



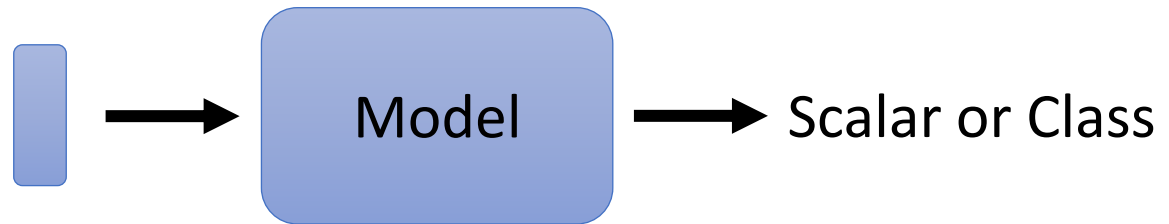
Self-attention

Hung-yi Lee

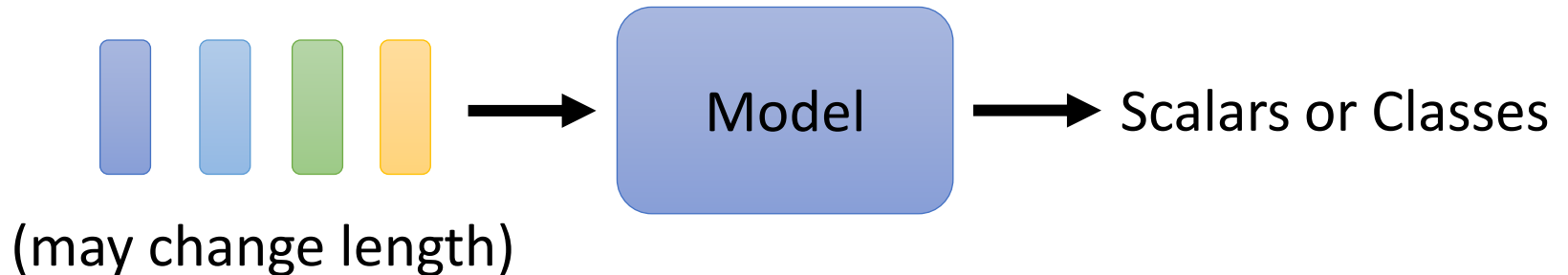
李宏毅

Sophisticated Input

- Input is a **vector**




- Input is a **set of vectors**



Vector Set as Input

this is a cat



One-hot Encoding

apple = [1 0 0 0 0]

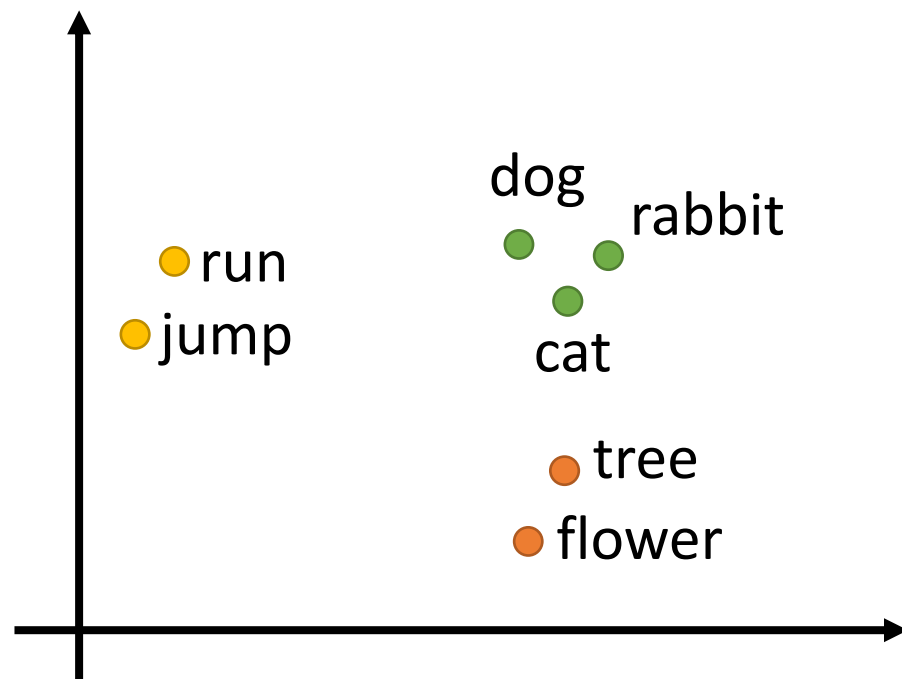
bag = [0 1 0 0 0]

cat = [0 0 1 0 0]

dog = [0 0 0 1 0]

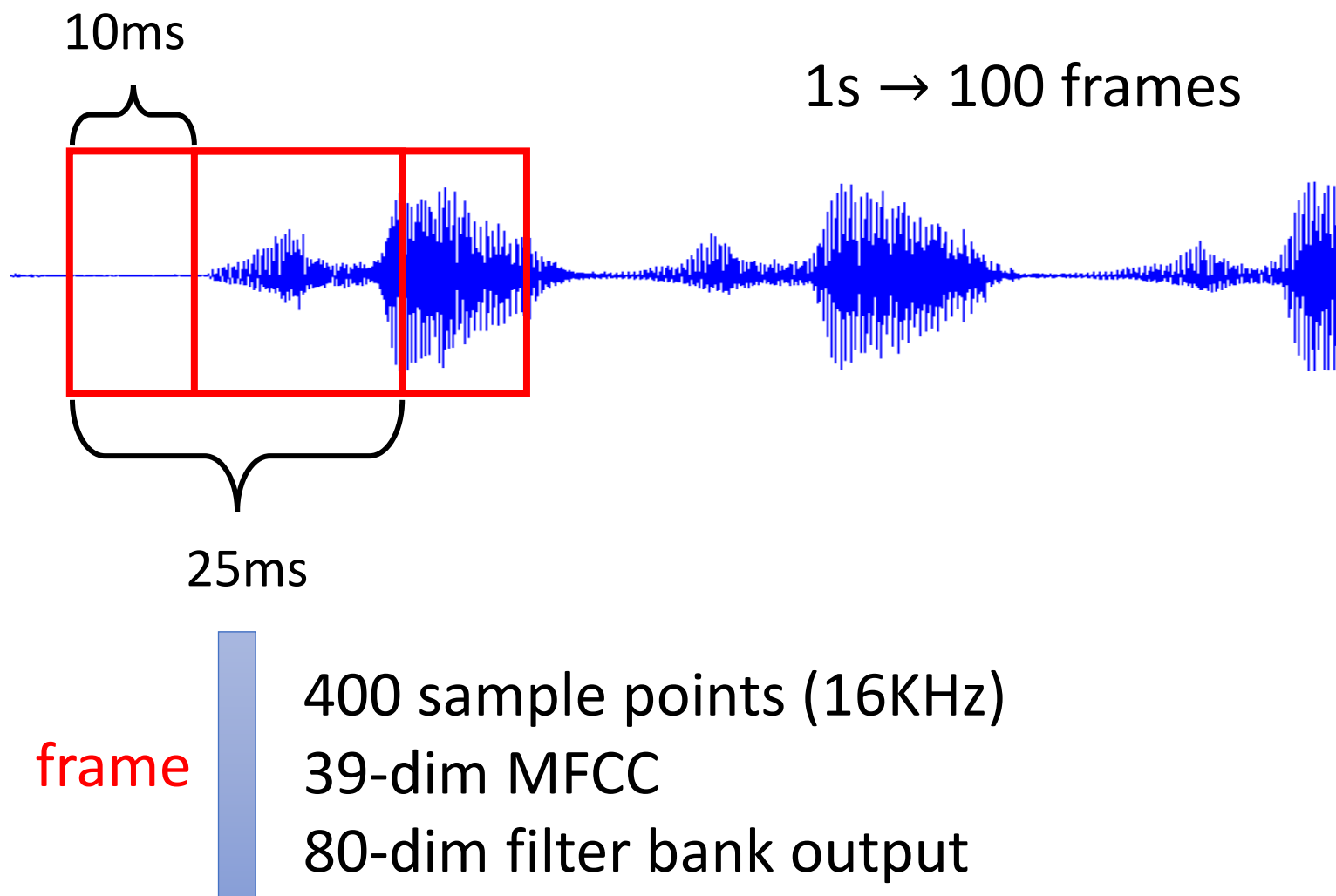
elephant = [0 0 0 0 1]

Word Embedding



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

Vector Set as Input



Vector Set as Input

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



Vector Set as Input

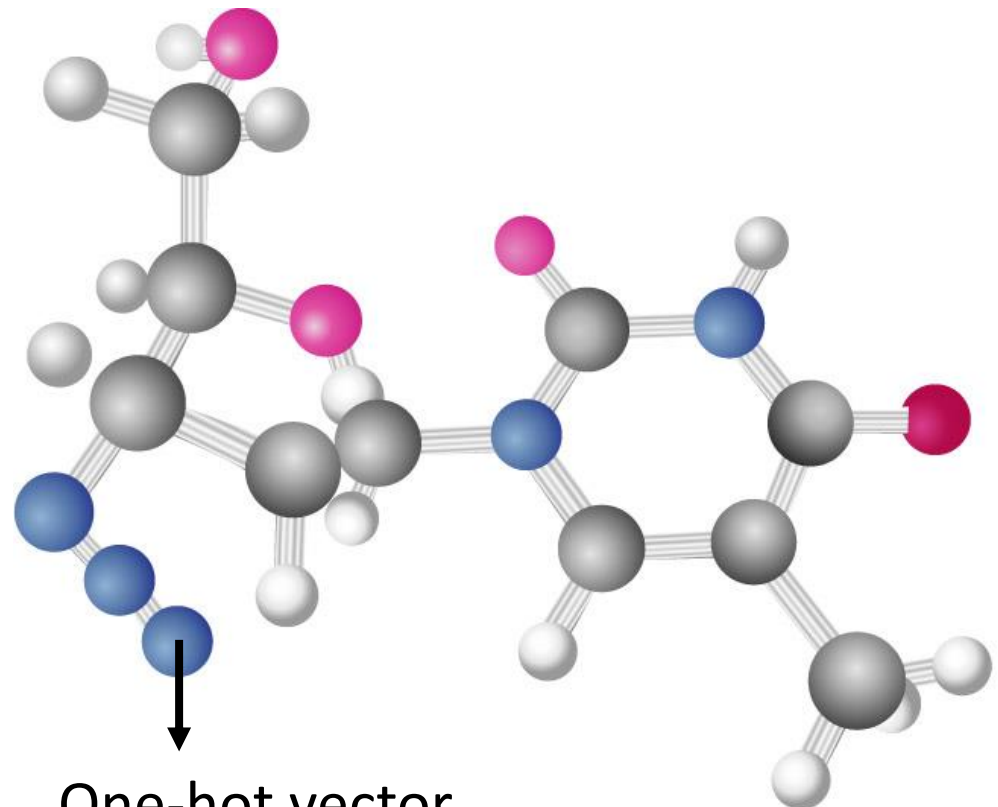
- Graph is also a set of vectors (consider each **node** as a **vector**)

$$H = [1 \ 0 \ 0 \ 0 \ 0 \ \dots]$$

$$C = [0 \ 1 \ 0 \ 0 \ 0 \ \dots]$$

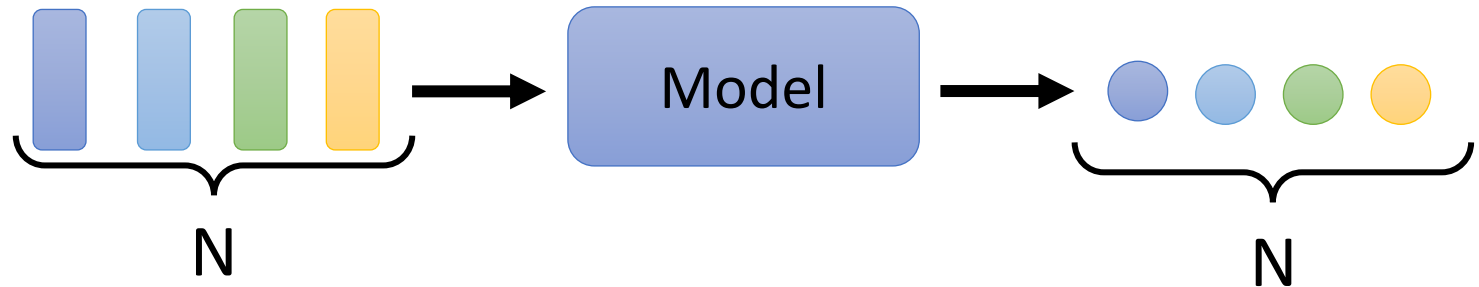
$$O = [0 \ 0 \ 1 \ 0 \ 0 \ \dots]$$

⋮



What is the output?

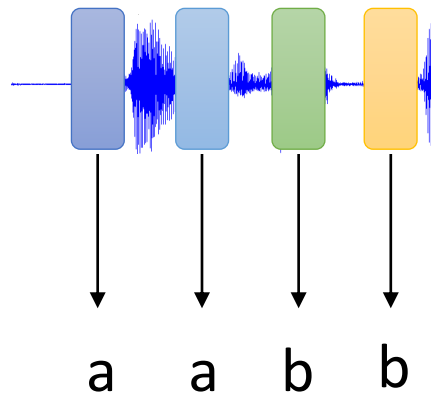
- Each vector has a label.



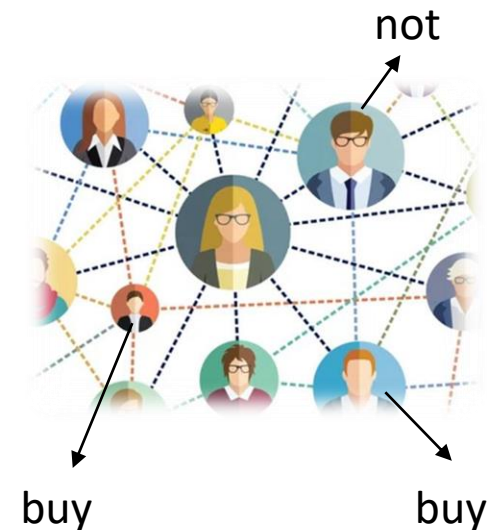
Example Applications

I saw a saw
↓ ↓ ↓ ↓
N V DET N

POS tagging

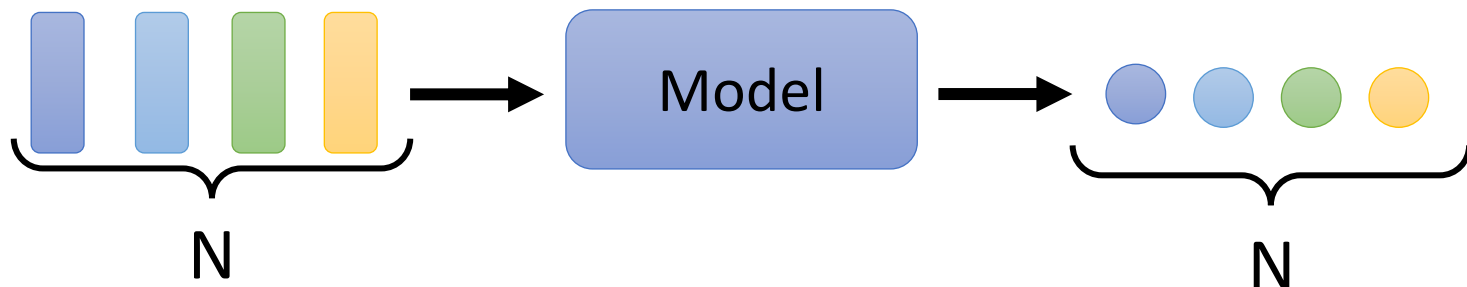


HW2

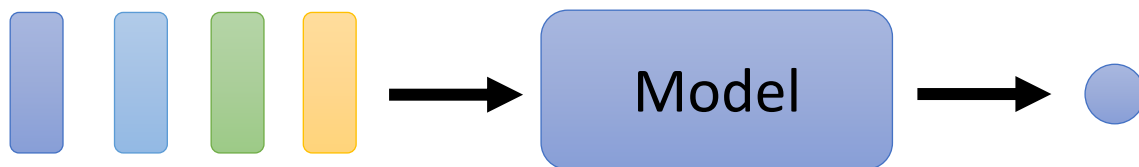


What is the output?

- Each vector has a label.

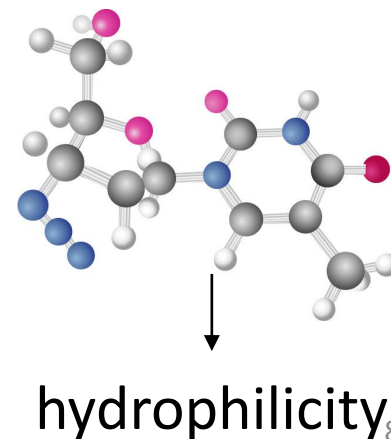
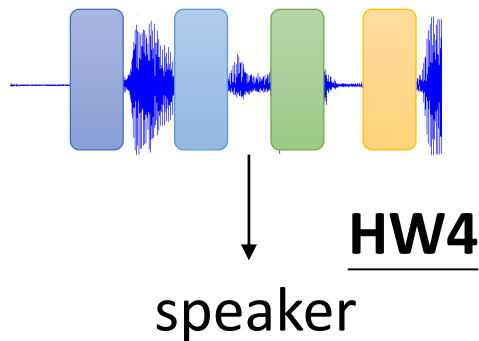


- The whole sequence has a label.



Example Applications

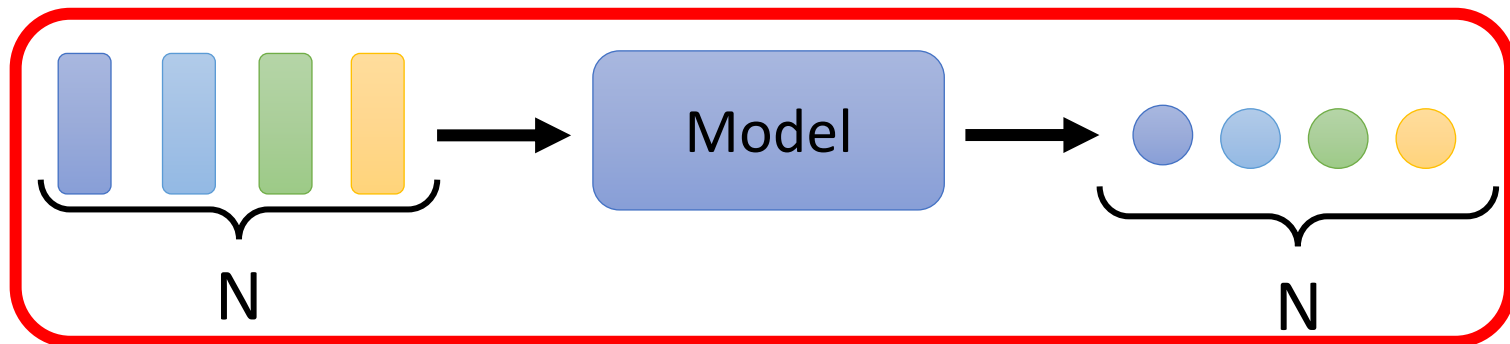
this is good
Sentiment
analysis
↓
positive



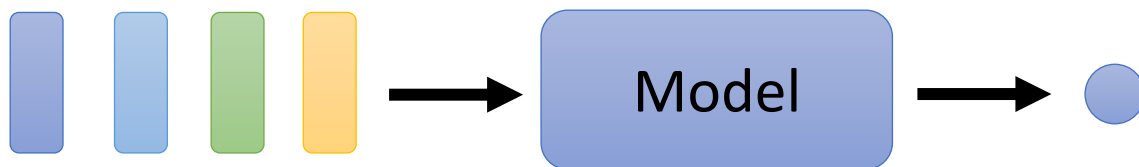
What is the output?

- Each vector has a label.

focus of this lecture



- The whole sequence has a label.



- Model decides the number of labels itself.

seq2seq



Translation (HW5)

Sequence Labeling

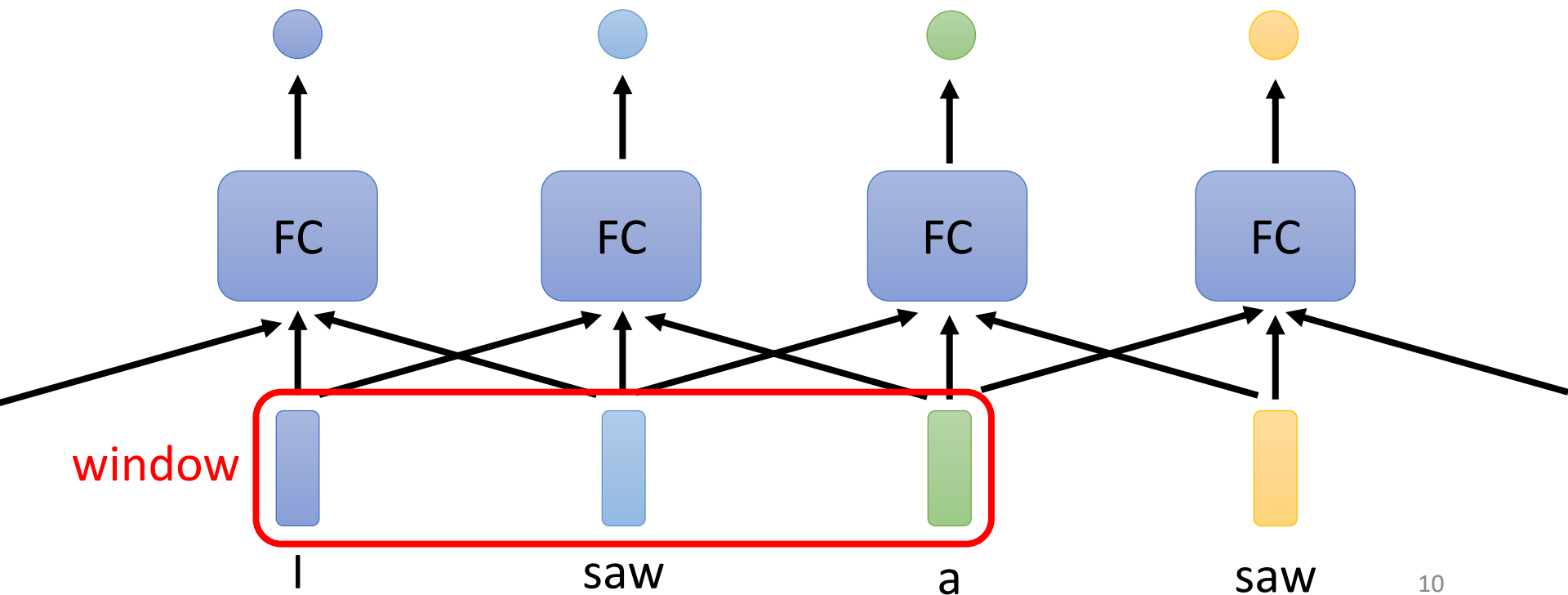
Is it possible to consider the context?

FC Fully-connected

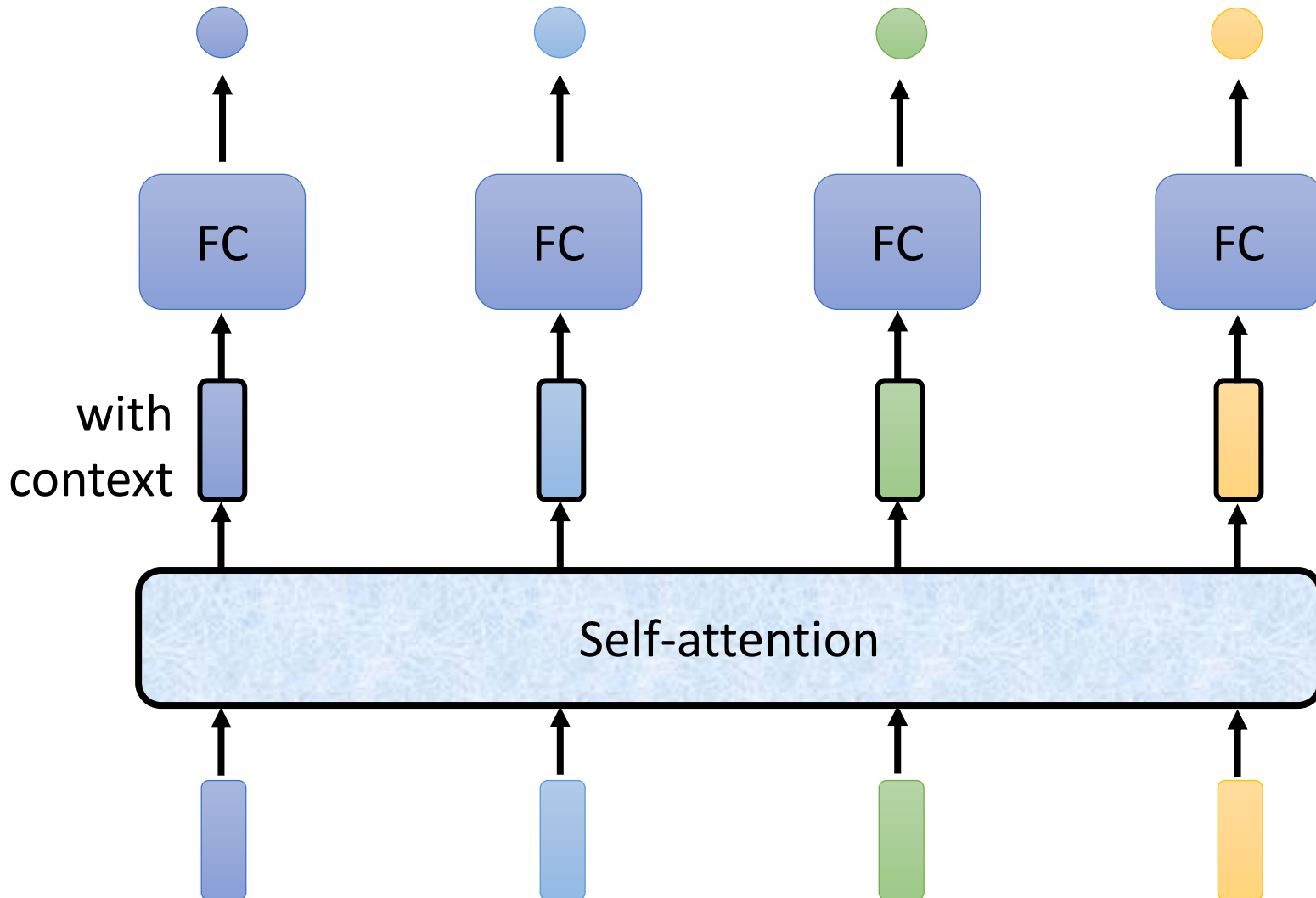
FC can consider the neighbor

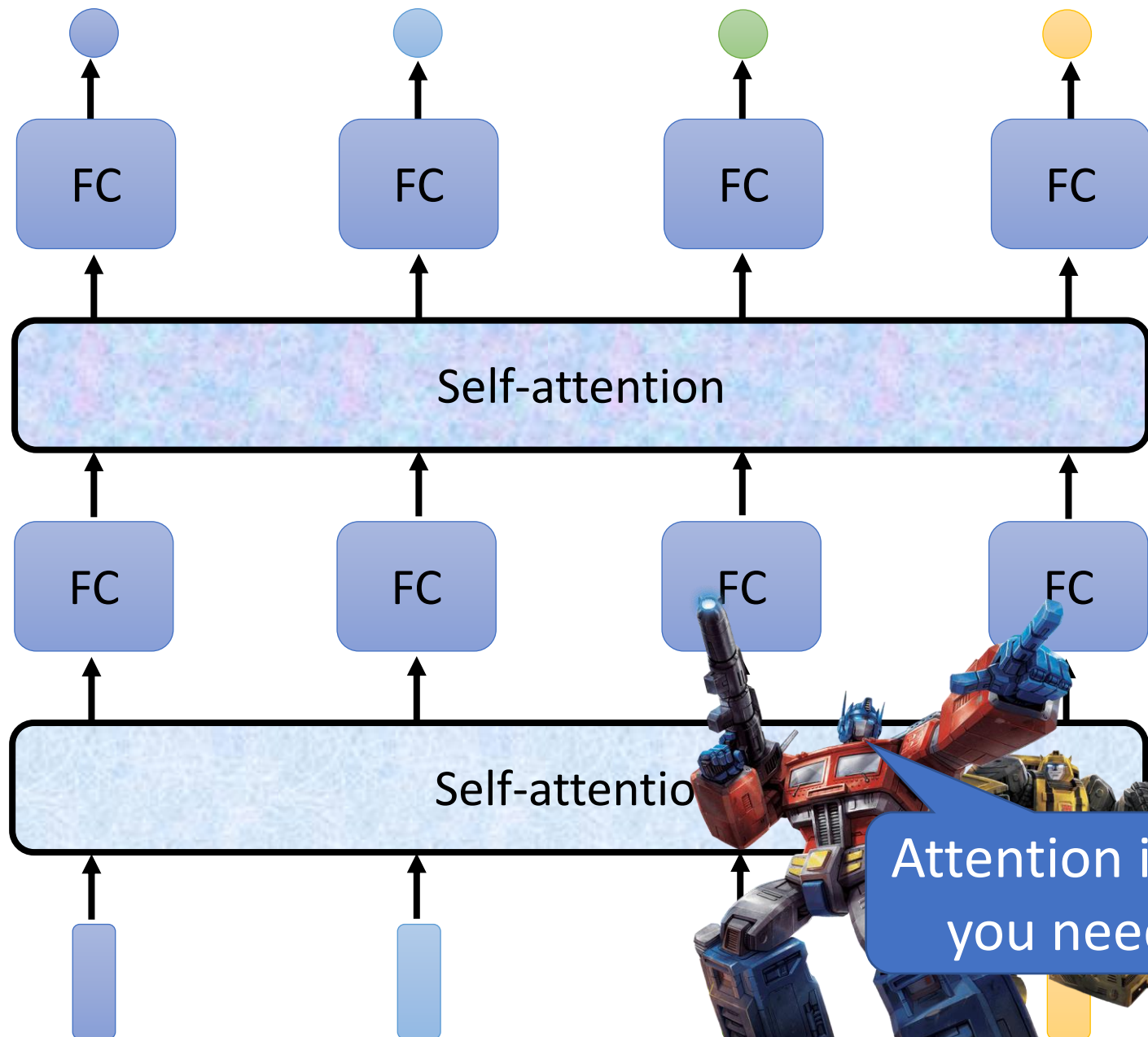
How to consider the whole sequence?

a window covers the whole sequence?



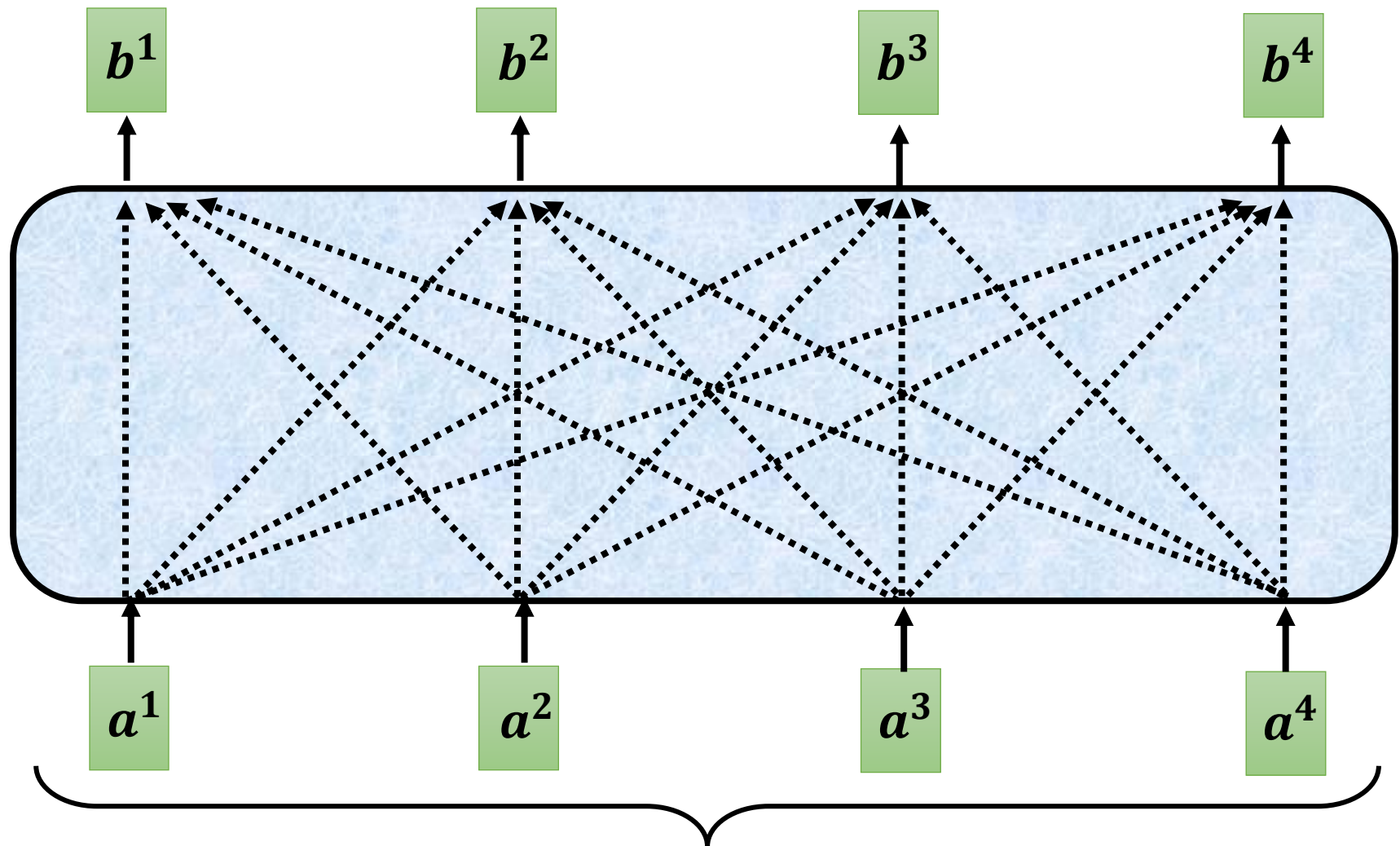
Self-attention





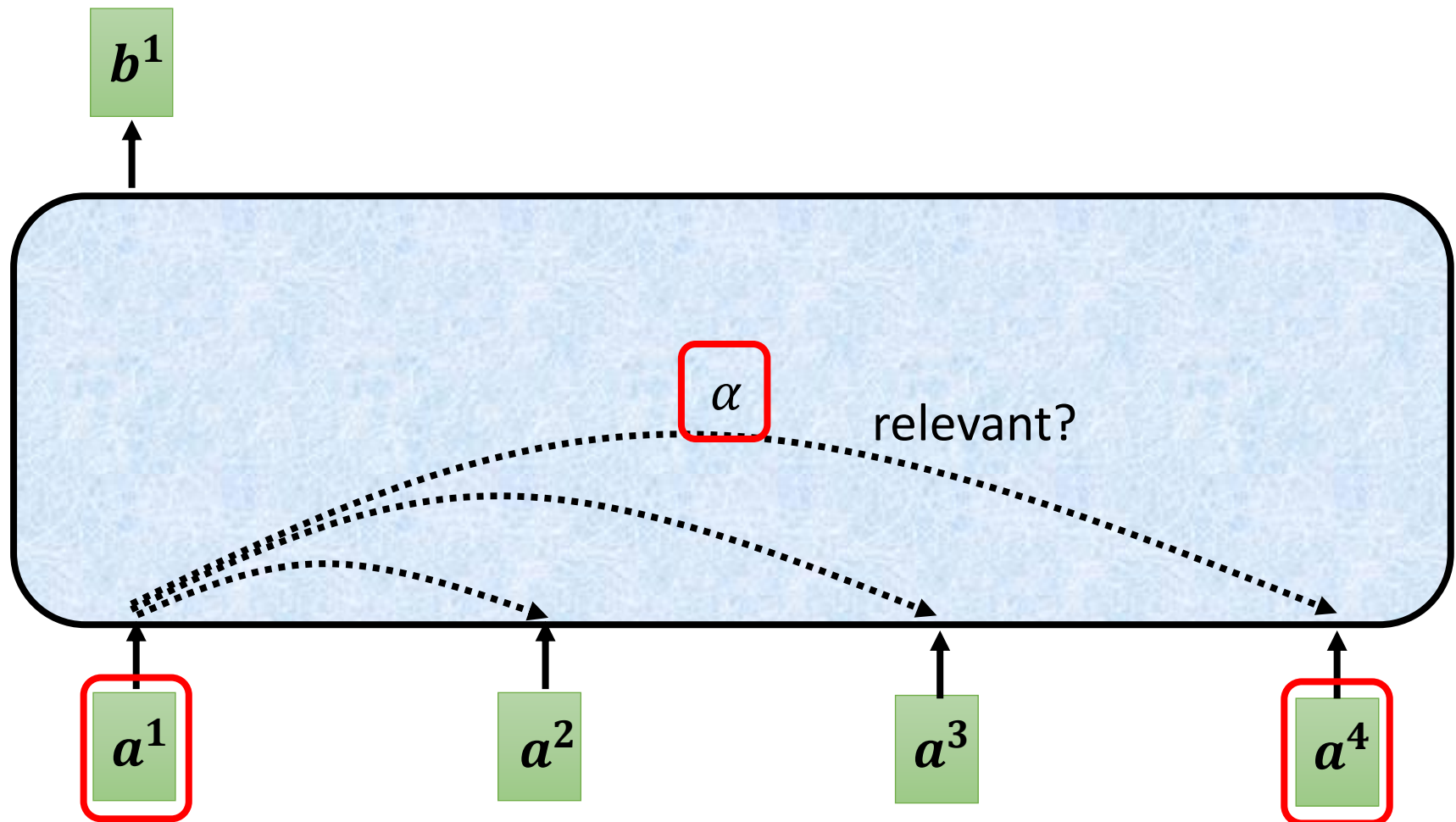
Attention is all
you need.

Self-attention



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

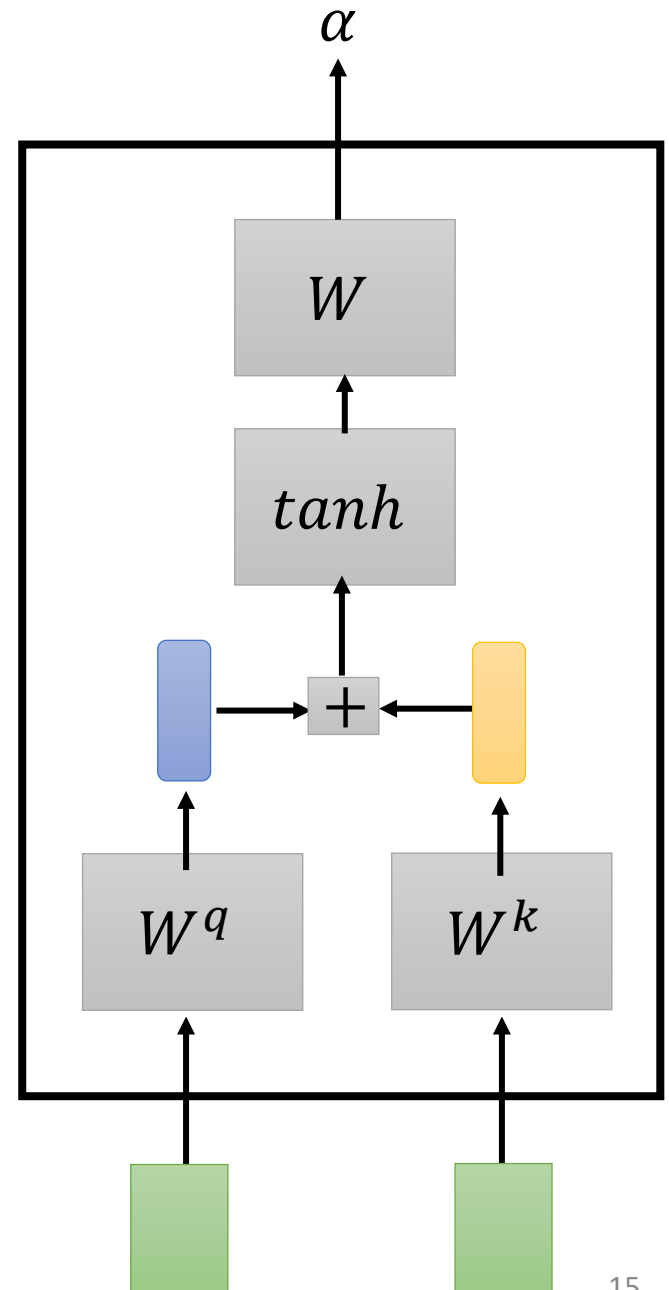
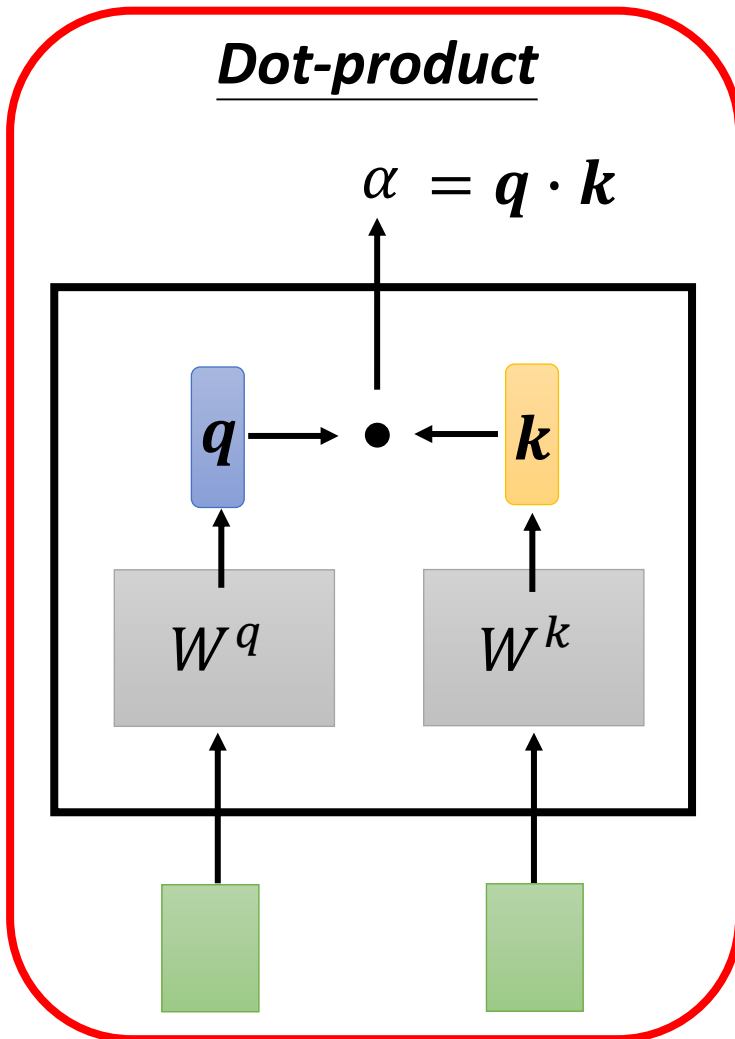
Self-attention



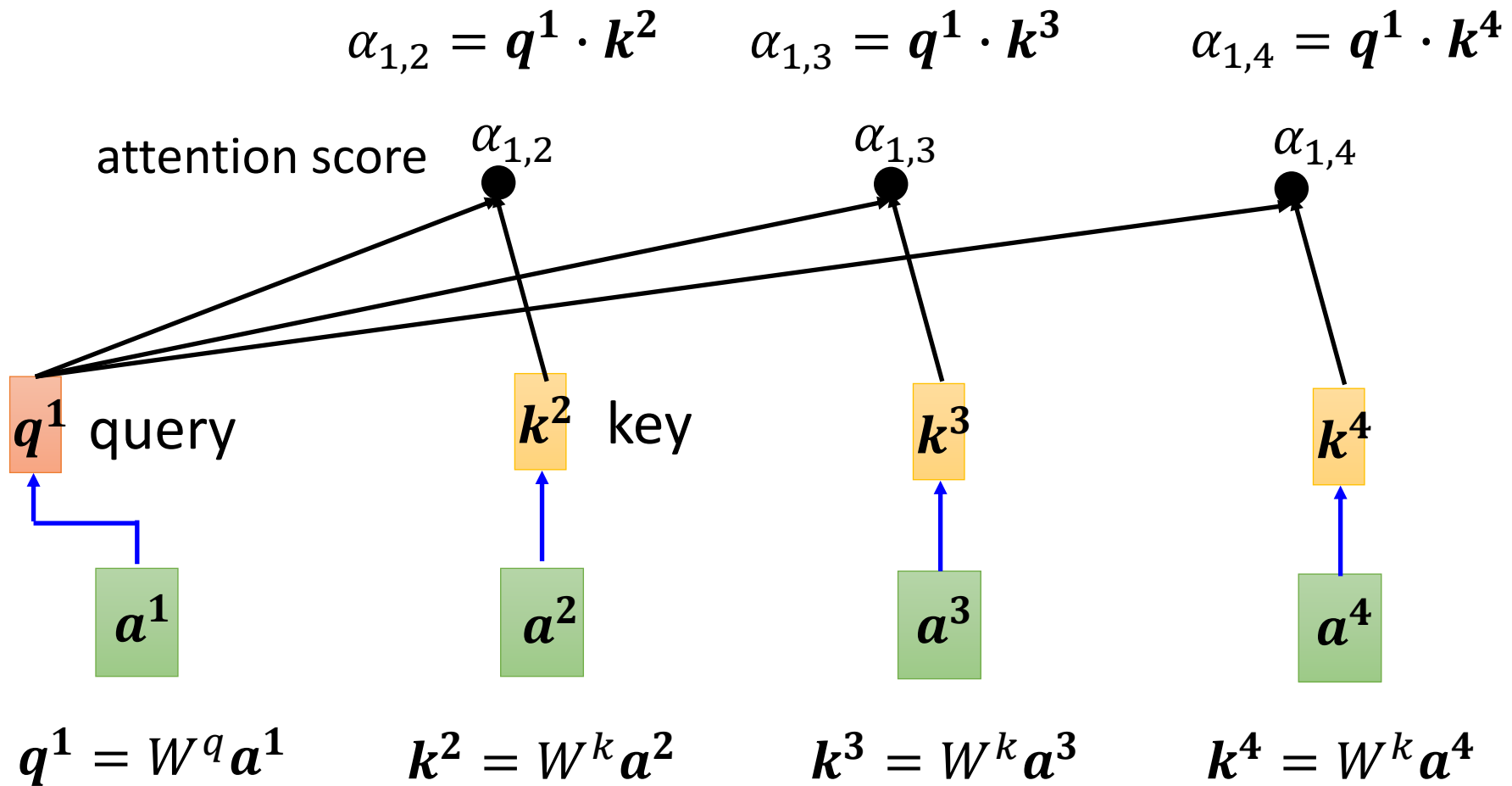
Find the relevant vectors in a sequence

Self-attention

Additive

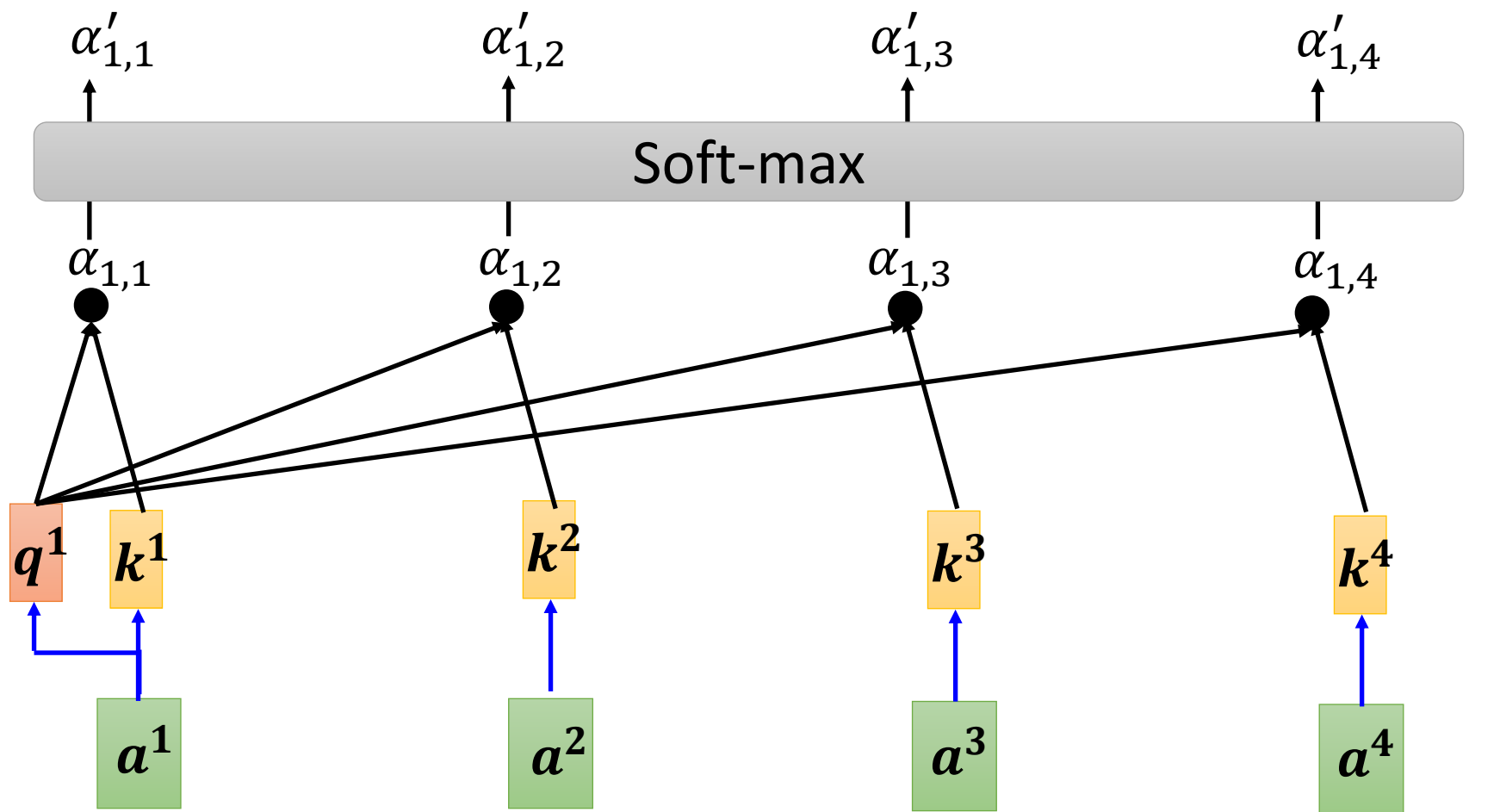


Self-attention



Self-attention

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



$$q^1 = W^q a^1$$

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

$$k^2 = W^k a^2$$

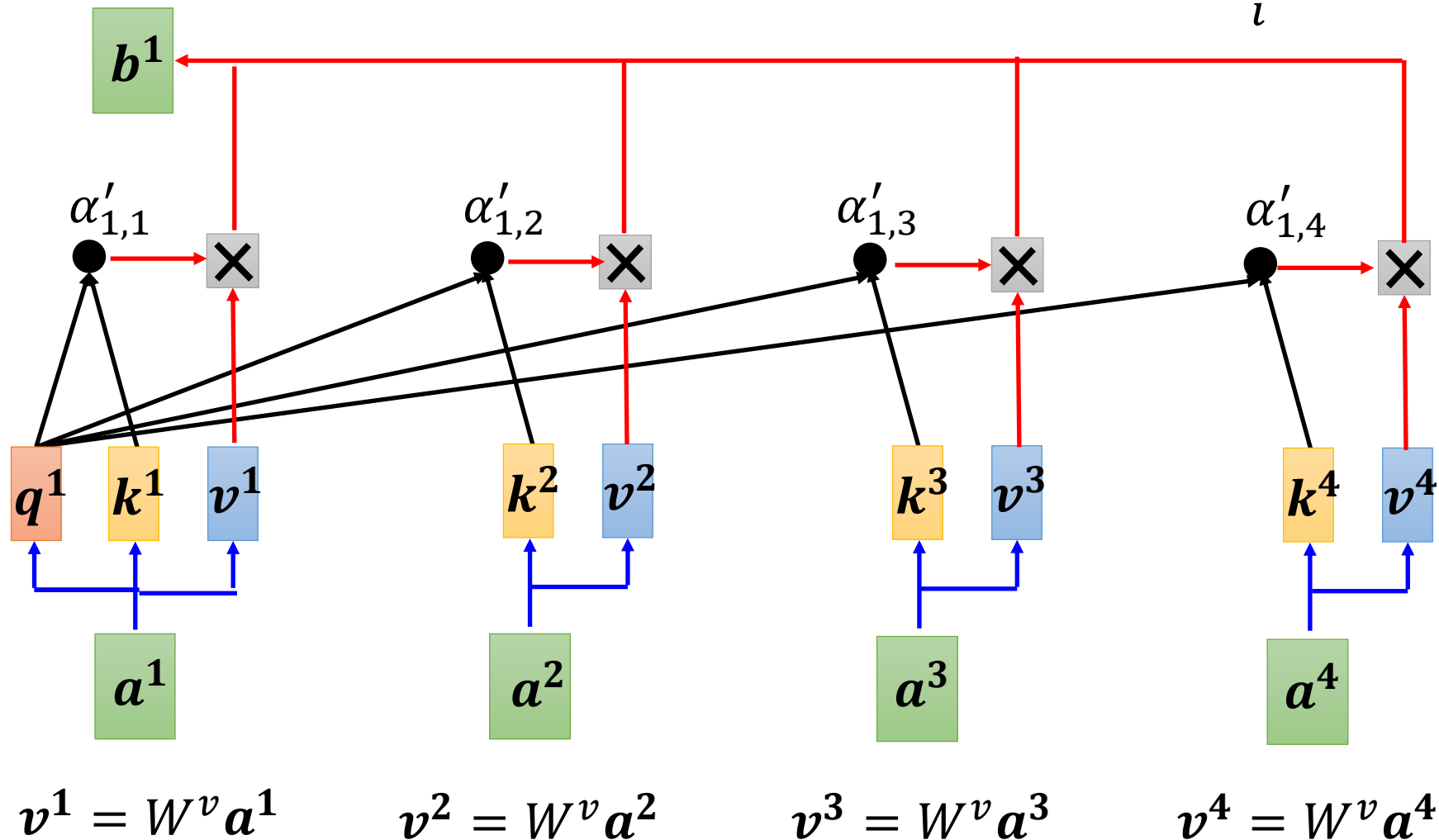
$$k^3 = W^k a^3$$

$$k^4 = W^k a^4$$

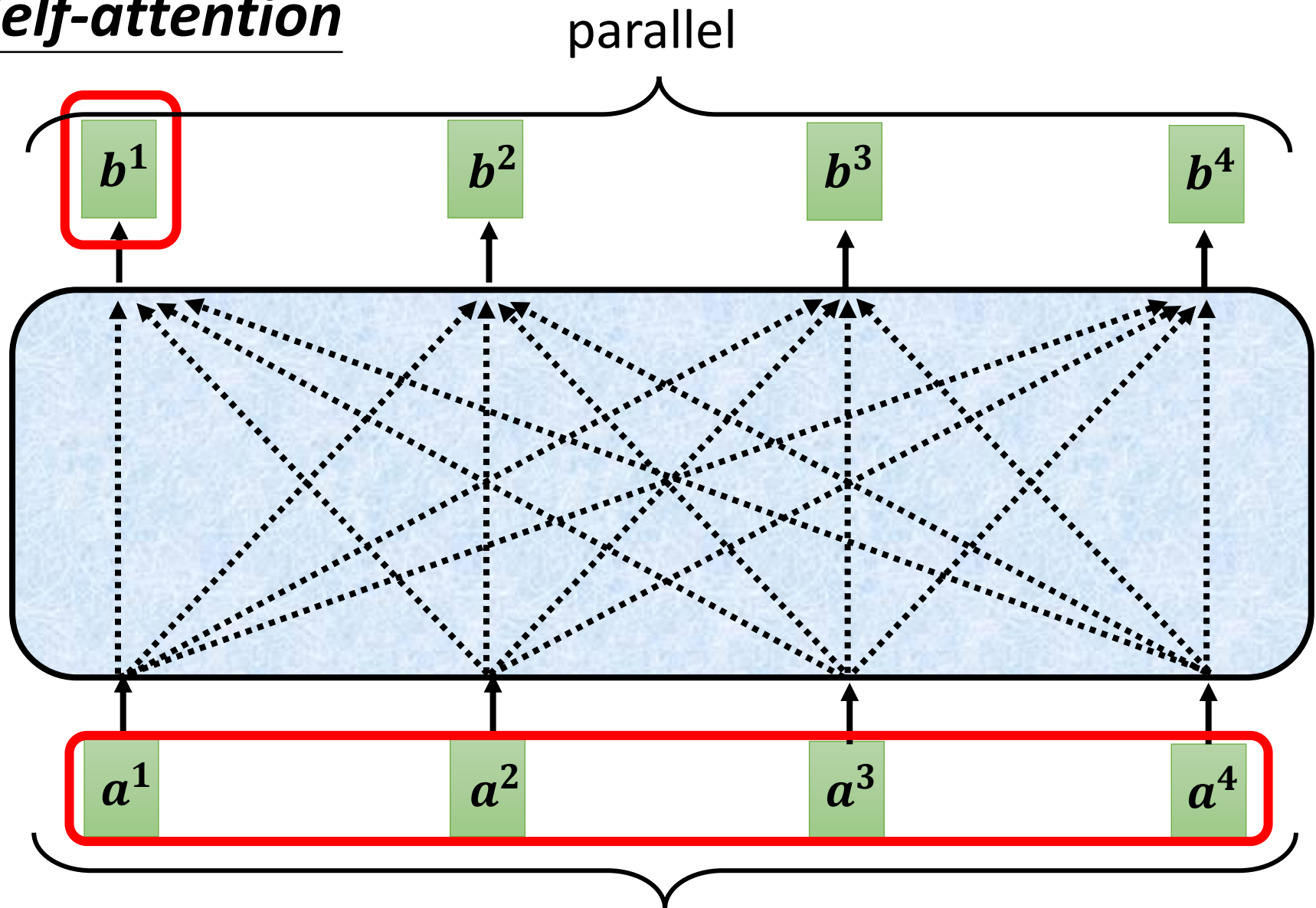
Self-attention

Extract information based on attention scores

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



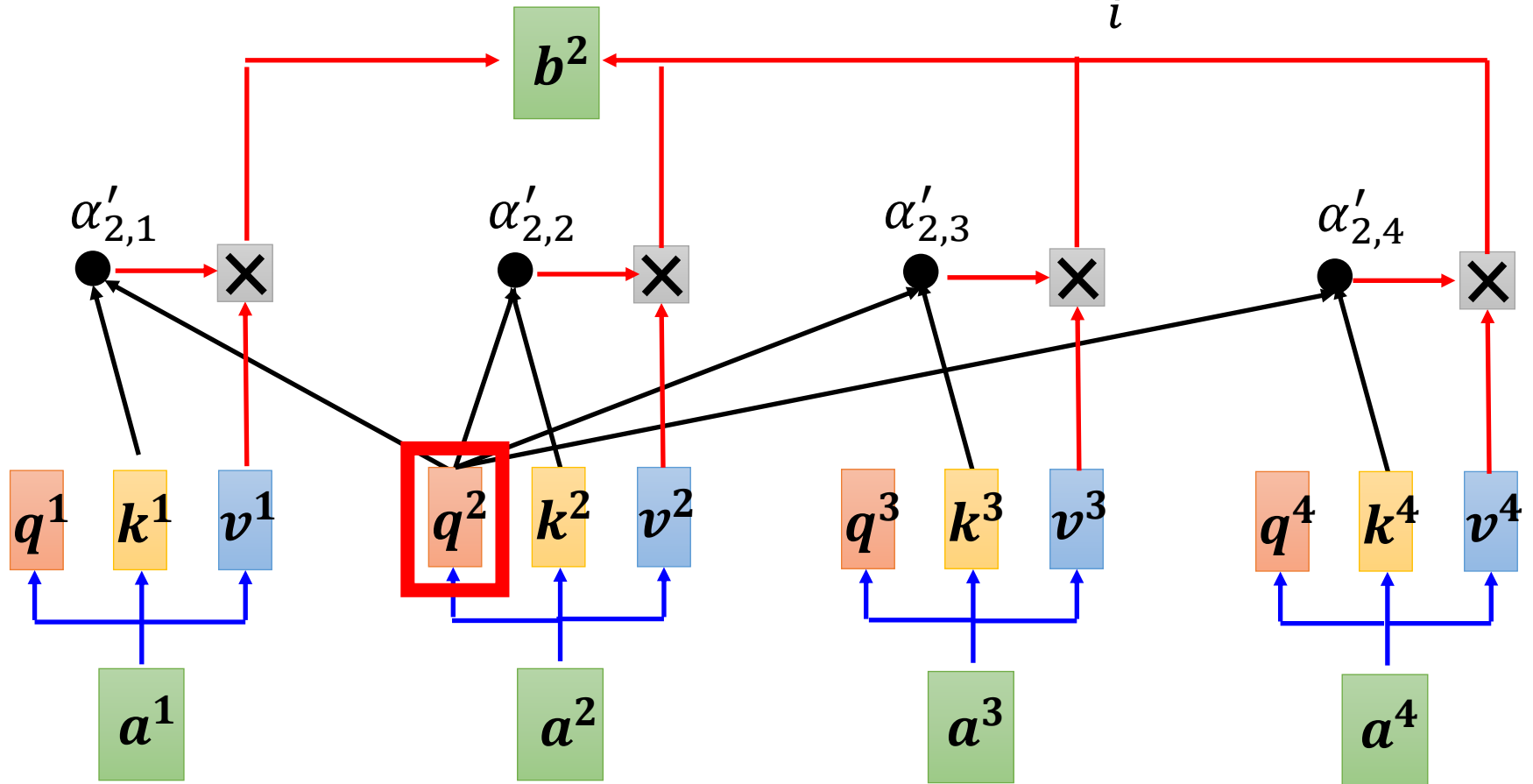
Self-attention



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

Self-attention

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



Self-attention

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

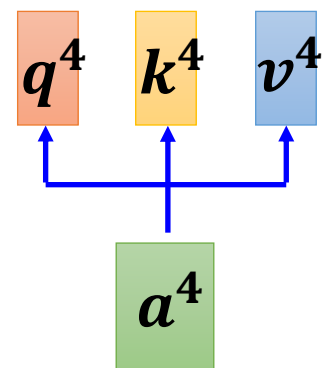
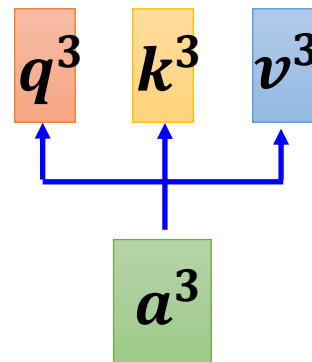
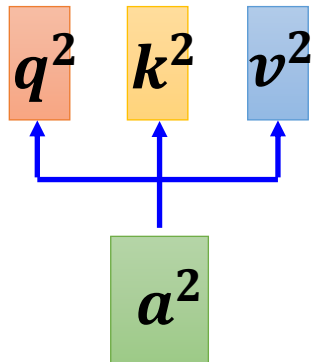
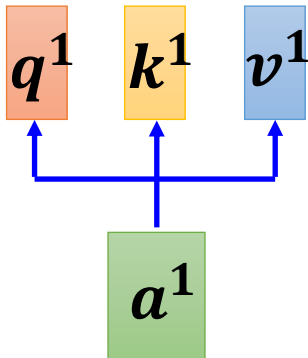
$$\begin{array}{c} \boxed{q^1} \boxed{q^2} \boxed{q^3} \boxed{q^4} \\ Q \end{array} = \begin{array}{c} \boxed{W^q} \begin{array}{c} \boxed{a^1} \boxed{a^2} \boxed{a^3} \boxed{a^4} \\ I \end{array} \end{array}$$

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

$$\begin{array}{c} \boxed{k^1} \boxed{k^2} \boxed{k^3} \boxed{k^4} \\ K \end{array} = \begin{array}{c} \boxed{W^k} \begin{array}{c} \boxed{a^1} \boxed{a^2} \boxed{a^3} \boxed{a^4} \\ I \end{array} \end{array}$$

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

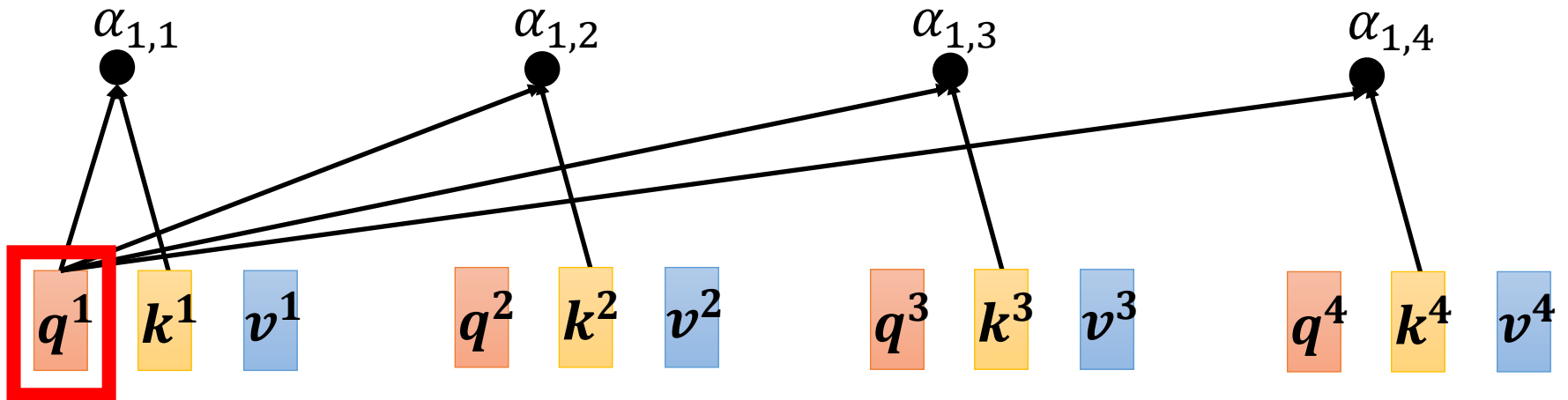
$$\begin{array}{c} \boxed{v^1} \boxed{v^2} \boxed{v^3} \boxed{v^4} \\ V \end{array} = \begin{array}{c} \boxed{W^v} \begin{array}{c} \boxed{a^1} \boxed{a^2} \boxed{a^3} \boxed{a^4} \\ I \end{array} \end{array}$$



Self-attention

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

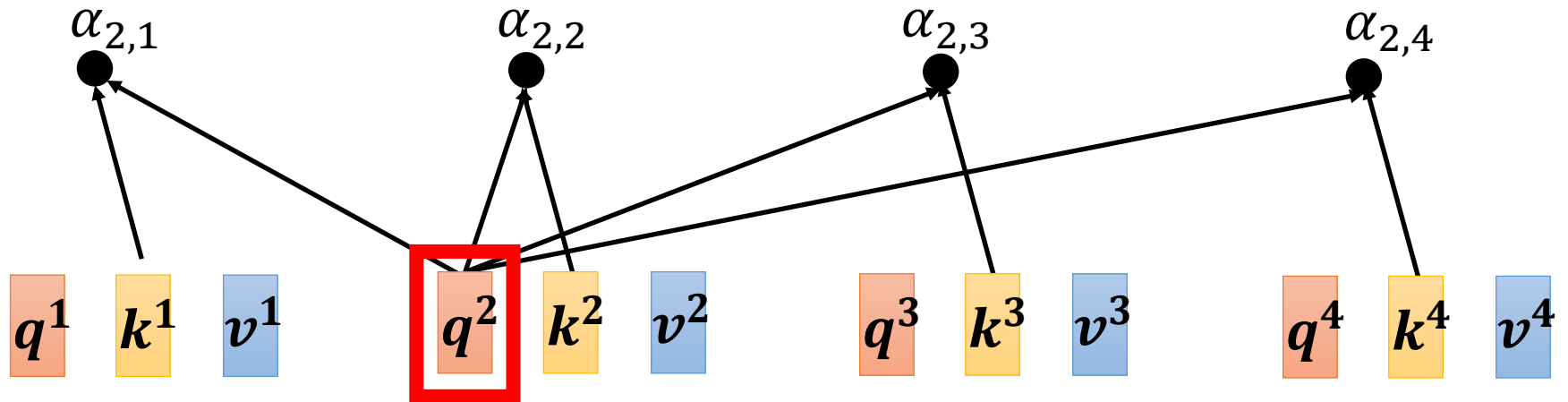
$$\begin{bmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \alpha_{1,3} \\ \alpha_{1,4} \end{bmatrix} = \begin{bmatrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{bmatrix} q^1$$



Self-attention

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

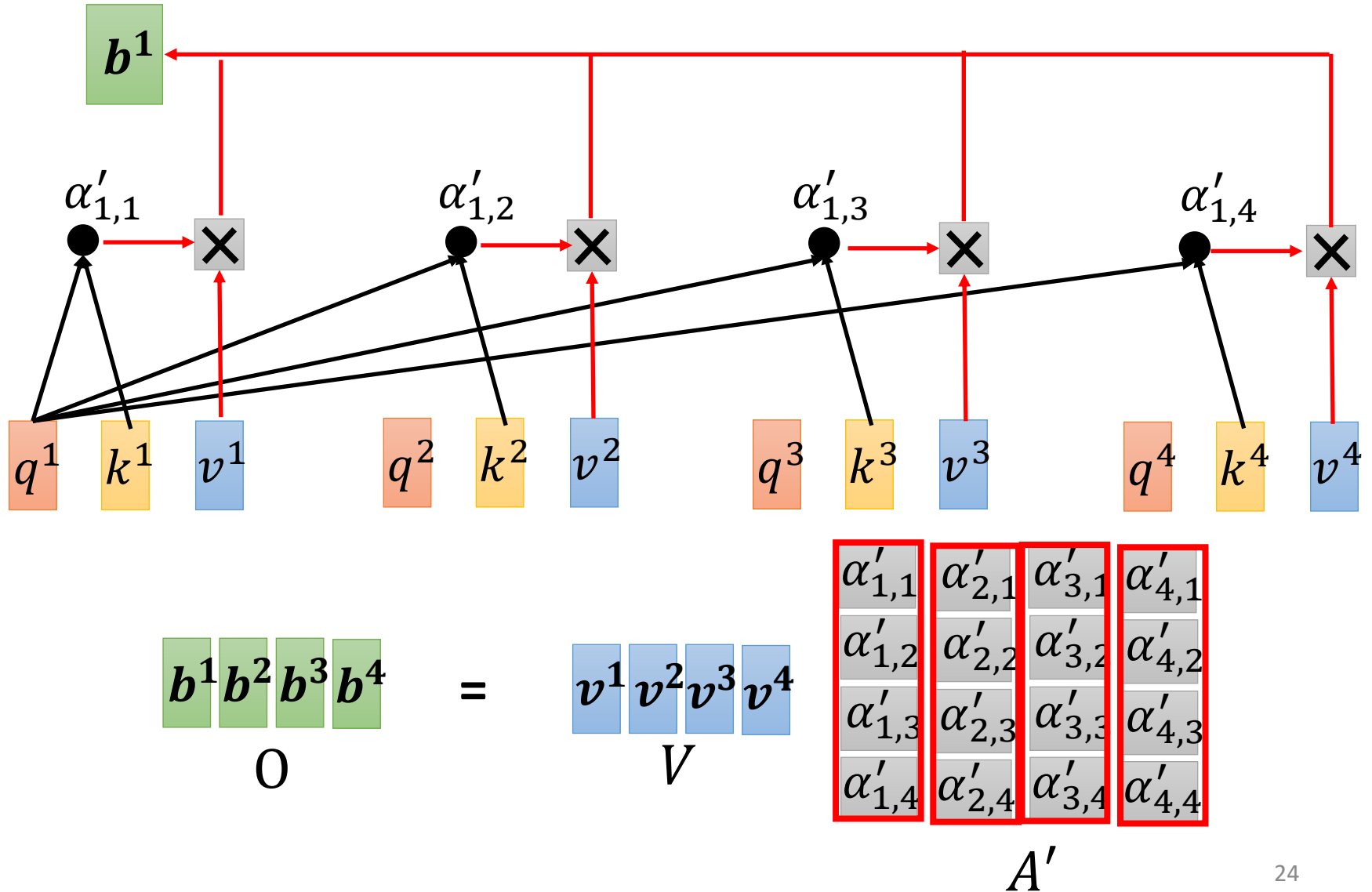
$$\begin{bmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \alpha_{1,3} \\ \alpha_{1,4} \end{bmatrix} = \begin{bmatrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{bmatrix} q^1$$



$$\begin{bmatrix} \alpha'_{1,1} & \alpha'_{2,1} & \alpha'_{3,1} & \alpha'_{4,1} \\ \alpha'_{1,2} & \alpha'_{2,2} & \alpha'_{3,2} & \alpha'_{4,2} \\ \alpha'_{1,3} & \alpha'_{2,3} & \alpha'_{3,3} & \alpha'_{4,3} \\ \alpha'_{1,4} & \alpha'_{2,4} & \alpha'_{3,4} & \alpha'_{4,4} \end{bmatrix} \xleftarrow{\text{softmax}} \begin{bmatrix} \alpha_{1,1} & \alpha_{2,1} & \alpha_{3,1} & \alpha_{4,1} \\ \alpha_{1,2} & \alpha_{2,2} & \alpha_{3,2} & \alpha_{4,2} \\ \alpha_{1,3} & \alpha_{2,3} & \alpha_{3,3} & \alpha_{4,3} \\ \alpha_{1,4} & \alpha_{2,4} & \alpha_{3,4} & \alpha_{4,4} \end{bmatrix} = \begin{bmatrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{bmatrix} \begin{bmatrix} q^1 & q^2 & q^3 & q^4 \end{bmatrix}$$

$A' \quad \text{softmax} \quad A \quad = \quad K^T \quad Q$

Self-attention



Self-attention

$$\begin{array}{lcl} Q & = & W^q I \\ K & = & W^k I \\ V & = & W^v I \end{array}$$

Parameters to be learned

$$A' \leftarrow A = K^T Q$$

Attention Matrix

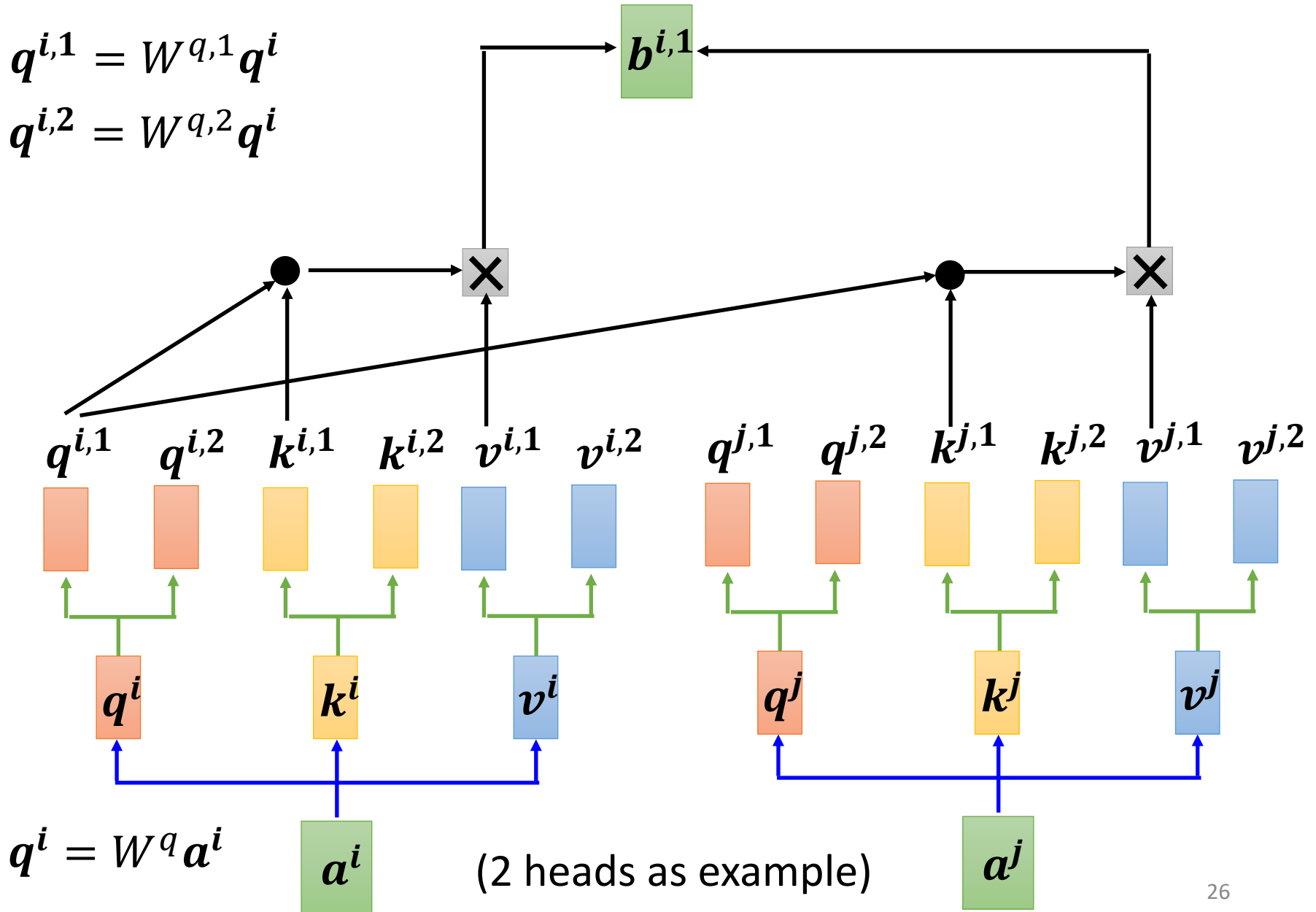
$$O = V A'$$

Multi-head Self-attention

Different types of relevance

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

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

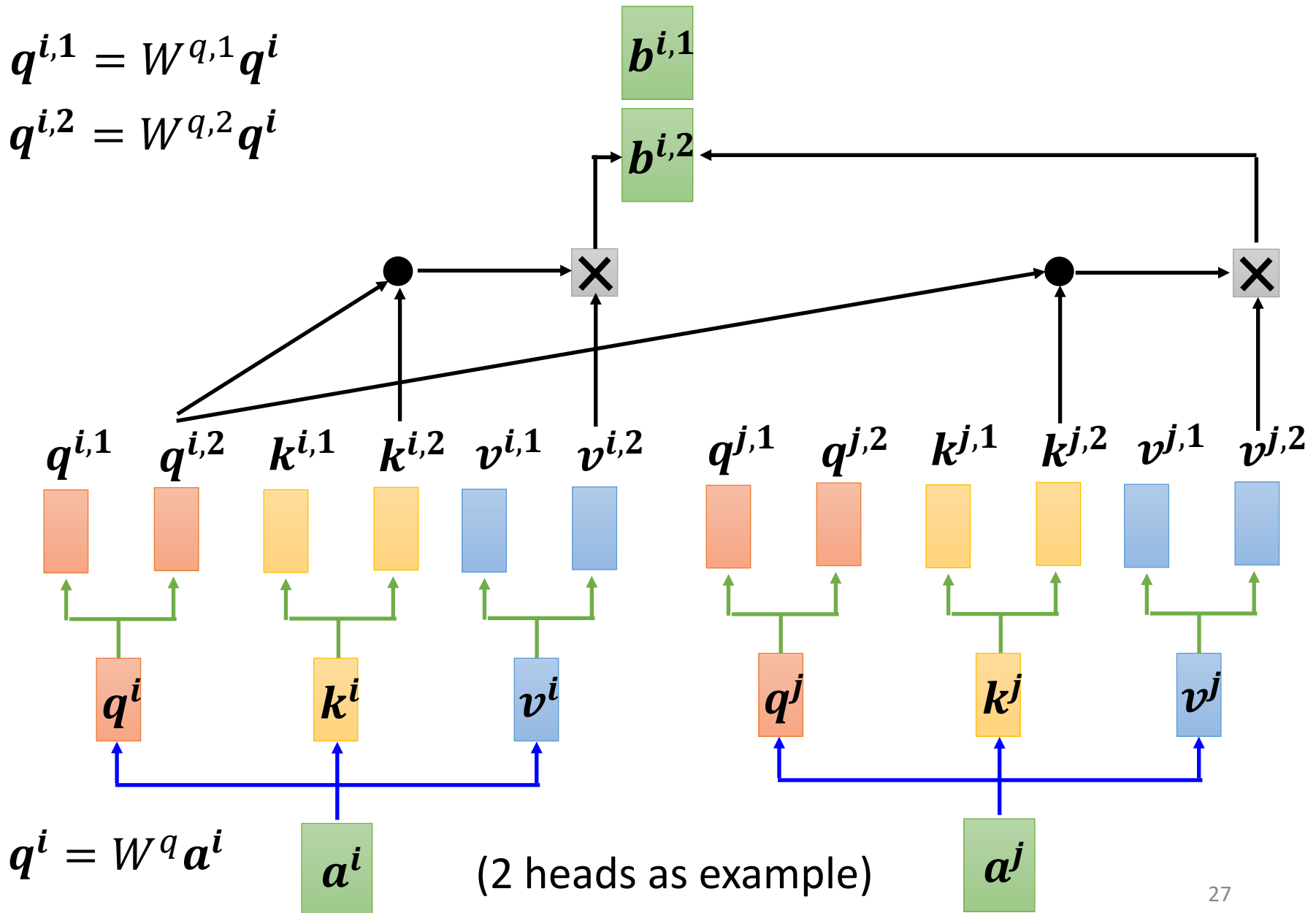


Multi-head Self-attention

Different types of relevance

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

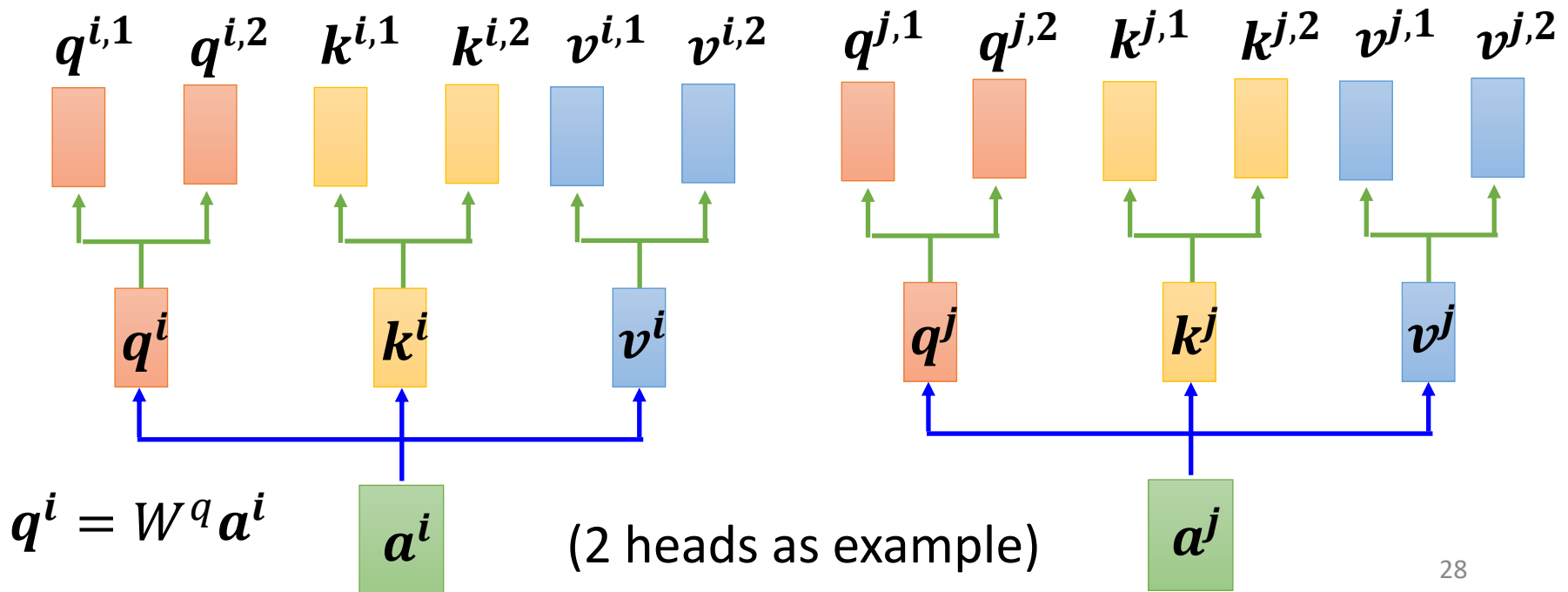
$$q^{i,2} = W^{q,2} q^i$$



Multi-head Self-attention

Different types of relevance

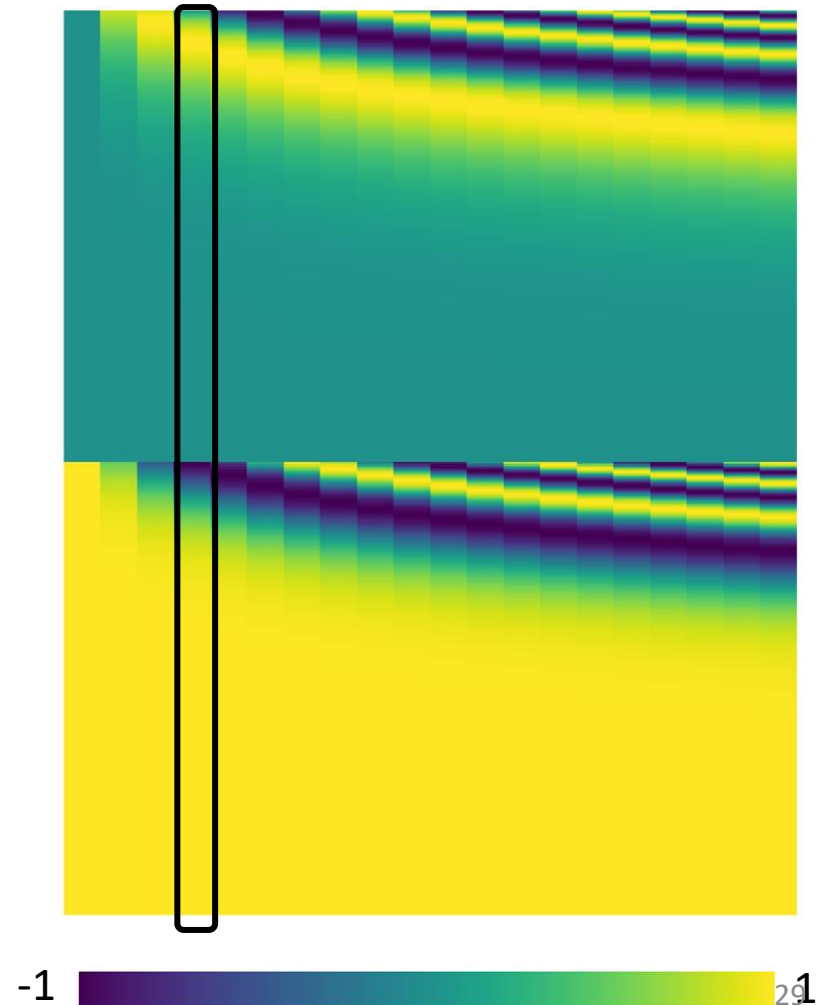
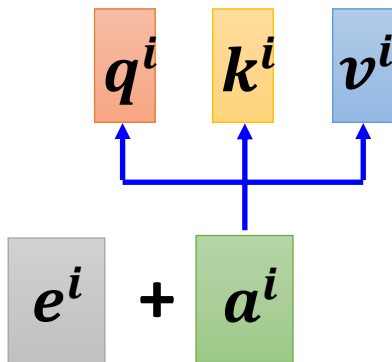
$$b^i = W^O \begin{bmatrix} b^{i,1} \\ b^{i,2} \end{bmatrix}$$



Positional Encoding

Each column represents a positional vector e^i

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

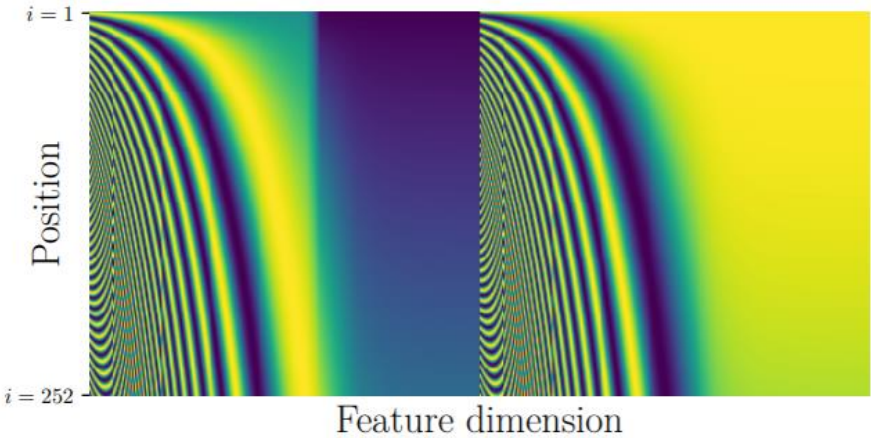


<https://arxiv.org/abs/2003.09229>

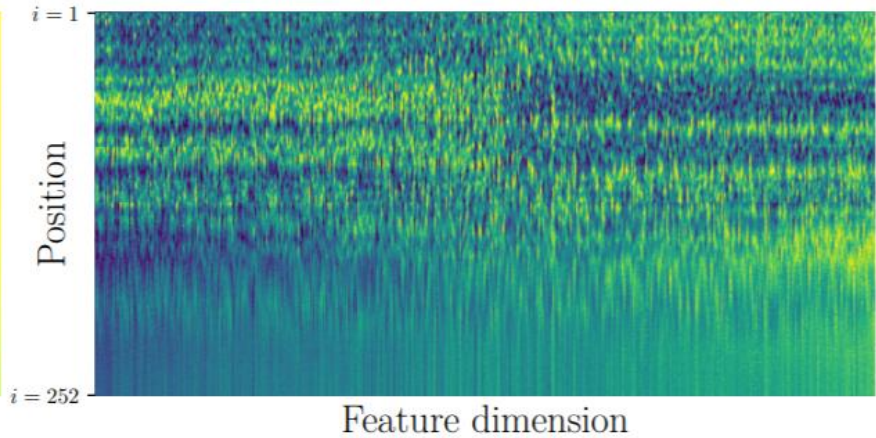
Table 1. Comparing position representation methods

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

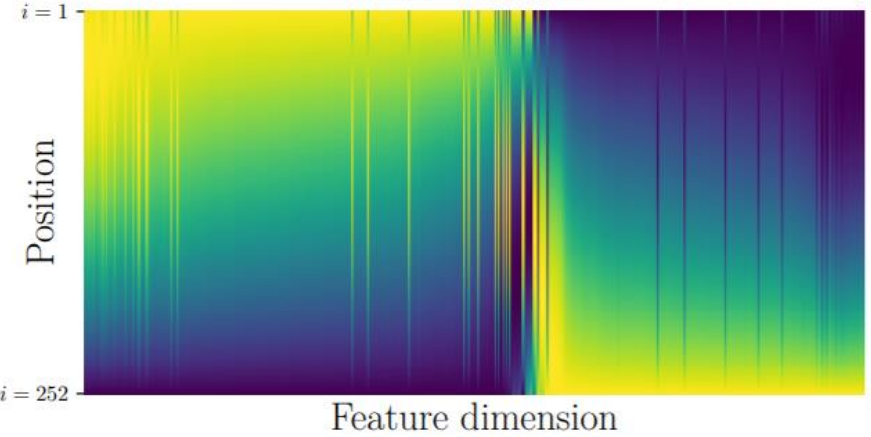
(a) Sinusoidal



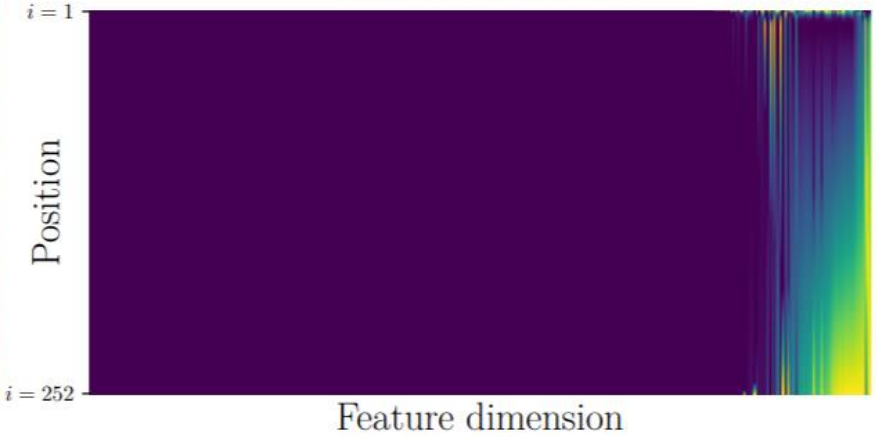
(b) Position embedding



(c) FLOATER



(d) RNN



Many applications ...



Transformer

<https://arxiv.org/abs/1706.03762>



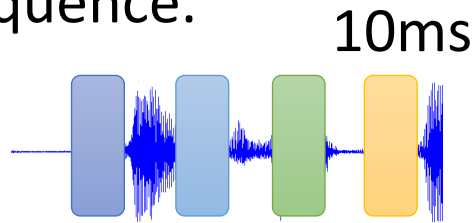
BERT

<https://arxiv.org/abs/1810.04805>

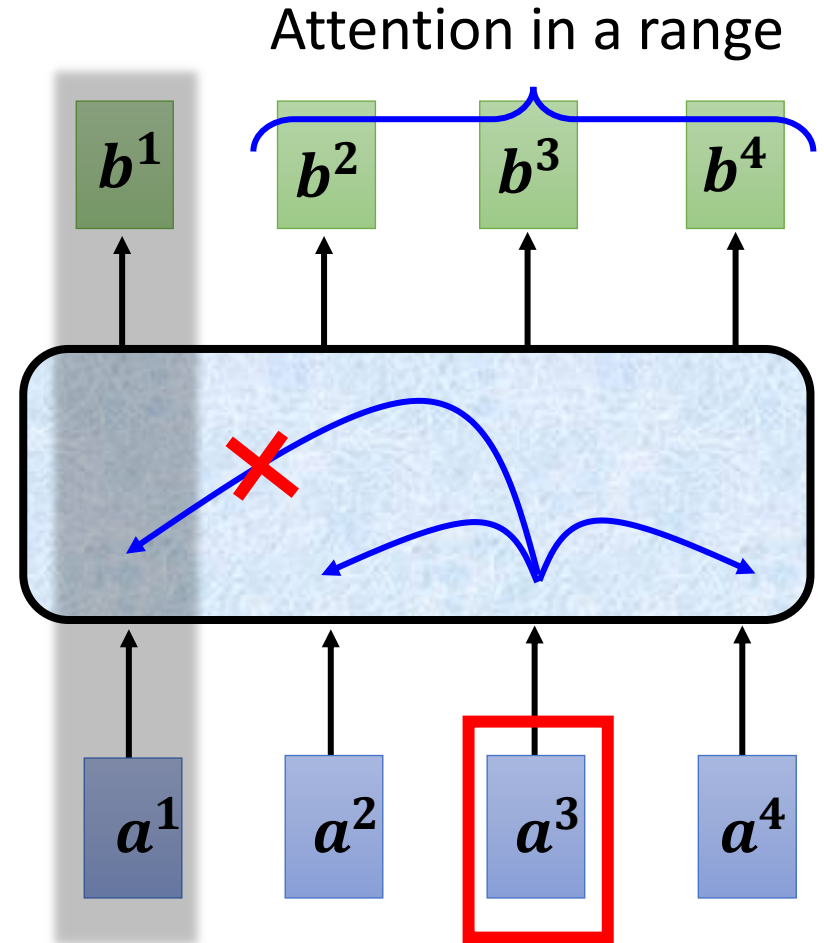
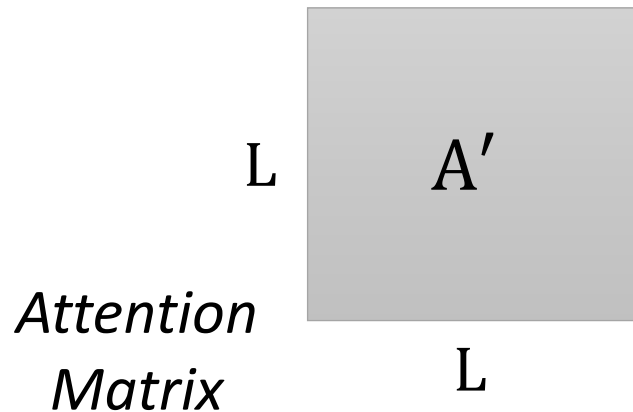
Widely used in Natural Language Processing (NLP)!

Self-attention for Speech

Speech is a very long vector sequence.



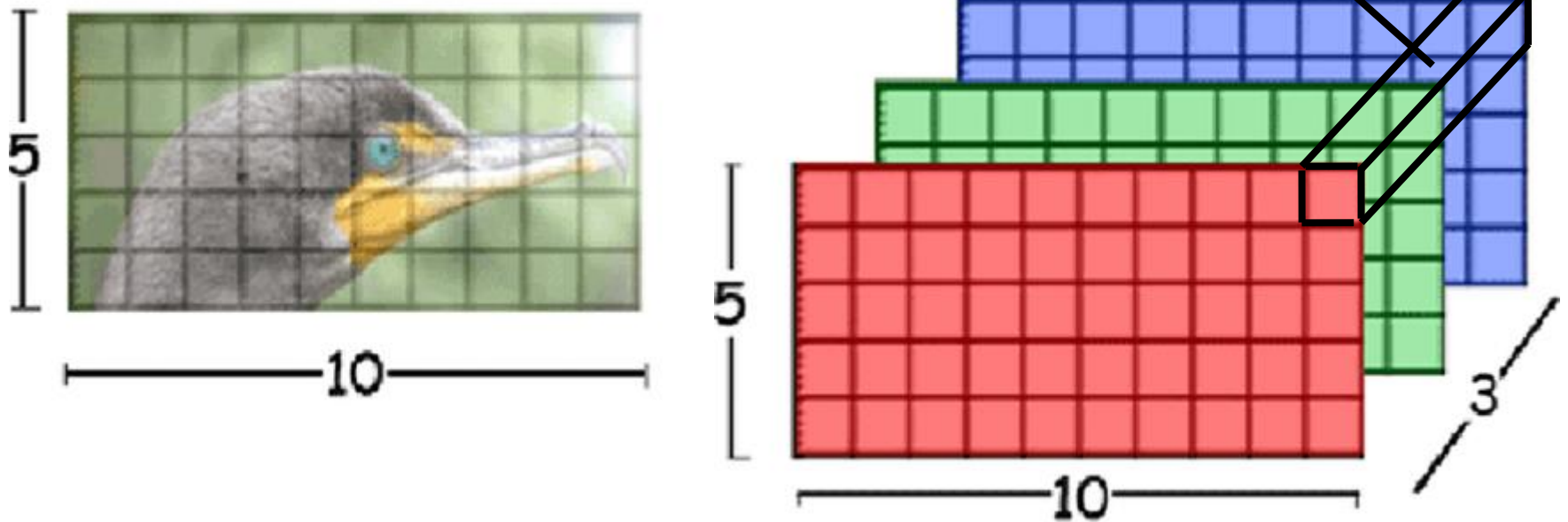
If input sequence is length L



Truncated Self-attention

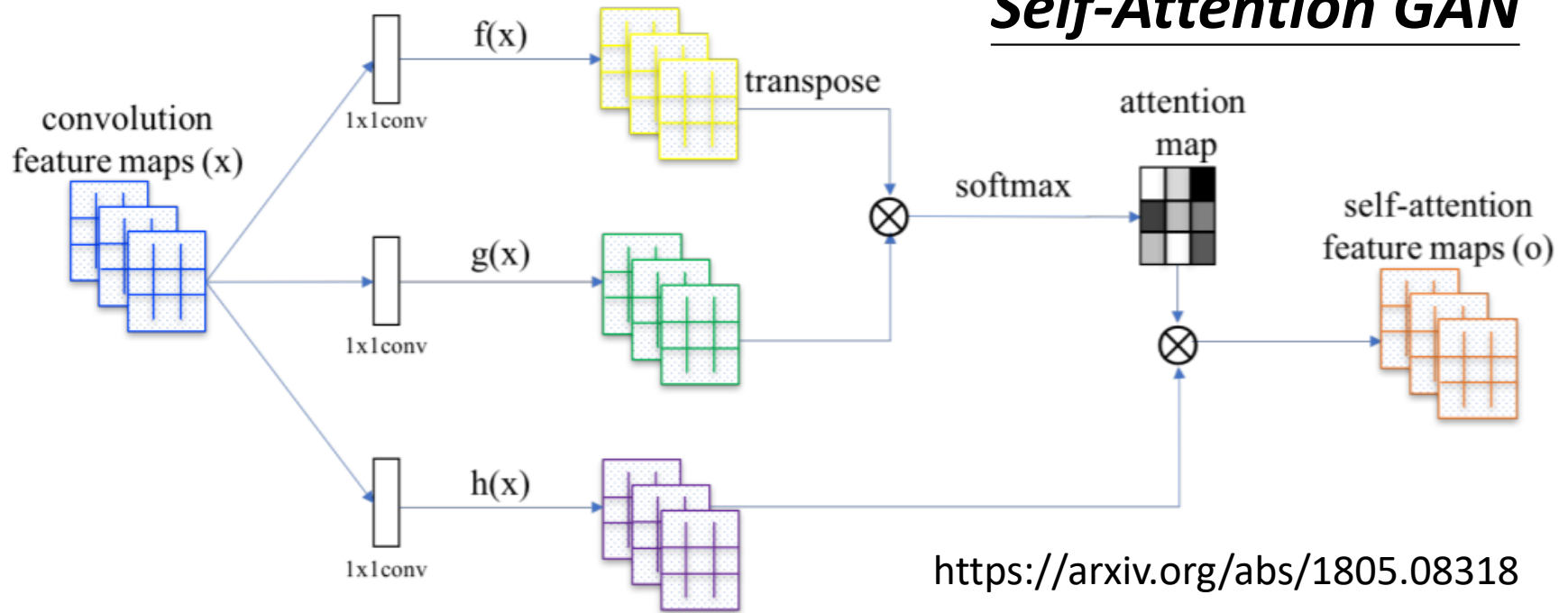
Self-attention for Image

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

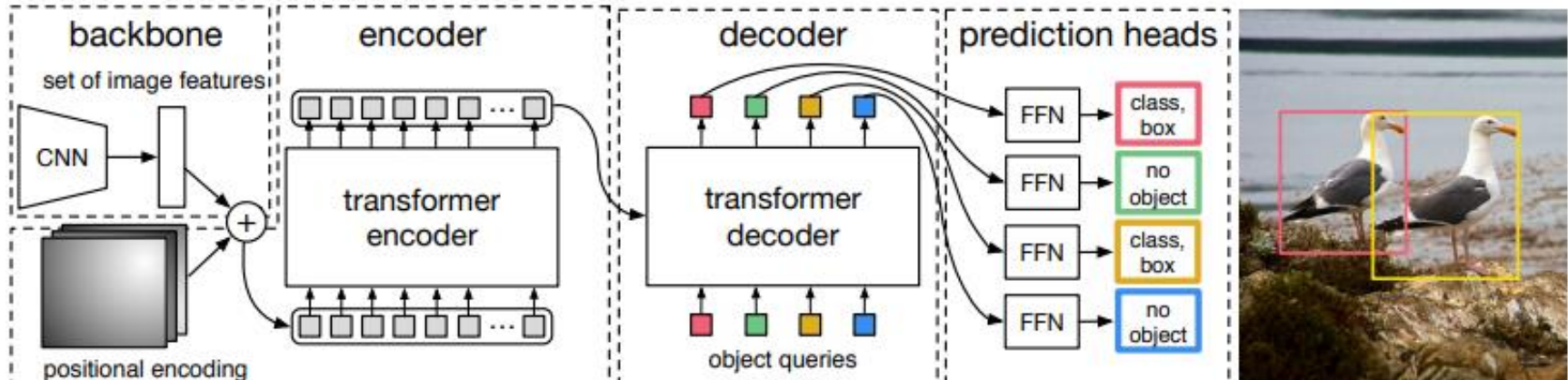


Source of image: https://www.researchgate.net/figure/Color-image-representation-and-RGB-matrix_fig15_282798184

Self-Attention GAN

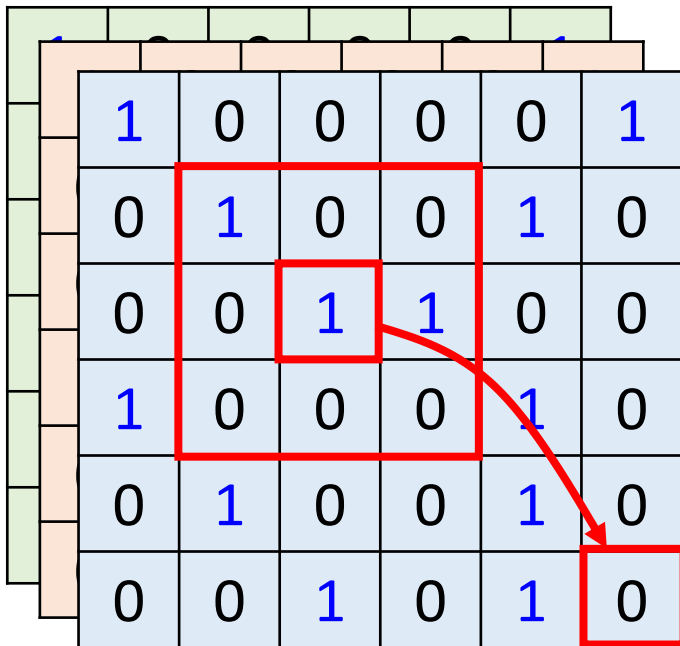


DEtection Transformer (DETR)



<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.

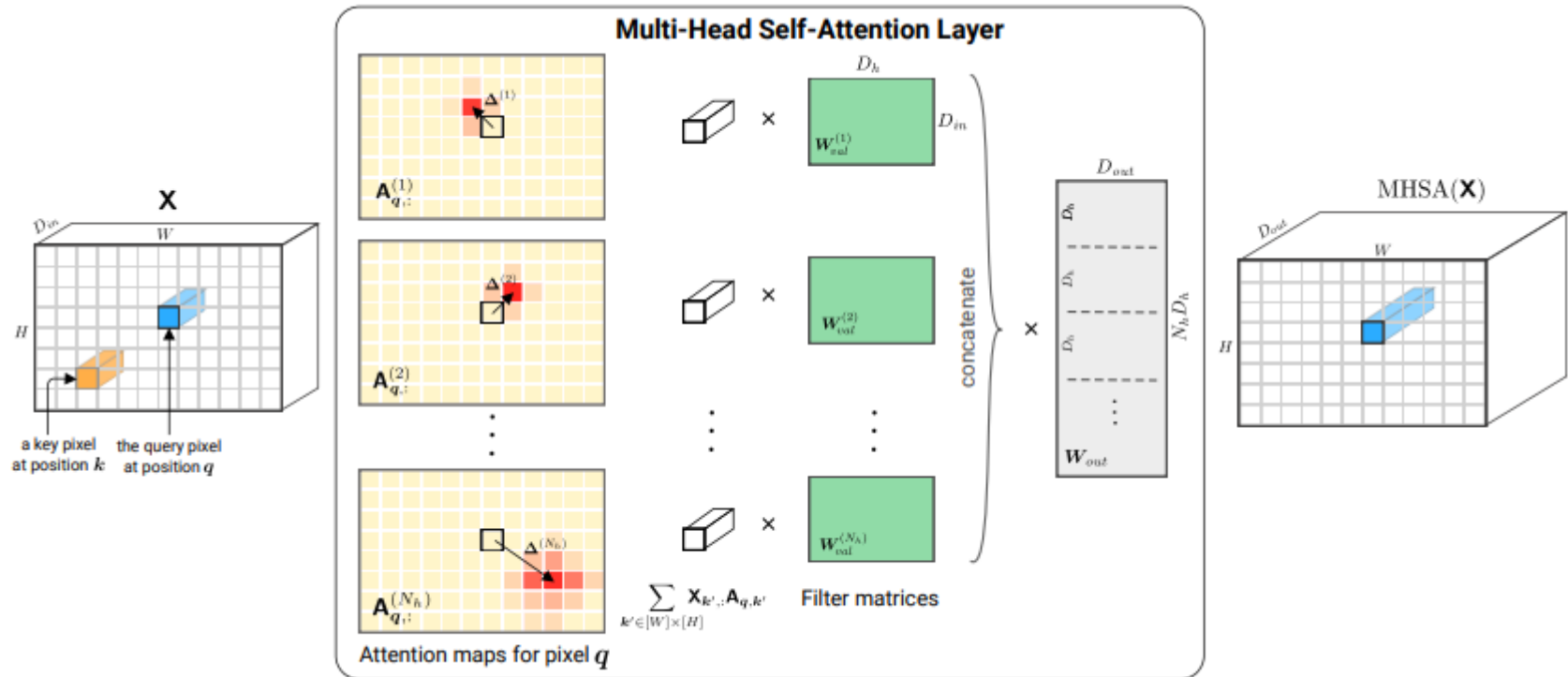
Self-attention v.s. CNN

這篇paper用嚴謹的數學證明
CNN就是self-attention的特例

越flexible的data會需要越多的data
才不容易overfitting

Self-attention

CNN



On the Relationship between Self-Attention and
Convolutional Layers

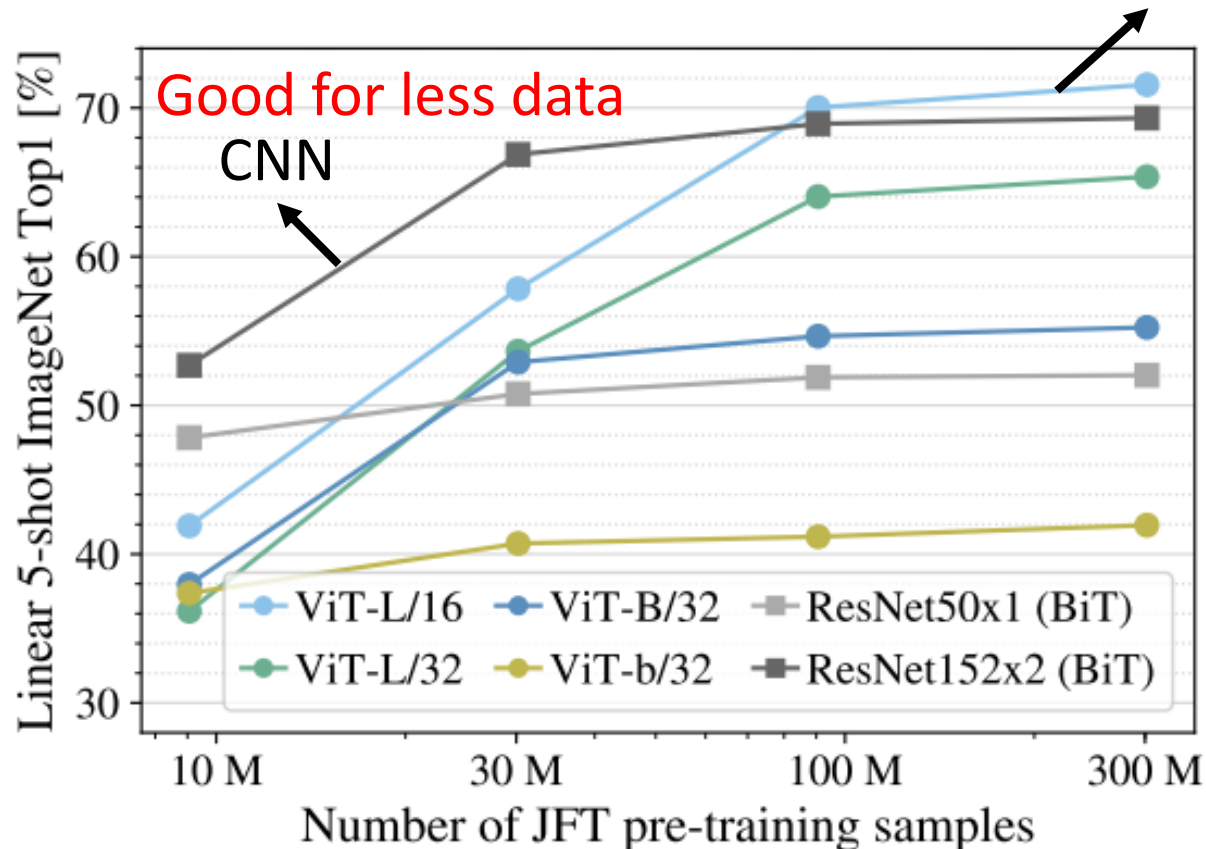
<https://arxiv.org/abs/1911.03584>

在資料量少的時候CNN表現比較好
但資料量來到3億筆的時候
self-attention的表現比較好

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>

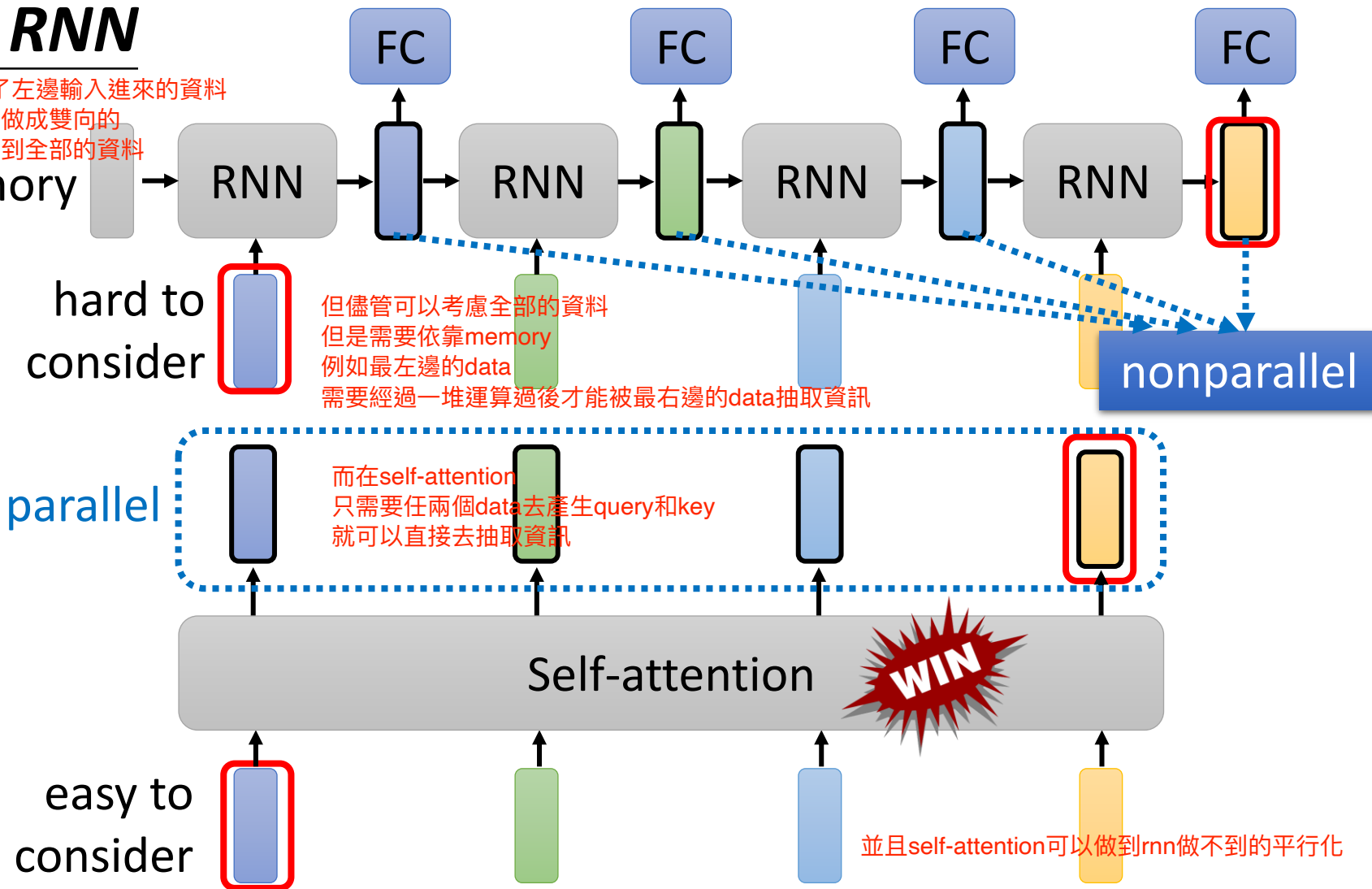
Self-attention

v.s. RNN

rnn僅考慮了左邊輸入進來的資料
但其實可以做成雙向的
就可以考慮到全部的資料

這門課不會講rnn因為rnn已經可以被self-attention取代了

Recurrent Neural Network (RNN)



Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention

這篇paper告訴你，其實transformer加上某些東西就會變成rnn

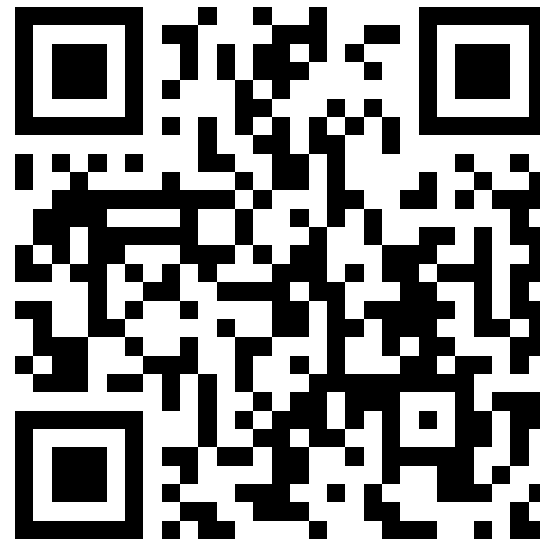
<https://arxiv.org/abs/2006.16236>

To learn more about RNN



<https://youtu.be/xCGidAeyS4M>

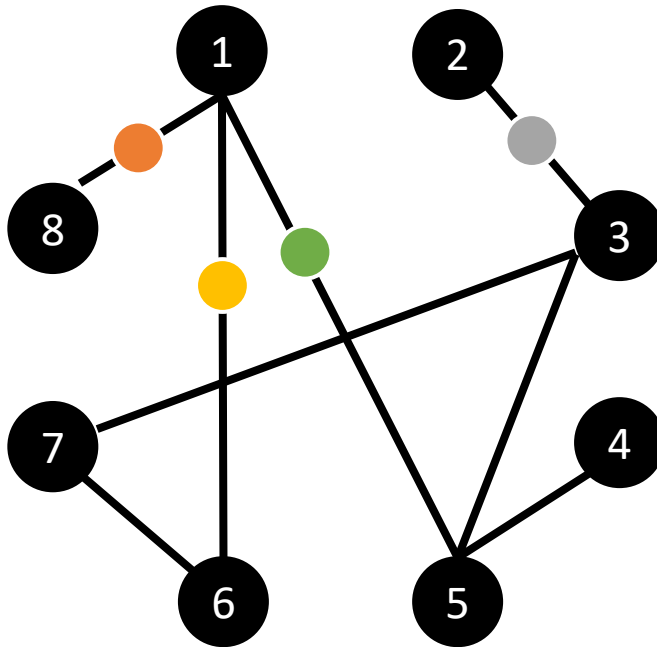
(in Mandarin)



<https://youtu.be/Jjy6ER0bHv8>

(in English)

Self-attention for Graph



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

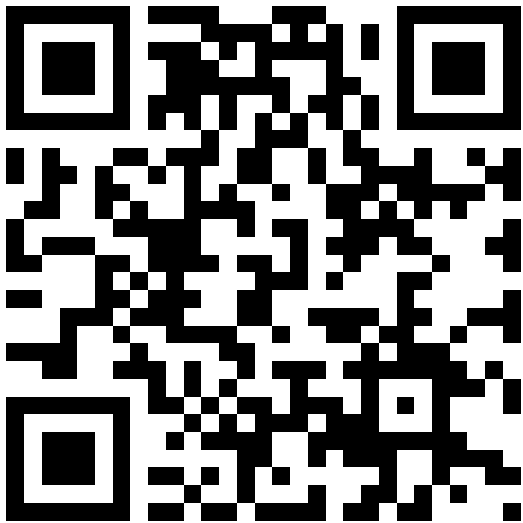
Attention Matrix

	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								
7								
8							0	

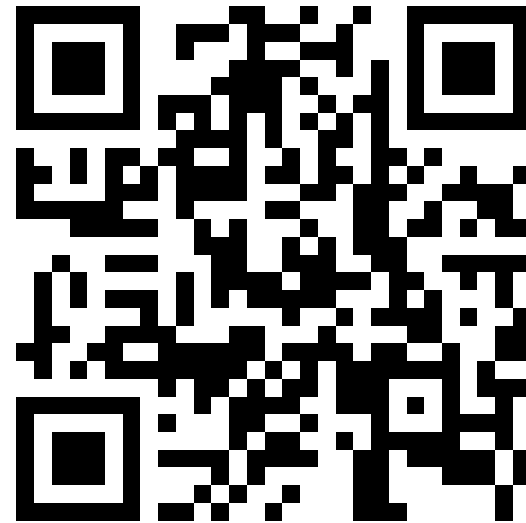
This is one type of **Graph Neural Network (GNN)**.

Self-attention for Graph

- To learn more about GNN ...



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



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

To Learn More ...

self-attention有非常多種變形
而這篇paper正是在比較個種不同的model

Long Range Arena: A Benchmark for Efficient Transformers

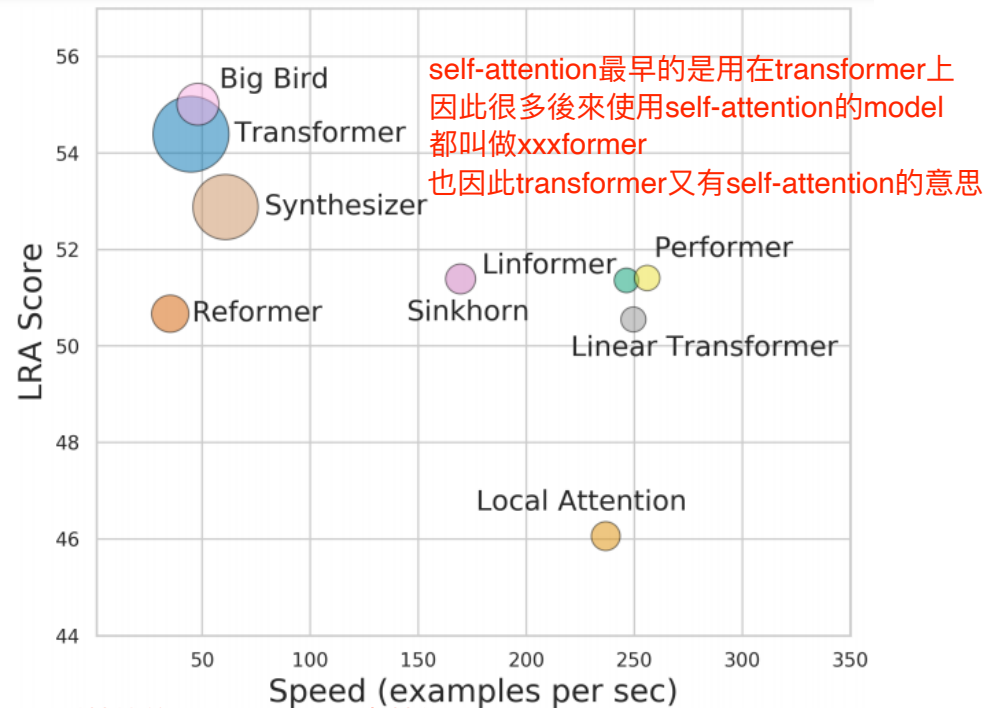
<https://arxiv.org/abs/2011.04006>

self-attention最大的問題就是運算量非常的大
所以如何efficient是未來探討的重點

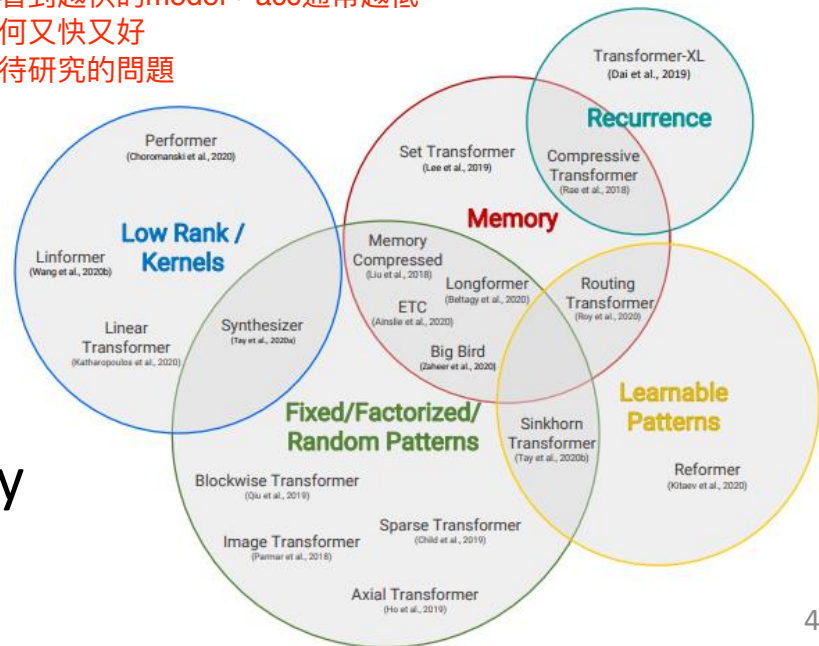
這篇paper介紹各式各樣transformer的變形

Efficient Transformers: A Survey

<https://arxiv.org/abs/2009.06732>



可以看到越快的model，acc通常越低
而如何又快又好
是個待研究的問題



Q&A