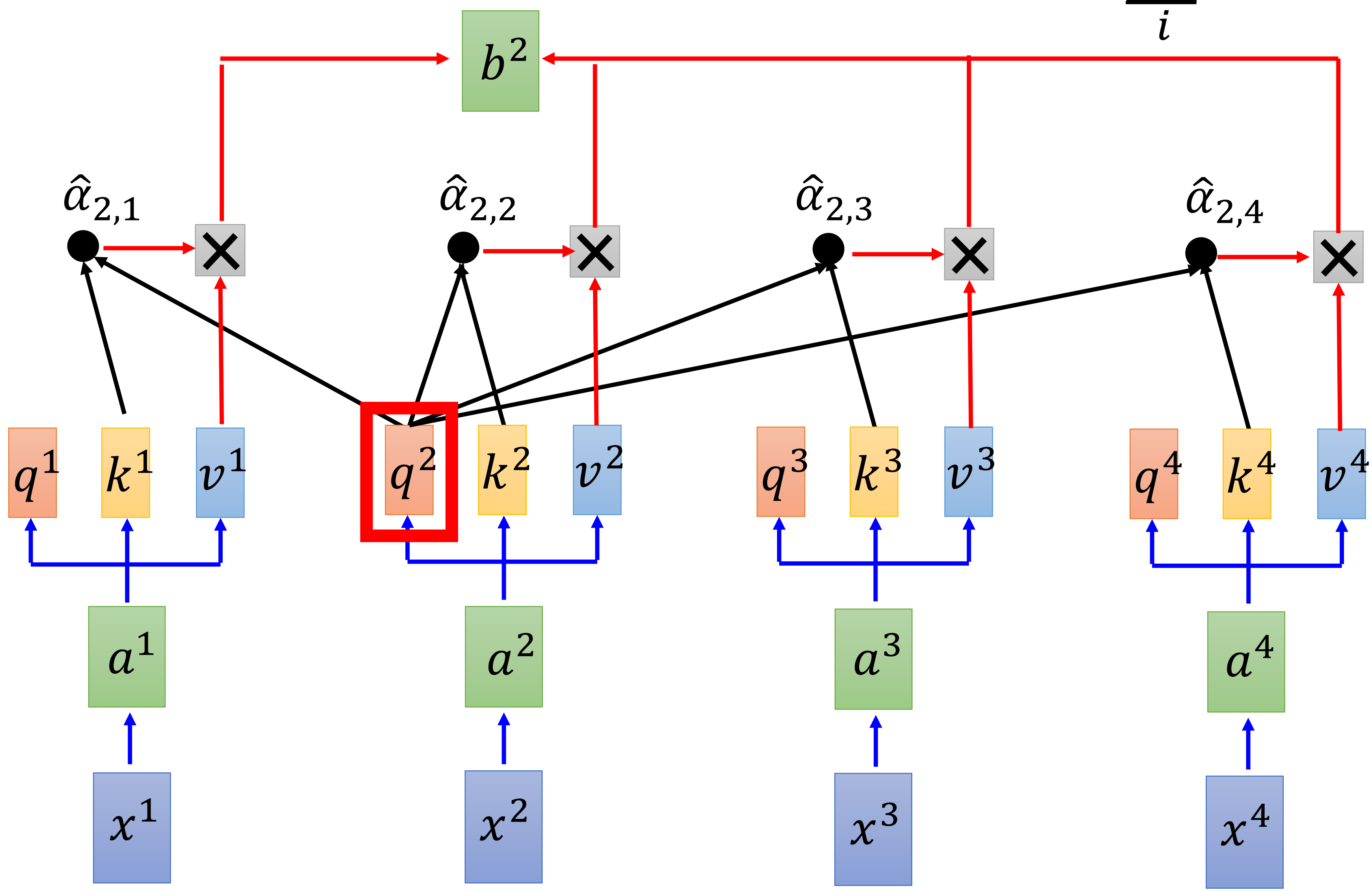
$$b^1 = \sum_i \hat{\alpha}_{1,i} v^i$$



Self-attention

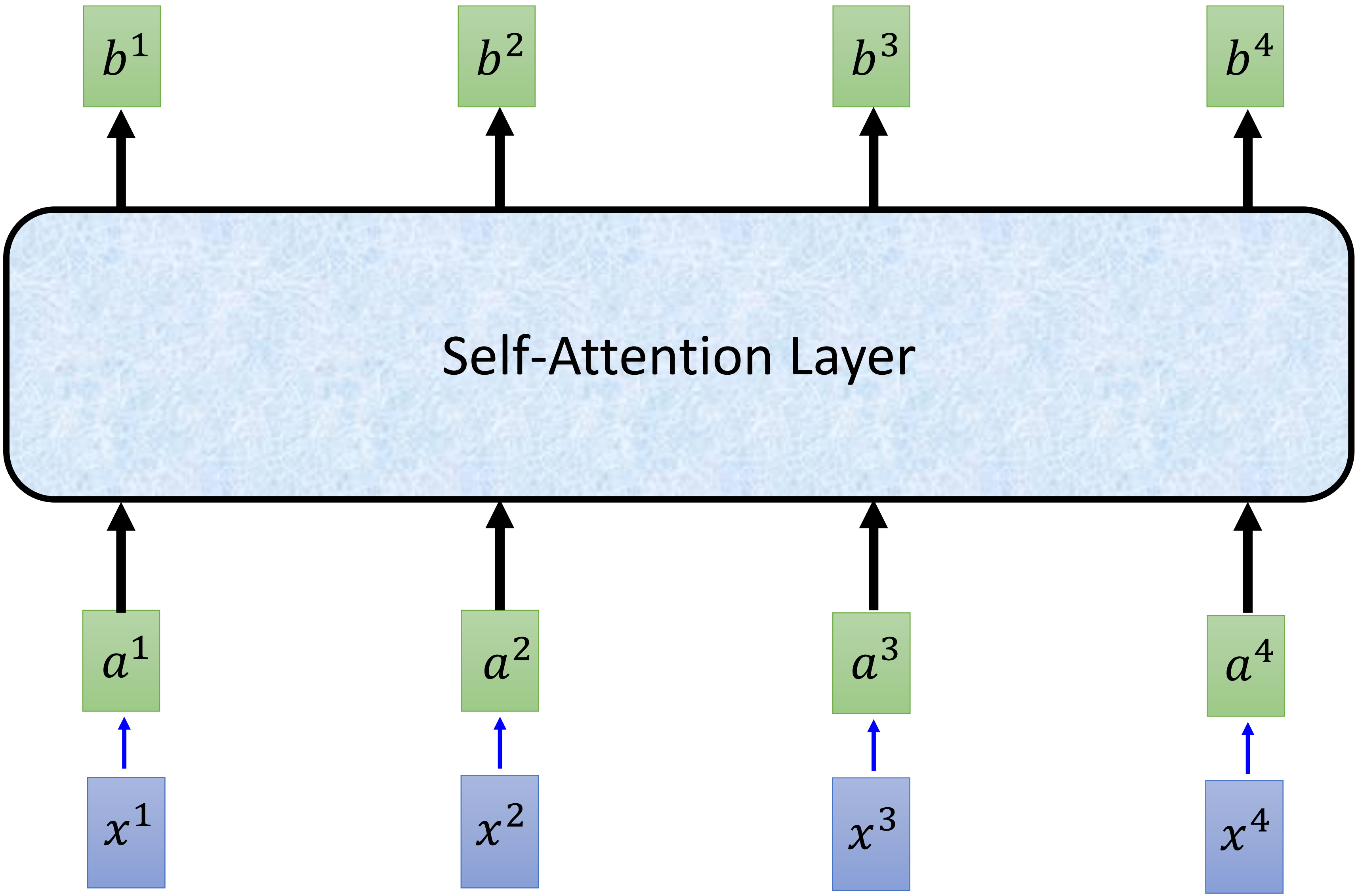
拿每個 query q 去對每個 key k 做 attention

$$b^2 = \sum_i \hat{\alpha}_{2,i} v^i$$



Self-attention

b^1, b^2, b^3, b^4 can be parallelly computed.



Self-attention

$$q^i = W^q a^i$$

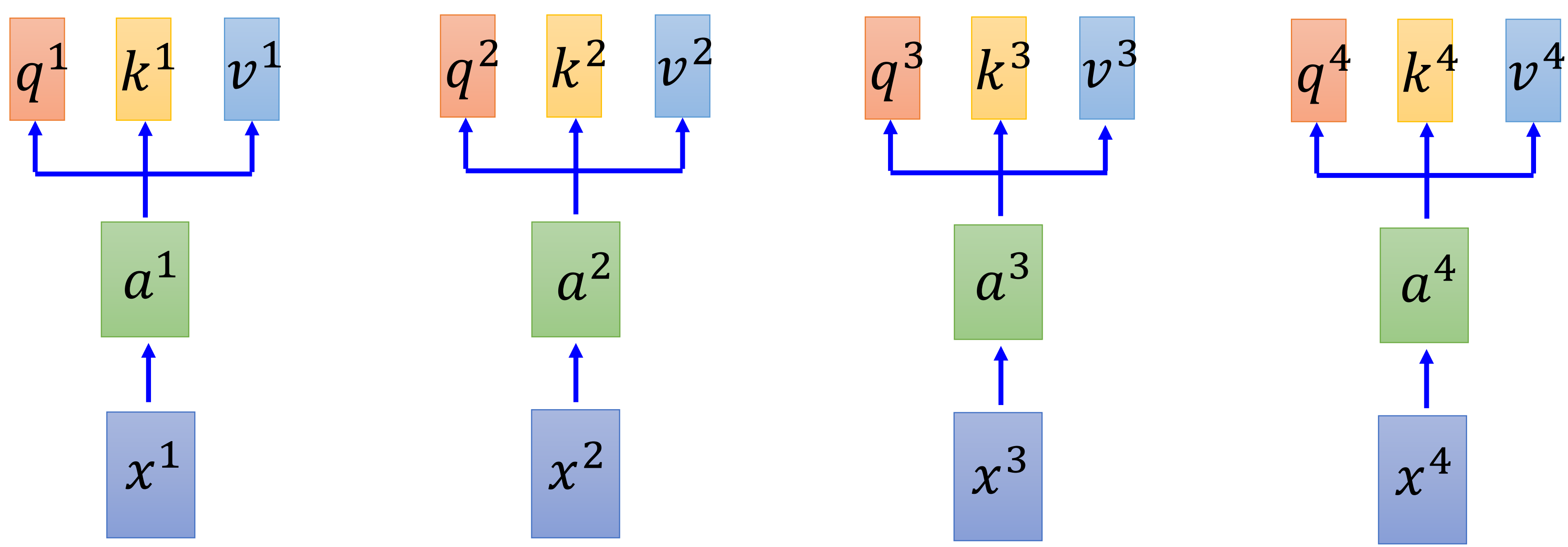
$$k^i = W^k a^i$$

$$v^i = W^v a^i$$

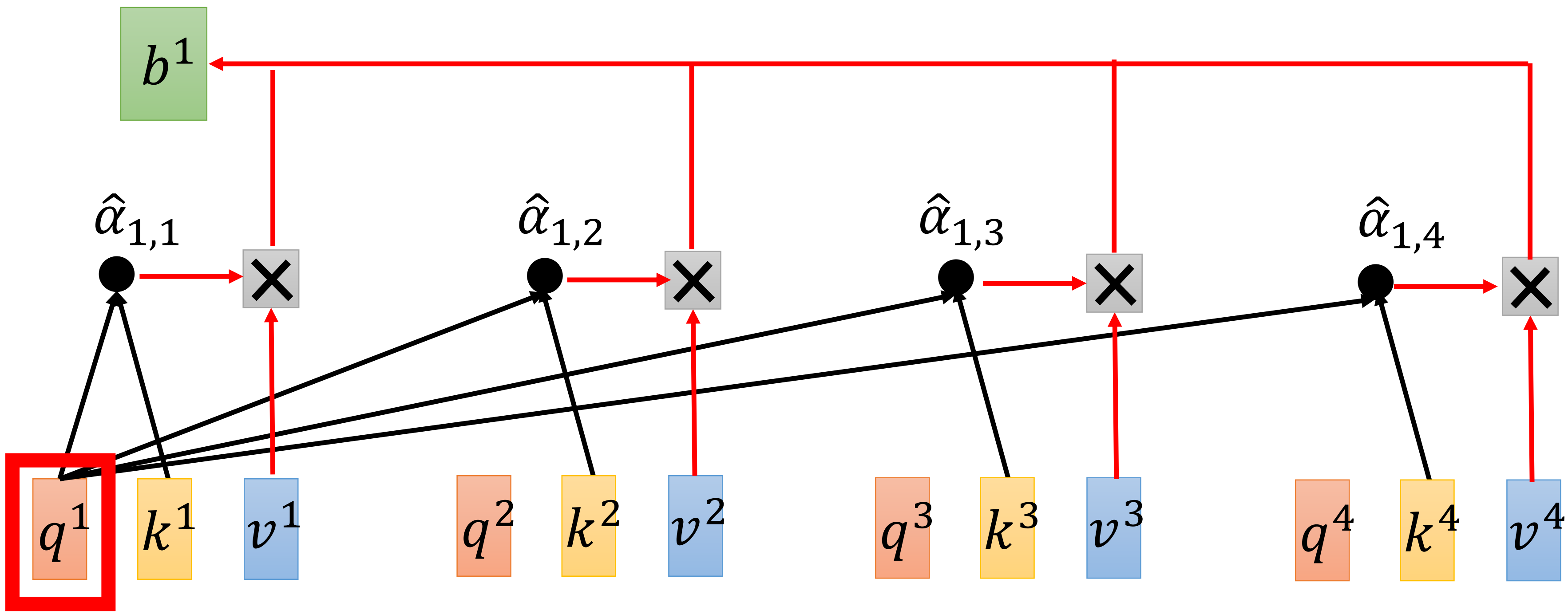
$$\underset{Q}{\begin{bmatrix} q^1 & q^2 & q^3 & q^4 \end{bmatrix}} = W^q \underset{\mathrm{I}}{\begin{bmatrix} a^1 & a^2 & a^3 & a^4 \end{bmatrix}}$$

$$\underset{K}{\begin{bmatrix} k^1 & k^2 & k^3 & k^4 \end{bmatrix}} = W^k \underset{\mathrm{I}}{\begin{bmatrix} a^1 & a^2 & a^3 & a^4 \end{bmatrix}}$$

$$\underset{V}{\begin{bmatrix} v^1 & v^2 & v^3 & v^4 \end{bmatrix}} = W^v \underset{\mathrm{I}}{\begin{bmatrix} a^1 & a^2 & a^3 & a^4 \end{bmatrix}}$$



Self-attention



$\alpha_{1,1} = k^1 \; q^1$ $\quad \alpha_{1,2} = k^2 \; q^1$

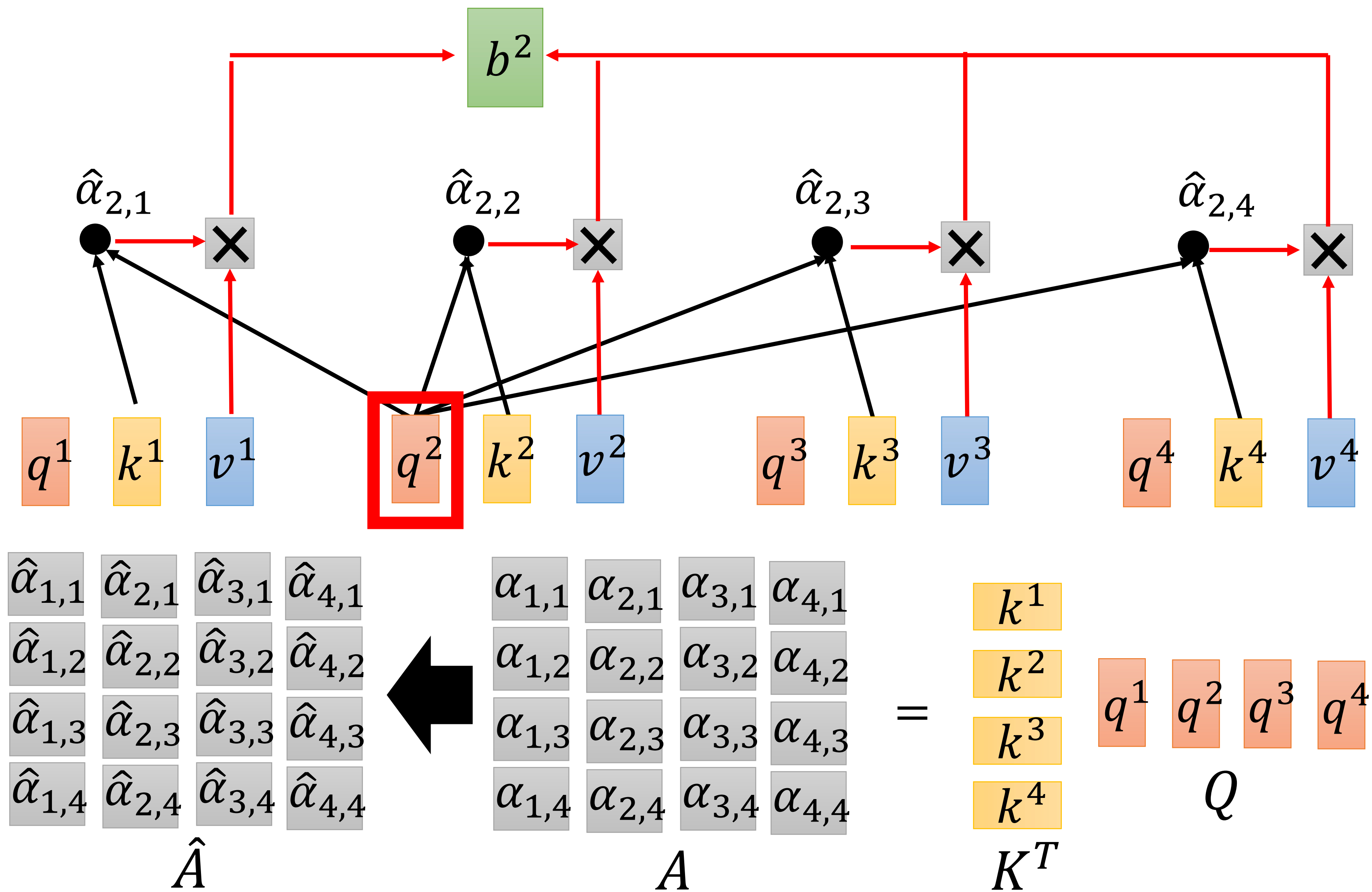
$\alpha_{1,3} = k^3 \; q^1$ $\quad \alpha_{1,4} = k^4 \; q^1$

$$\begin{bmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \alpha_{1,3} \\ \alpha_{1,4} \end{bmatrix} = \begin{bmatrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{bmatrix} q^1$$

(ignore $\sqrt{d}$ for simplicity)

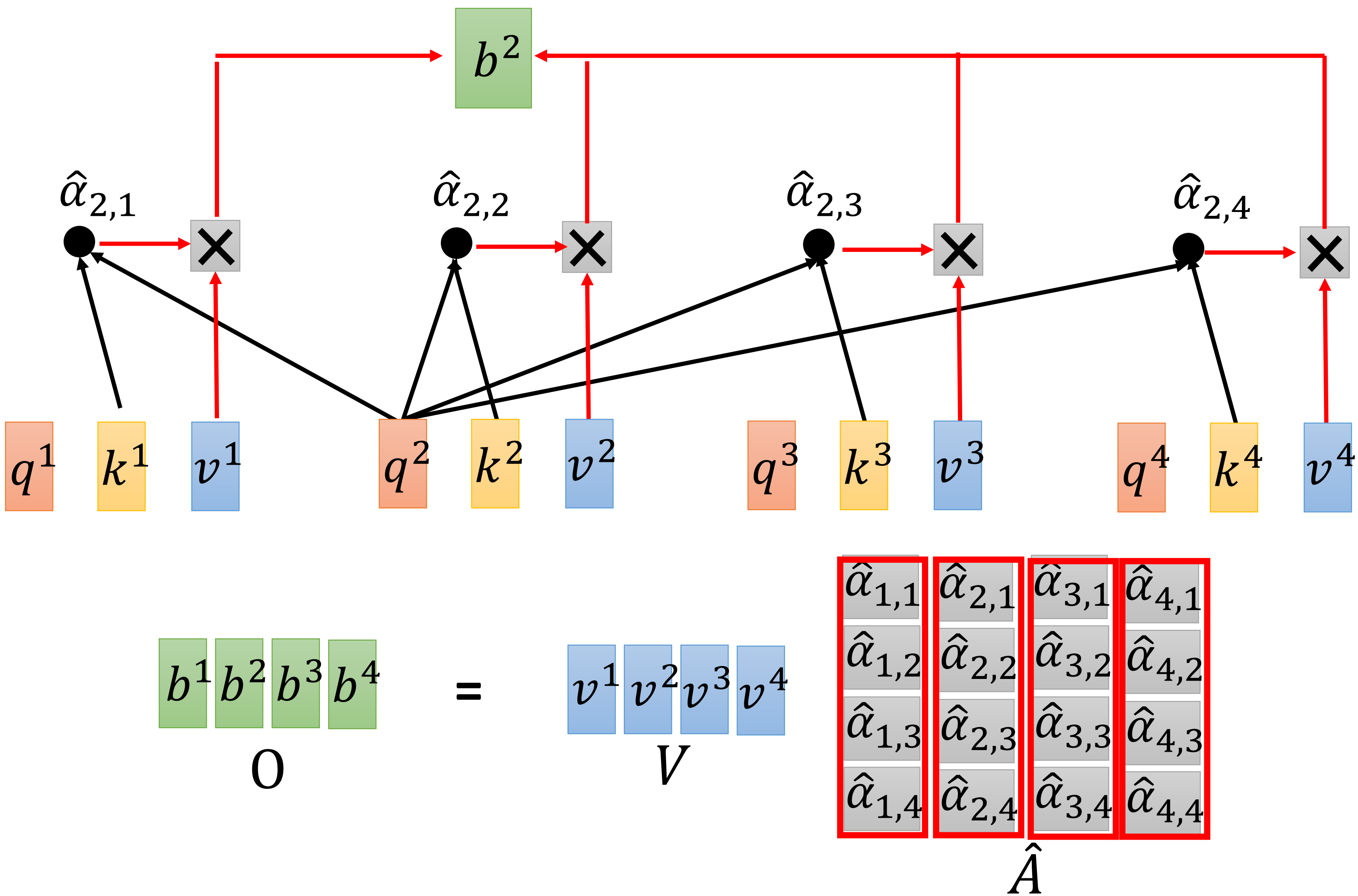
Self-attention

$$b^2 = \sum_i \hat{\alpha}_{2,i} v^i$$

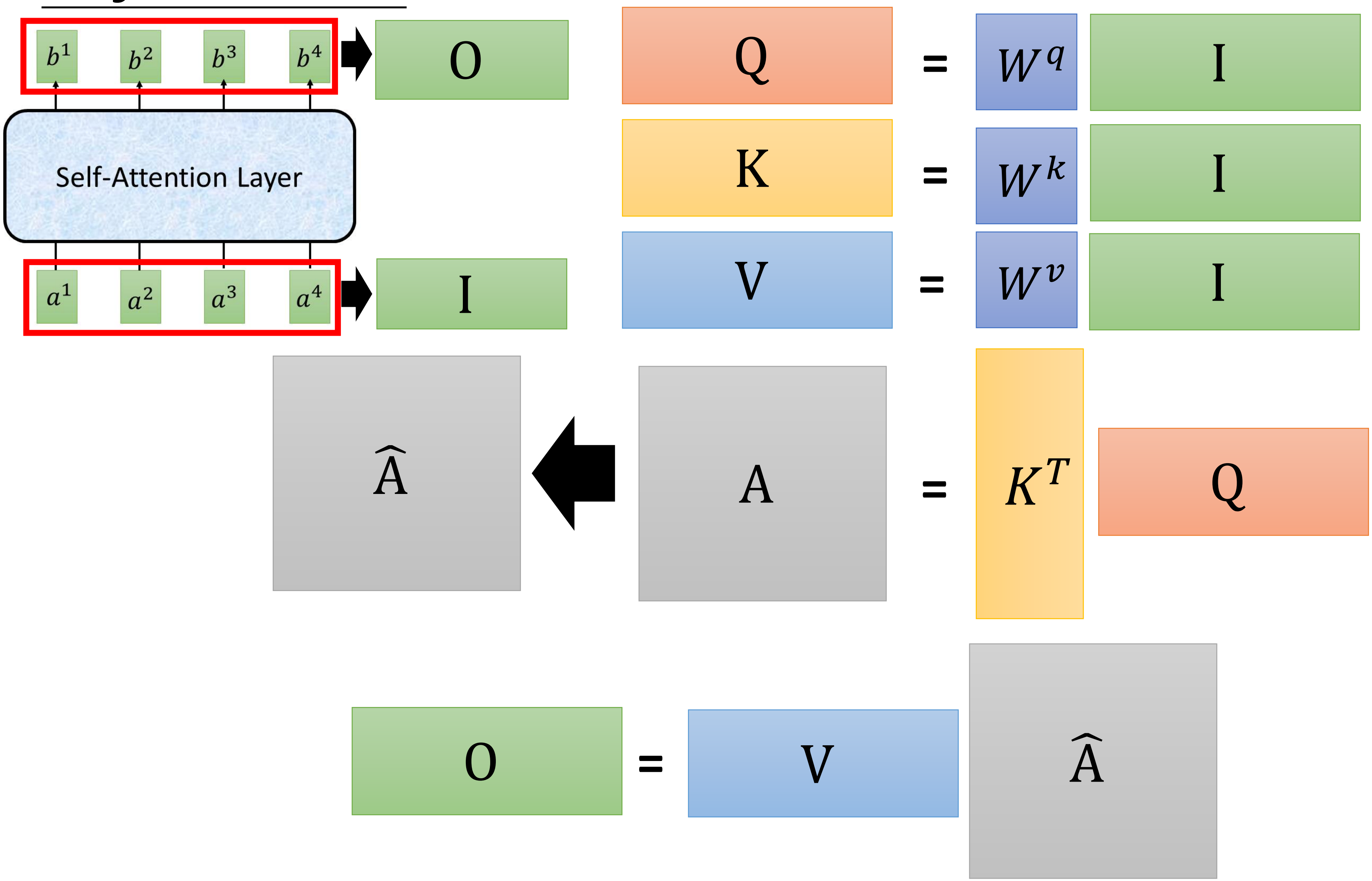


Self-attention

$$b^2 = \sum_i \hat{\alpha}_{2,i} v^i$$



Self-attention

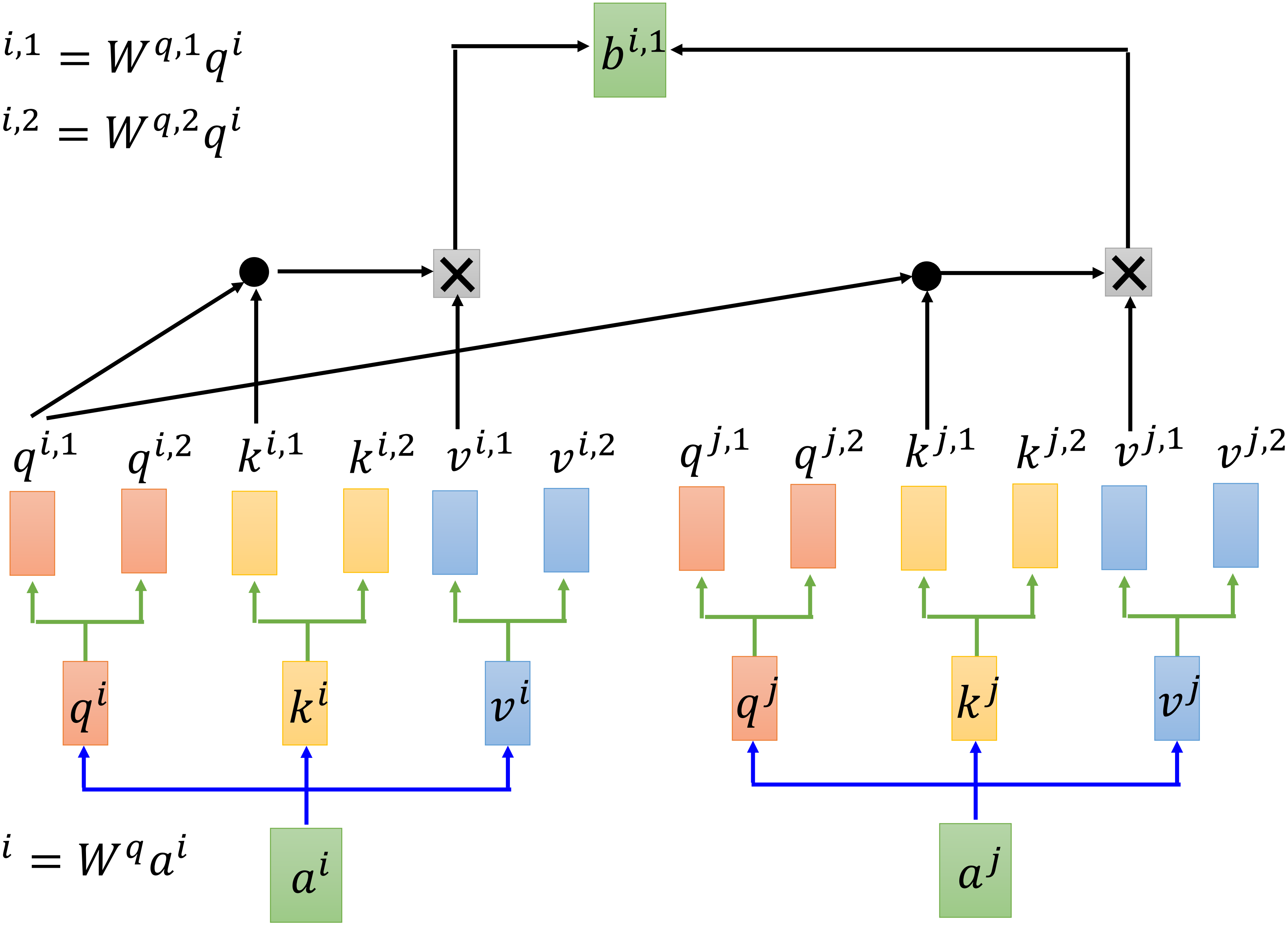


反正就是一堆矩陣乘法，用 GPU 可以加速

Multi-head Self-attention (2 heads as example)

$$q^{i,1} = W^{q,1} q^i$$

$$q^{i,2} = W^{q,2} q^i$$

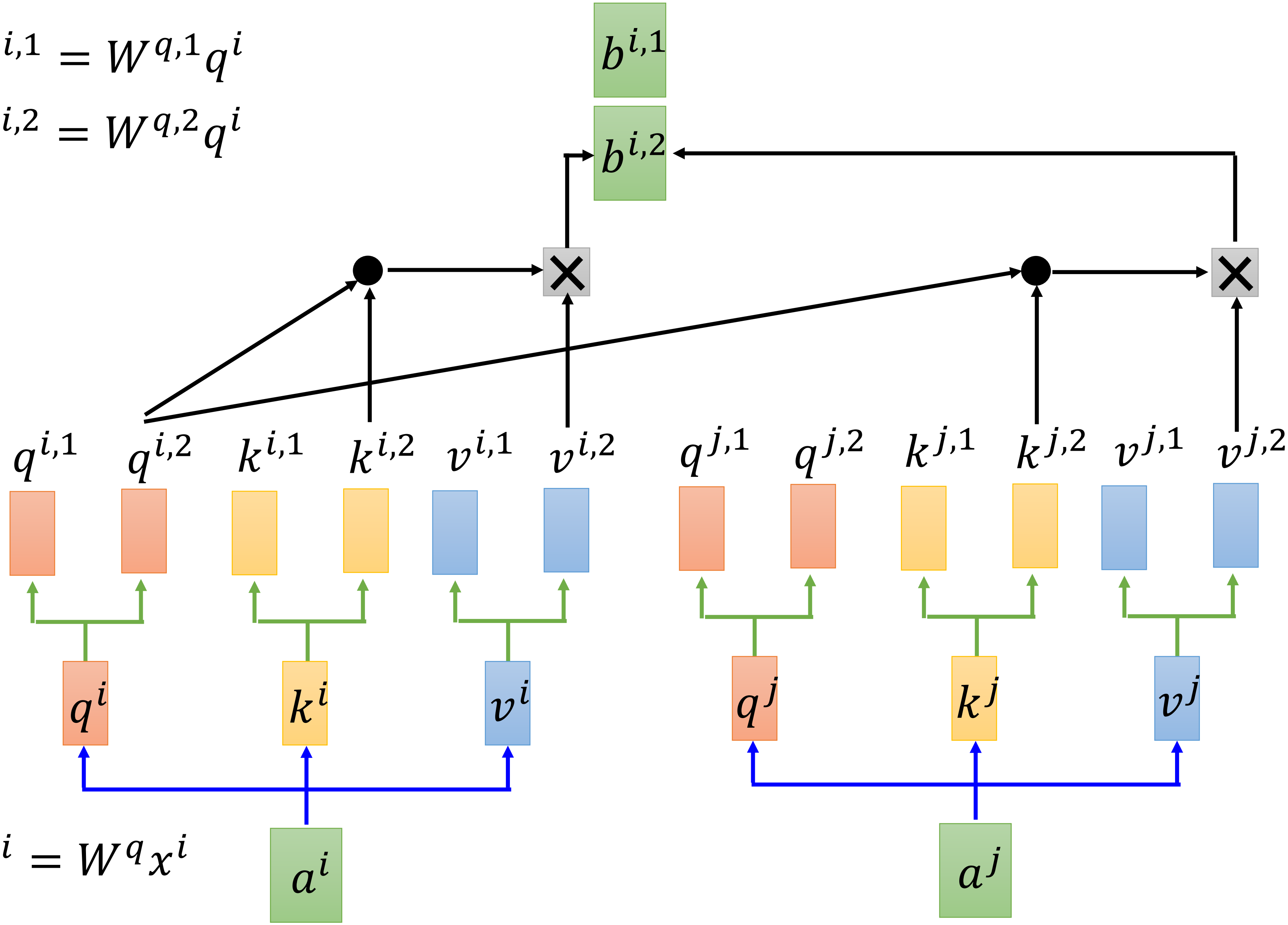


$$q^i = W^q a^i$$

Multi-head Self-attention (2 heads as example)

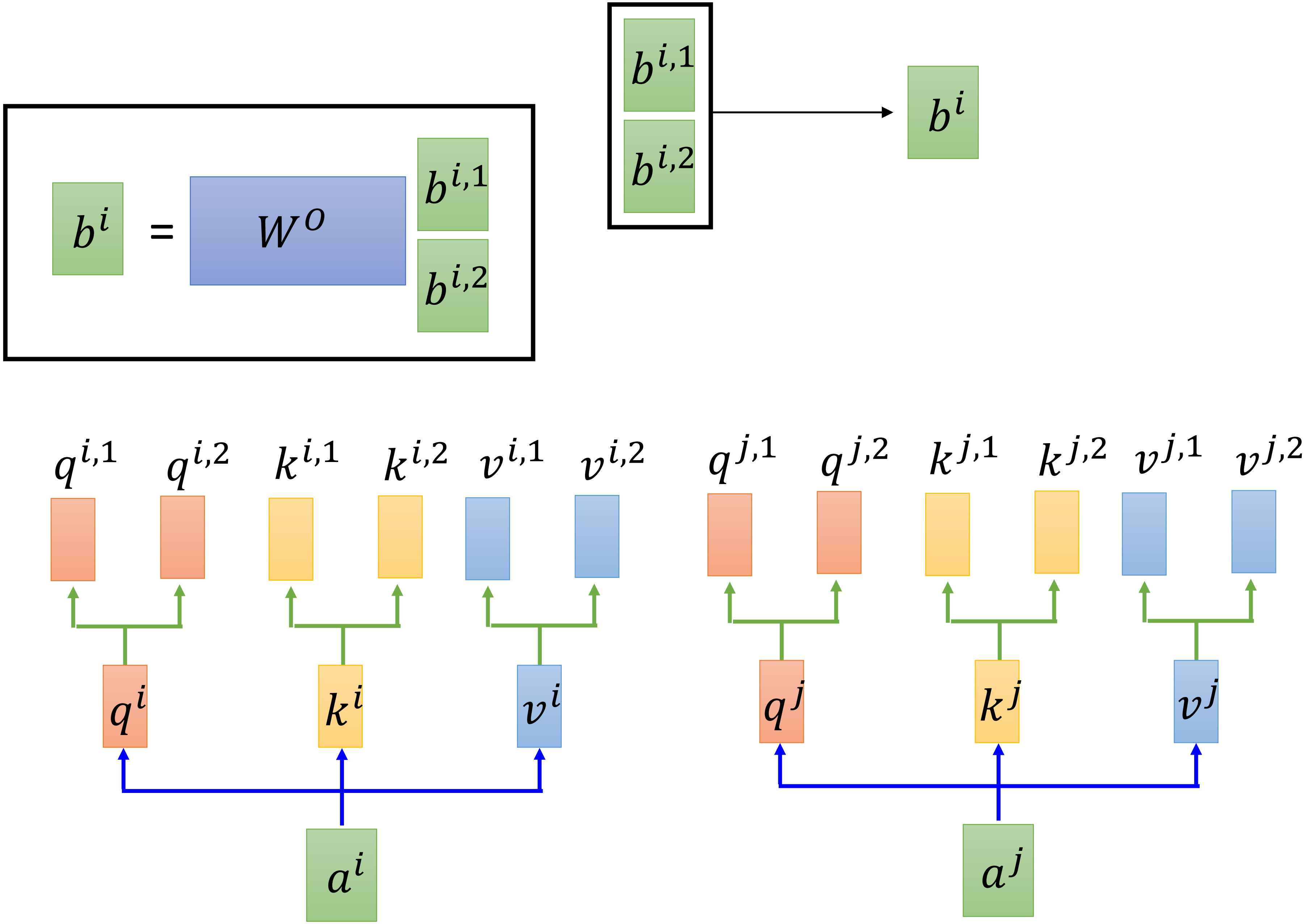
$$q^{i,1} = W^{q,1} q^i$$

$$q^{i,2} = W^{q,2} q^i$$

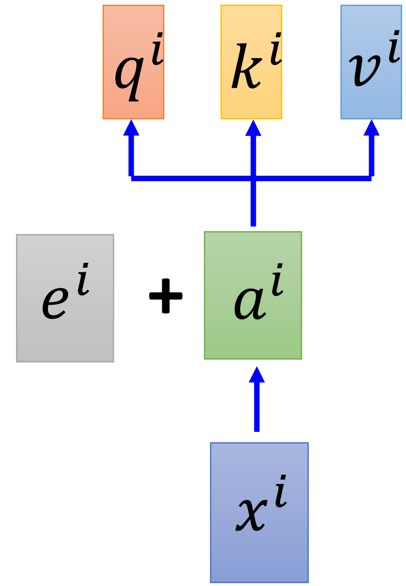


$$q^i = W^q x^i$$

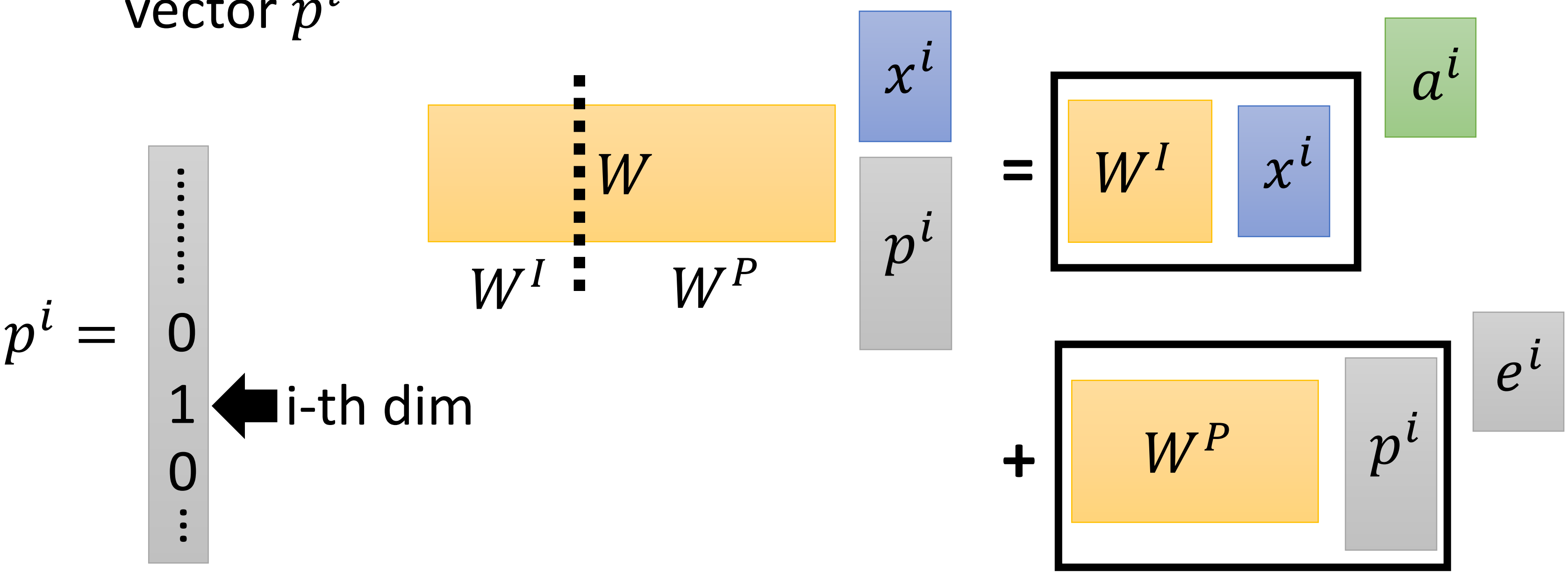
Multi-head Self-attention (2 heads as example)

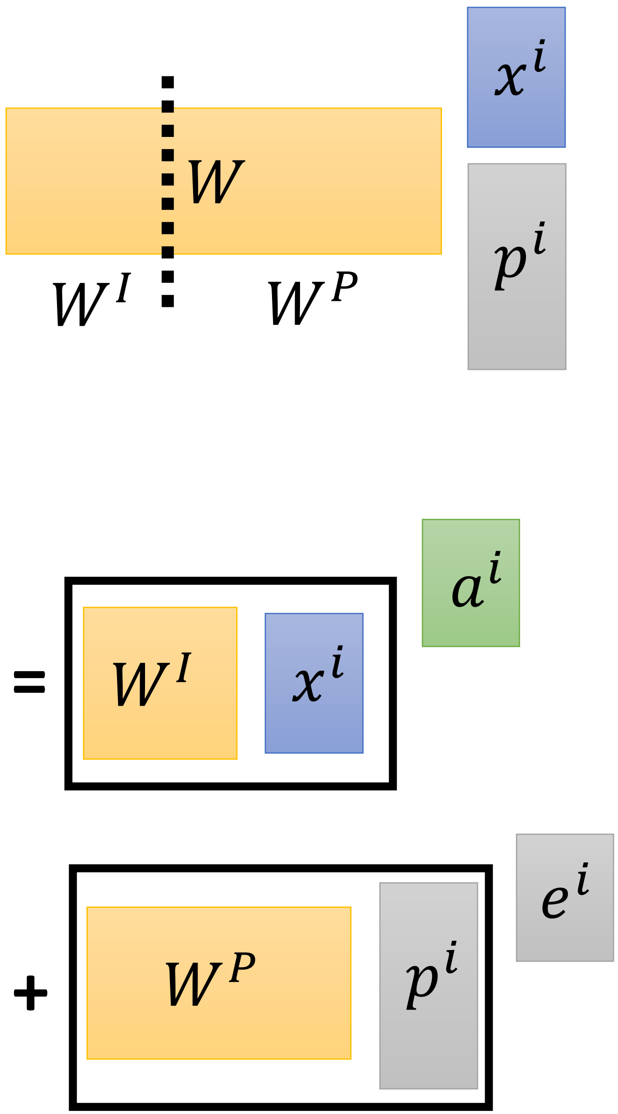


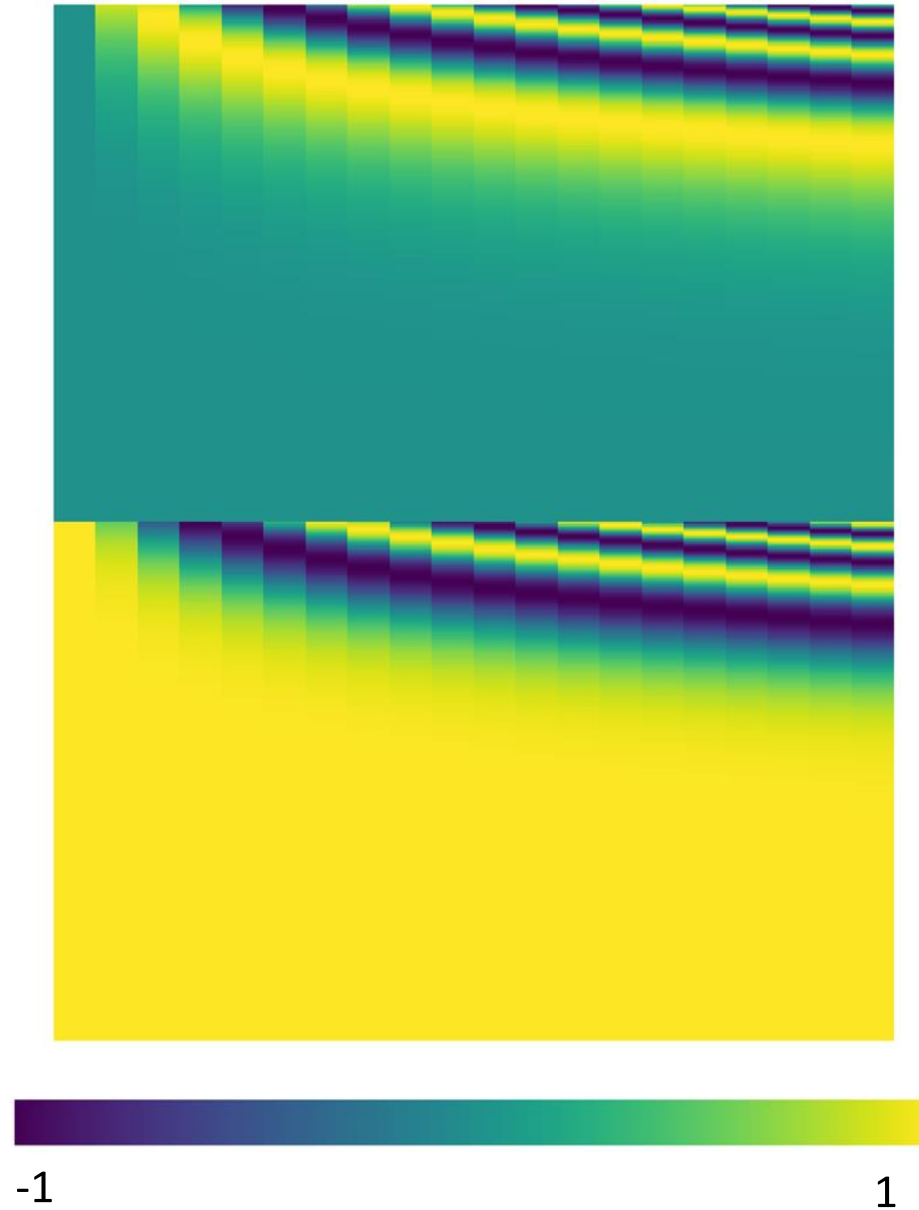
Positional Encoding



- No position information in self-attention.
- Original paper: each position has a unique positional vector e^i (not learned from data)
- In other words: each x^i appends a one-hot vector p^i



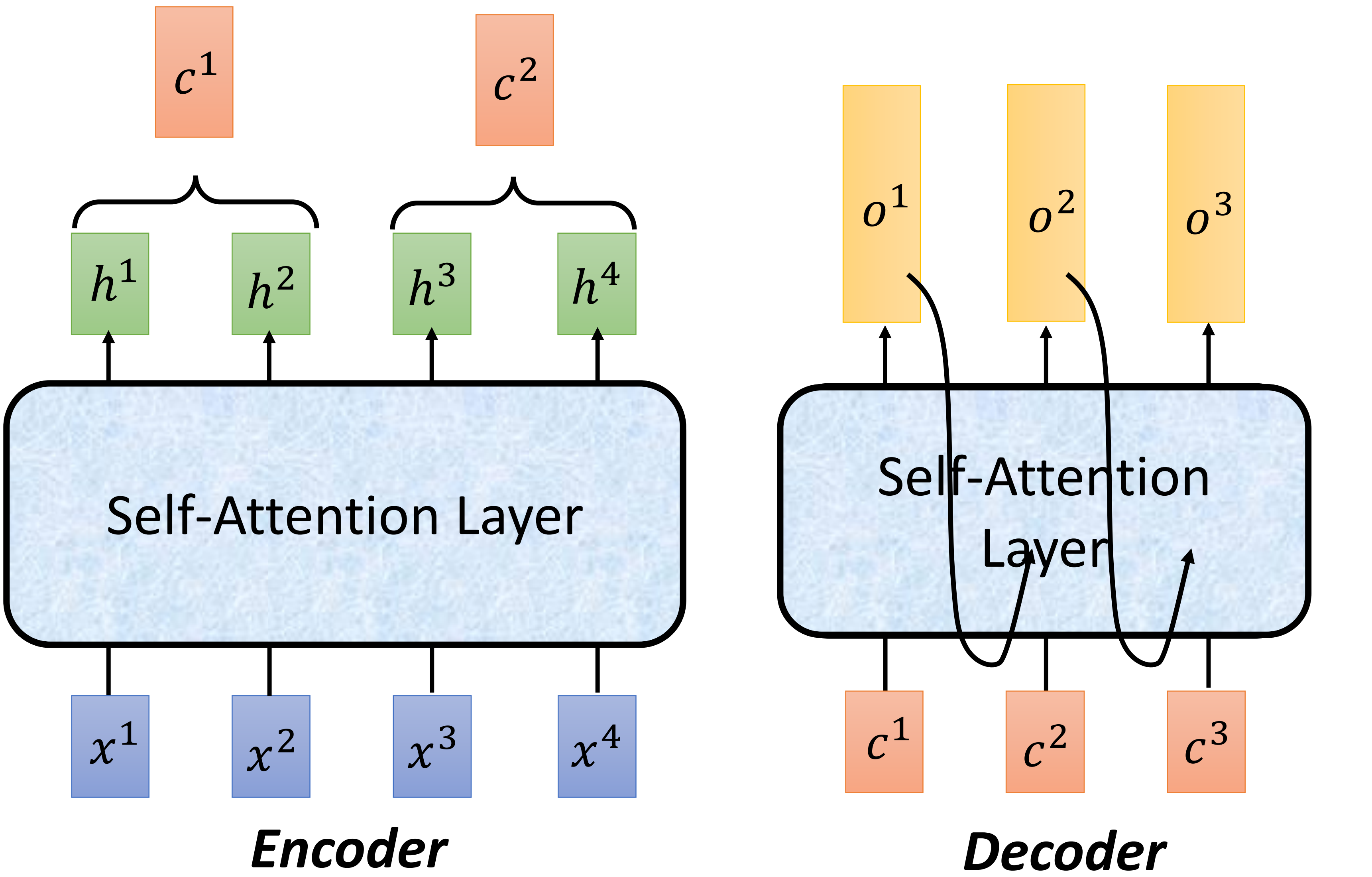




source of image: http://jalammar.github.io/illustrated-transformer/

Review: https://www.youtube.com/watch?v=ZjfjPzXw6og&feature=youtu.be

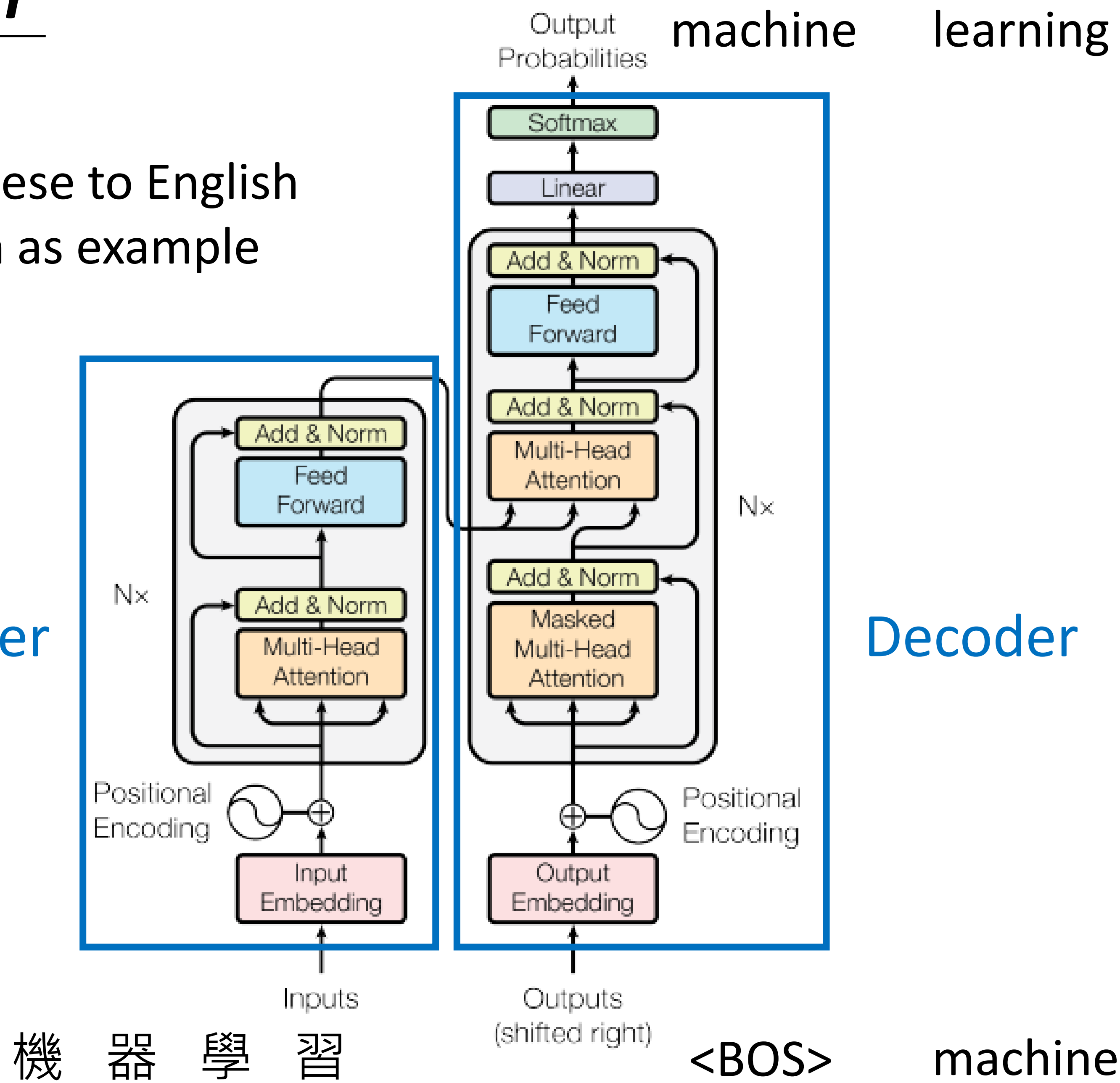
Seq2seq with Attention



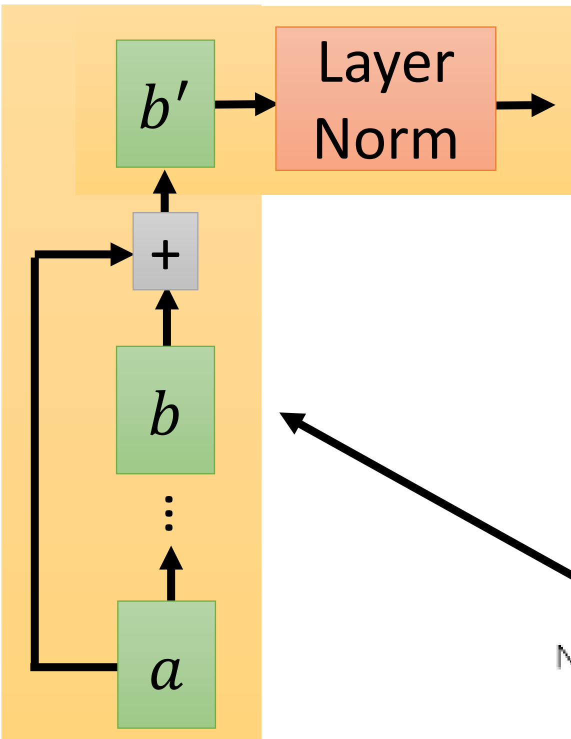
https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

Transformer

Using Chinese to English translation as example

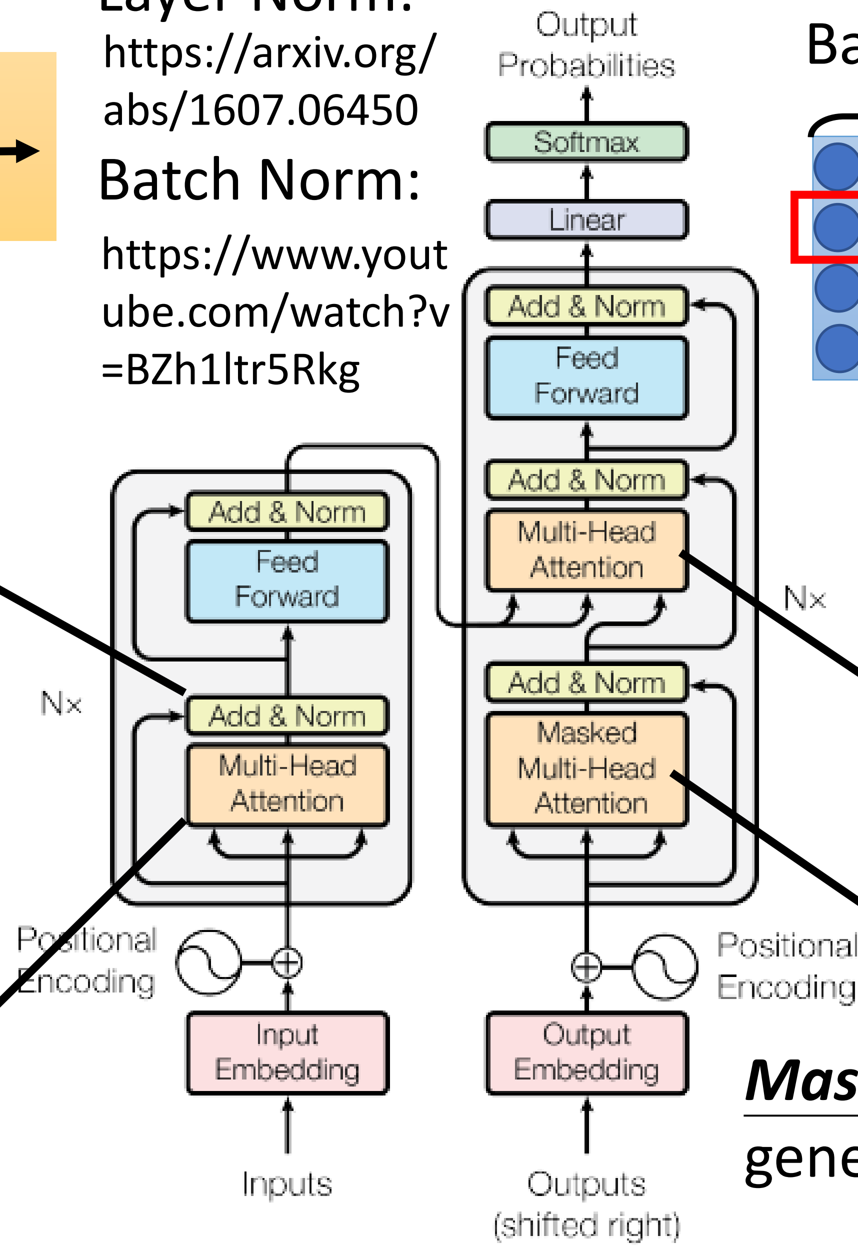


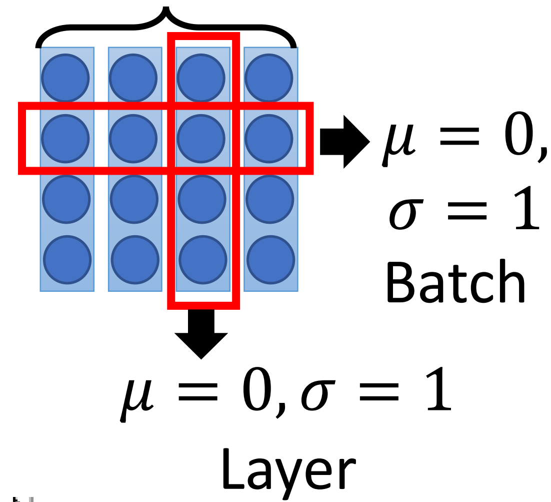
Transformer



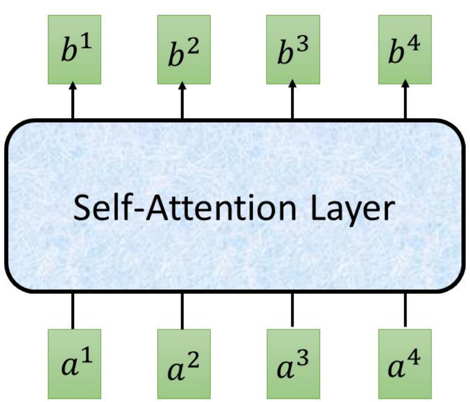
Layer Norm:
https://arxiv.org/abs/1607.06450

Batch Norm:
https://www.youtube.com/watch?v=BZh1ltr5Rkg



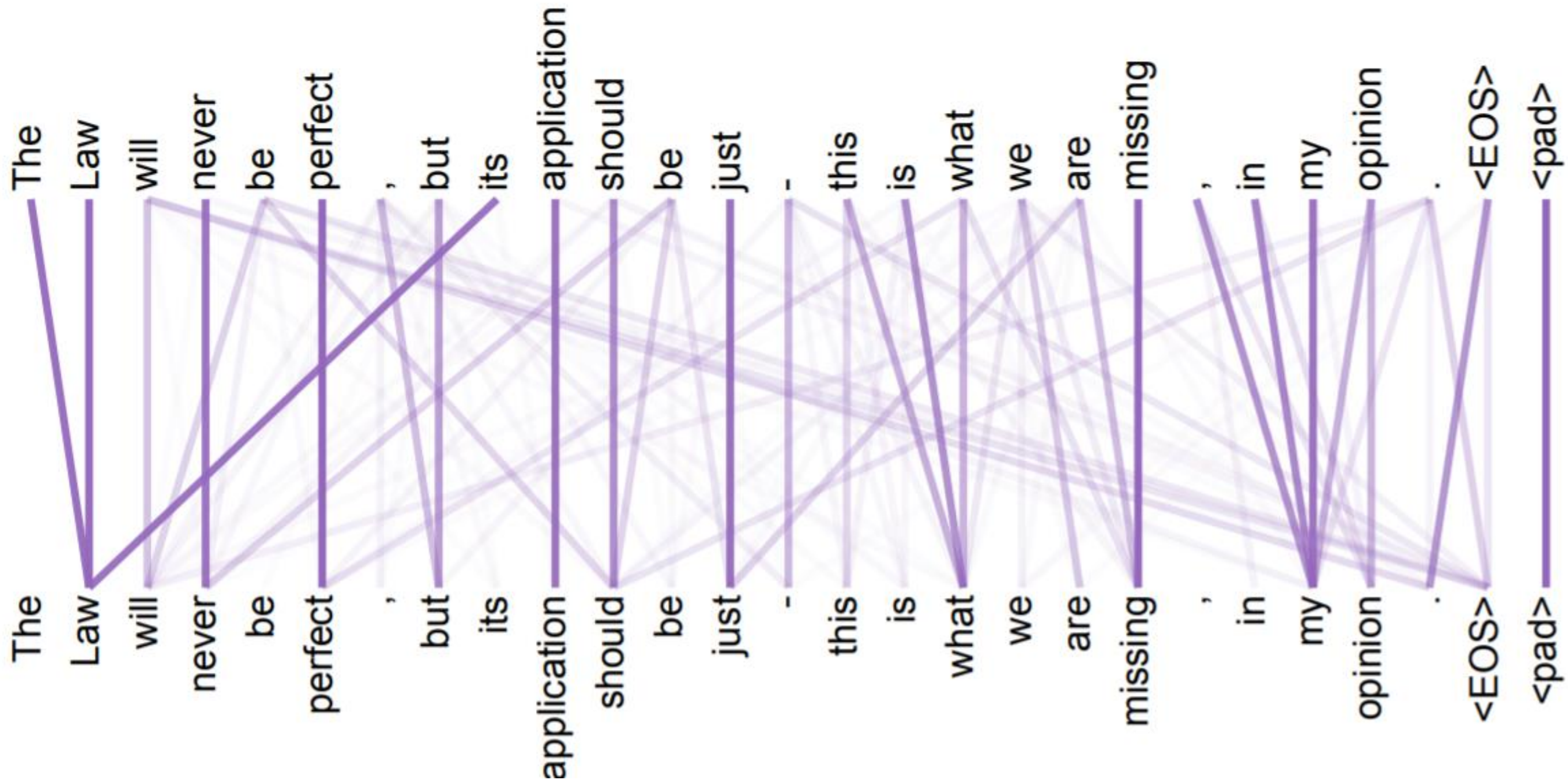


attend on the input sequence



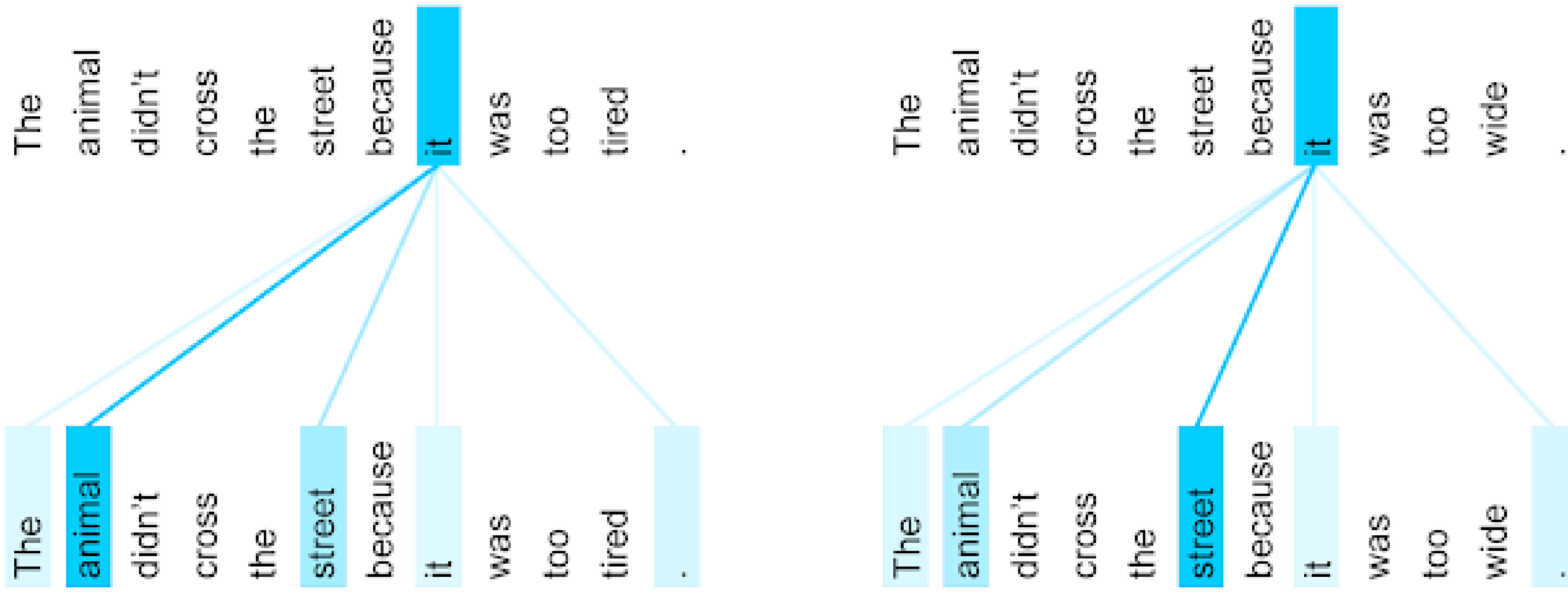
Masked: attend on the generated sequence

Attention Visualization



https://arxiv.org/abs/1706.03762

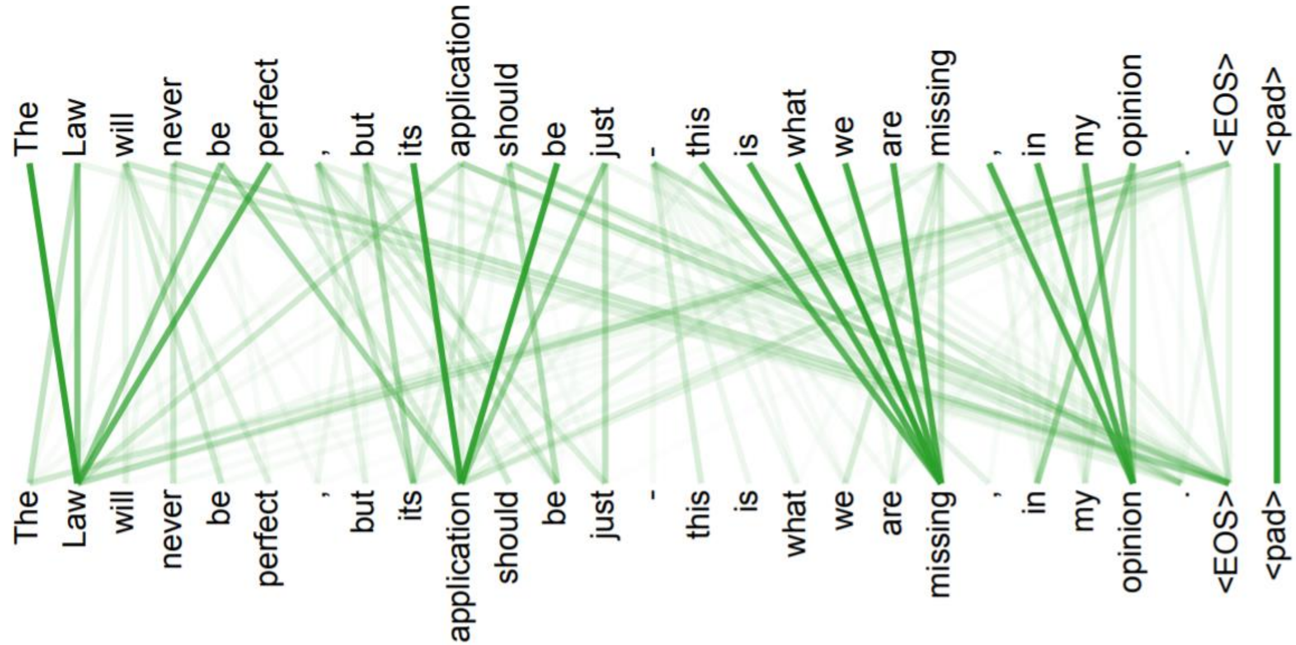
Attention Visualization

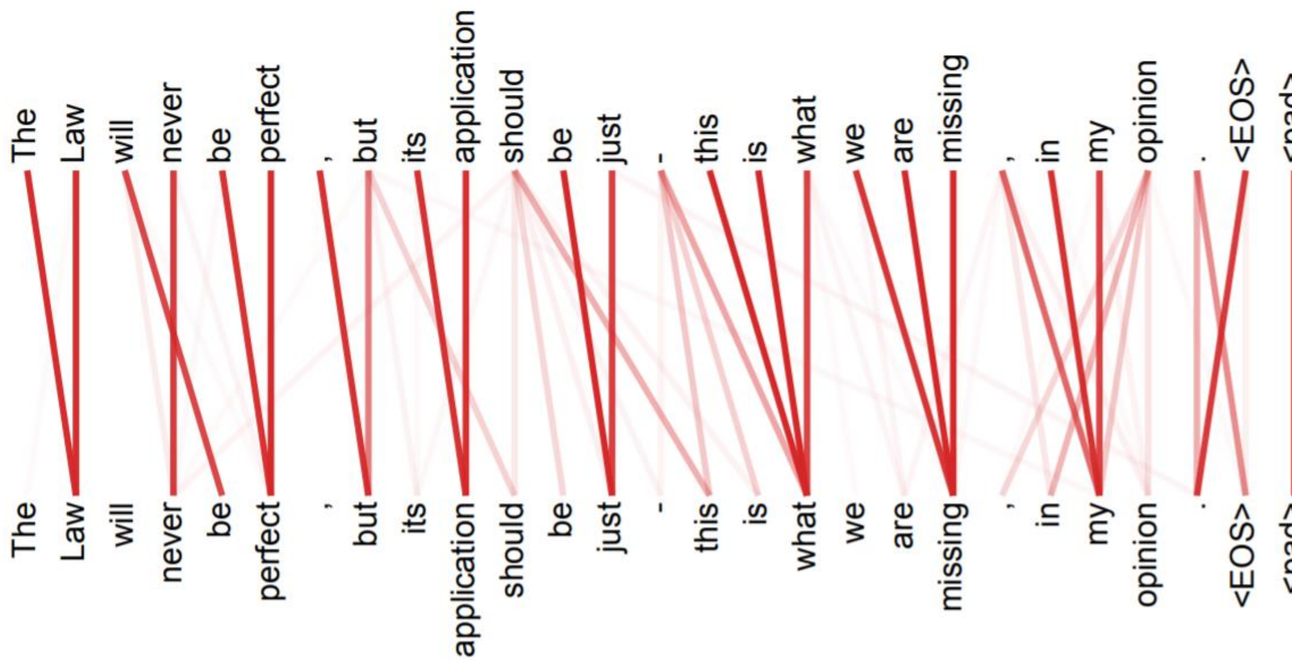


The encoder self-attention distribution for the word "it" from the 5th to the 6th layer of a Transformer trained on English to French translation (one of eight attention heads).

https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

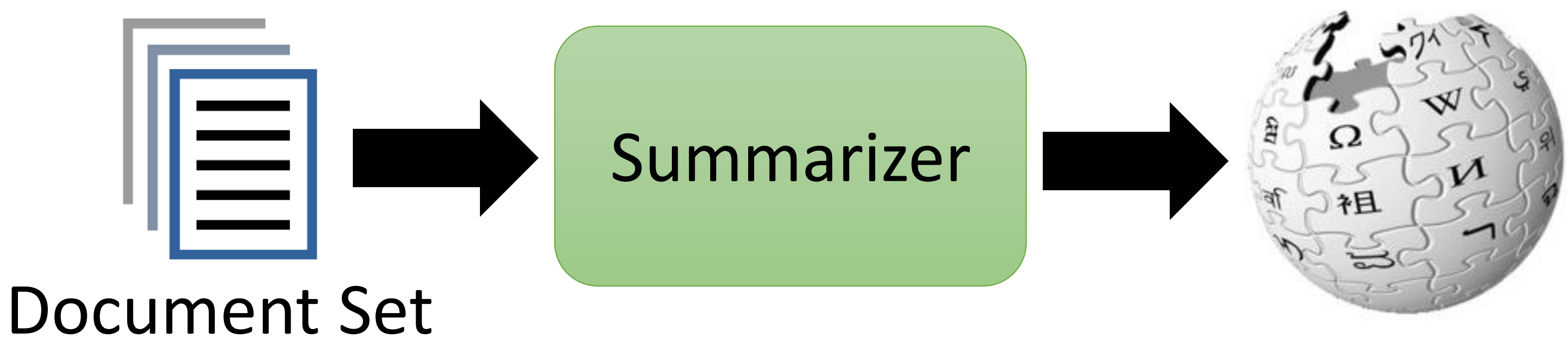
Multi-head Attention





Example Application

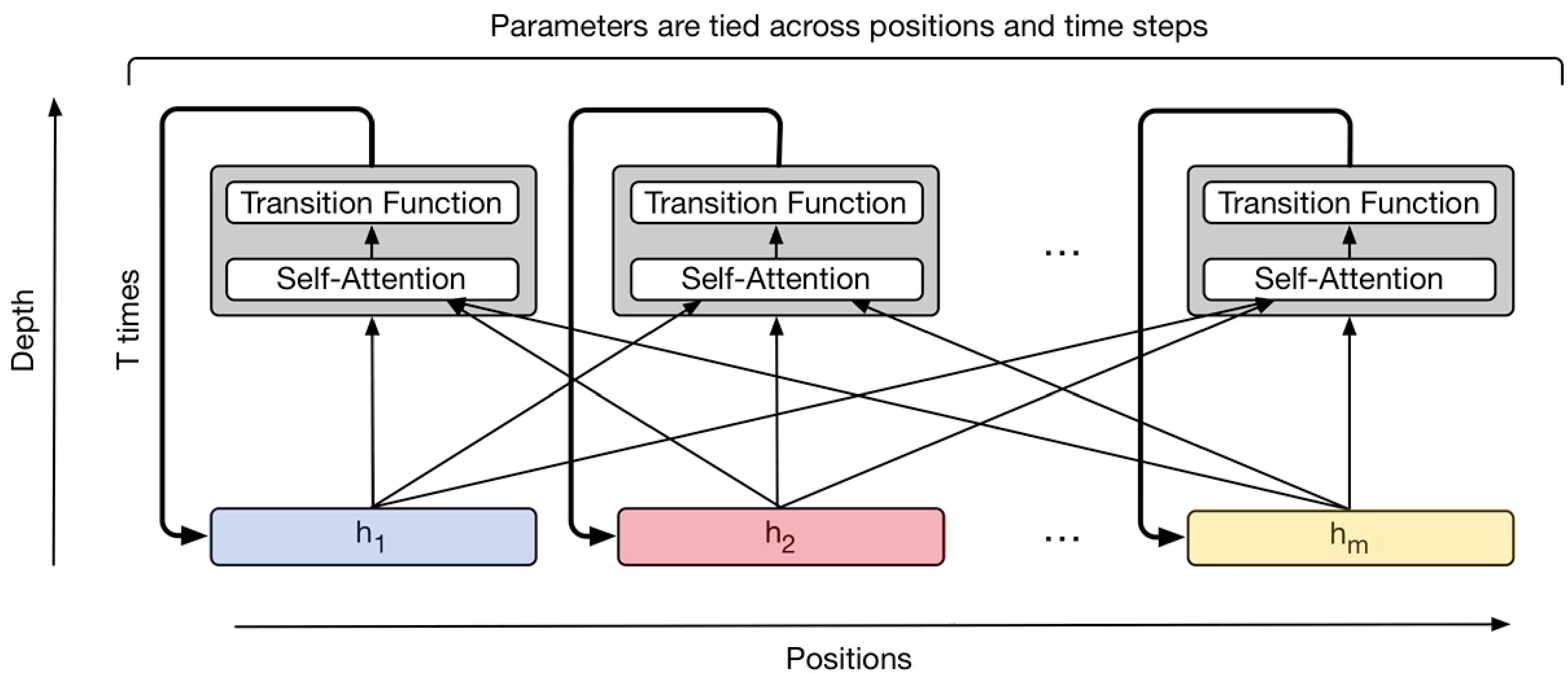
- If you can use seq2seq, you can use transformer.



Dataset	Input	Output	# examples
Gigaword (Graff & Cieri, 2003)	10^1	10^1	10^6
CNN/DailyMail (Nallapati et al., 2016)	10^2–10^3	10^1	10^5
WikiSum (ours)	10^2–10^6	10^1–10^3	10^6

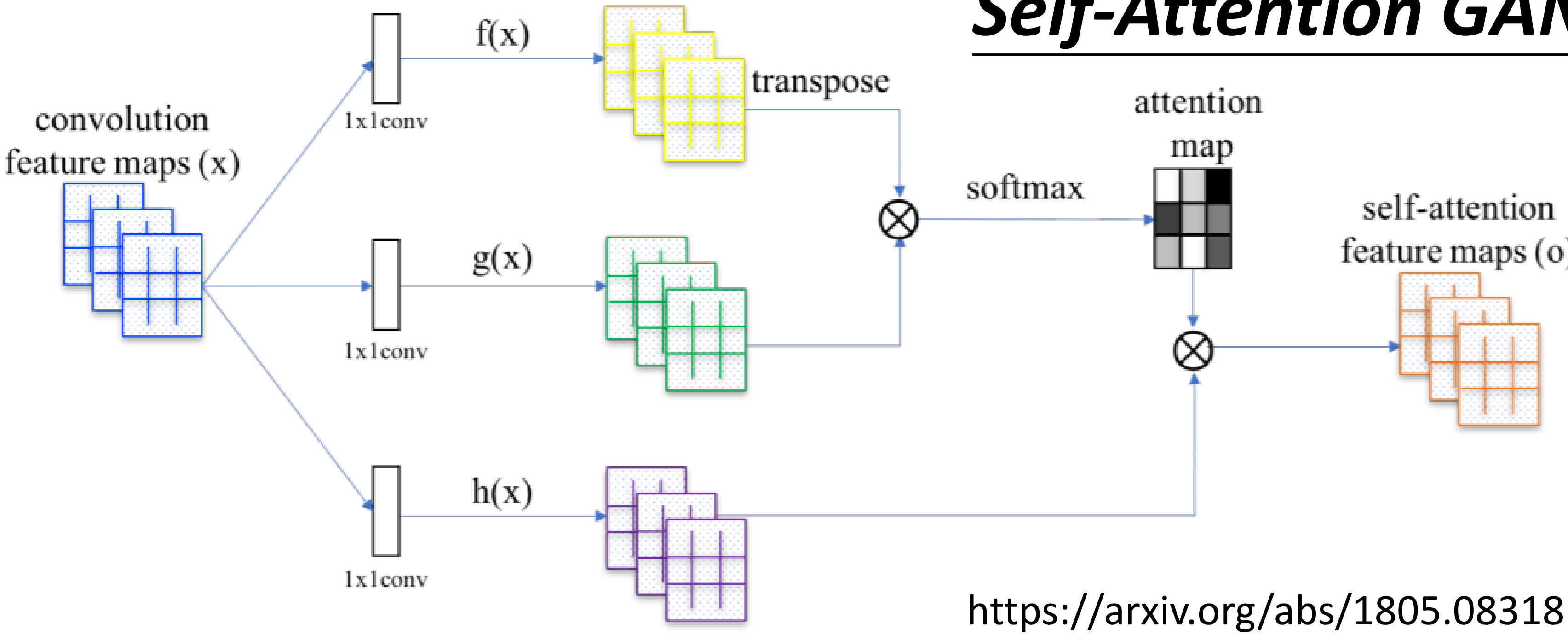
https://arxiv.org/abs/1801.10198

Universal Transformer



https://ai.googleblog.com/2018/08/moving-beyond-translation-with.html

Self-Attention GAN



https://arxiv.org/abs/1805.08318