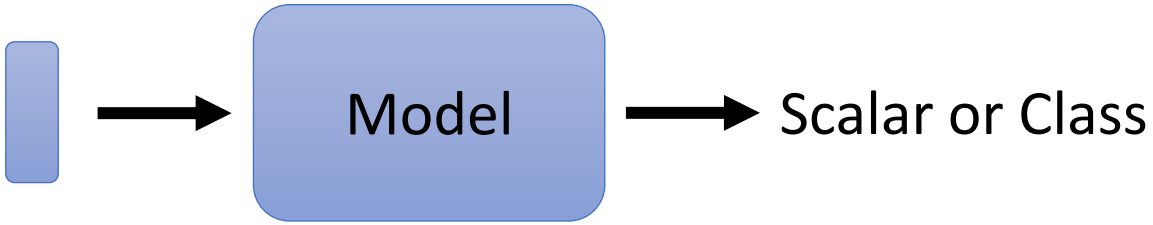
Self-attention

Hung-yi Lee

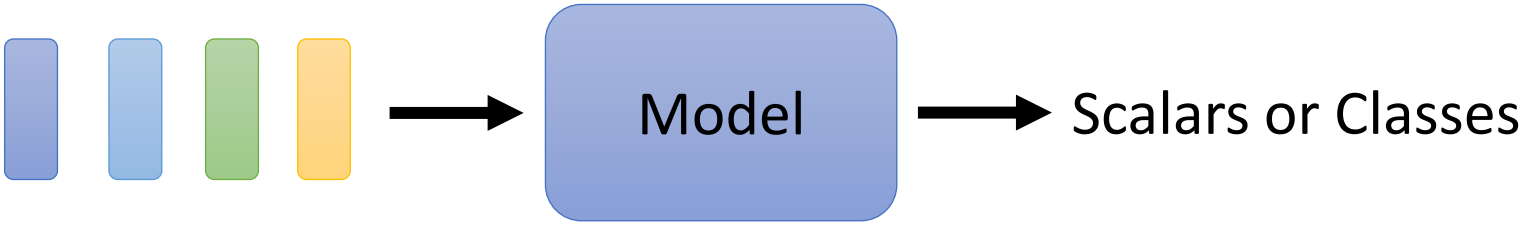
李宏毅

Sophisticated Input

- Input is **a vector**

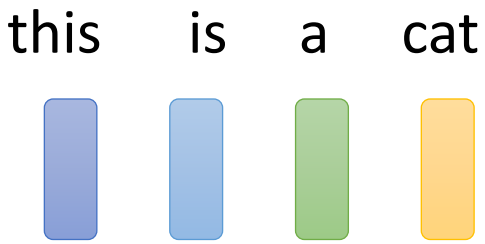


- Input is **a set of vectors**



(may change length)

Vector Set as Input



One-hot Encoding

apple = [1 0 0 0 0 ……]

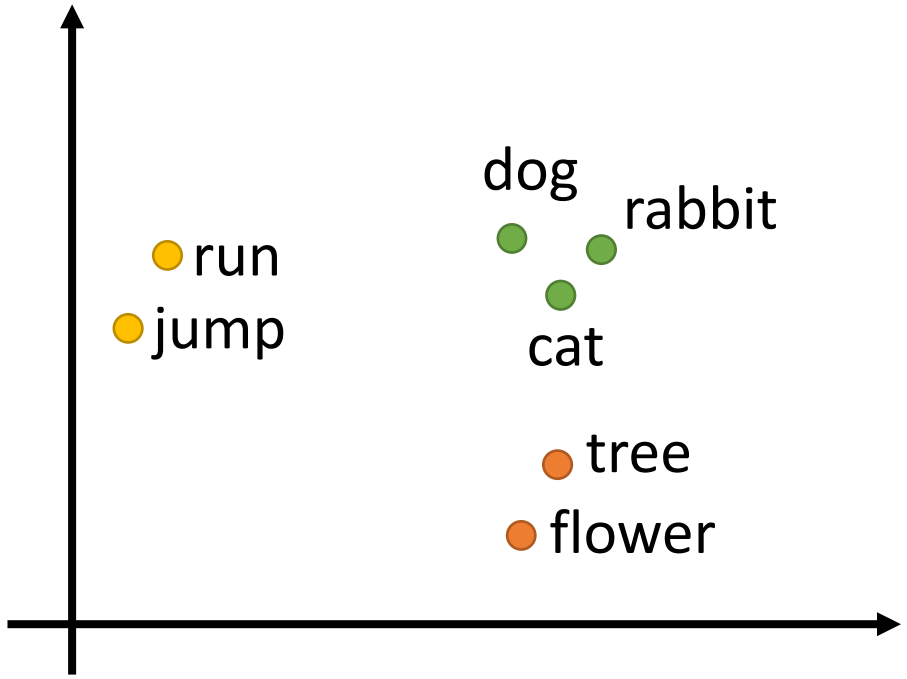
bag = [0 1 0 0 0 ……]

cat = [0 0 1 0 0 ……]

dog = [0 0 0 1 0 ……]

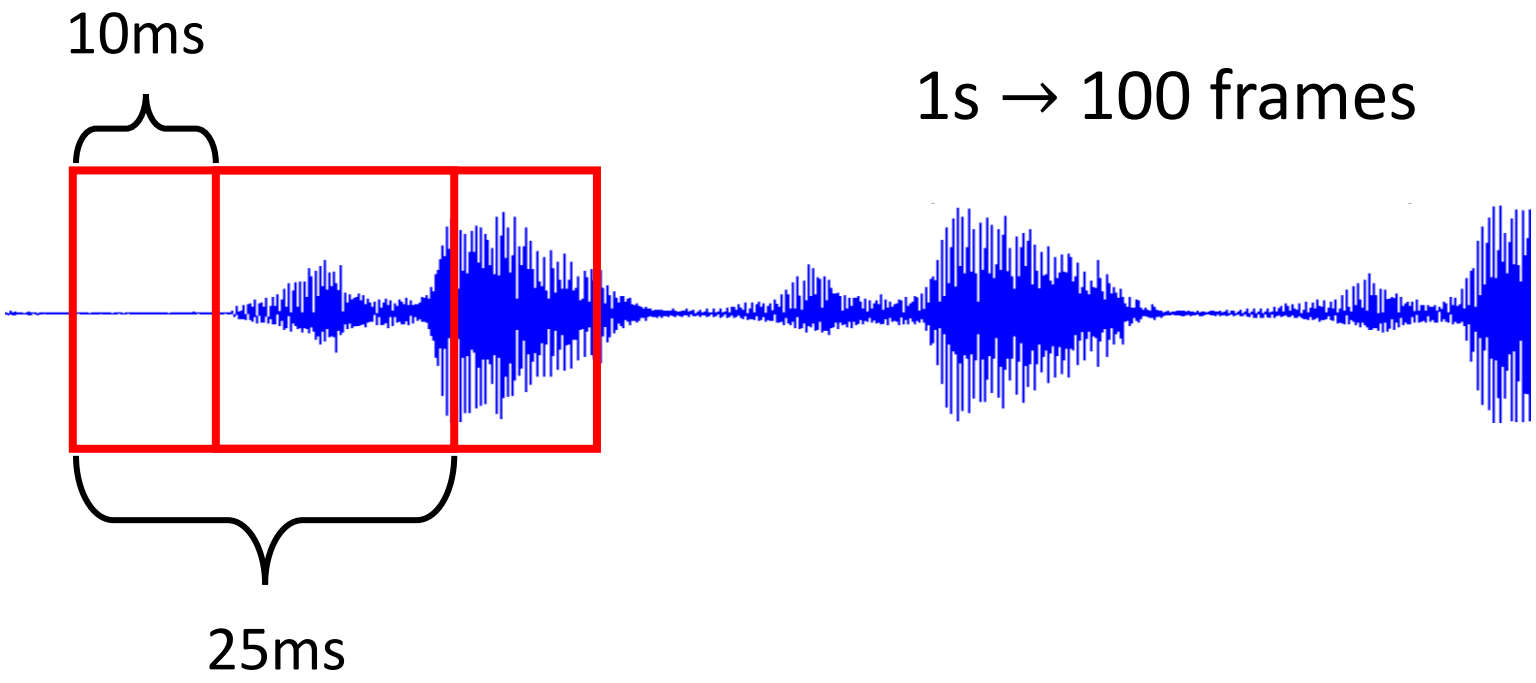
elephant = [0 0 0 0 1 ……]

Word Embedding



To learn more: https://youtu.be/X7PH3NuYW0Q (in Mandarin)

Vector Set as Input



frame

400 sample points (16KHz)

39-dim MFCC

80-dim filter bank output

https://medium.com/analytics-vidhya/social-network-analytics-f082f4e21b16

Vector Set as Input

- Graph is also a set of vectors (consider each **node** as **a vector**)

Each profile is a vector

http://www.twword.com/wiki/%E5%88%86%E5%AD%90

Vector Set as Input

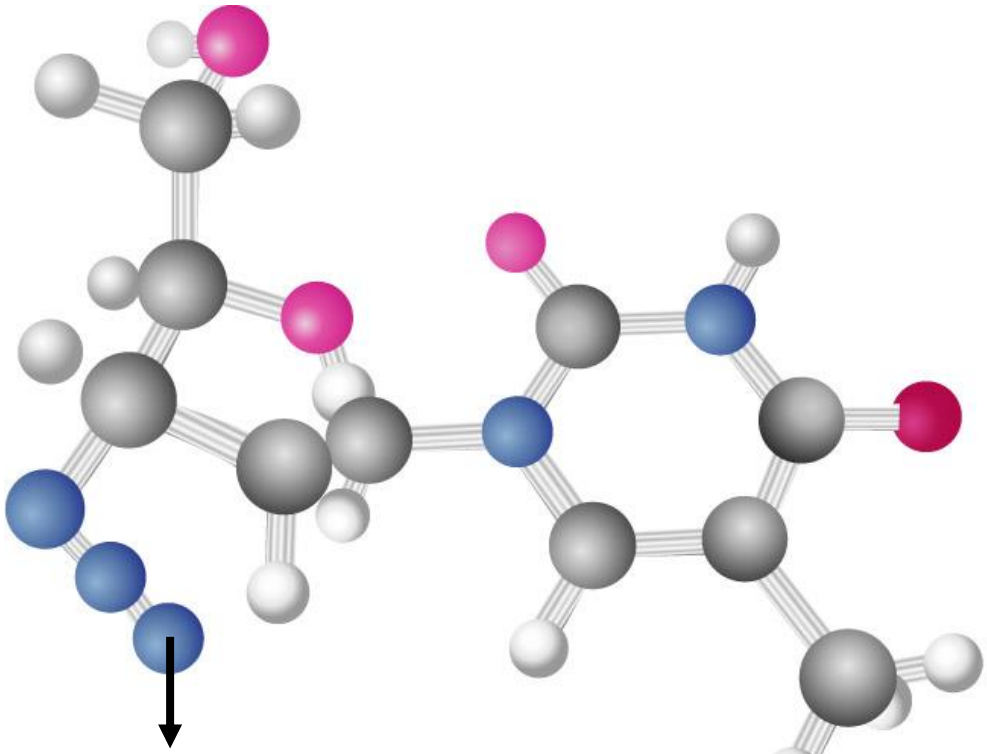
- Graph is also a set of vectors (consider each **node** as **a vector**)

H = [1 0 0 0 0 ……]

C = [0 1 0 0 0 ……]

O = [0 0 1 0 0 ……]

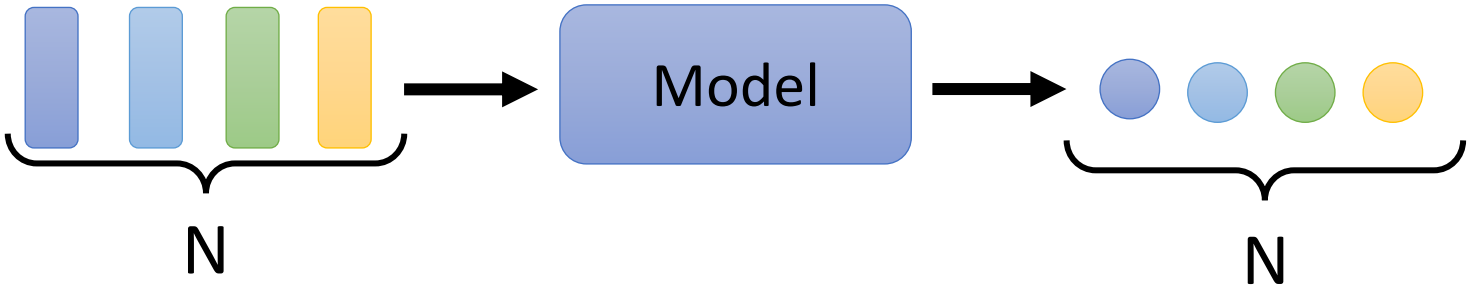
⋮



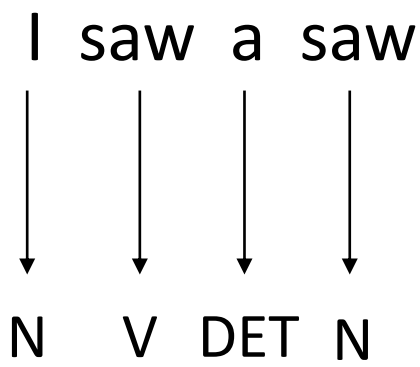
One-hot vector

What is the output?

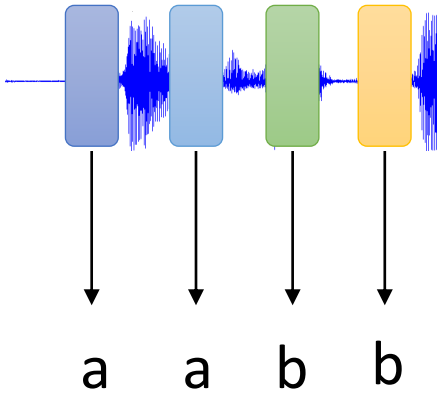
- Each vector has a label.



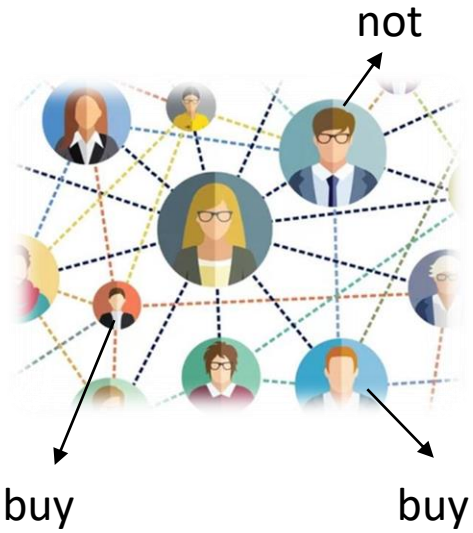
Example Applications



POS tagging

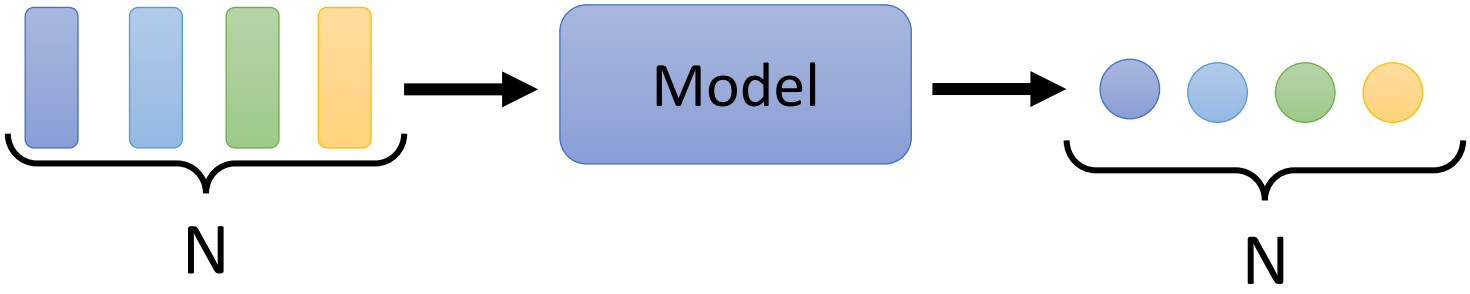


HW2

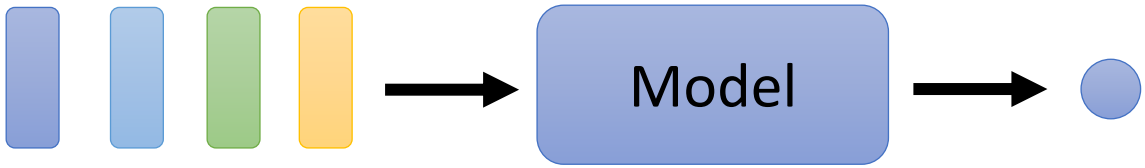


What is the output?

- Each vector has a label.



- The whole sequence has a label.

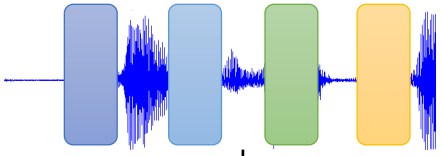


Example Applications

this is good

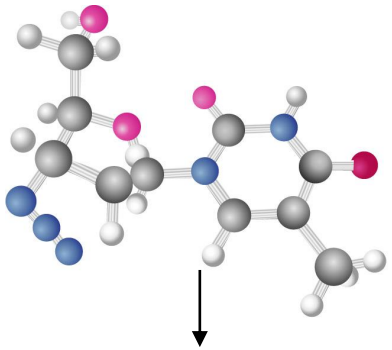
Sentiment analysis

positive



HW4

speaker

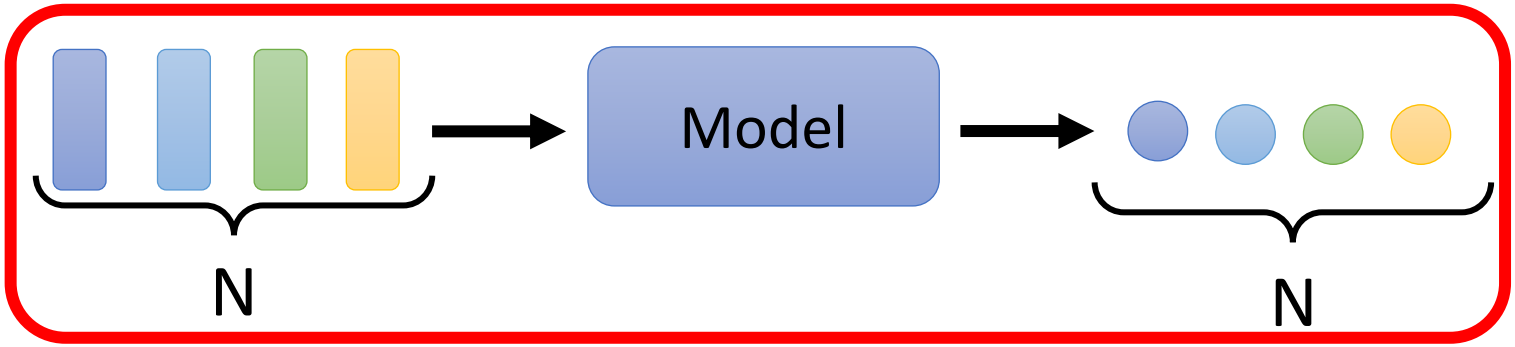


hydrophilicity

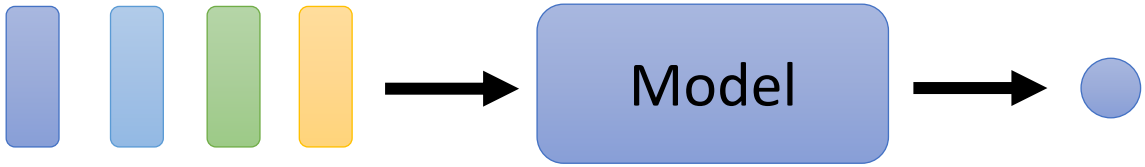
What is the output?

- Each vector has a label.

focus of this lecture

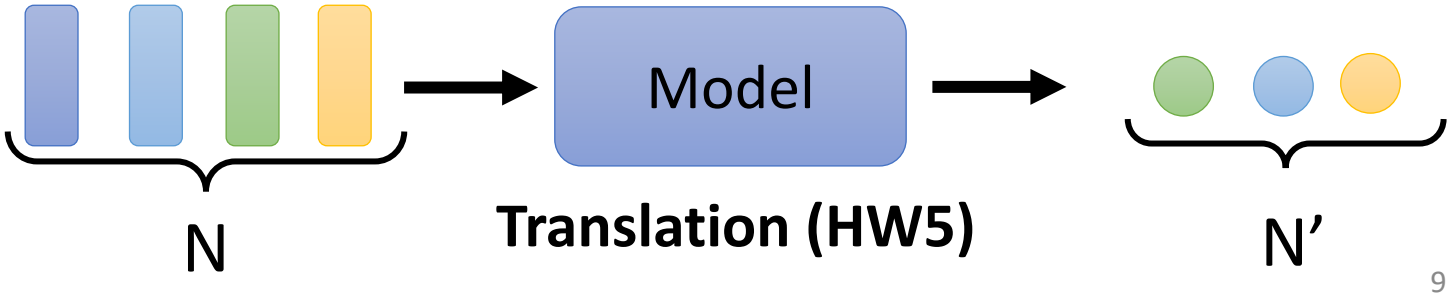


- The whole sequence has a label.



- Model decides the number of labels itself.

seq2seq



Sequence Labeling

FC

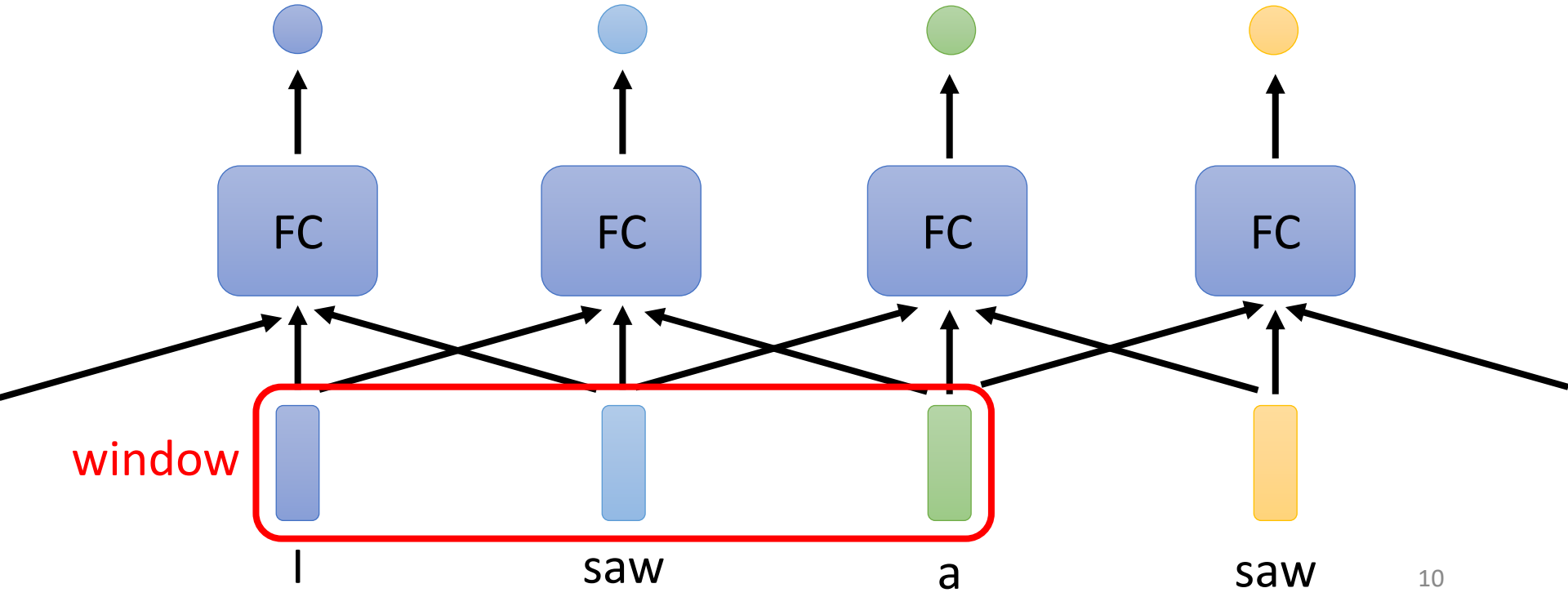
Fully-connected

Is it possible to consider the context?

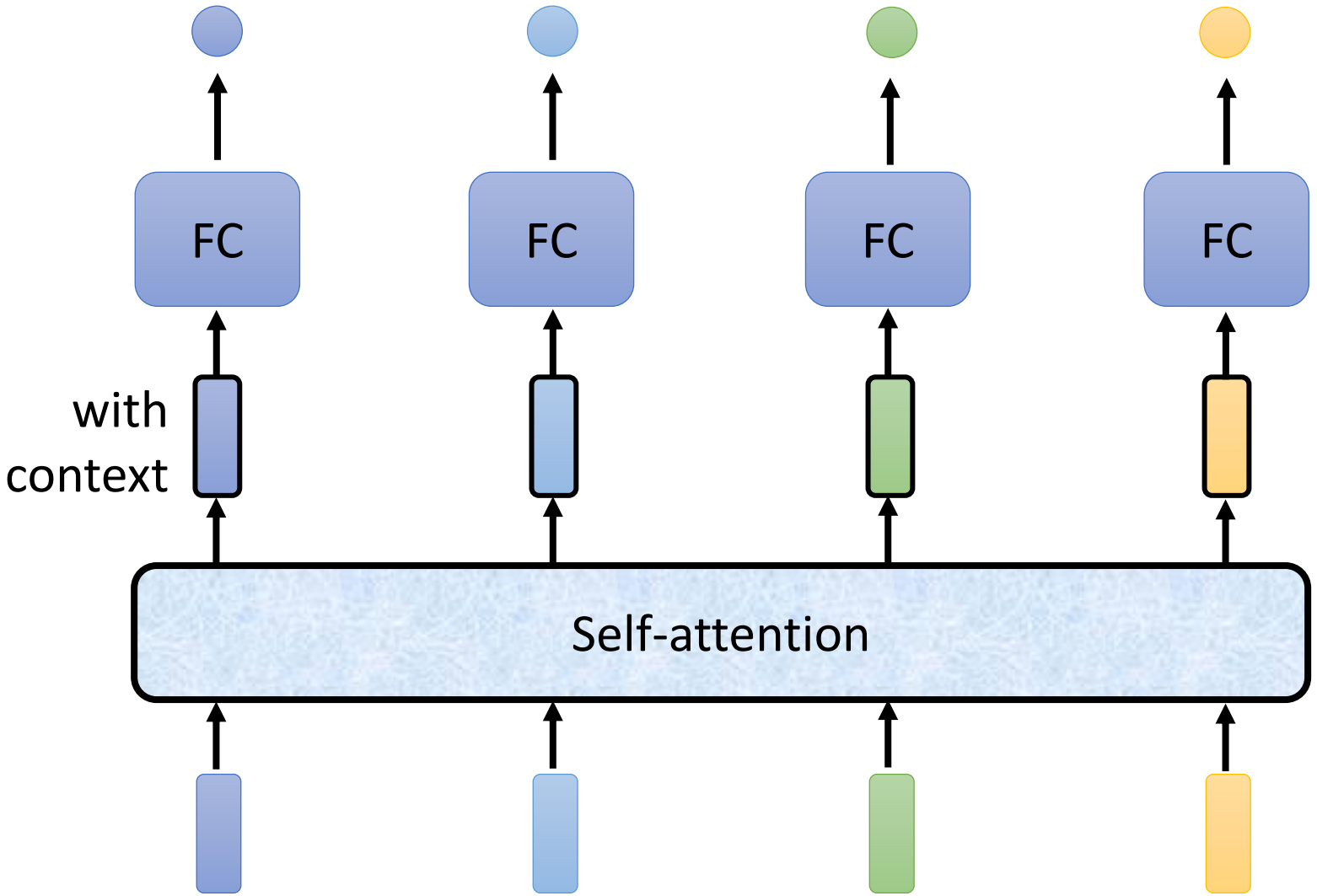
FC can consider the neighbor

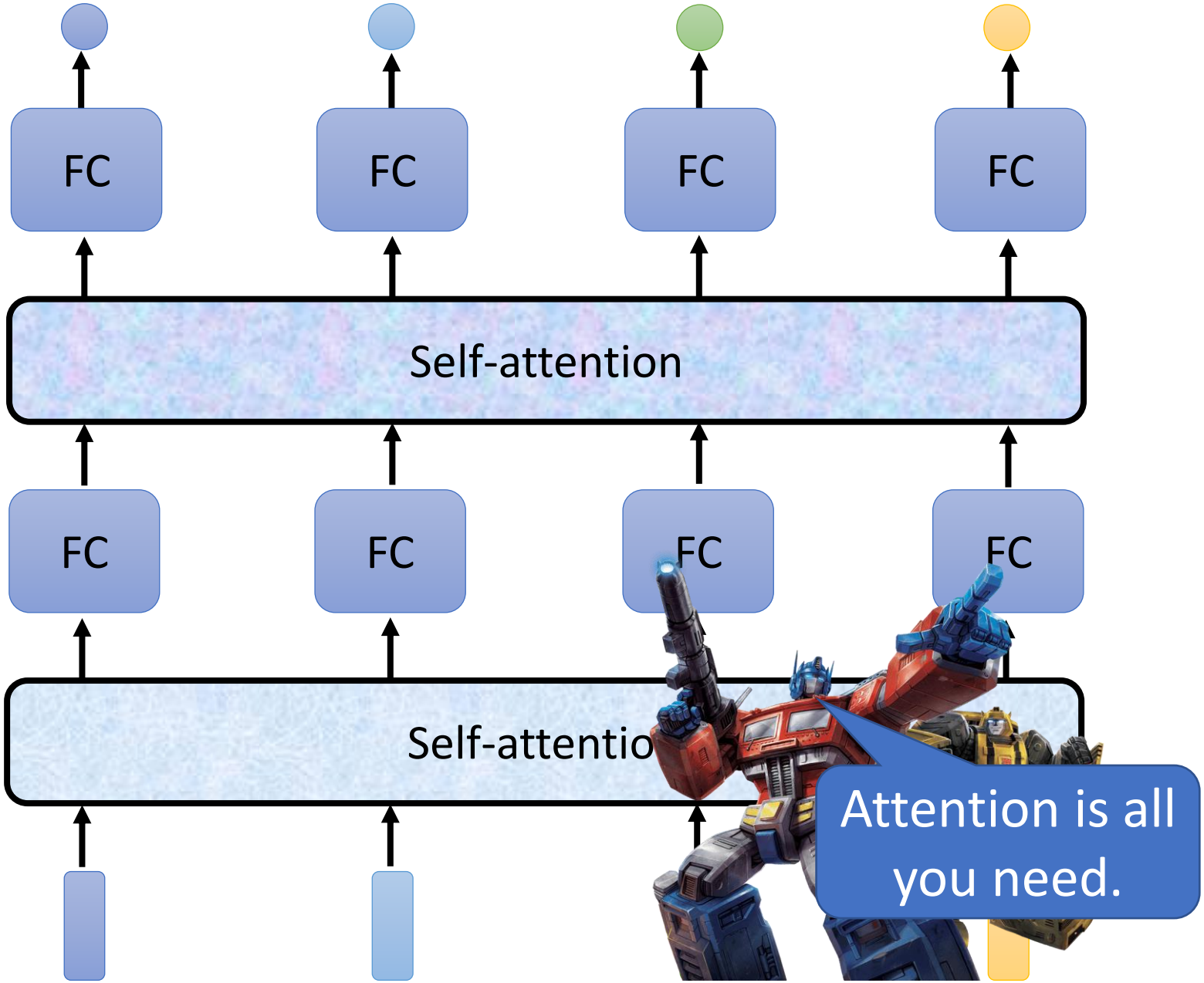
How to consider the whole sequence?

a window covers the whole sequence?



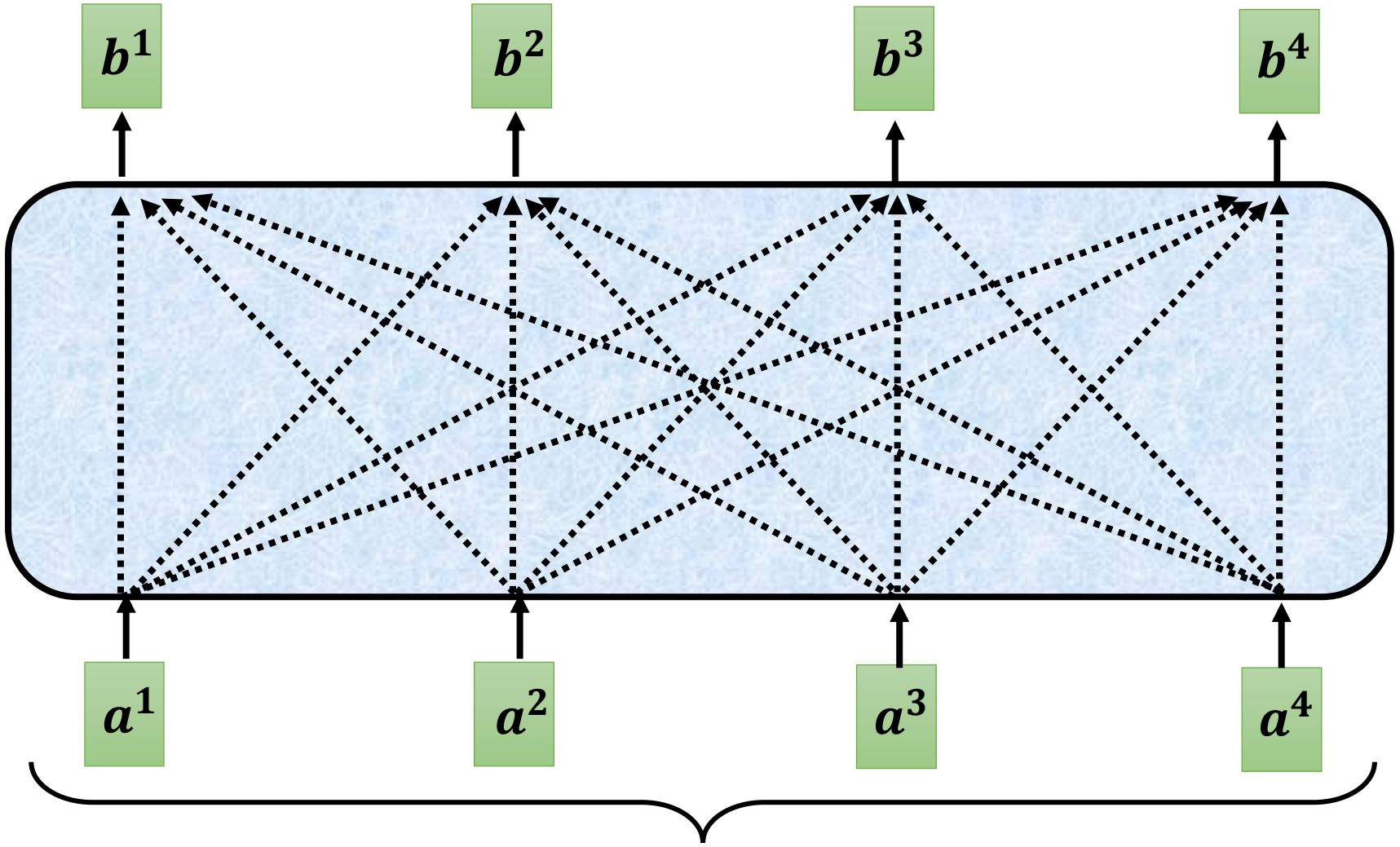
Self-attention





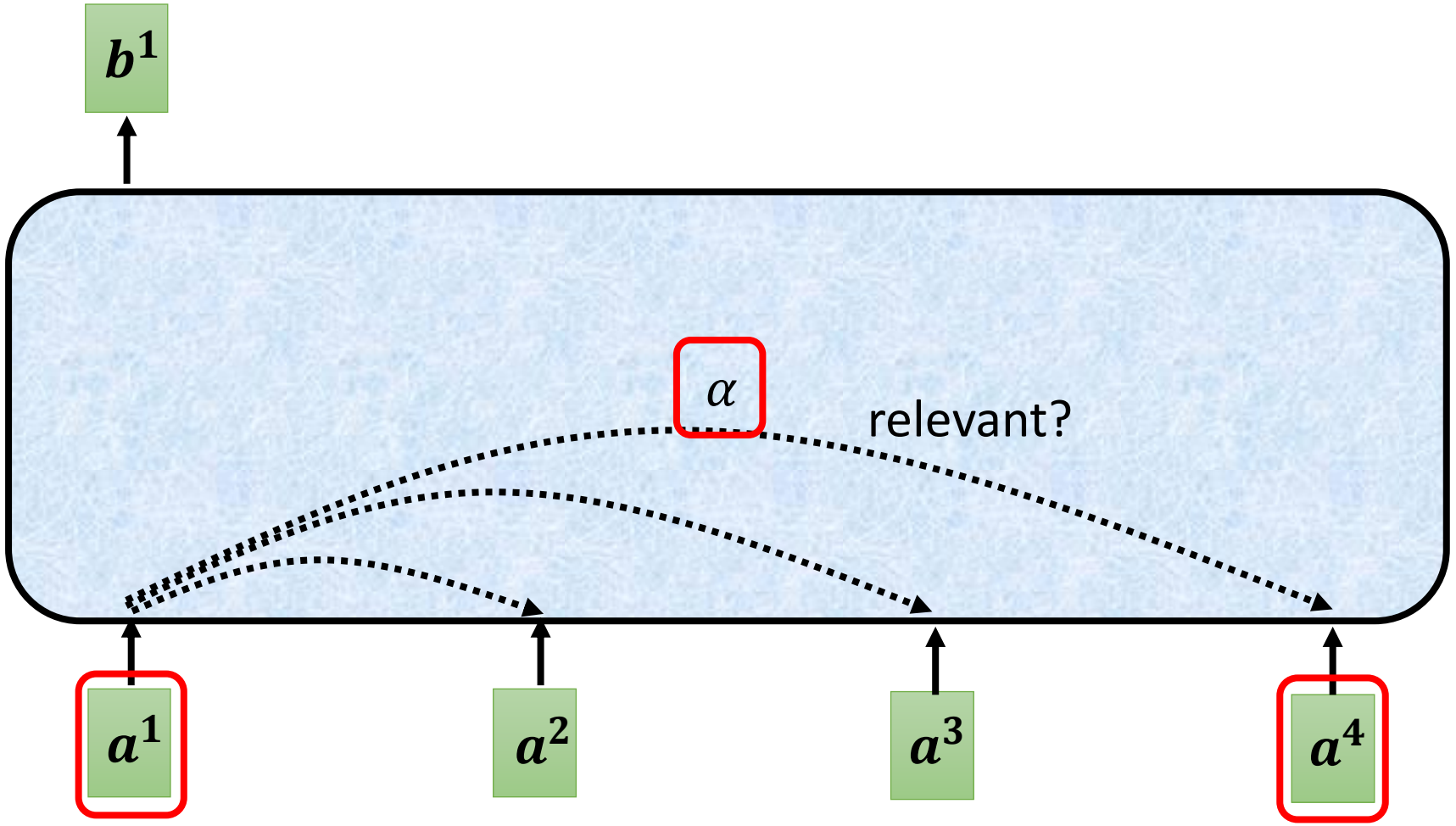
https://arxiv.org/abs/1706.03762

Self-attention



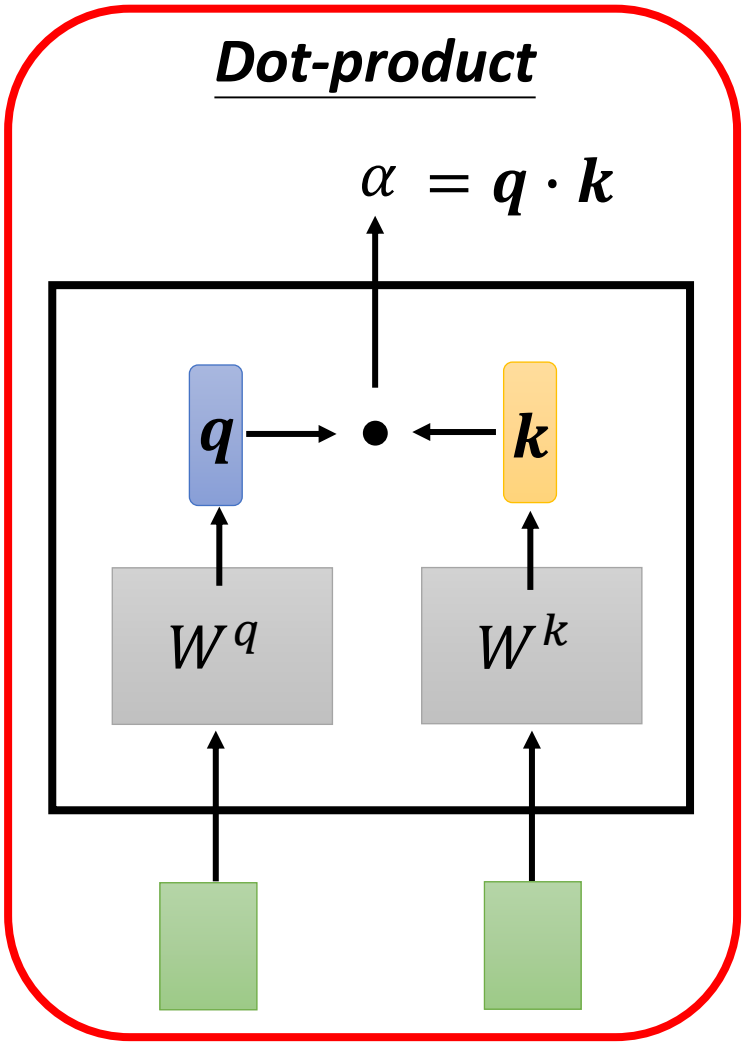
Can be either **input** or **a hidden layer**

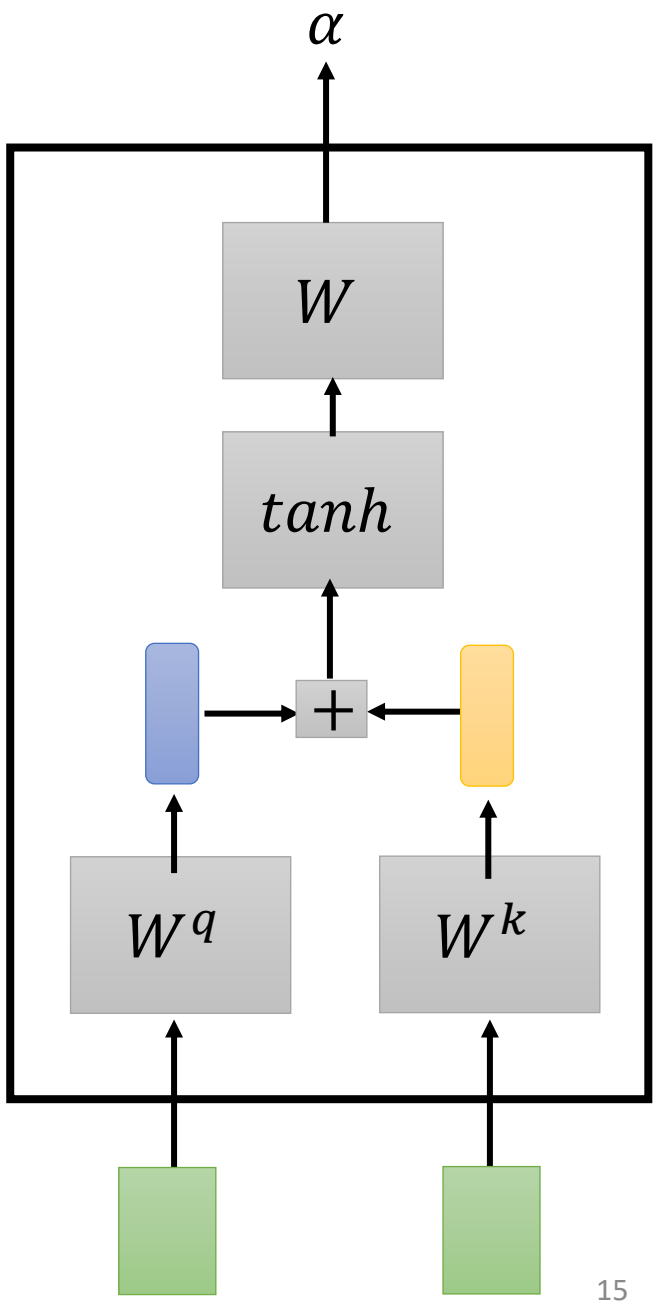
Self-attention



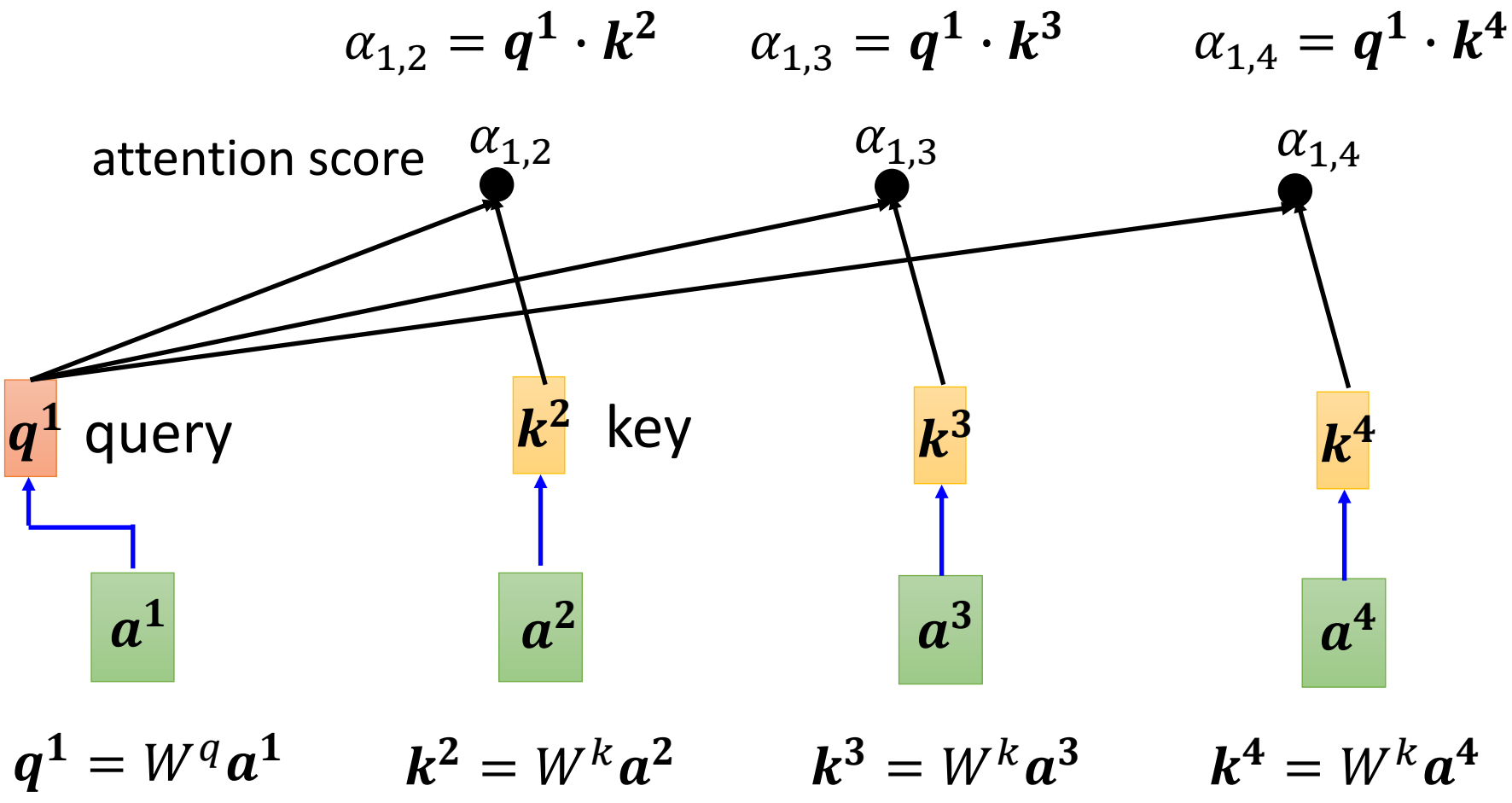
Find the relevant vectors in a sequence

Self-attention



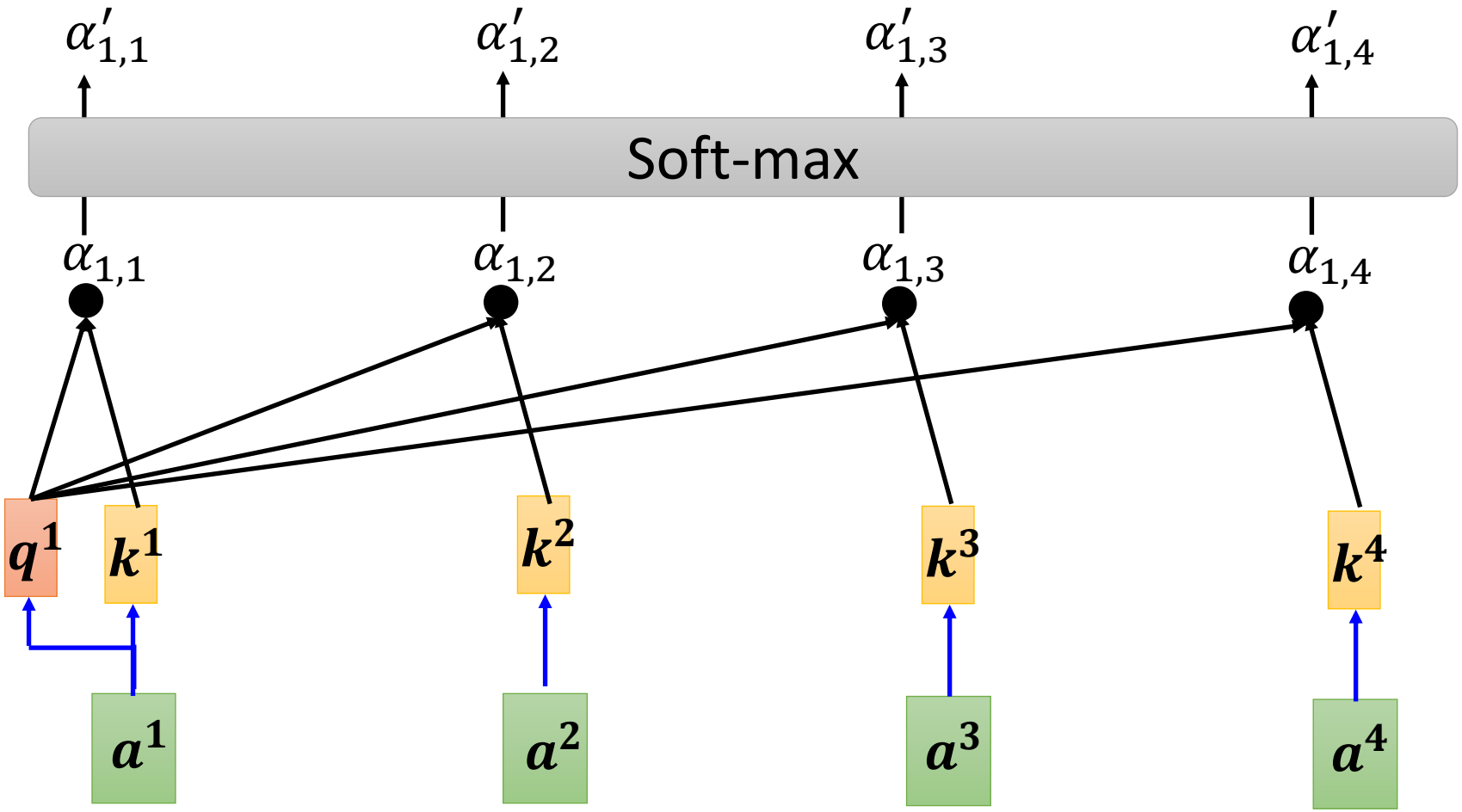


Self-attention



Self-attention

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



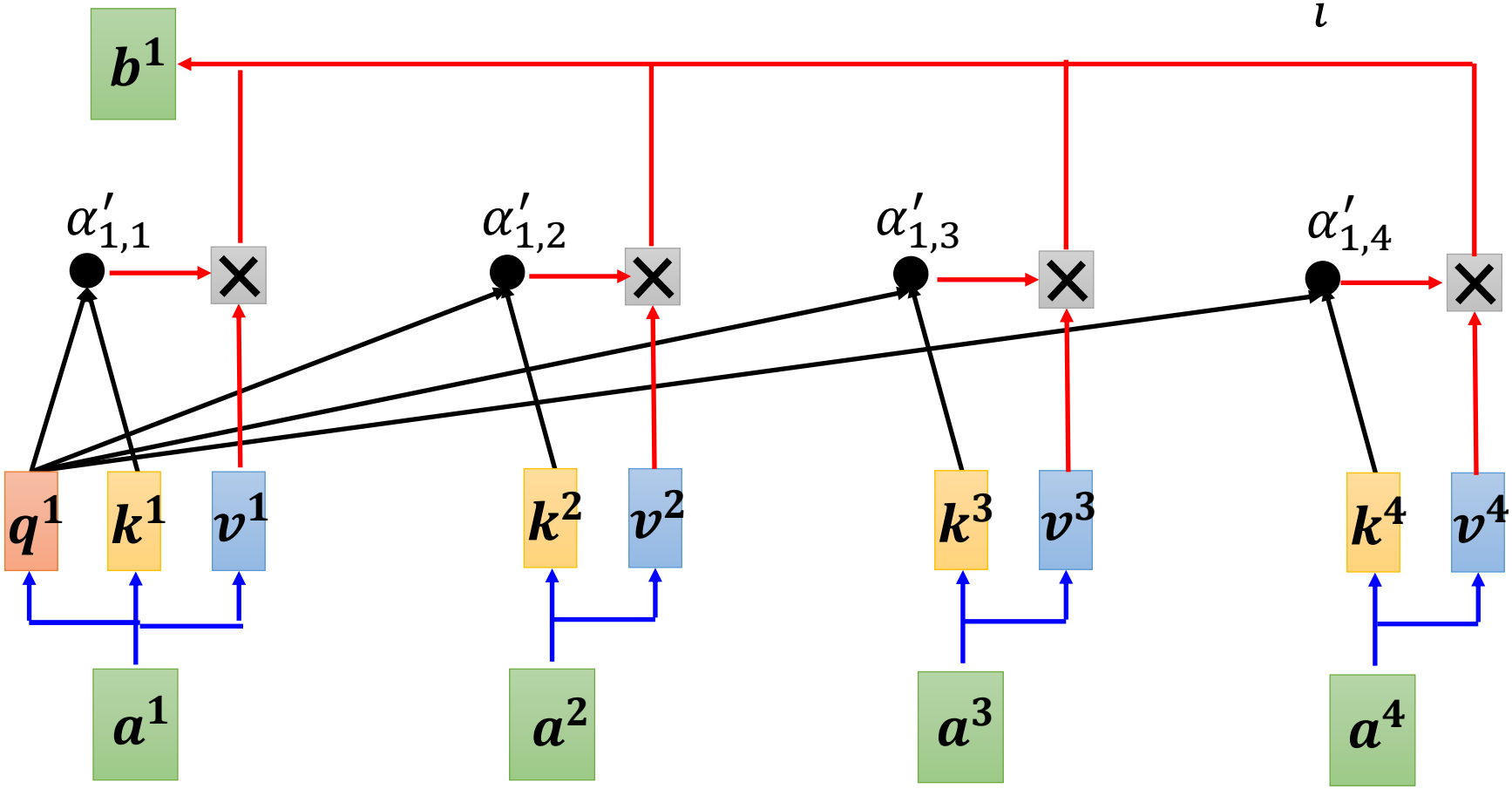
$q^1 = W^q a^1$ $k^2 = W^k a^2$ $k^3 = W^k a^3$ $k^4 = W^k a^4$

$k^1 = W^k a^1$

Self-attention

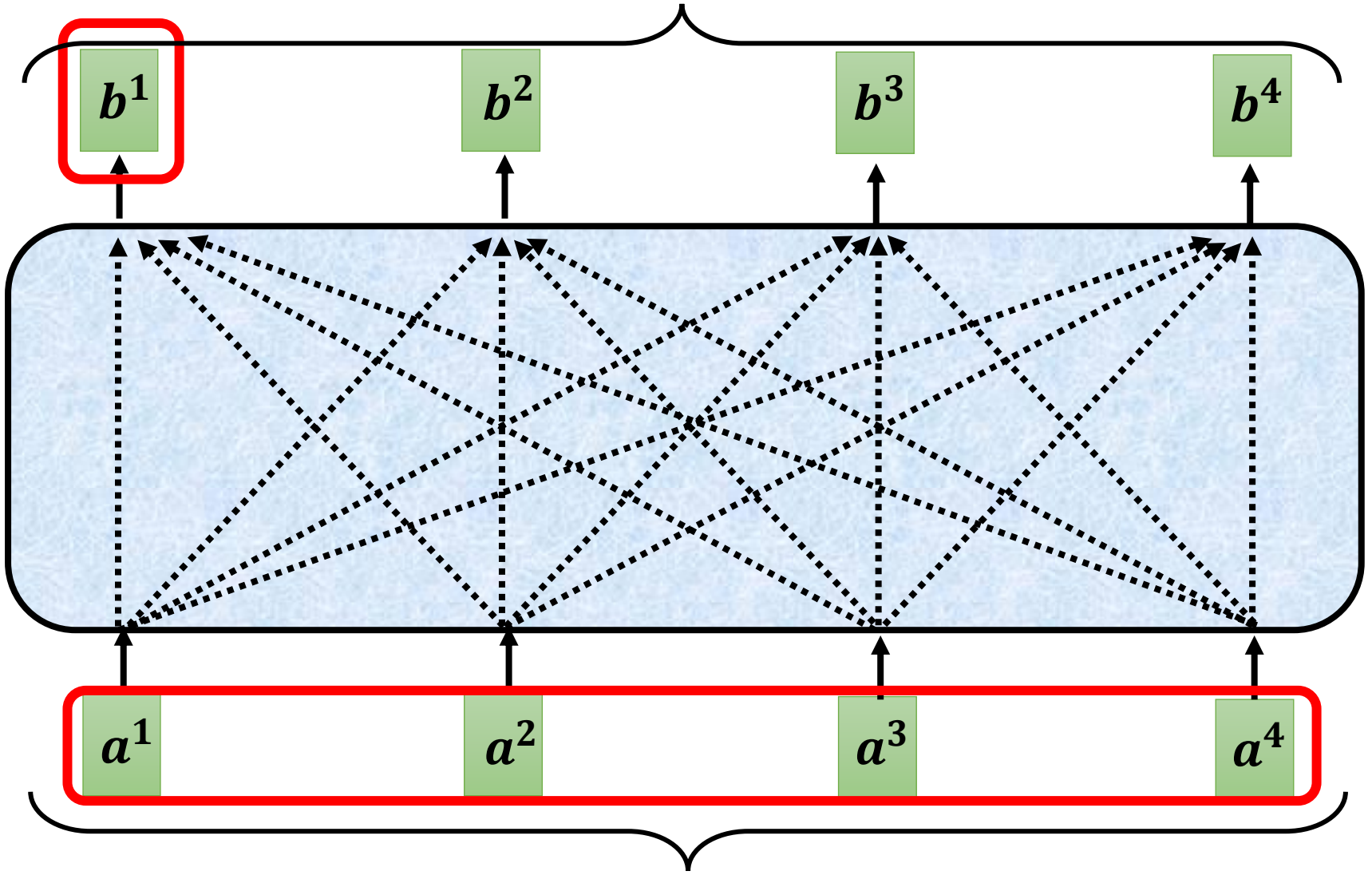
Extract information based on attention scores

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



$v^1 = W^v a^1$ $v^2 = W^v a^2$ $v^3 = W^v a^3$ $v^4 = W^v a^4$

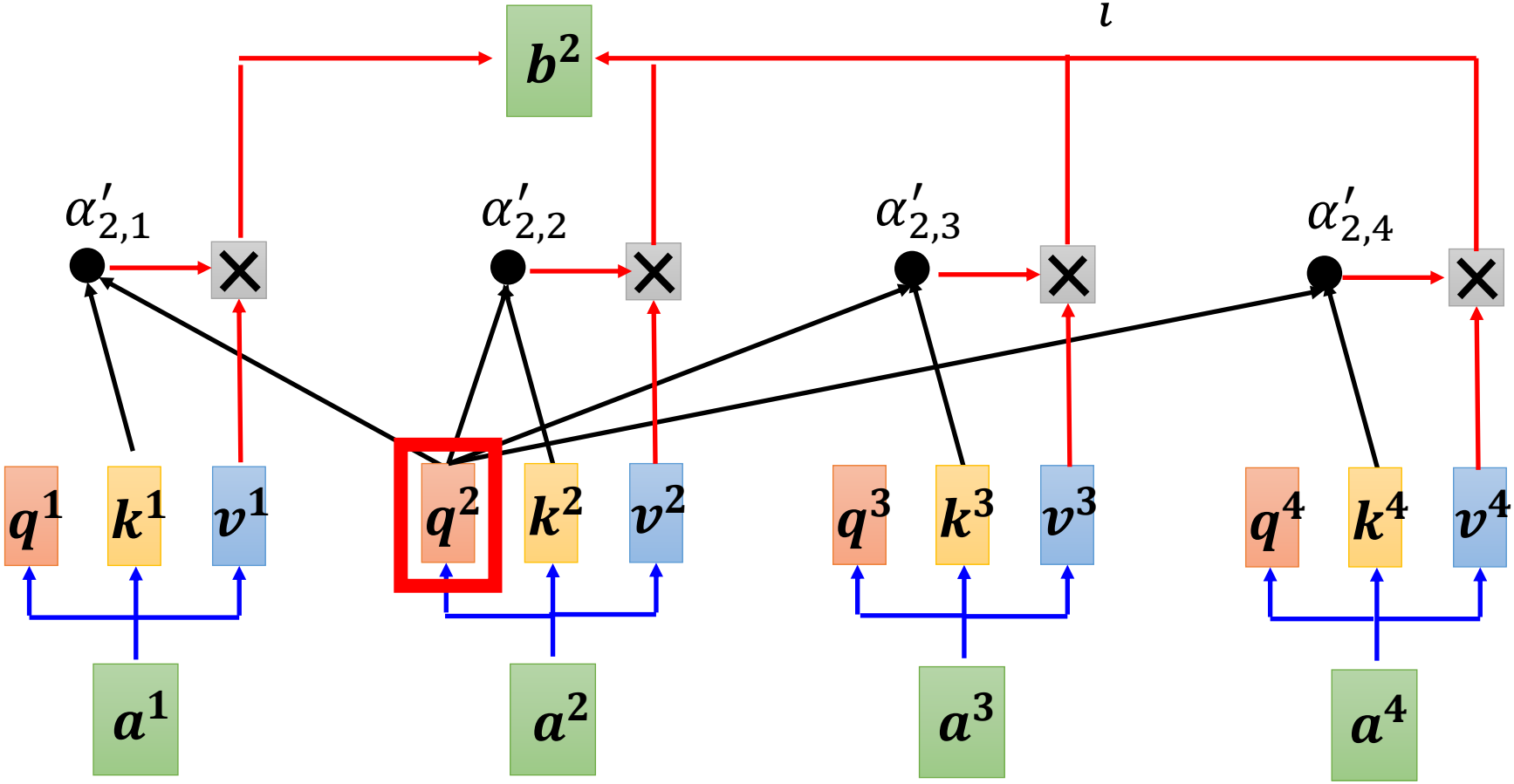
Self-attention



Can be either **input** or **a hidden layer**

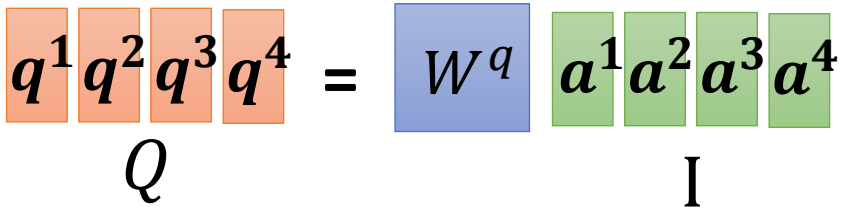
Self-attention

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

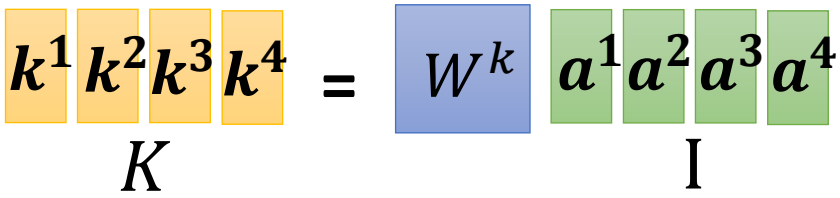


Self-attention

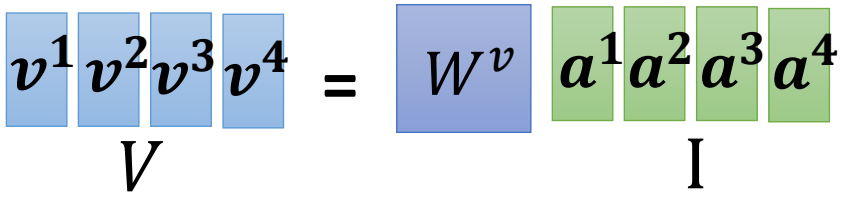
$$q^i = W^q a^i$$

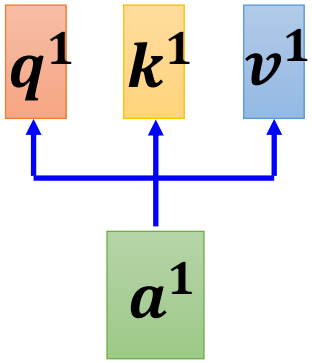


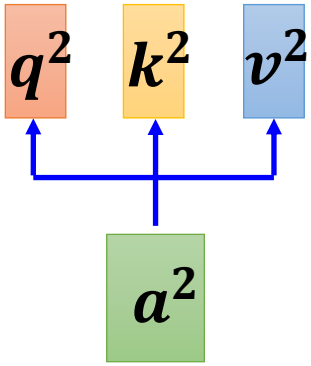
$$k^i = W^k a^i$$

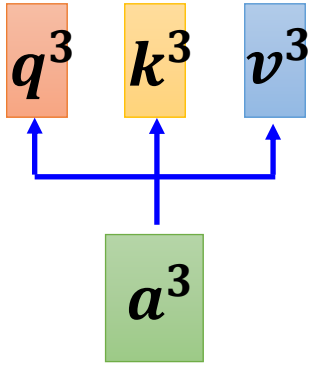


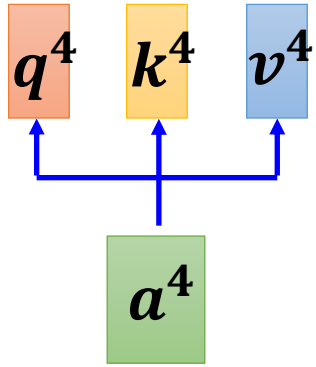
$$v^i = W^v a^i$$









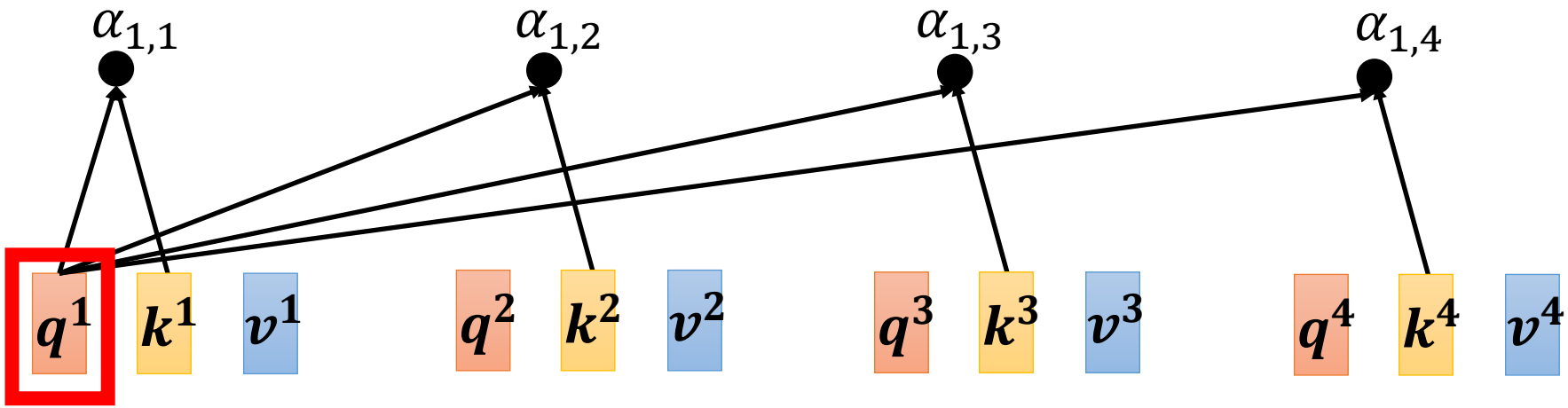


Self-attention

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

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

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

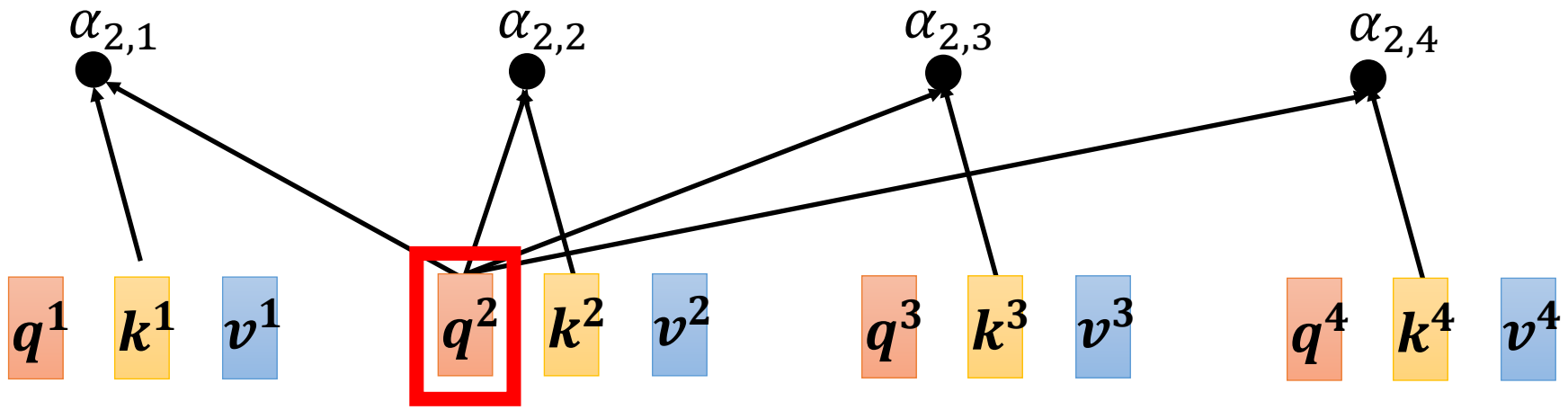


Self-attention

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

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

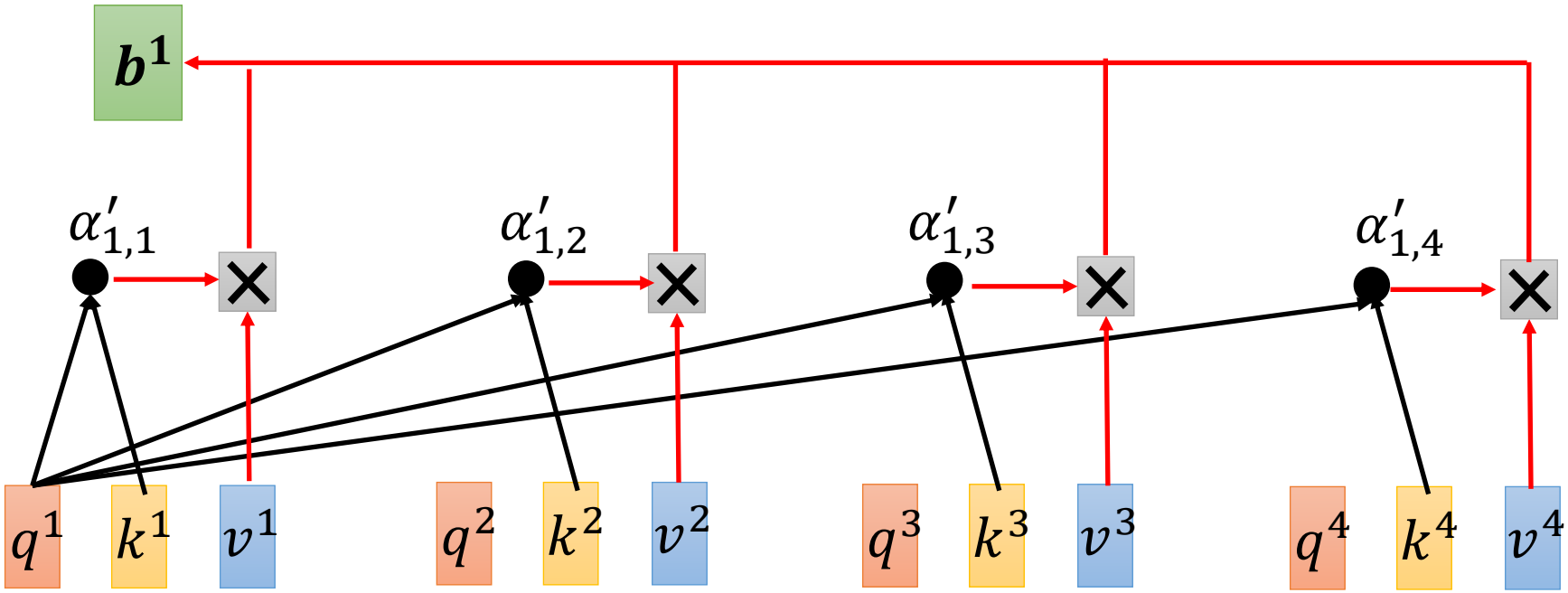
$$\begin{bmatrix}\alpha_{1,1}\\ \alpha_{1,2}\\ \alpha_{1,3}\\ \alpha_{1,4}\end{bmatrix} = \begin{bmatrix}\boldsymbol{k^1}\\ \boldsymbol{k^2}\\ \boldsymbol{k^3}\\ \boldsymbol{k^4}\end{bmatrix} \boldsymbol{q^1}$$



$$\begin{bmatrix}\alpha'_{1,1} & \alpha'_{2,1} & \alpha'_{3,1} & \alpha'_{4,1}\\ \alpha'_{1,2} & \alpha'_{2,2} & \alpha'_{3,2} & \alpha'_{4,2}\\ \alpha'_{1,3} & \alpha'_{2,3} & \alpha'_{3,3} & \alpha'_{4,3}\\ \alpha'_{1,4} & \alpha'_{2,4} & \alpha'_{3,4} & \alpha'_{4,4}\end{bmatrix} = A' \quad \text{softmax} \quad \begin{bmatrix}\alpha_{1,1} & \alpha_{2,1} & \alpha_{3,1} & \alpha_{4,1}\\ \alpha_{1,2} & \alpha_{2,2} & \alpha_{3,2} & \alpha_{4,2}\\ \alpha_{1,3} & \alpha_{2,3} & \alpha_{3,3} & \alpha_{4,3}\\ \alpha_{1,4} & \alpha_{2,4} & \alpha_{3,4} & \alpha_{4,4}\end{bmatrix} = A$$

$$\begin{bmatrix}\boldsymbol{k^1}\\ \boldsymbol{k^2}\\ \boldsymbol{k^3}\\ \boldsymbol{k^4}\end{bmatrix} = K^T \qquad \begin{bmatrix}\boldsymbol{q^1} & \boldsymbol{q^2} & \boldsymbol{q^3} & \boldsymbol{q^4}\end{bmatrix} = Q$$

Self-attention

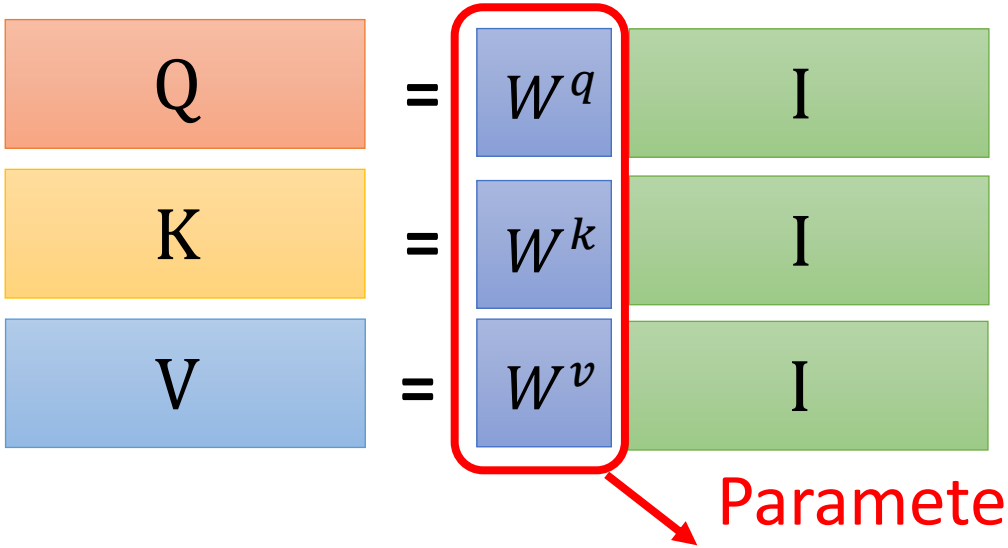


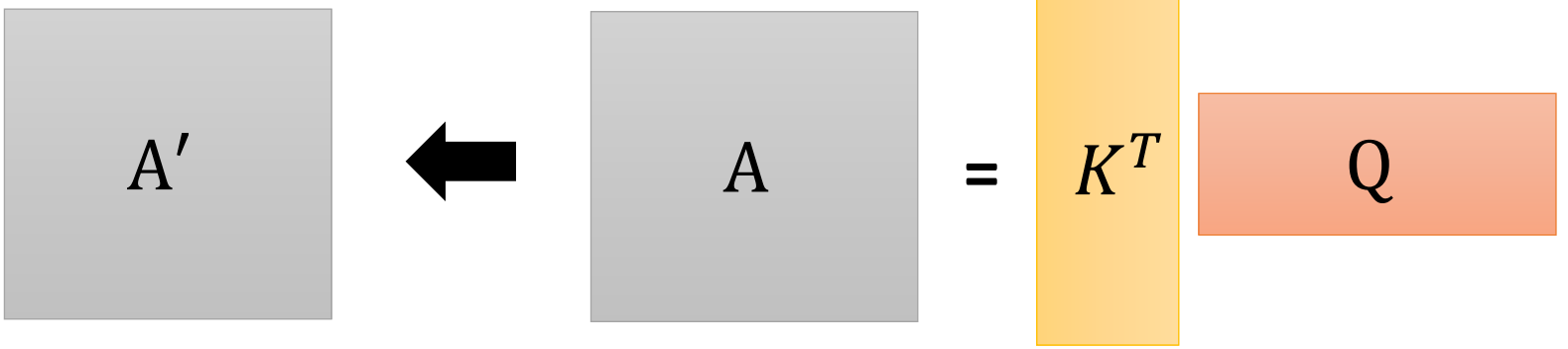
$b^1 b^2 b^3 b^4$ = $v^1 v^2 v^3 v^4$

O V

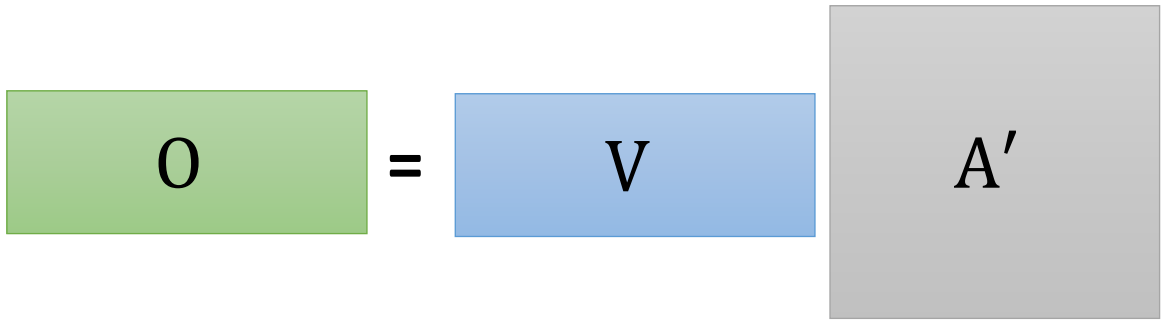
$$A' = \begin{bmatrix} \alpha'_{1,1} & \alpha'_{2,1} & \alpha'_{3,1} & \alpha'_{4,1} \\ \alpha'_{1,2} & \alpha'_{2,2} & \alpha'_{3,2} & \alpha'_{4,2} \\ \alpha'_{1,3} & \alpha'_{2,3} & \alpha'_{3,3} & \alpha'_{4,3} \\ \alpha'_{1,4} & \alpha'_{2,4} & \alpha'_{3,4} & \alpha'_{4,4} \end{bmatrix}$$

Self-attention





Attention Matrix

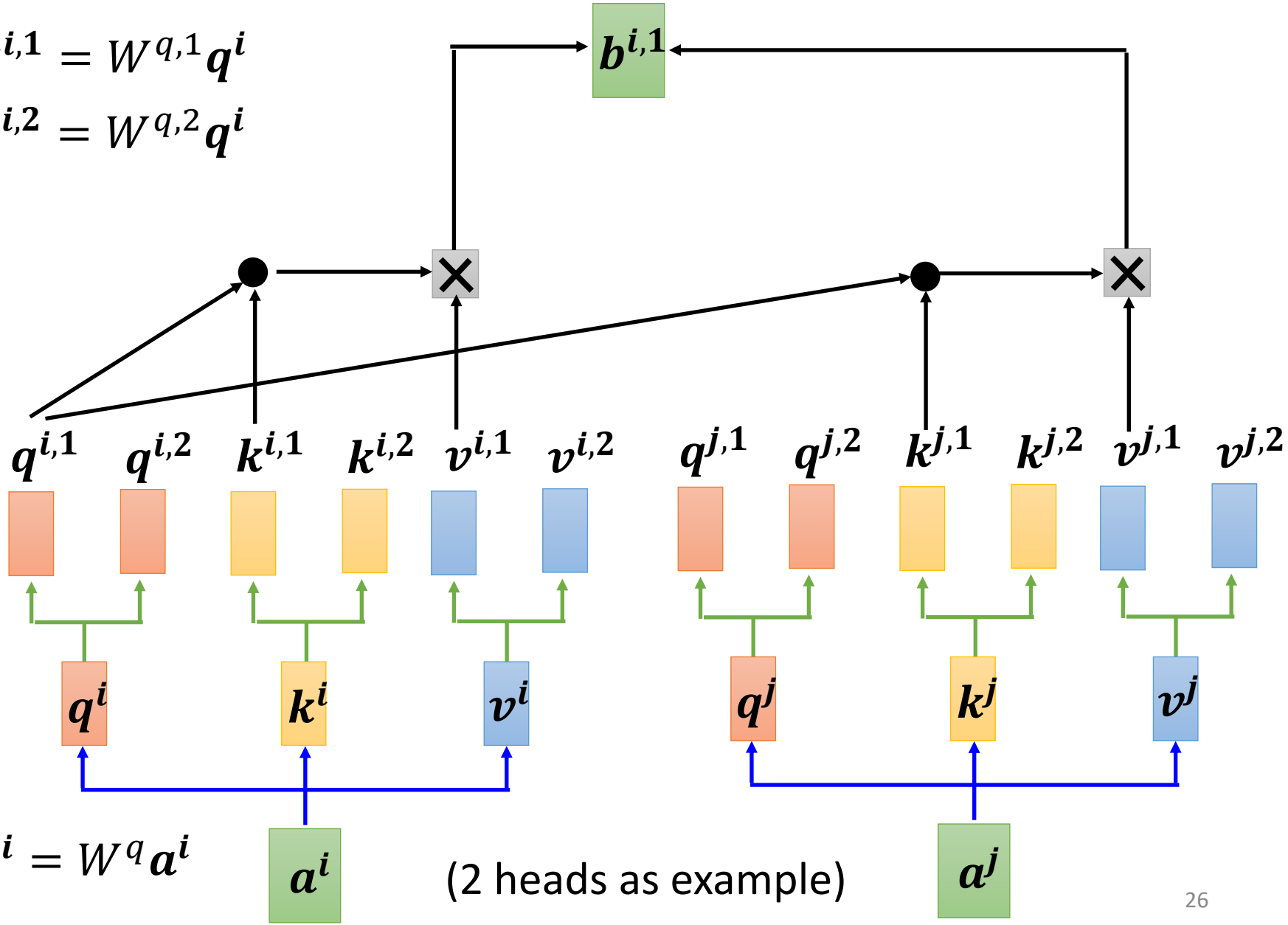


Multi-head Self-attention

Different types of relevance

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

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



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

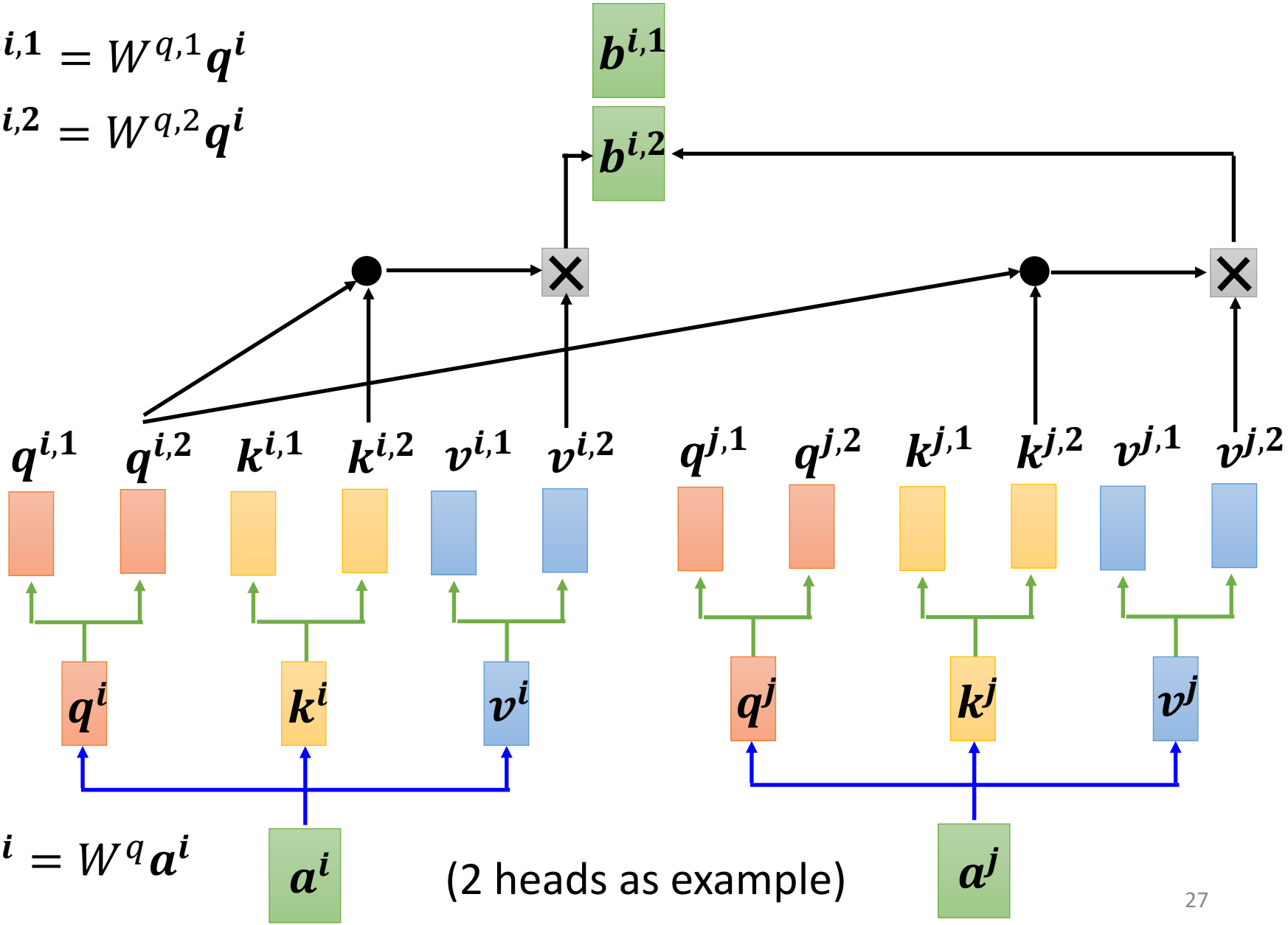
(2 heads as example)

Multi-head Self-attention

Different types of relevance

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

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

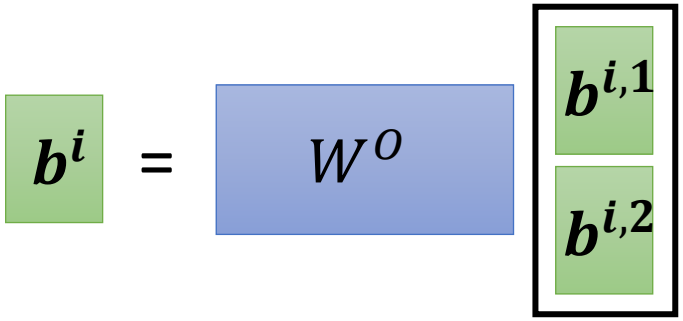


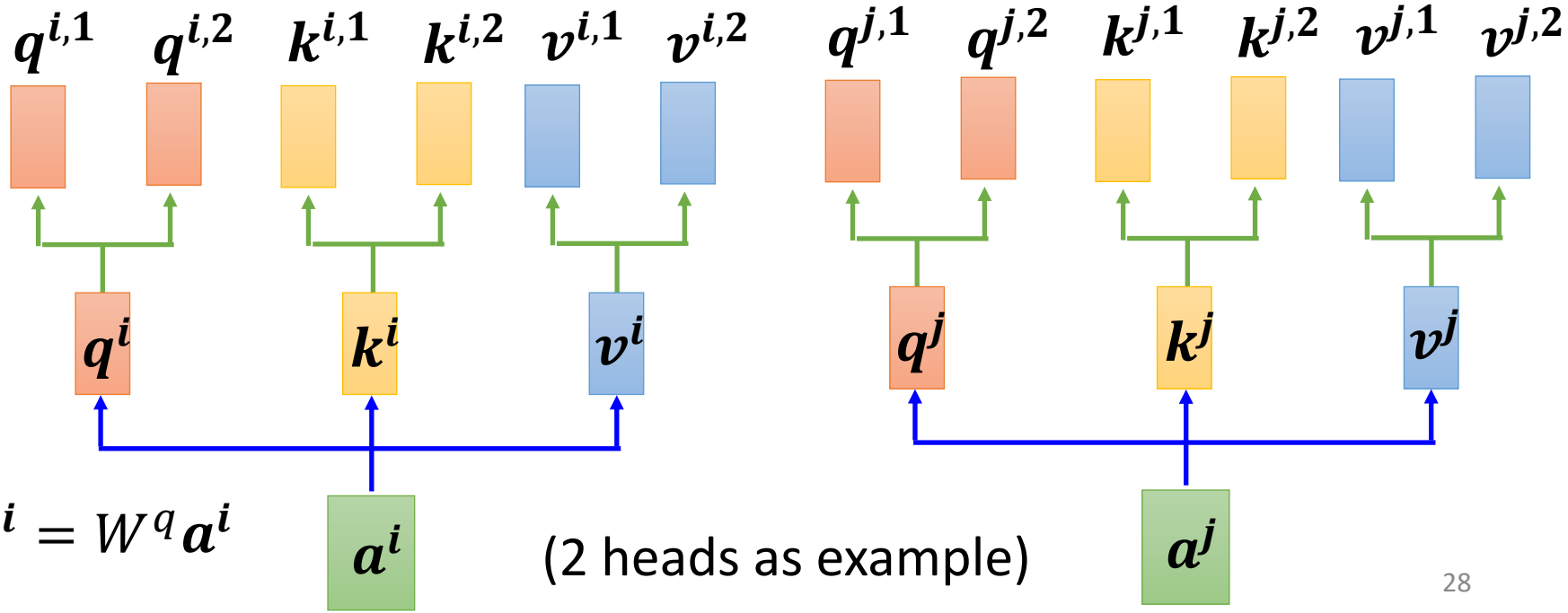
$$q^i = W^q a^i$$

(2 heads as example)

Multi-head Self-attention

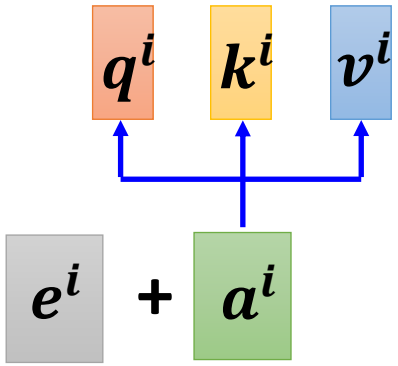
Different types of relevance



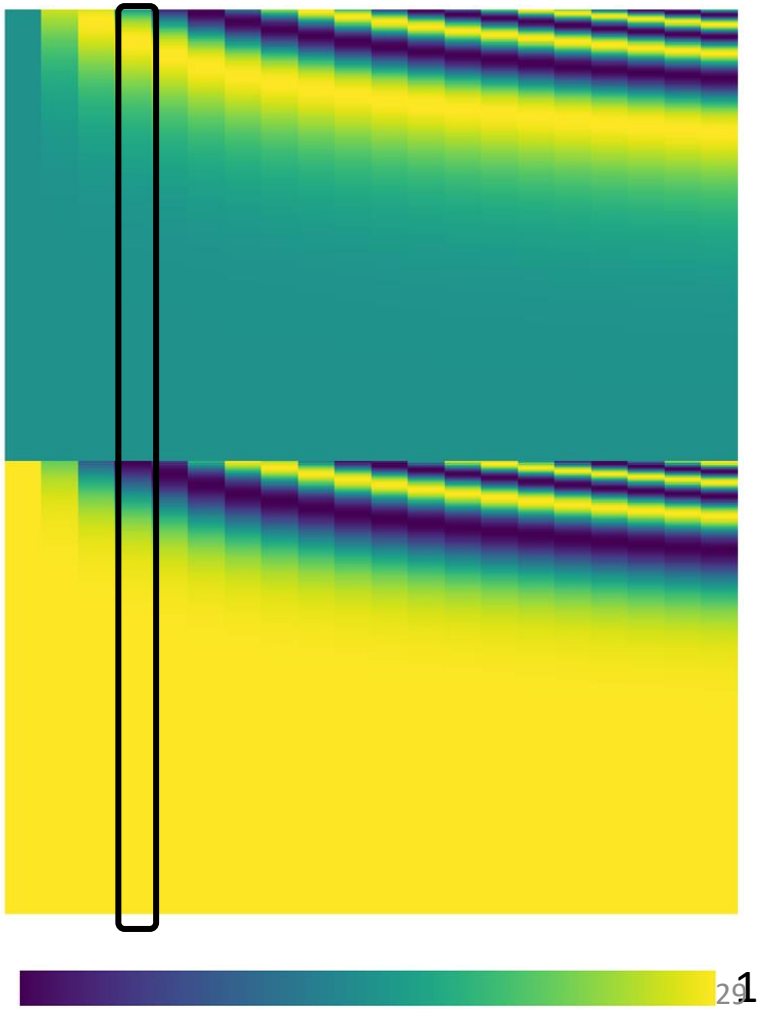


Positional Encoding

- No position information in self-attention.
- Each position has a unique positional vector e^i
- **hand-crafted**
- **learned from data**



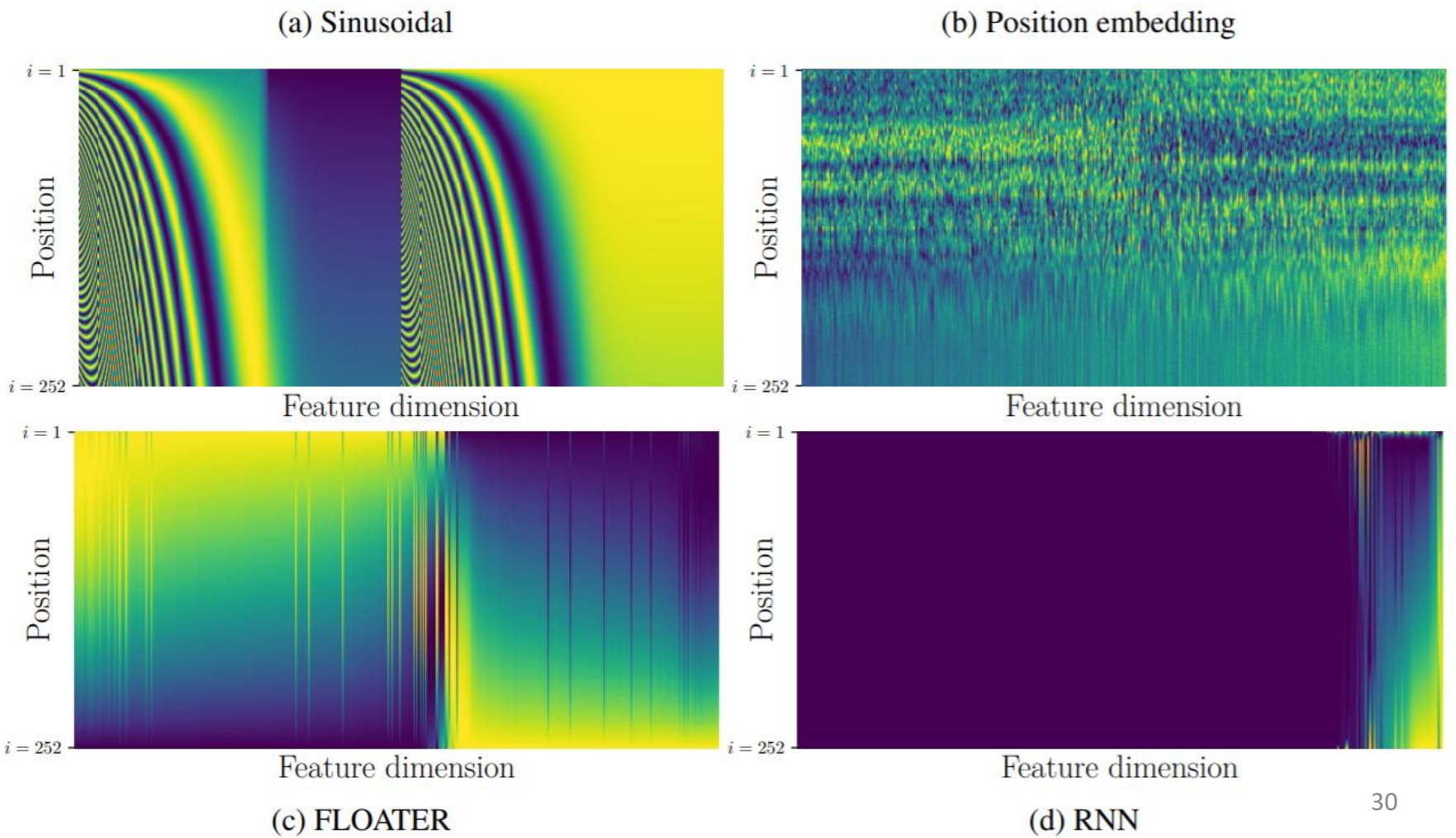
Each column represents a positional vector e^i



https://arxiv.org/abs/2003.09229

Table 1. Comparing position representation methods

Methods	Inductive	Data-Driven	Parameter Efficient
Sinusoidal (Vaswani et al., 2017)	✓	✗	✓
Embedding (Devlin et al., 2018)	✗	✓	✗
Relative (Shaw et al., 2018)	✗	✓	✓
This paper	✓	✓	✓



Many applications …

Transformer

https://arxiv.org/abs/1706.03762

BERT

https://arxiv.org/abs/1810.04805

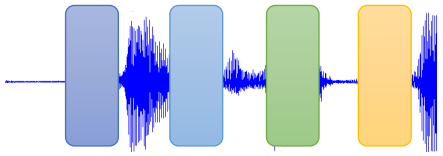
Widely used in Natural Langue Processing (NLP)!

https://arxiv.org/abs/1910.12977

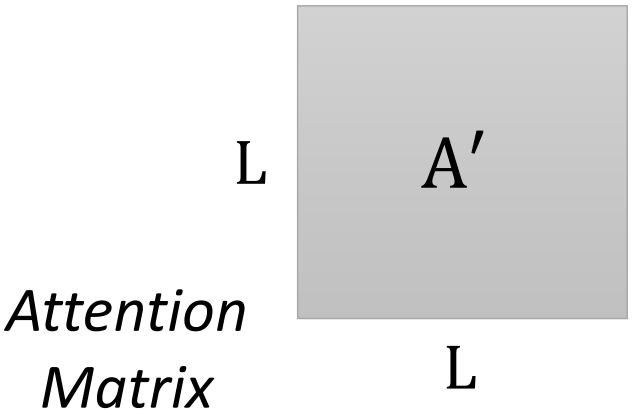
Self-attention for Speech

Speech is a very long vector sequence.

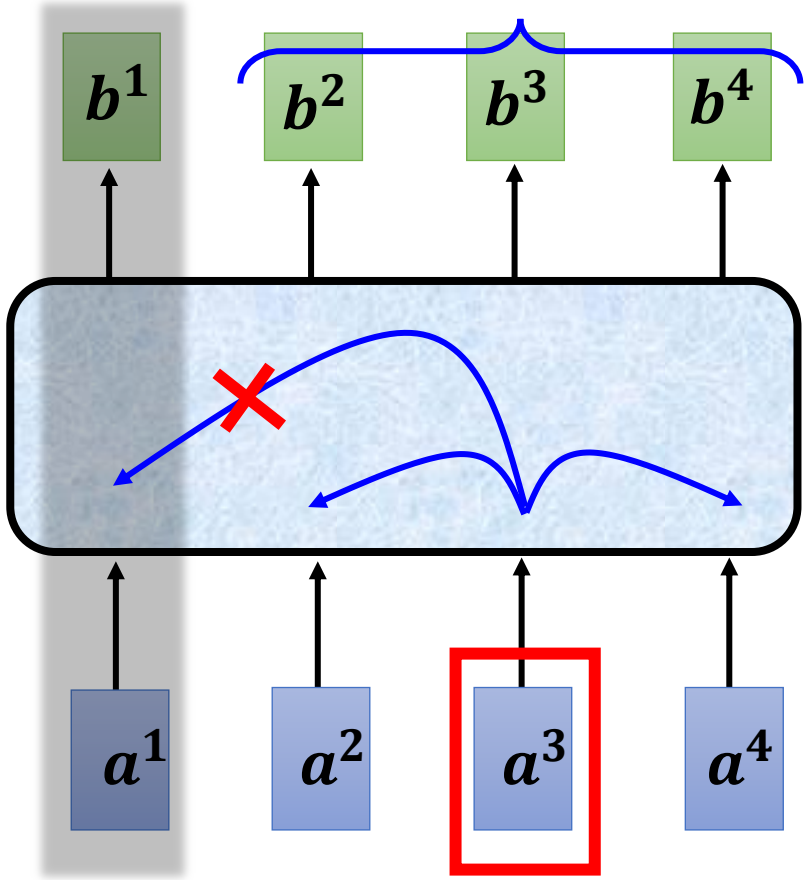
10ms



If input sequence is length L



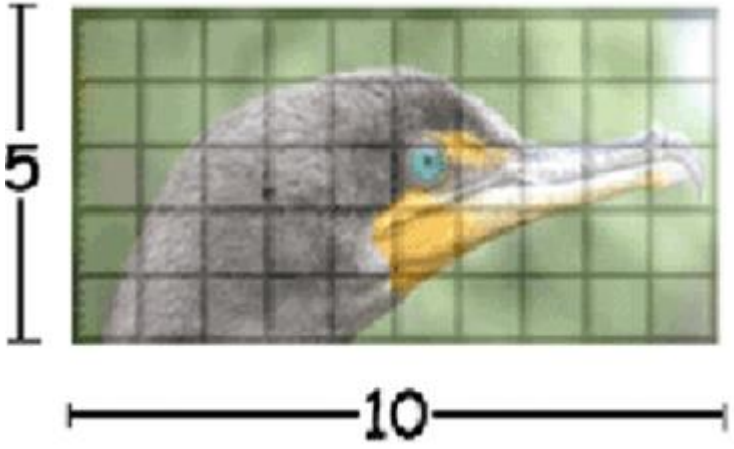
Attention in a range



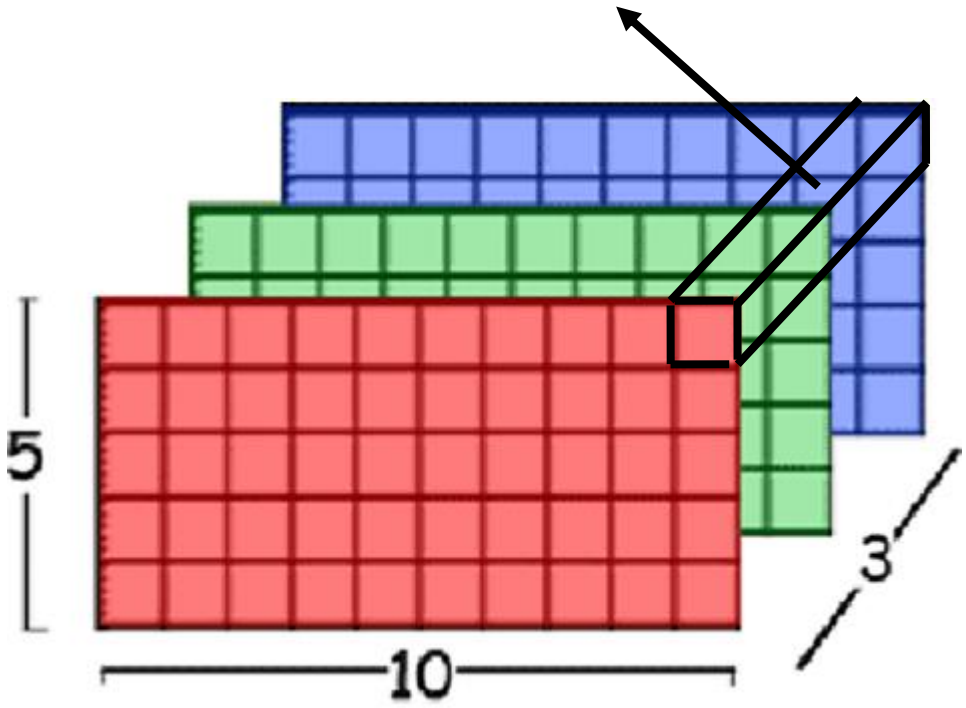
Truncated Self-attention

Self-attention for Image

An **image** can also be considered as a **vector set**.

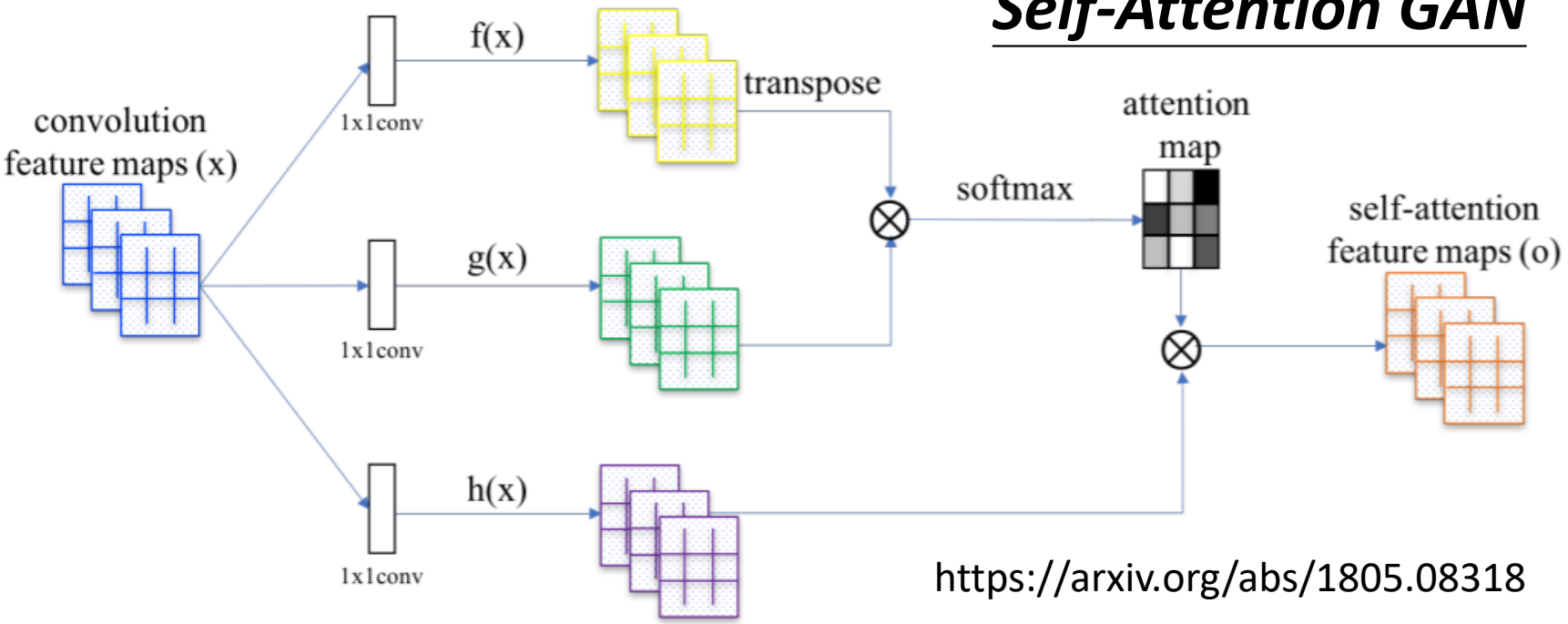


This is a vector.



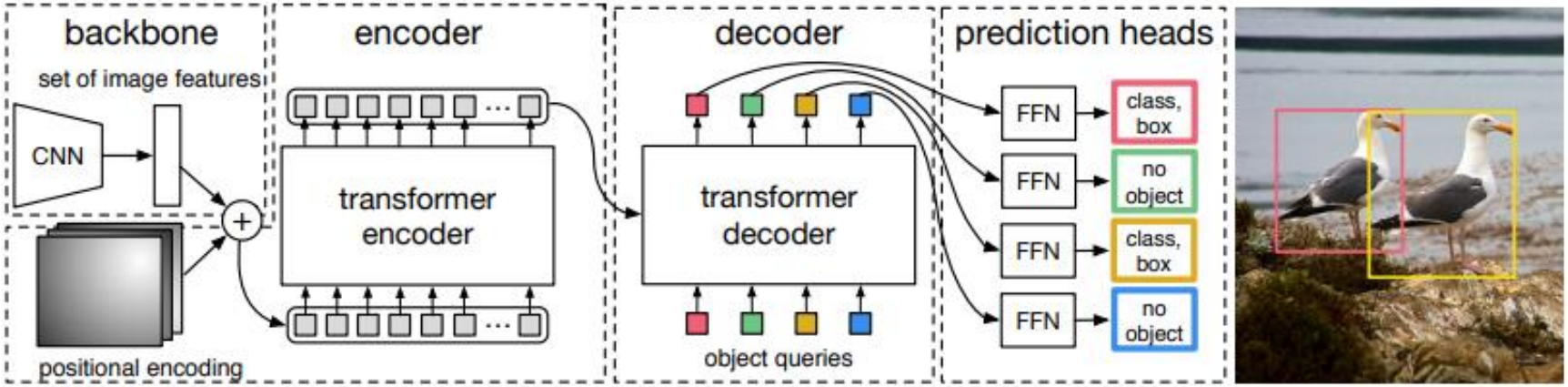
Source of image: https://www.researchgate.net/figure/Color-image-representation-and-RGB-matrix_fig15_282798184

Self-Attention GAN



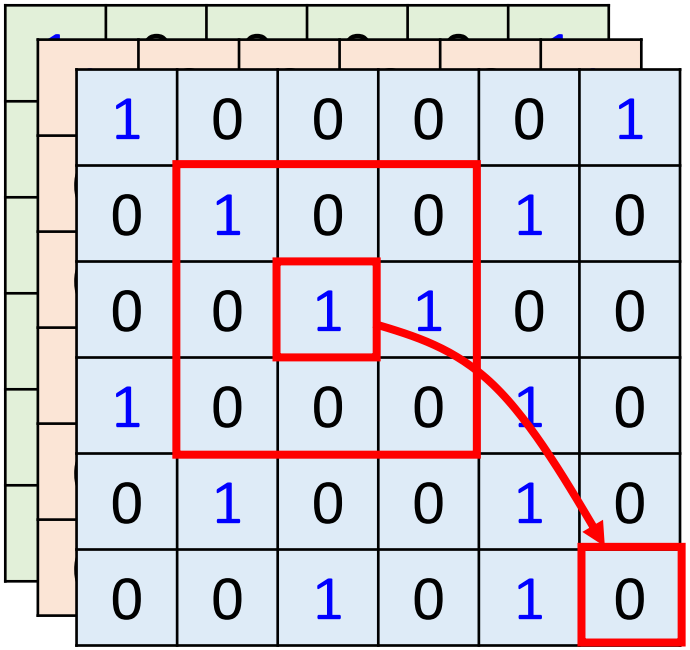
https://arxiv.org/abs/1805.08318

DEtection Transformer (DETR)



https://arxiv.org/abs/2005.12872

Self-attention v.s. CNN



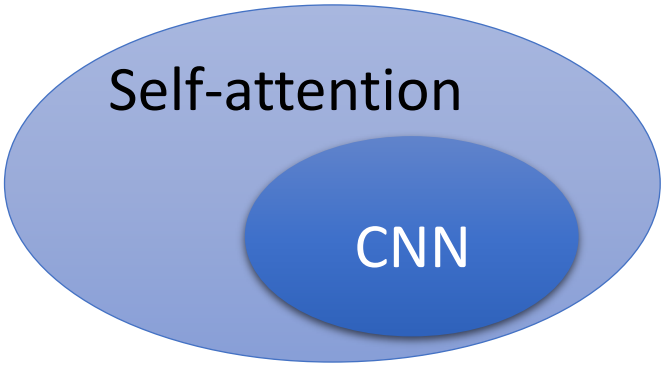
CNN: self-attention that can only attends in a receptive field

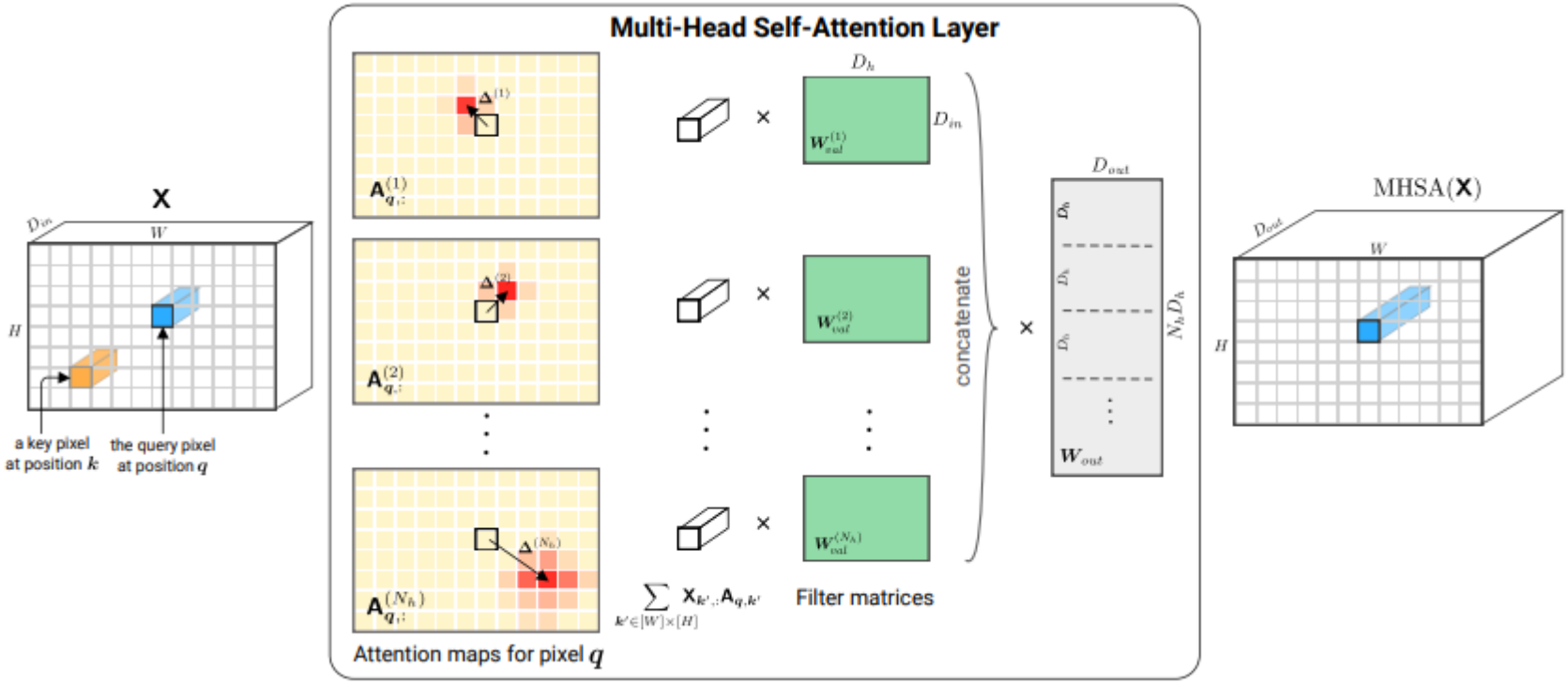
- CNN is simplified self-attention.

Self-attention: CNN with learnable receptive field

- Self-attention is the complex version of CNN.

Self-attention v.s. CNN





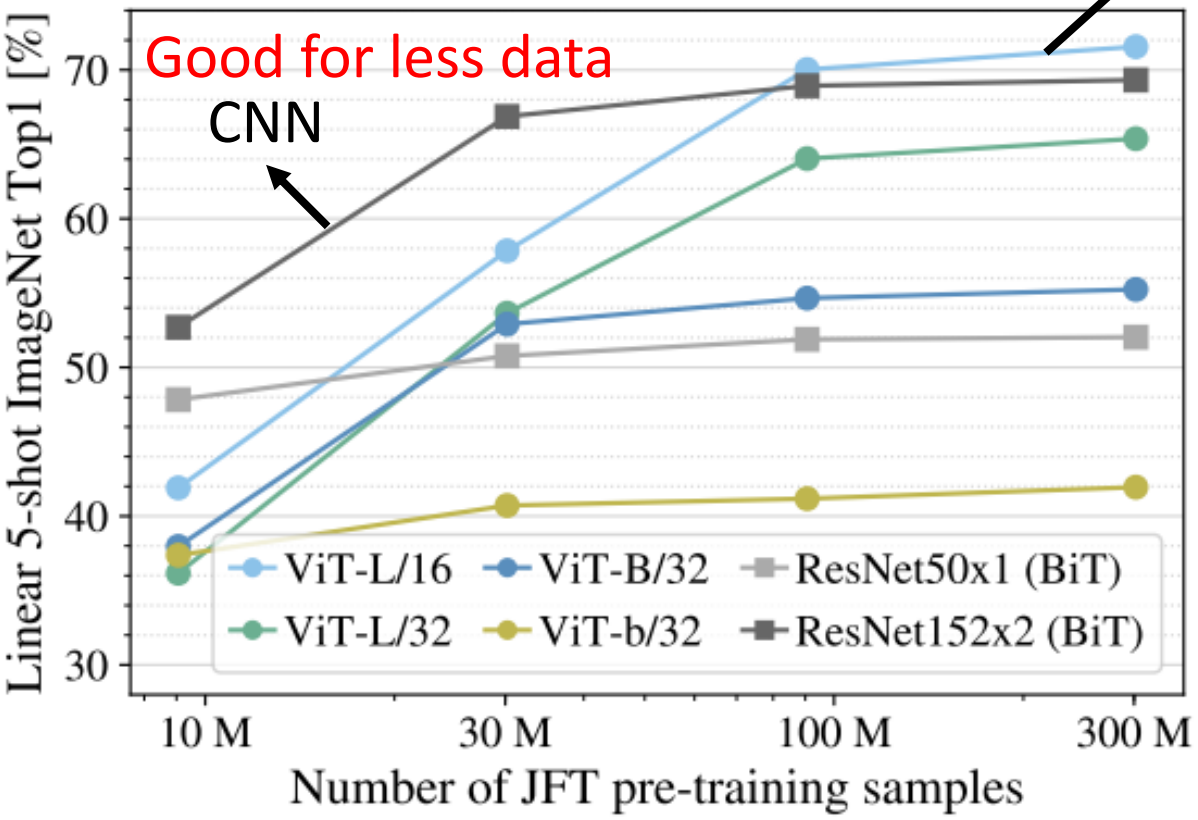
On the Relationship between Self-Attention and Convolutional Layers

https://arxiv.org/abs/1911.03584

Self-attention v.s. CNN

Good for more data

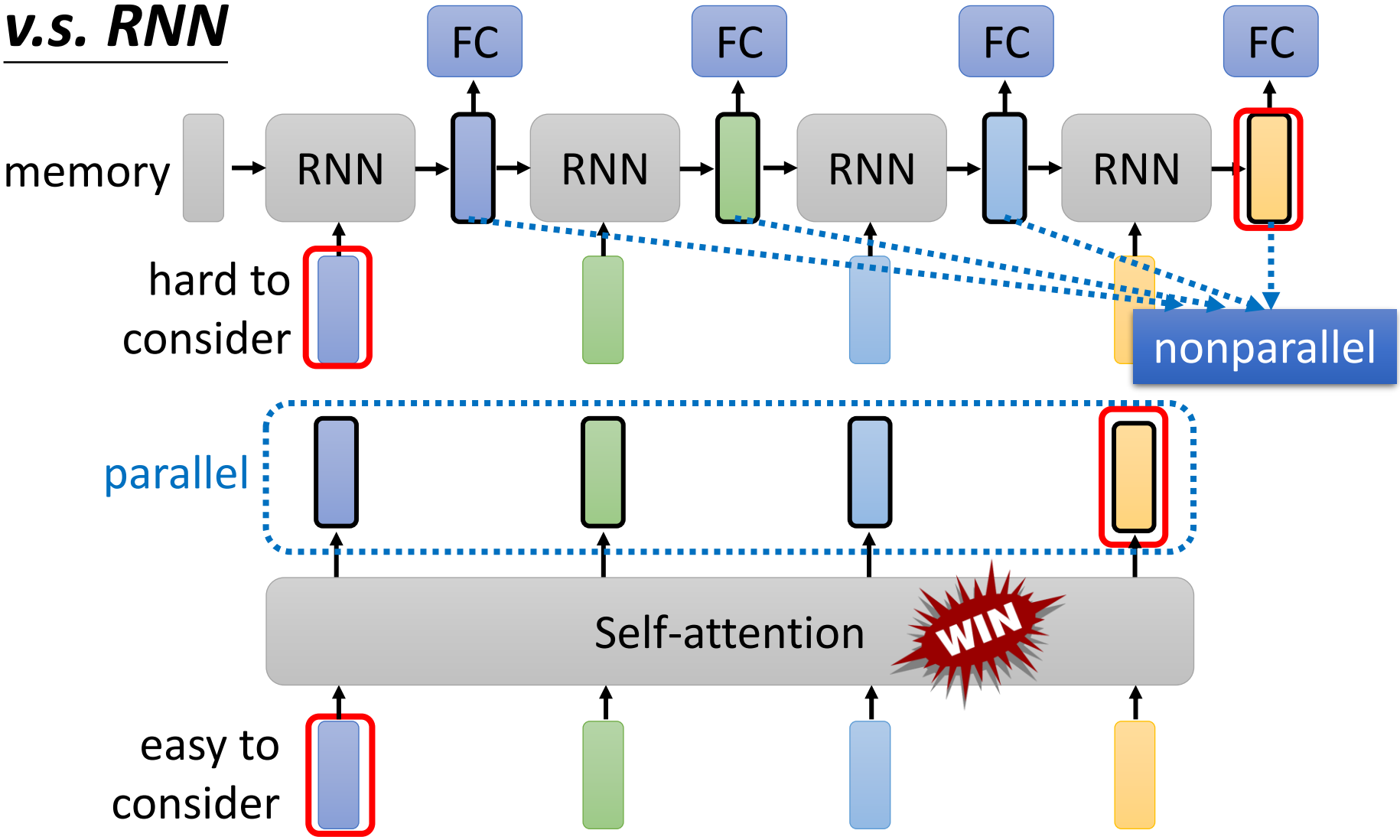
Self-attention



An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

https://arxiv.org/pdf/2010.11929.pdf

Self-attention v.s. RNN



Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention

https://arxiv.org/abs/2006.16236

To learn more about RNN ……

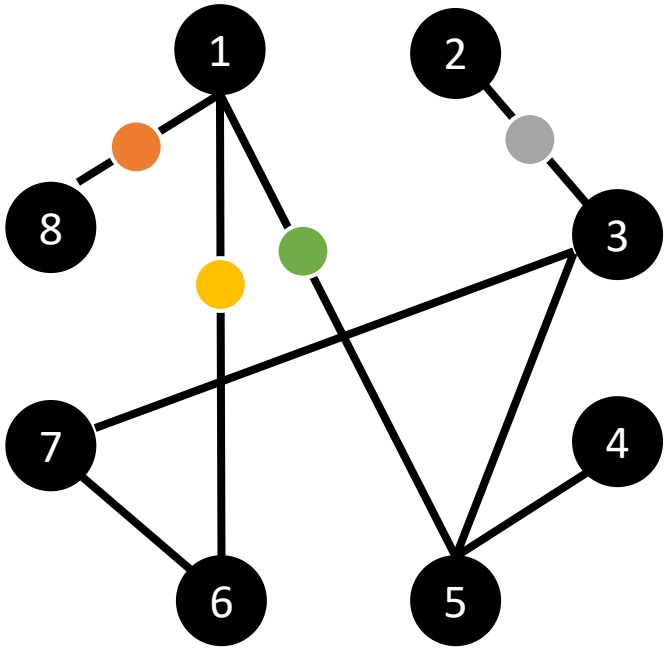
https://youtu.be/xCGidAeyS4M

(in Mandarin)

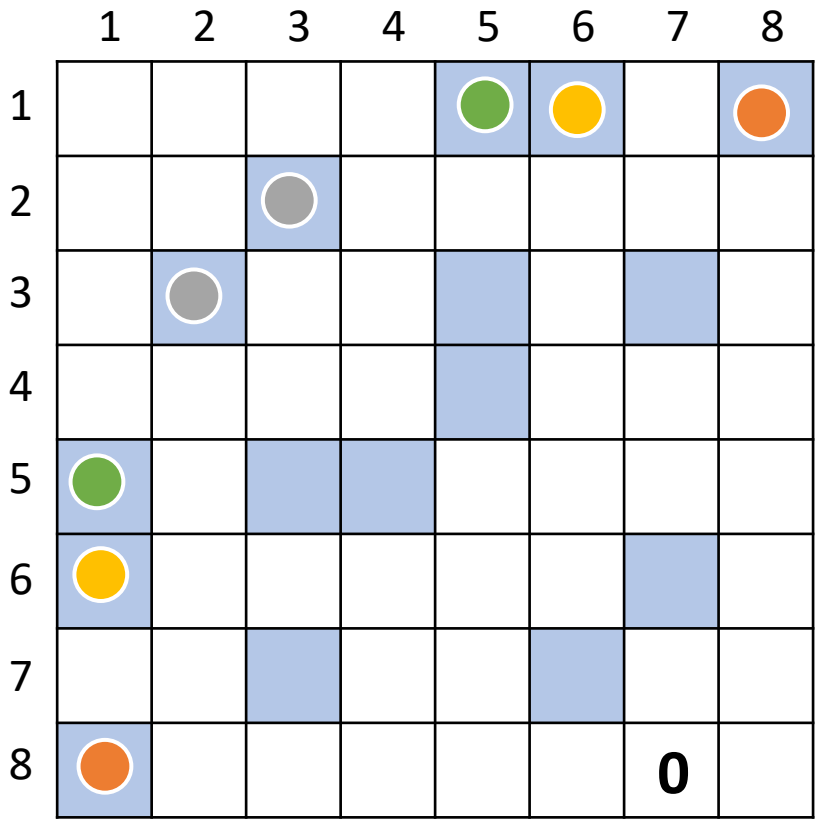
https://youtu.be/Jjy6ER0bHv8

(in English)

Self-attention for Graph



Attention Matrix



Consider **edge**: only attention to connected nodes

This is one type of **Graph Neural Network (GNN)**.

Self-attention for Graph

- To learn more about GNN …

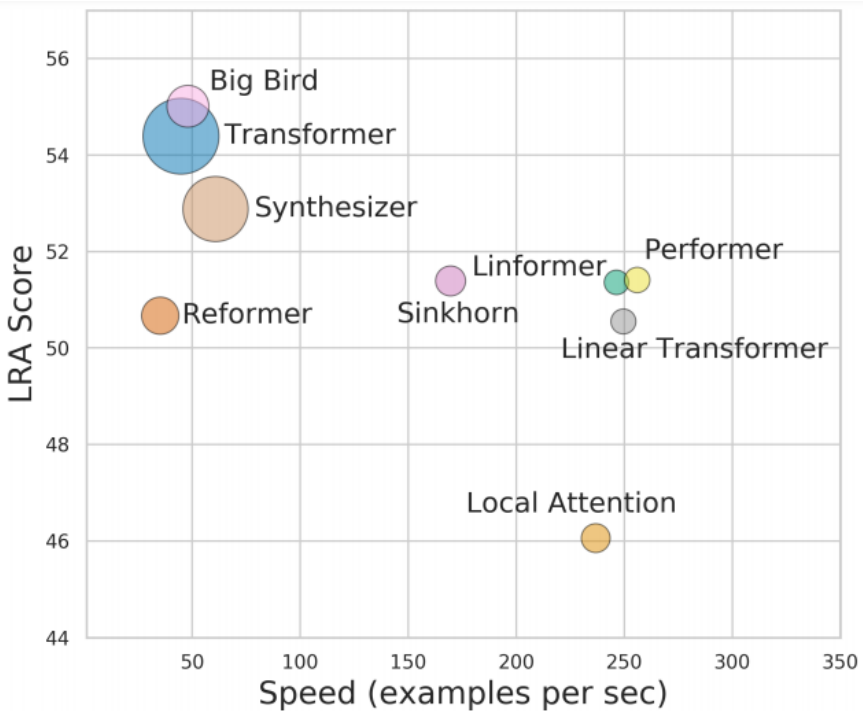
https://youtu.be/eybCCtNKwzA
(in Mandarin)

https://youtu.be/M9ht8vsVEw8
(in Mandarin)

To Learn More …

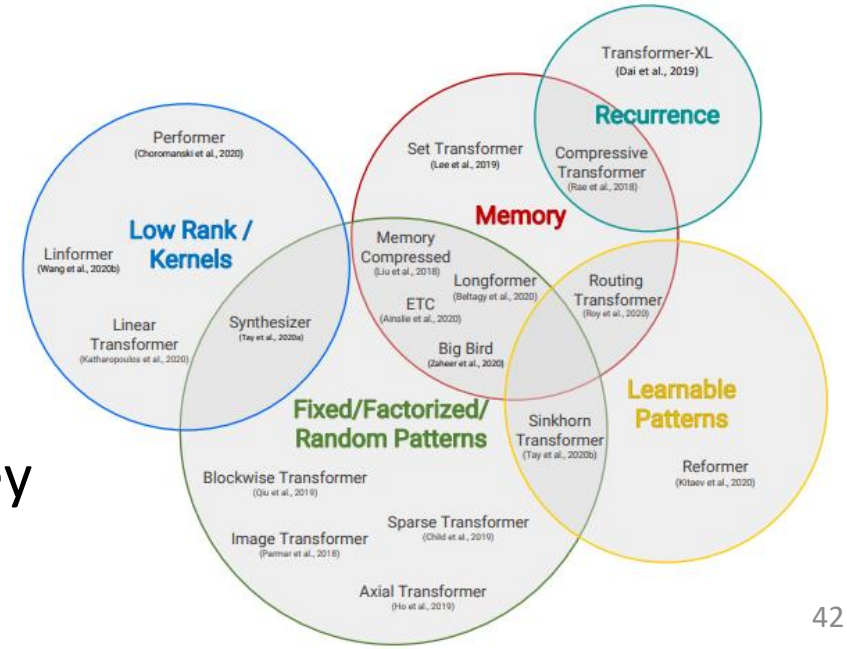
Long Range Arena: A Benchmark for Efficient Transformers

https://arxiv.org/abs/2011.04006



Efficient Transformers: A Survey

https://arxiv.org/abs/2009.06732




Q&A