Transformer 系列模型

Presenter: 王浩

Advisor: 林嘉文

Date: 2023/07/26

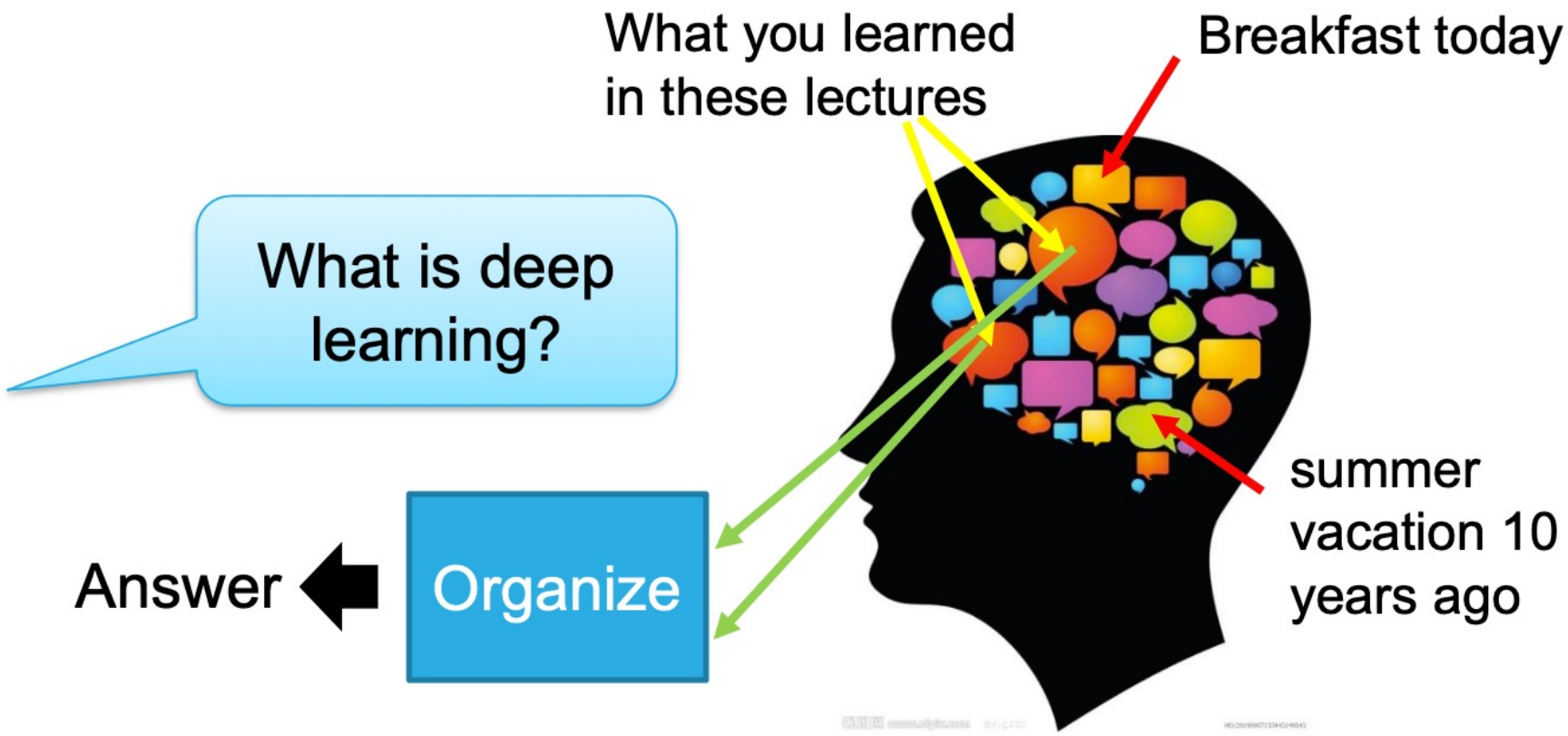
Outlines

- Attention Mechanism
- Transformer
- Vision Transformer
- Swin Transformer

Outlines

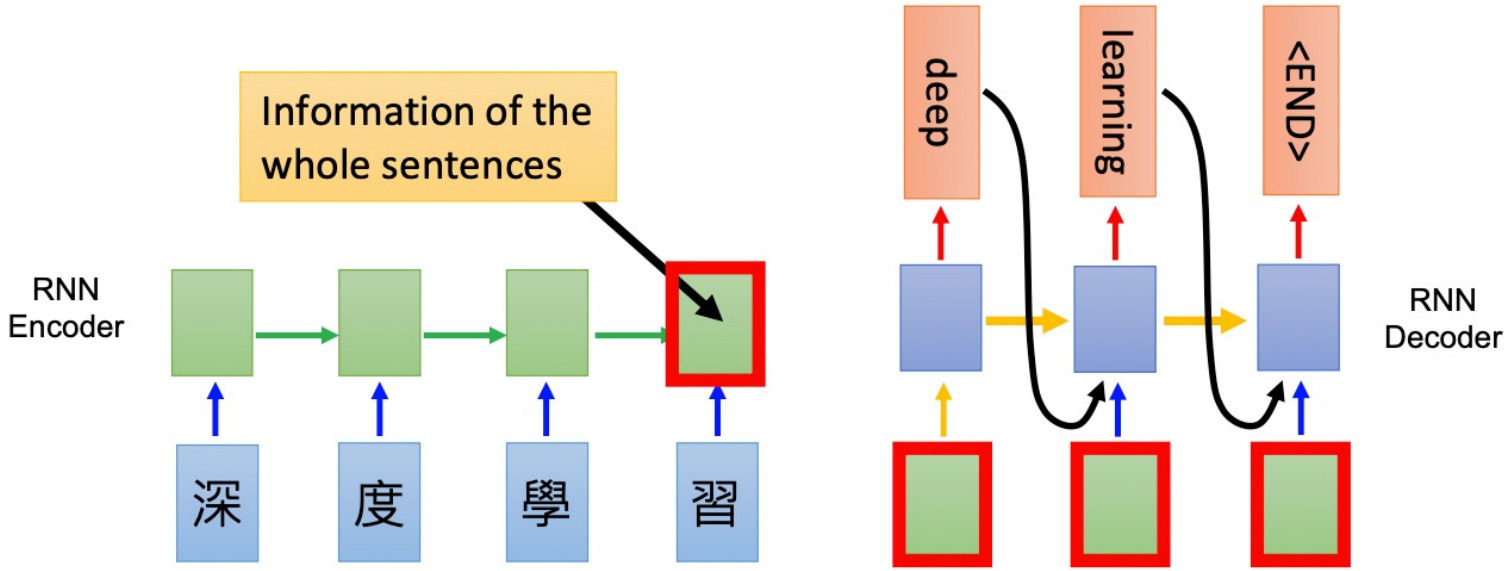
- Attention Mechanism
- Transformer
- Vision Transformer
- Swin Transformer

Attention Mechanism

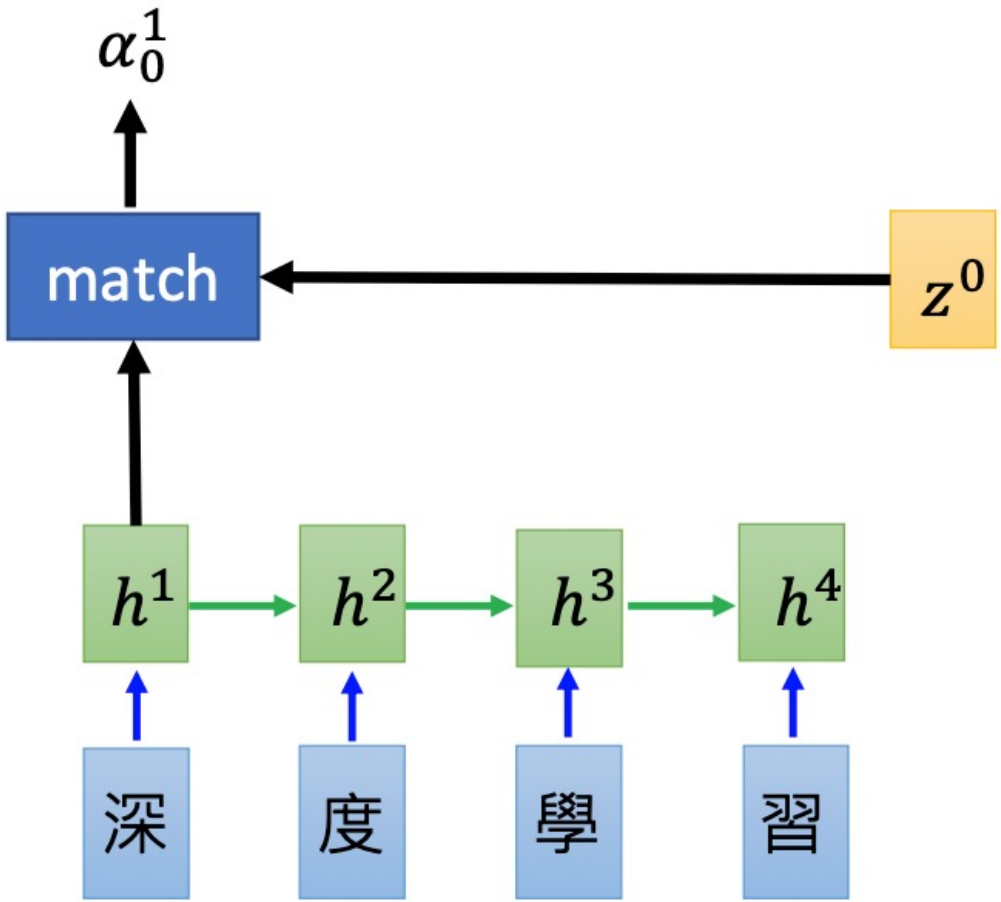


Machine Translation

- Sequence-to-sequence learning: both input and output are both sequences with different lengths.
- E.g. 深度學習 → deep learning



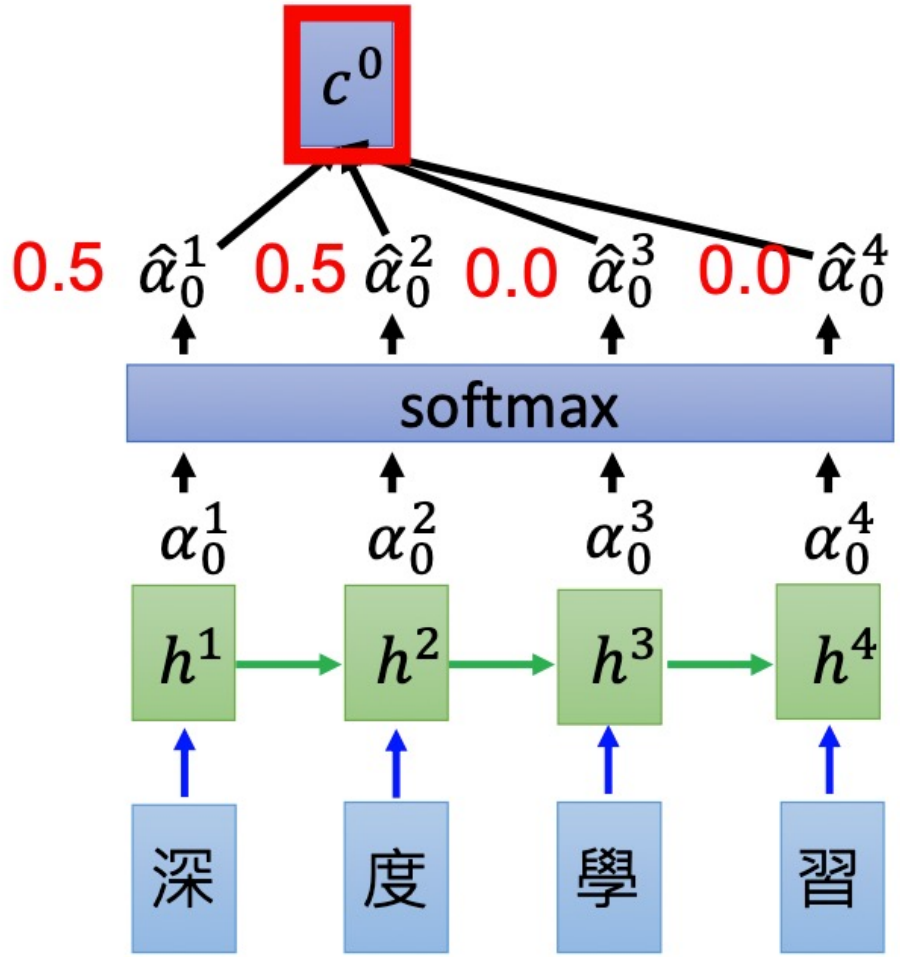
Machine Translation with Attention

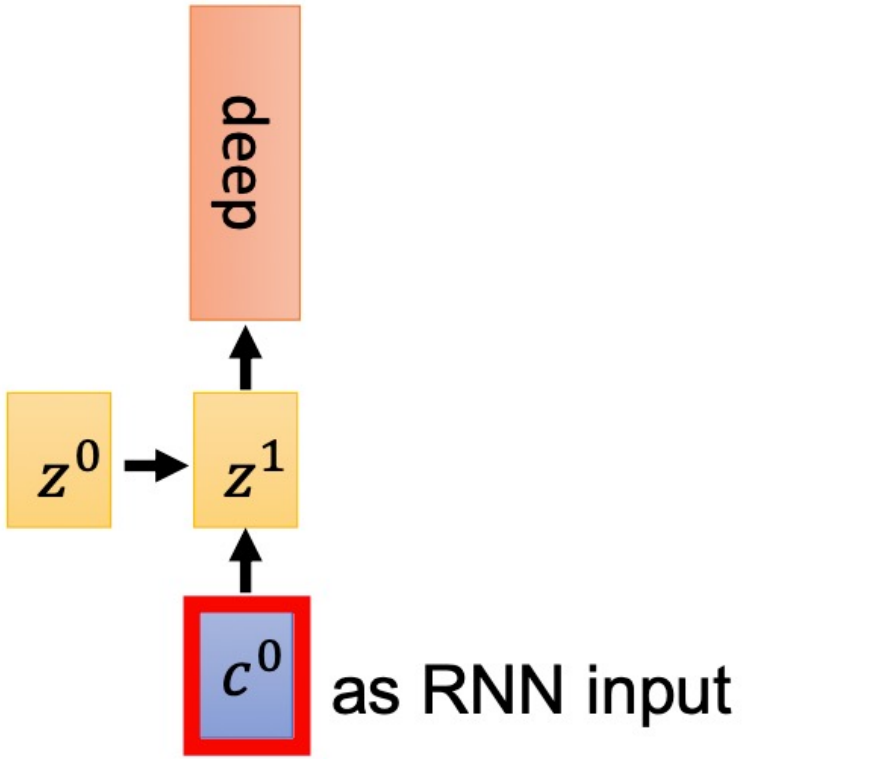


What is **match** ?

- Cosine similarity of z and h
- Small NN whose input is z and h, output a scalar
- $\alpha = h^T W z$

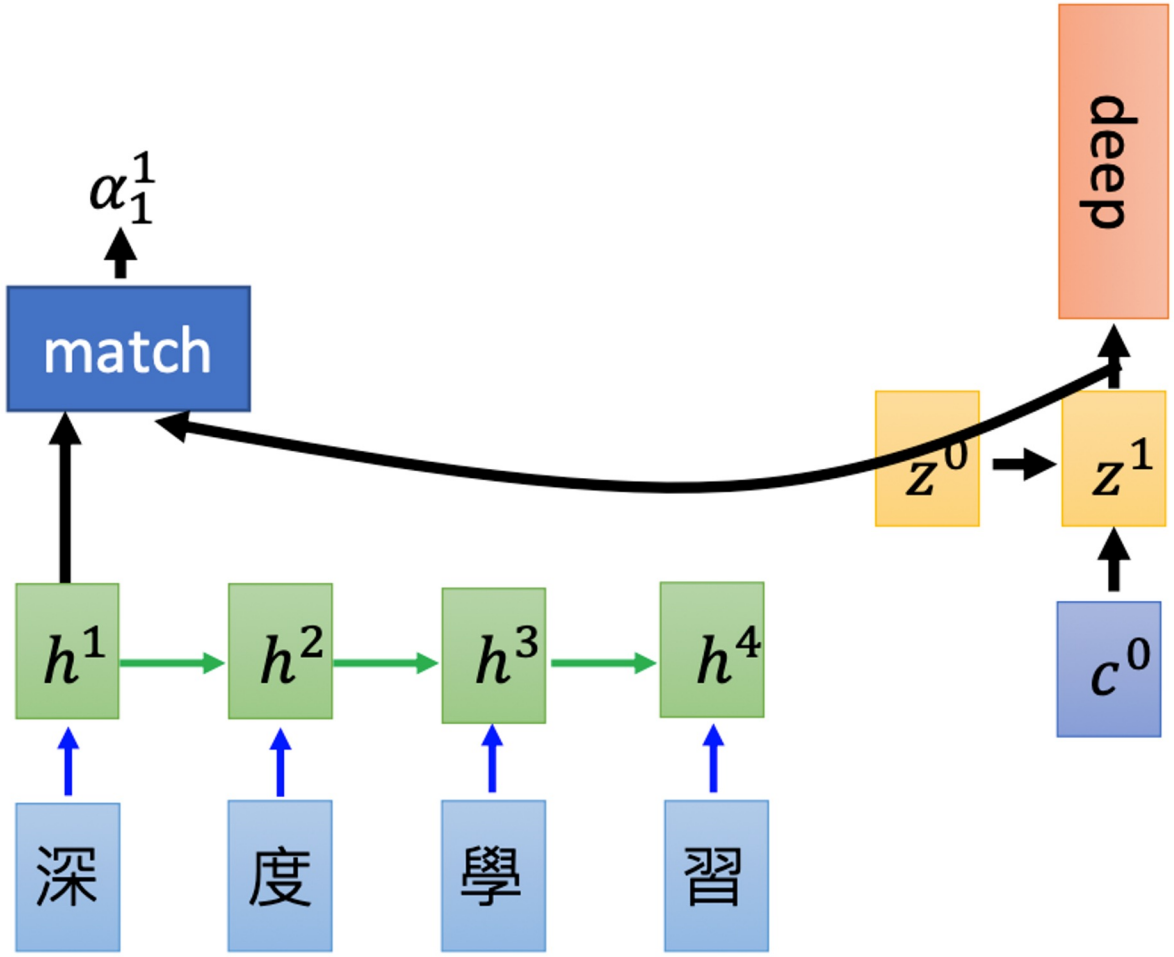
Machine Translation with Attention



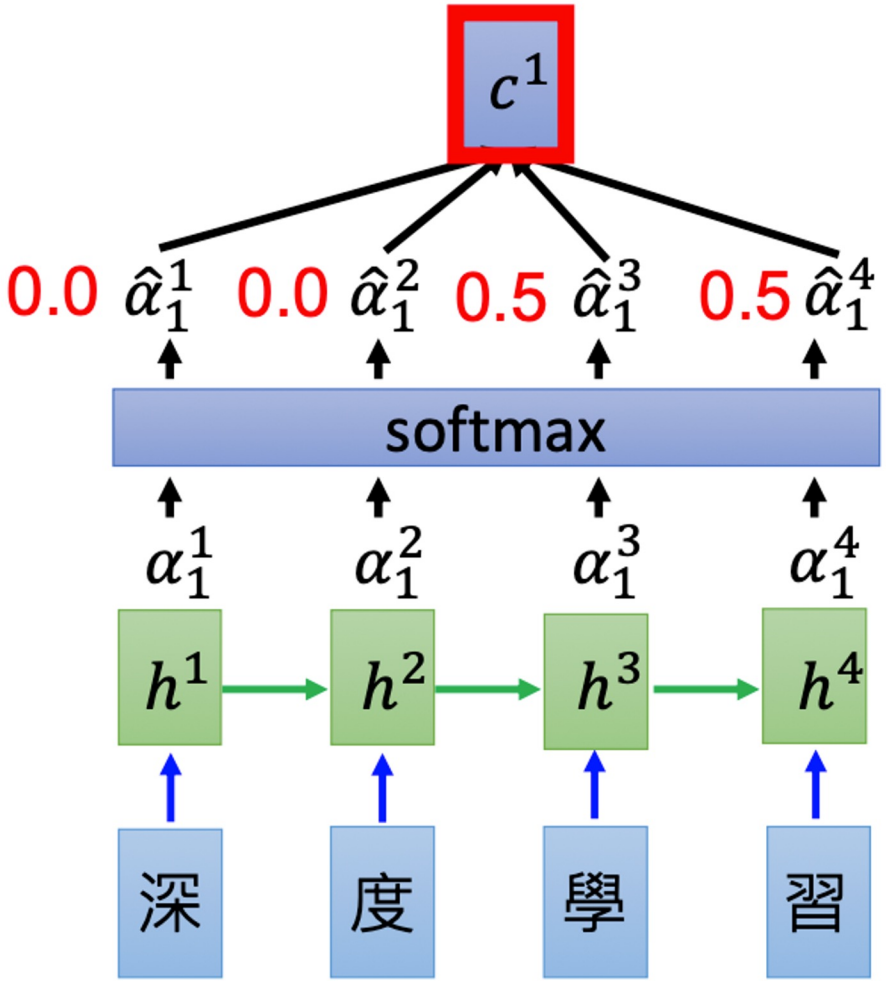


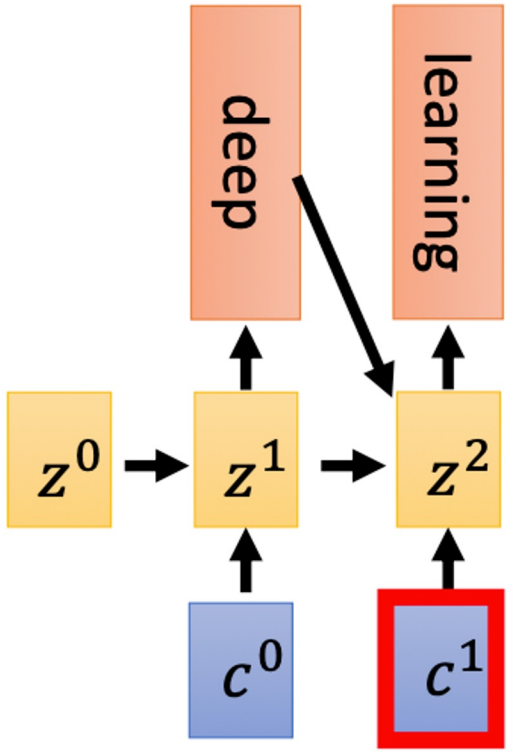
$$c^0 = \sum \hat{\alpha}_0^i h^i = 0.5h^1 + 0.5h^2$$

Machine Translation with Attention



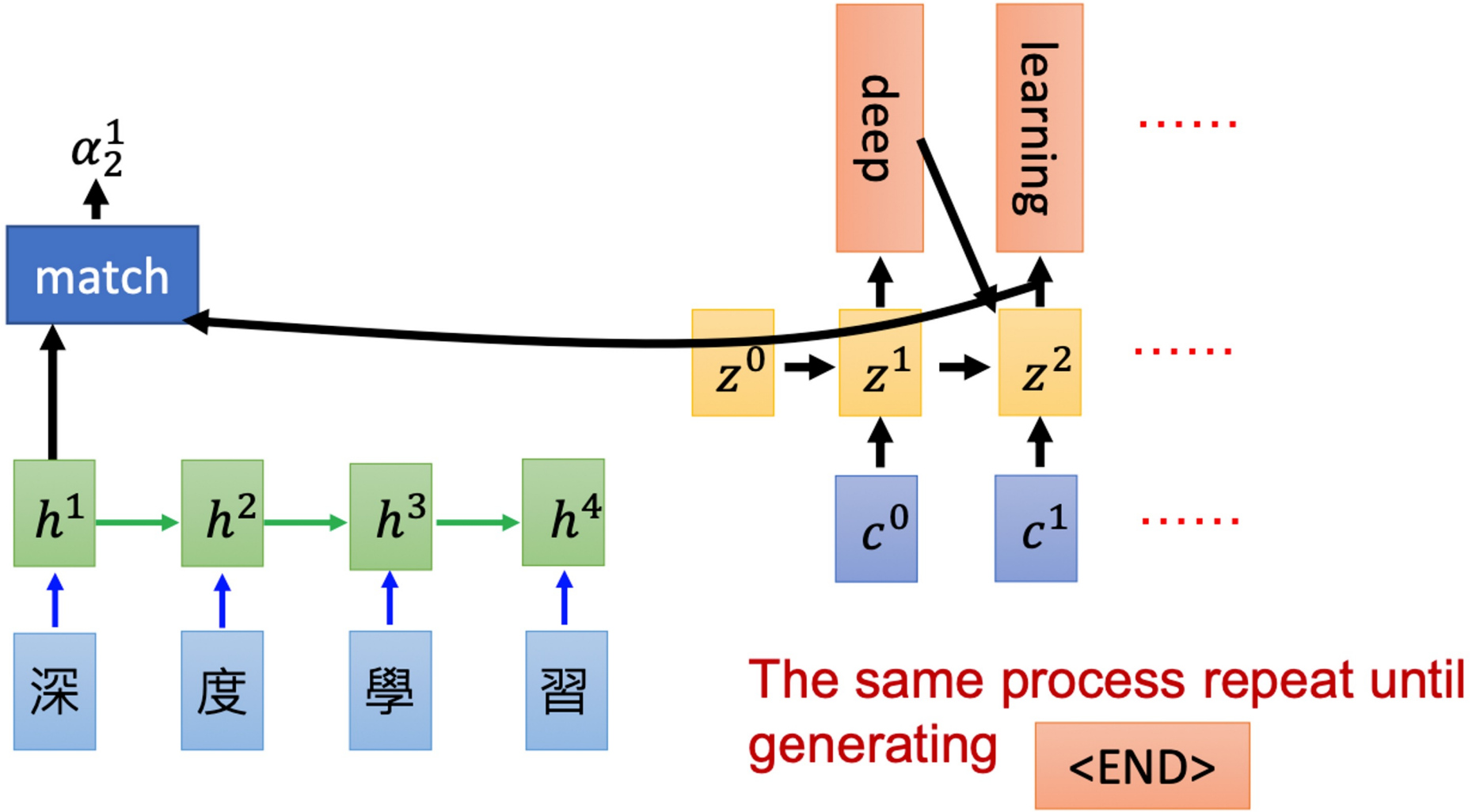
Machine Translation with Attention



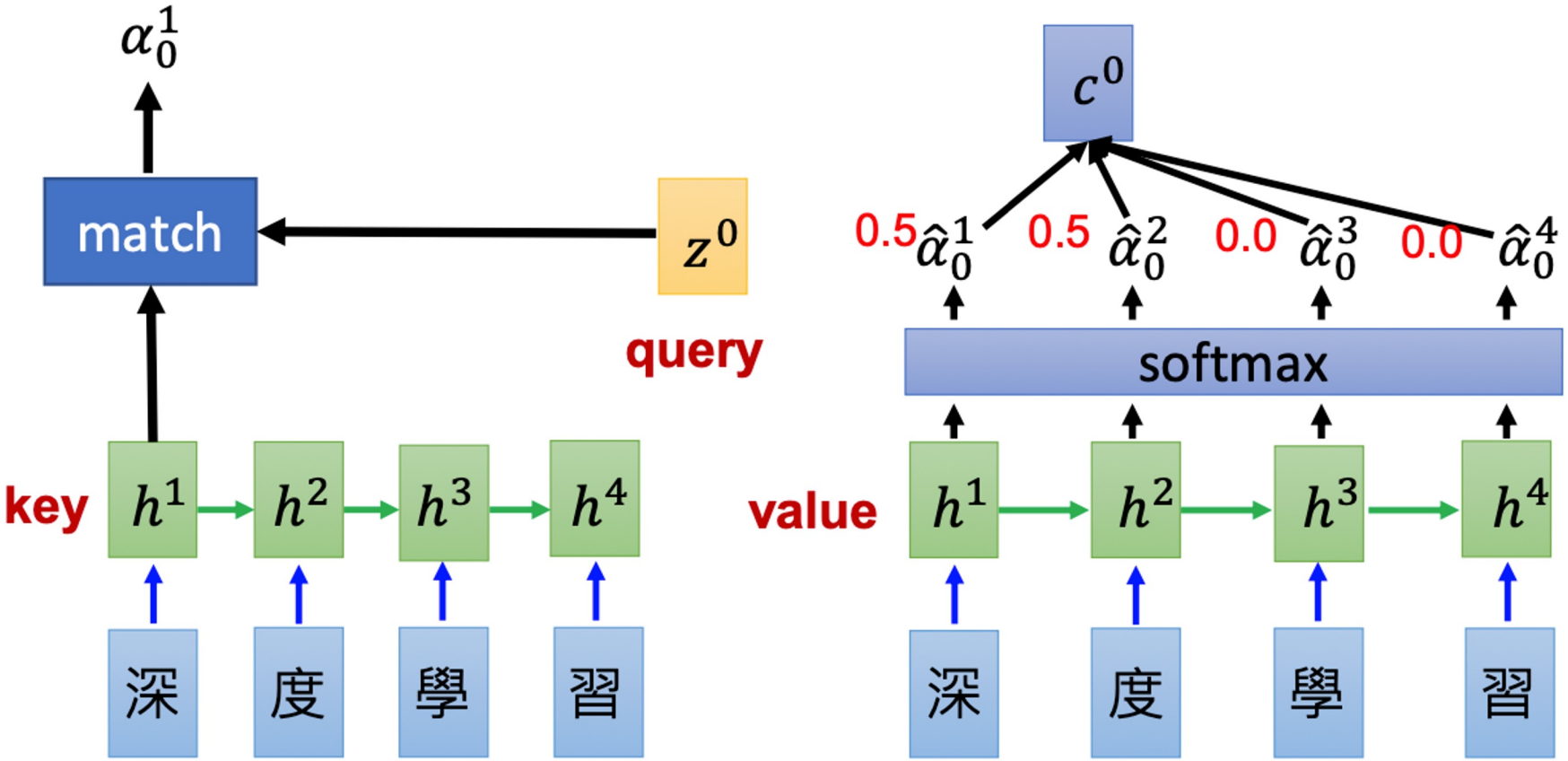


$$c^1 = \sum \hat{\alpha}_1^i h^i = 0.5h^3 + 0.5h^4$$

Machine Translation with Attention



Machine Translation with Attention



Dot-Product Attention

- Input: a query q and a set of key-value (k-v) pairs to an output
- Output: weighted sum of values

Inner product of
query and corresponding key

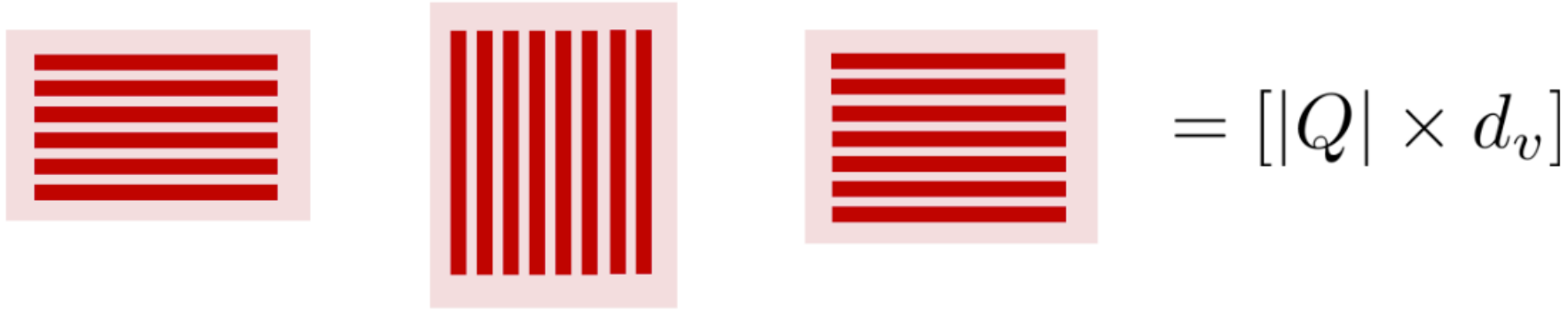
$$A(q, K, V) = \sum_i \frac{\exp(q \cdot k_i)}{\sum_j \exp(q \cdot k_j)} v_i$$

Dot-Product Attention in Matrix

$$A(Q, K, V) = \text{softmax}(QK^T)V$$

$$[|Q| \times d_k] \times [d_k \times |K|] \times [|K| \times d_v]$$

softmax
row-wise



$$= [|Q| \times d_v]$$

Image Captioning with Attention

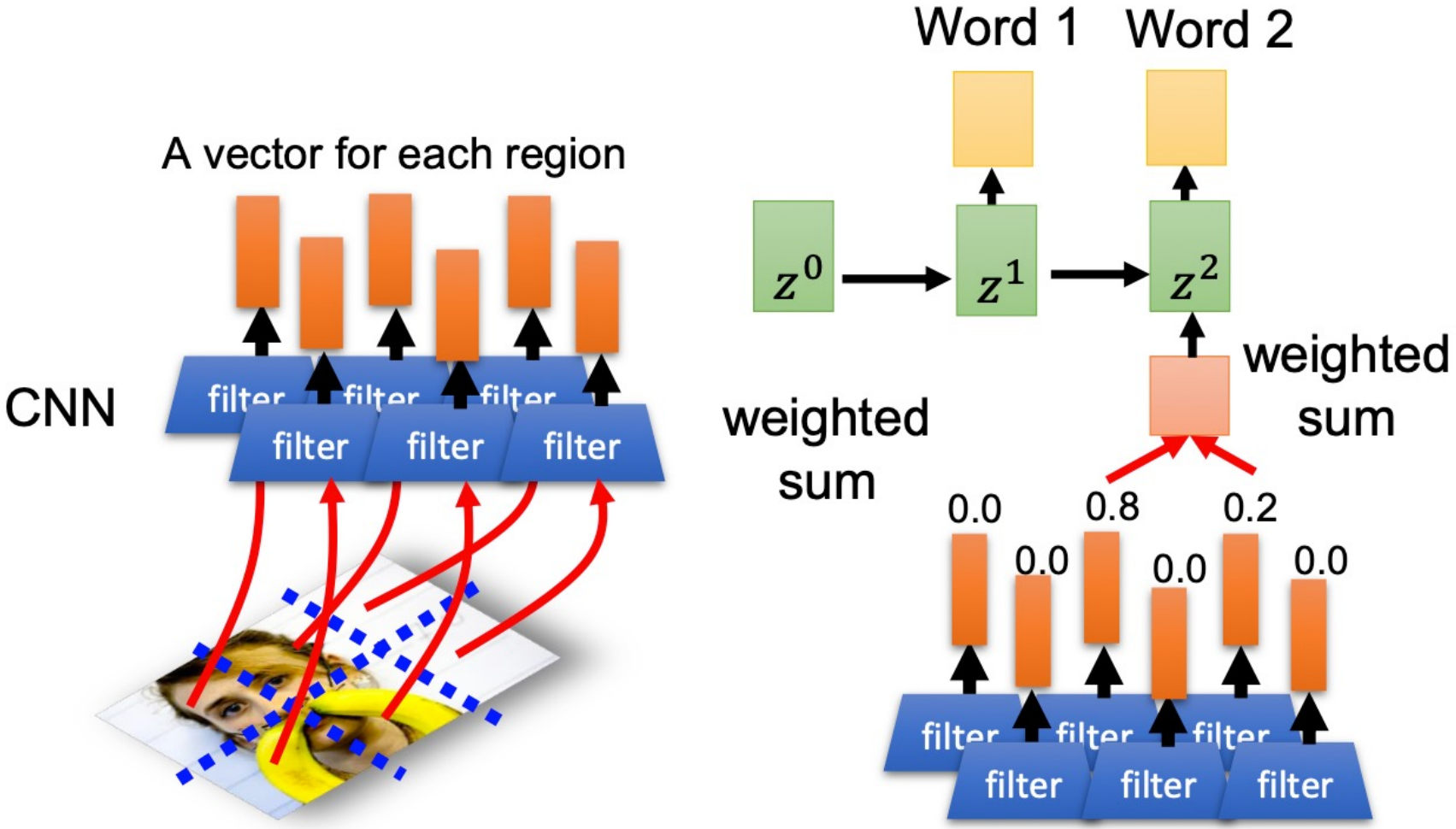


Image Captioning

- Good examples

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Image Captioning

- Bad examples

A large white bird standing in a forest.

A woman holding a clock in her hand.

A man wearing a hat and
a hat on a skateboard.

A person is standing on a beach
with a surfboard.

A woman is sitting at a table
with a large pizza.

A man is talking on his cell phone
while another man watches.

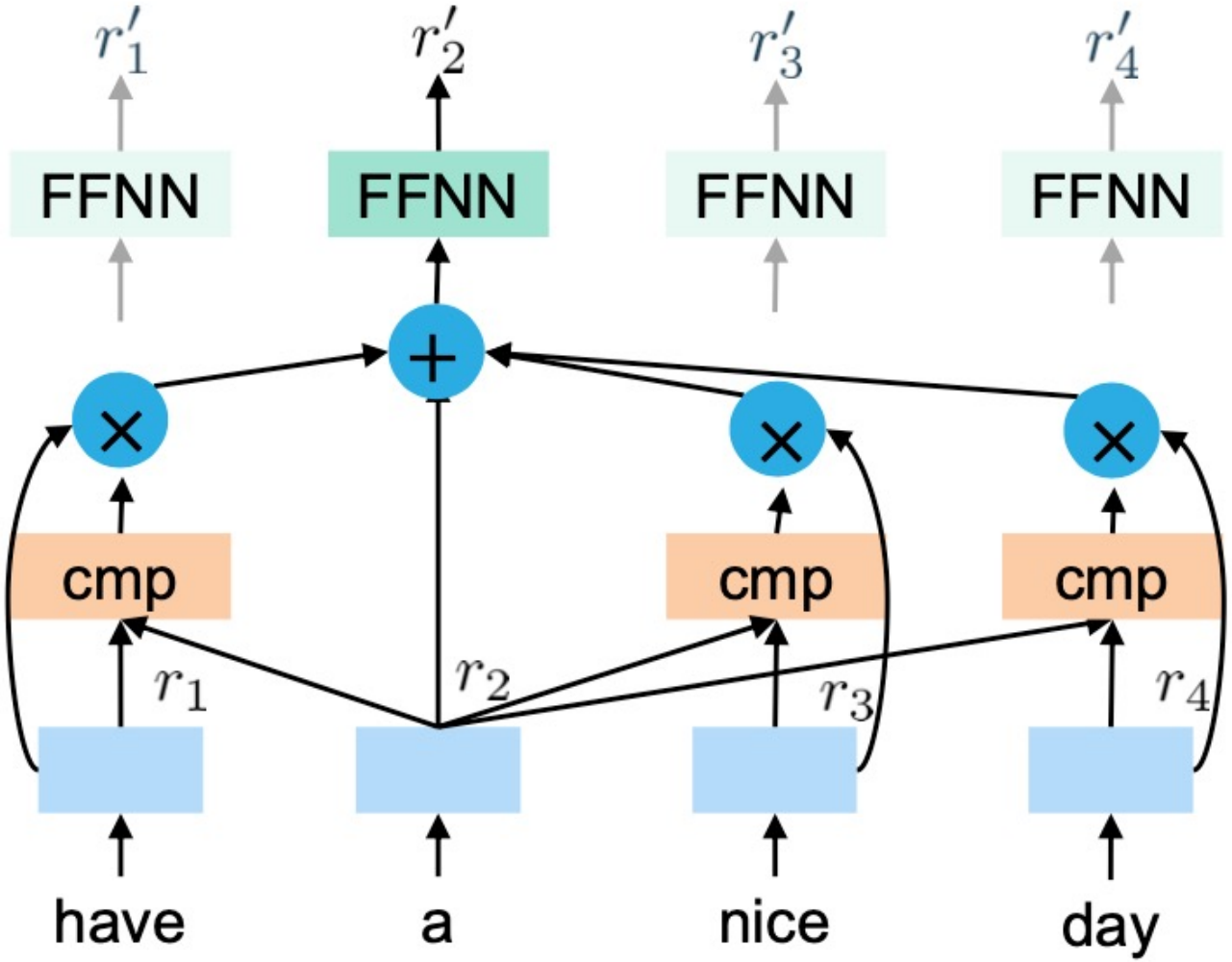
Outlines

- Attention Mechanism
- Transformer
- Vision Transformer
- Swin Transformer

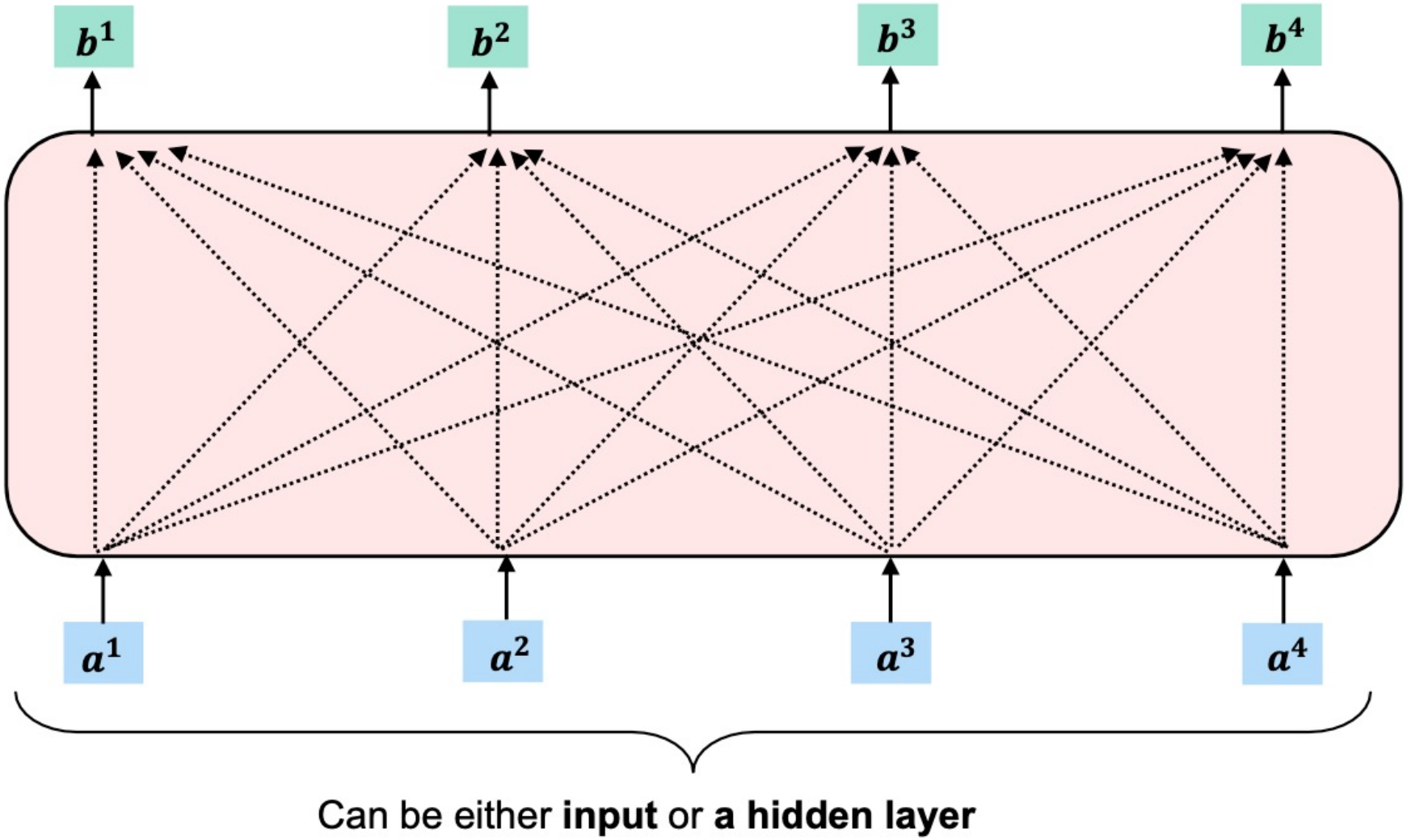
Comparison

- RNN
  - No explicit modeling of long and short range dependencies
  - LSTM?
- CNN
  - Long-distance dependencies require many layers
- Attention allows us to access any state

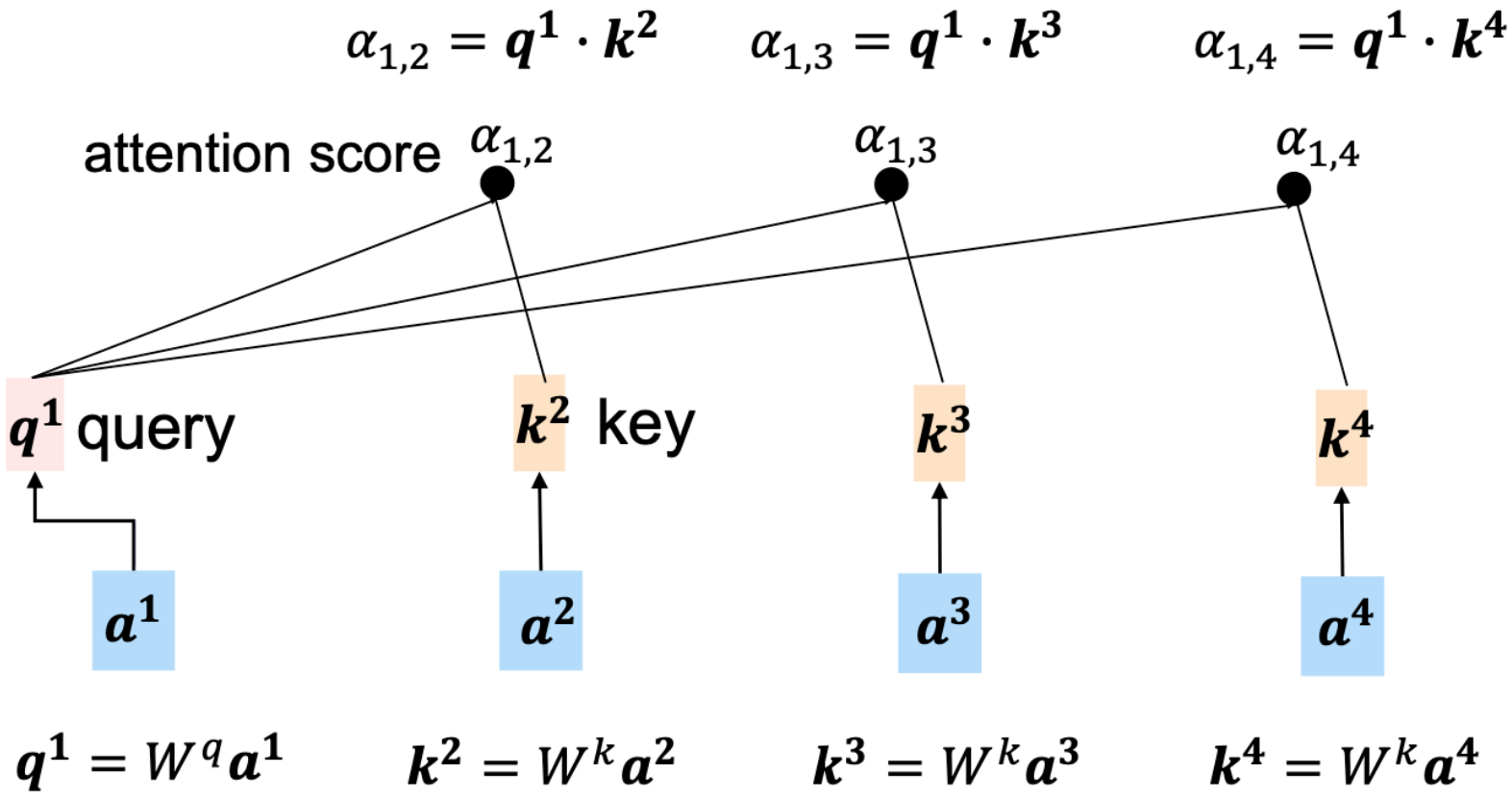
Self-Attention



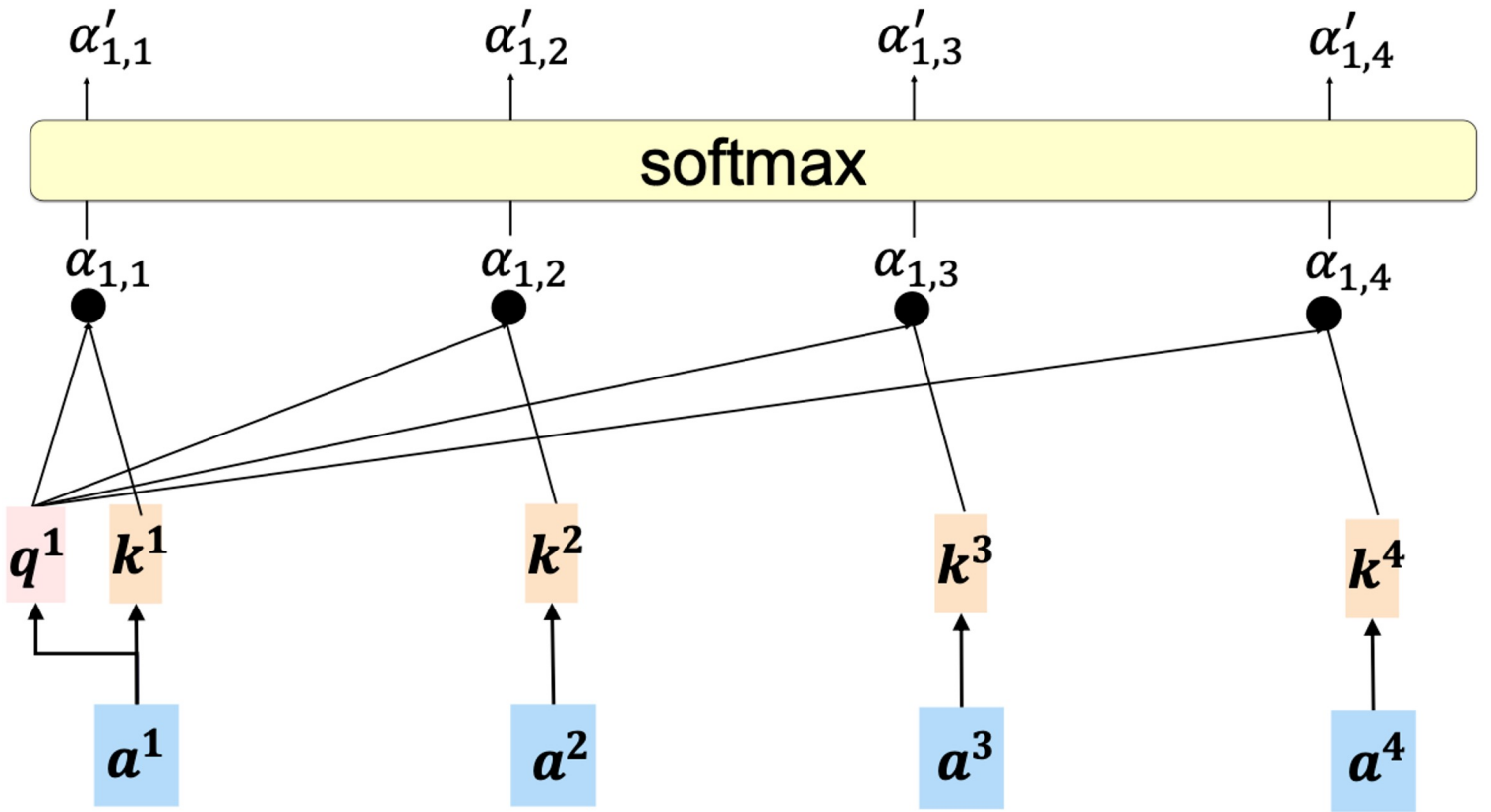
Self-Attention



Self-Attention



Self-Attention

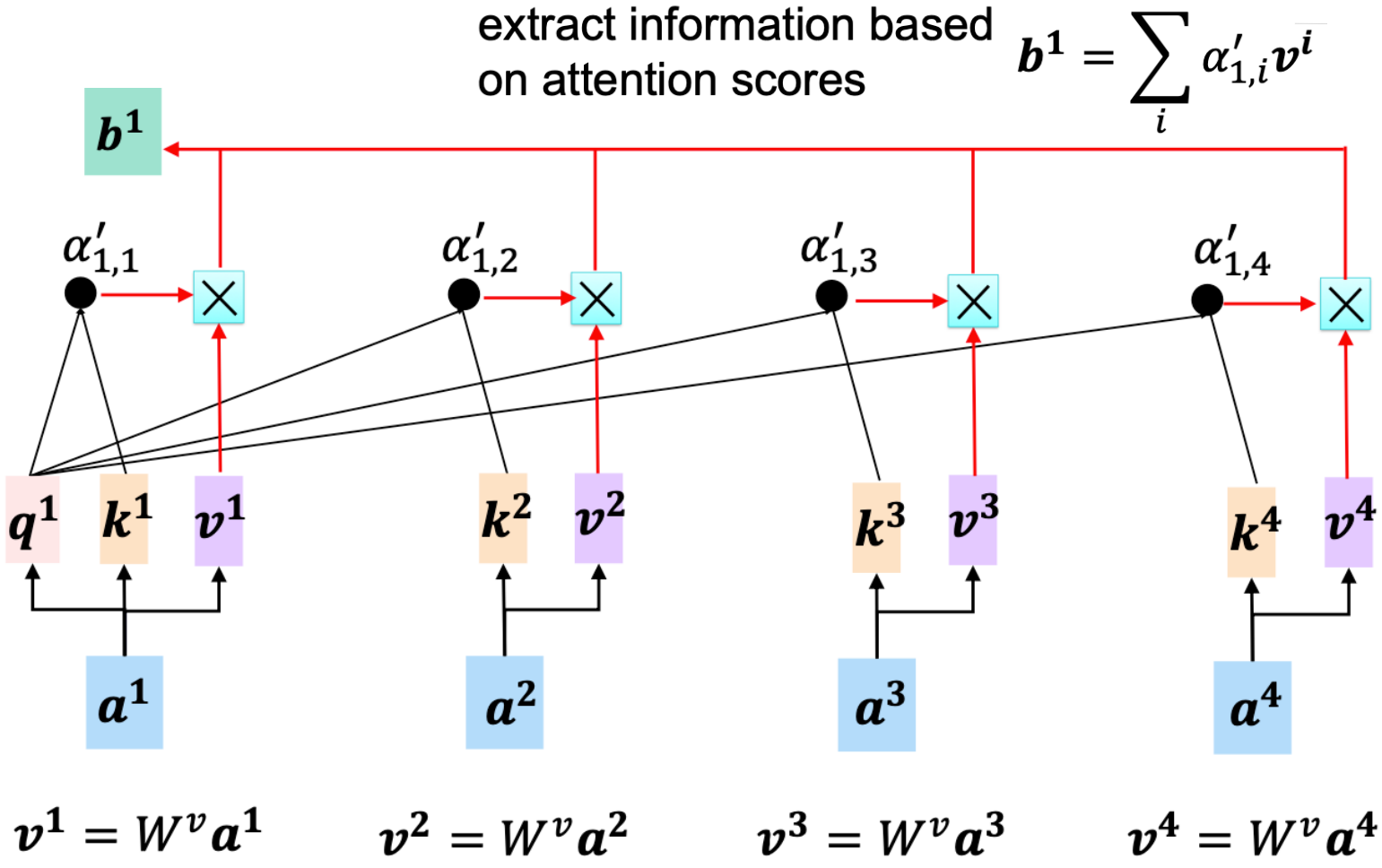


$$\alpha'_{1,i} = e^{\alpha_{1,i}} / \sum_j e^{\alpha_{1,j}}$$

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

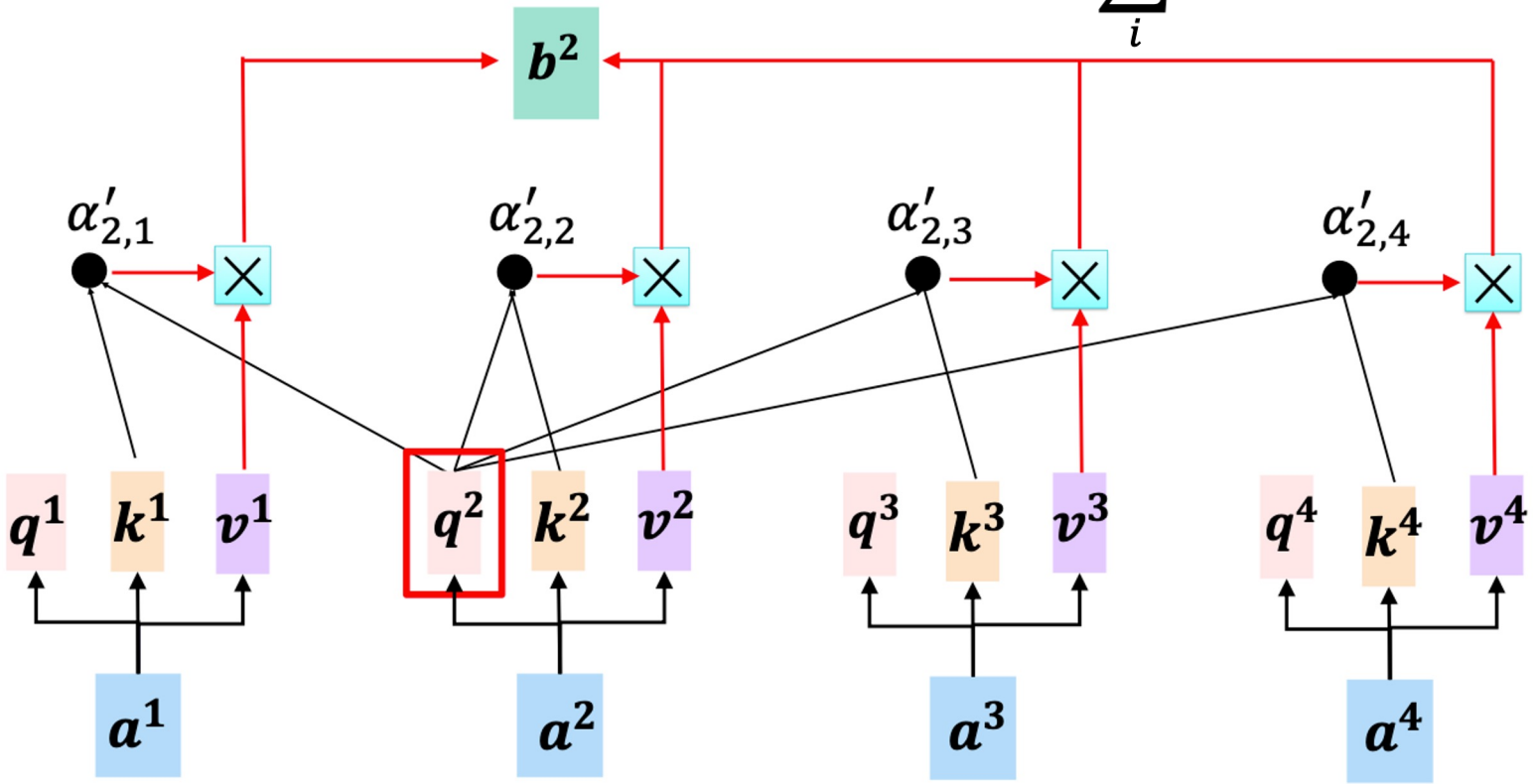
$$k^1 = W^k a^1$$

Self-Attention

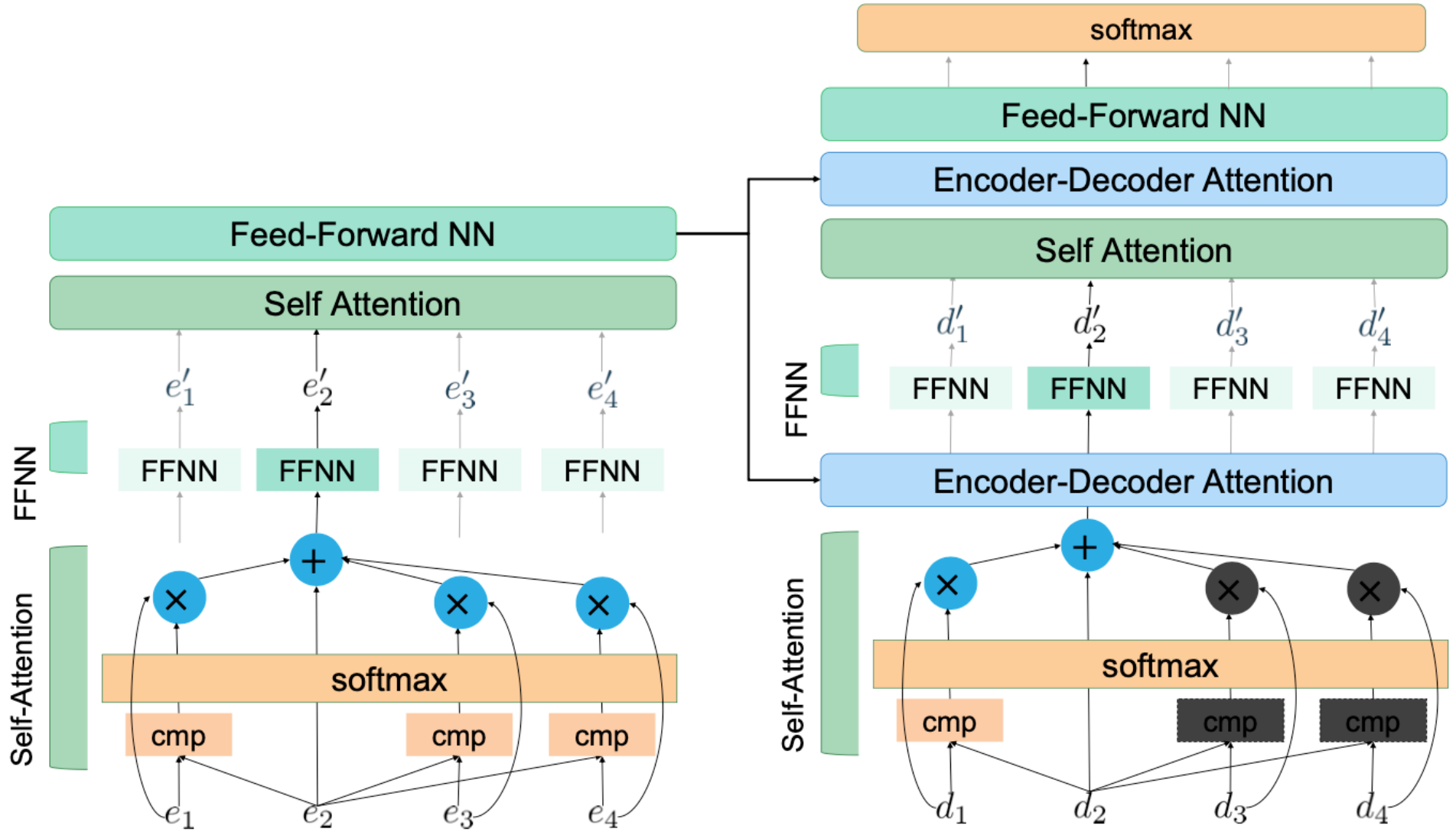


Self-Attention

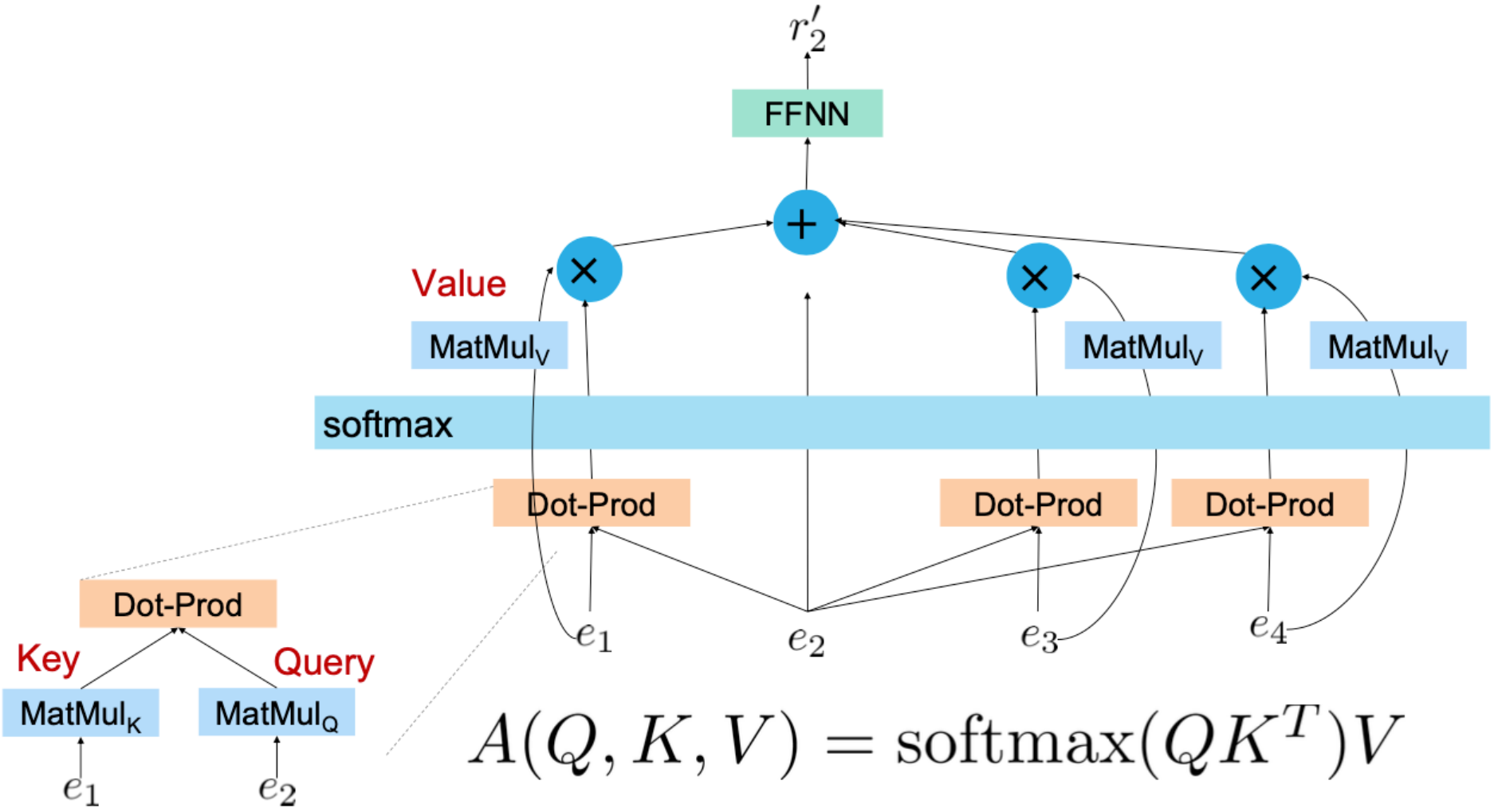
$$b^2 = \sum_i \alpha'_{2,i} v^i$$



Transformer Idea

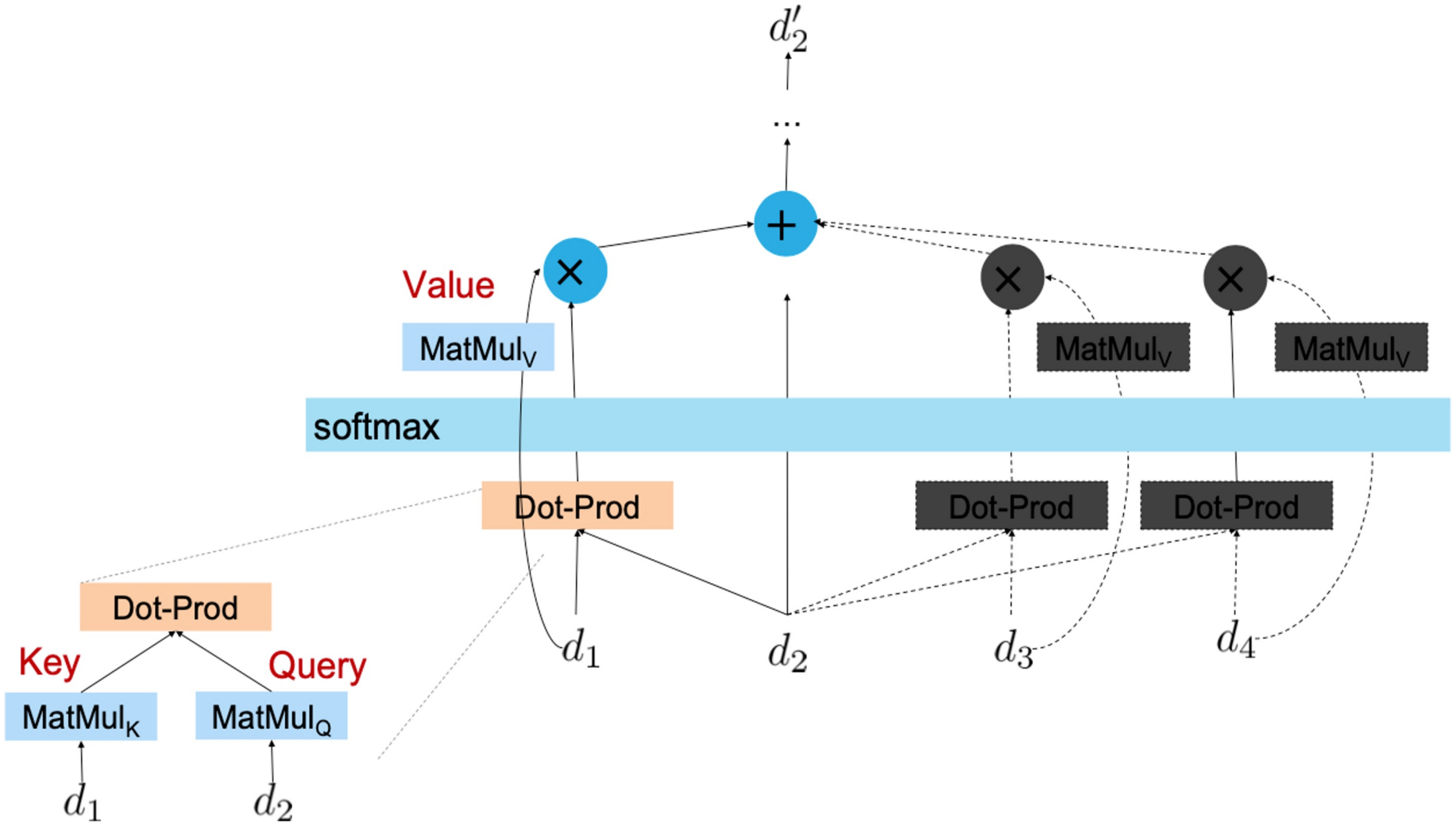


Encoder Self-Attention

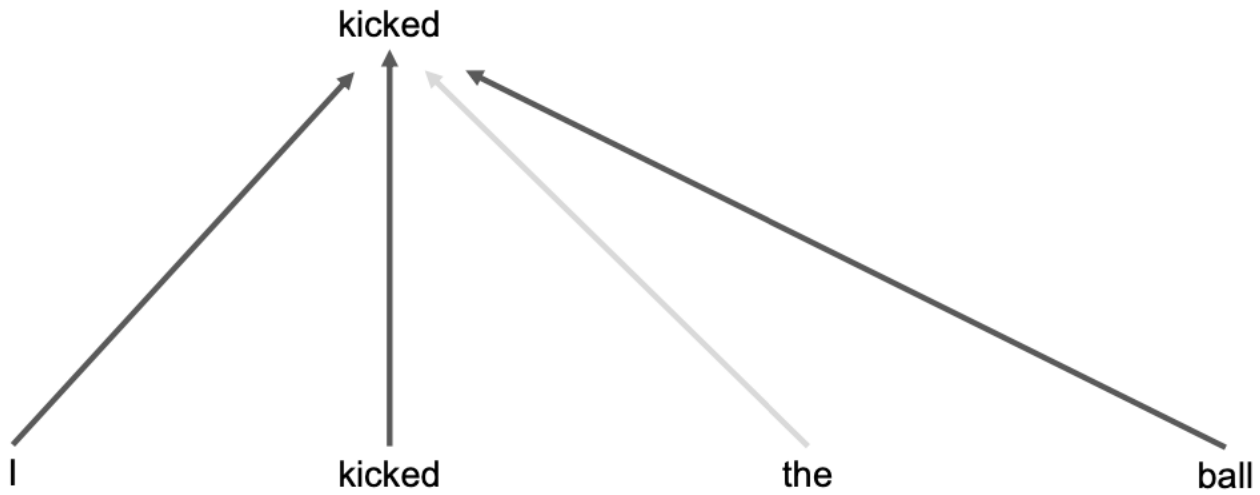


$$A(Q, K, V) = \text{softmax}(QK^T)V$$

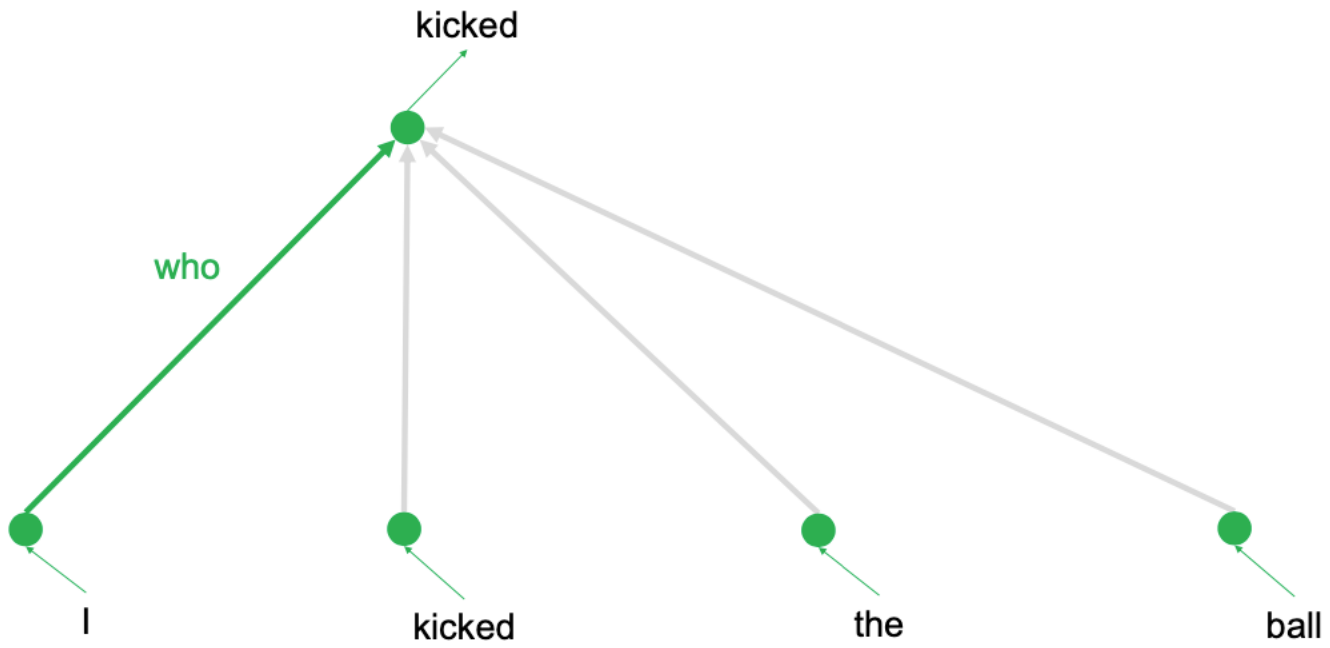
Decoder Self-Attention



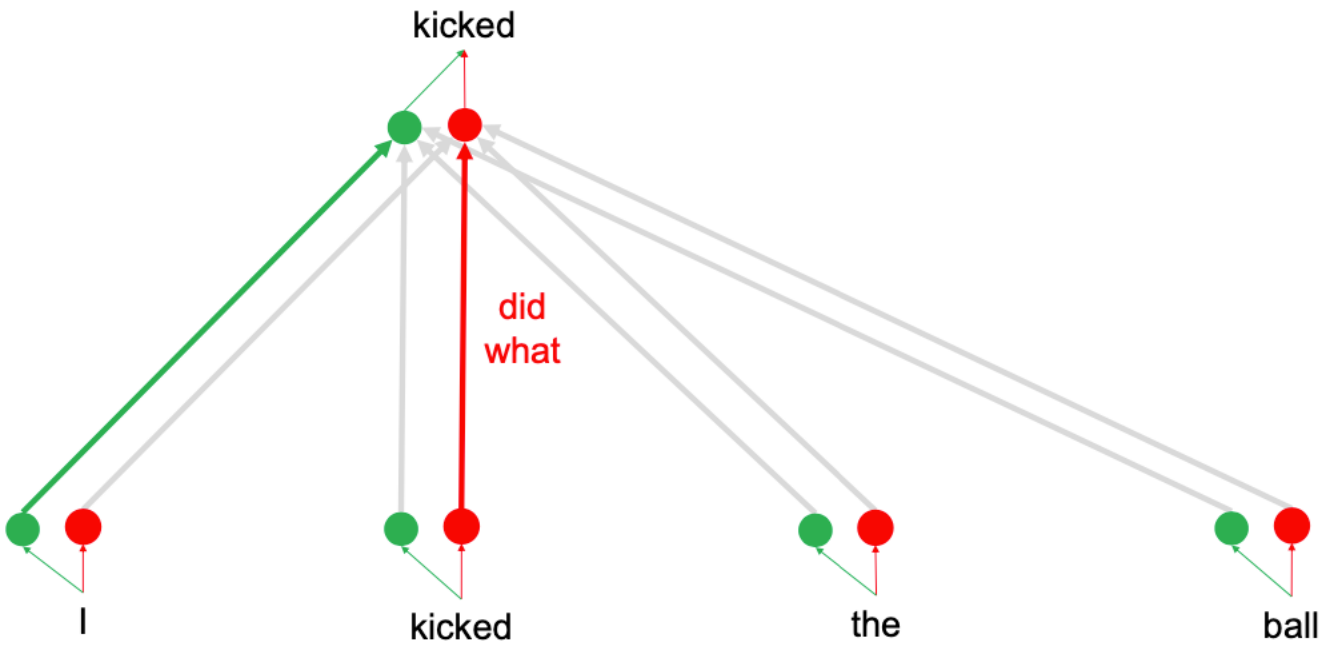
Self-Attention



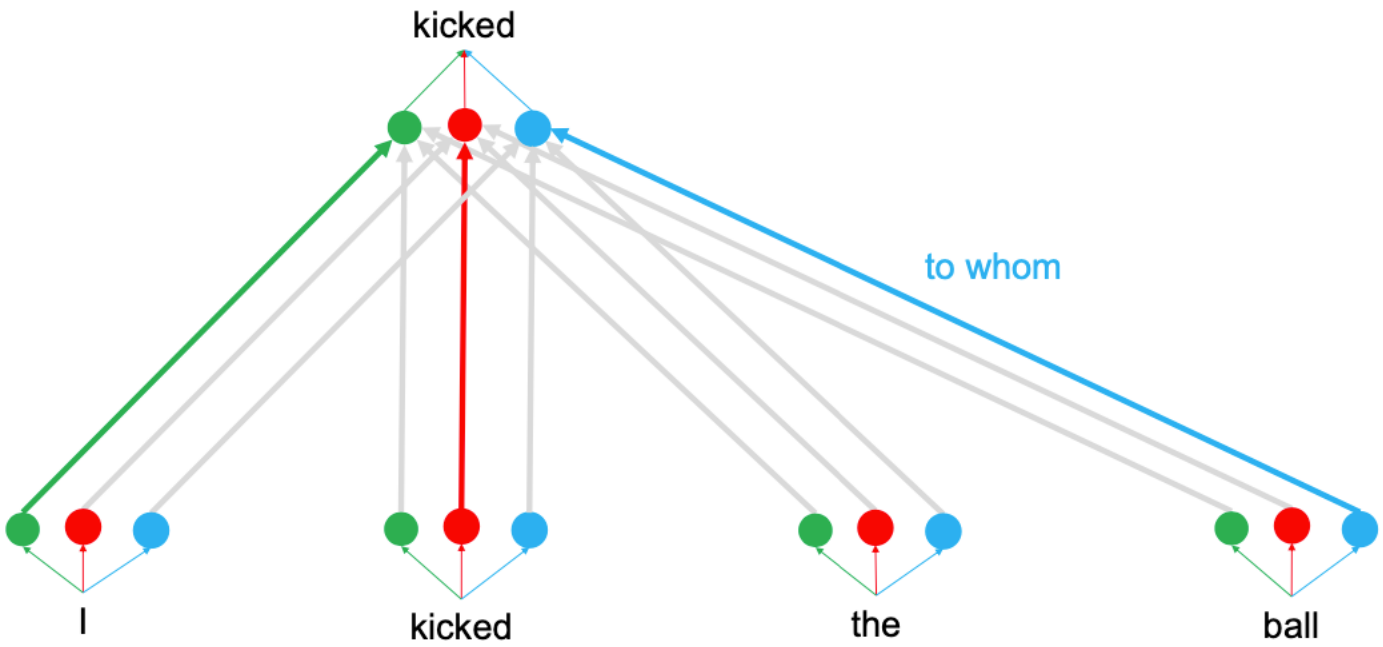
Self-Attention



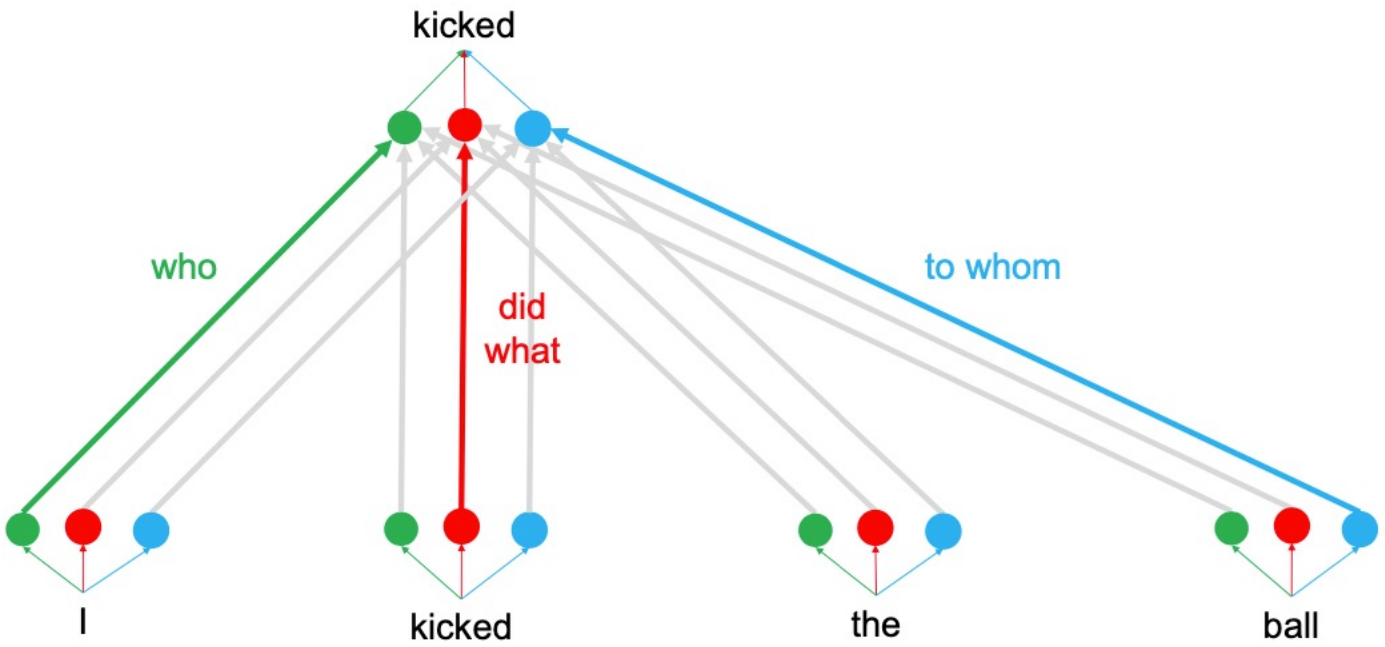
Self-Attention



Self-Attention



Self-Attention

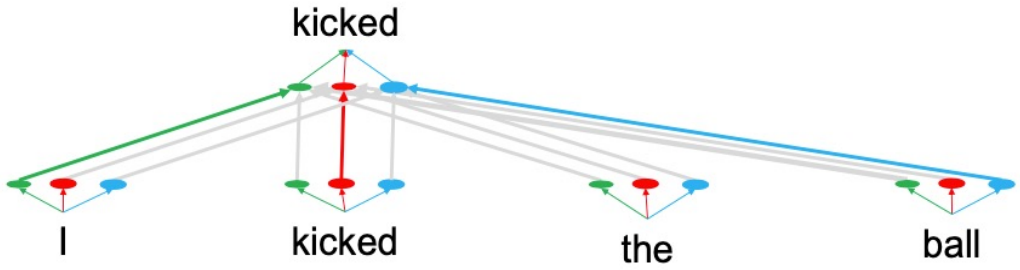


Comparison

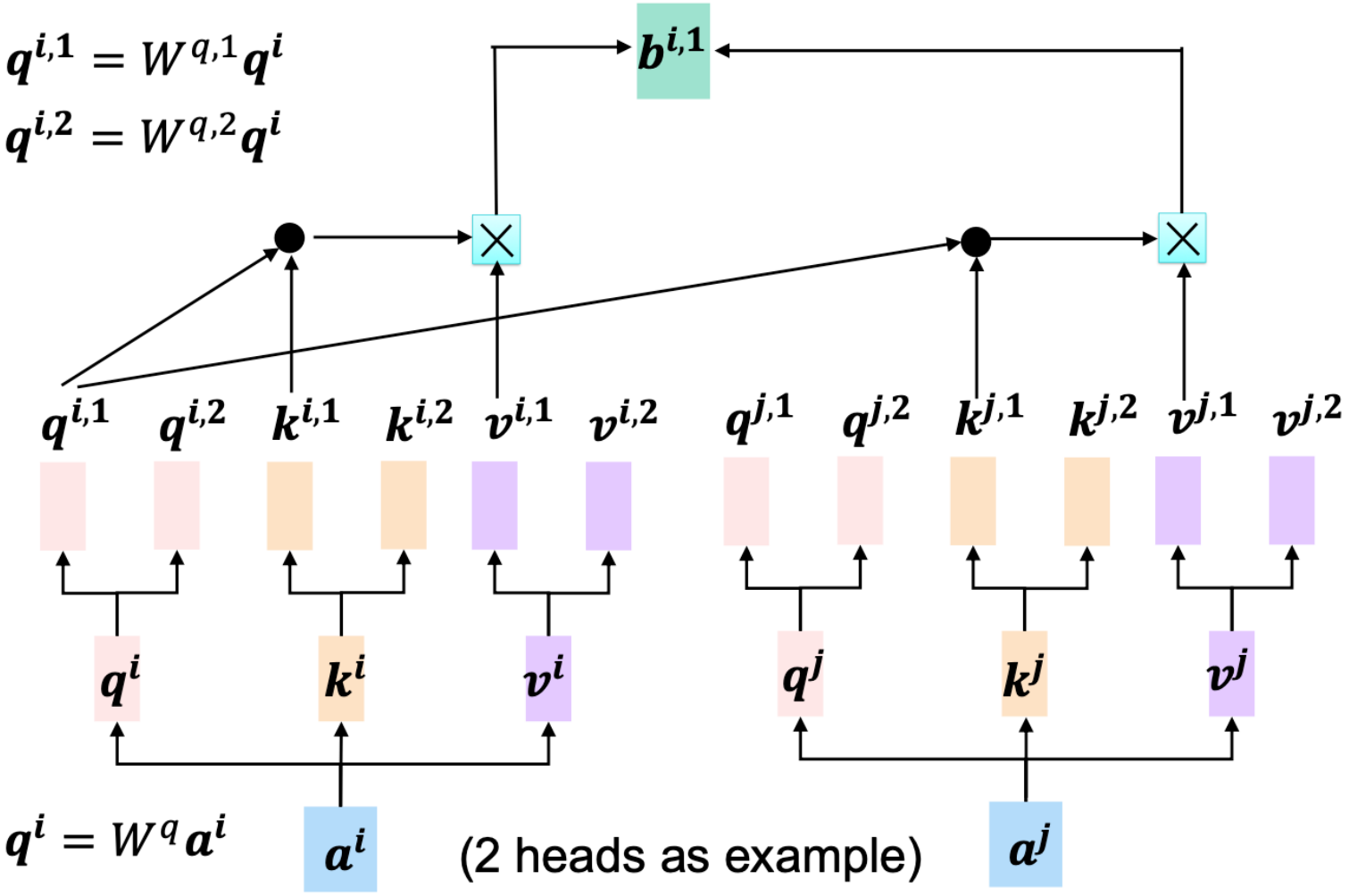
- Convolution: different linear transformations by relative positions

- Attention: a weighted average

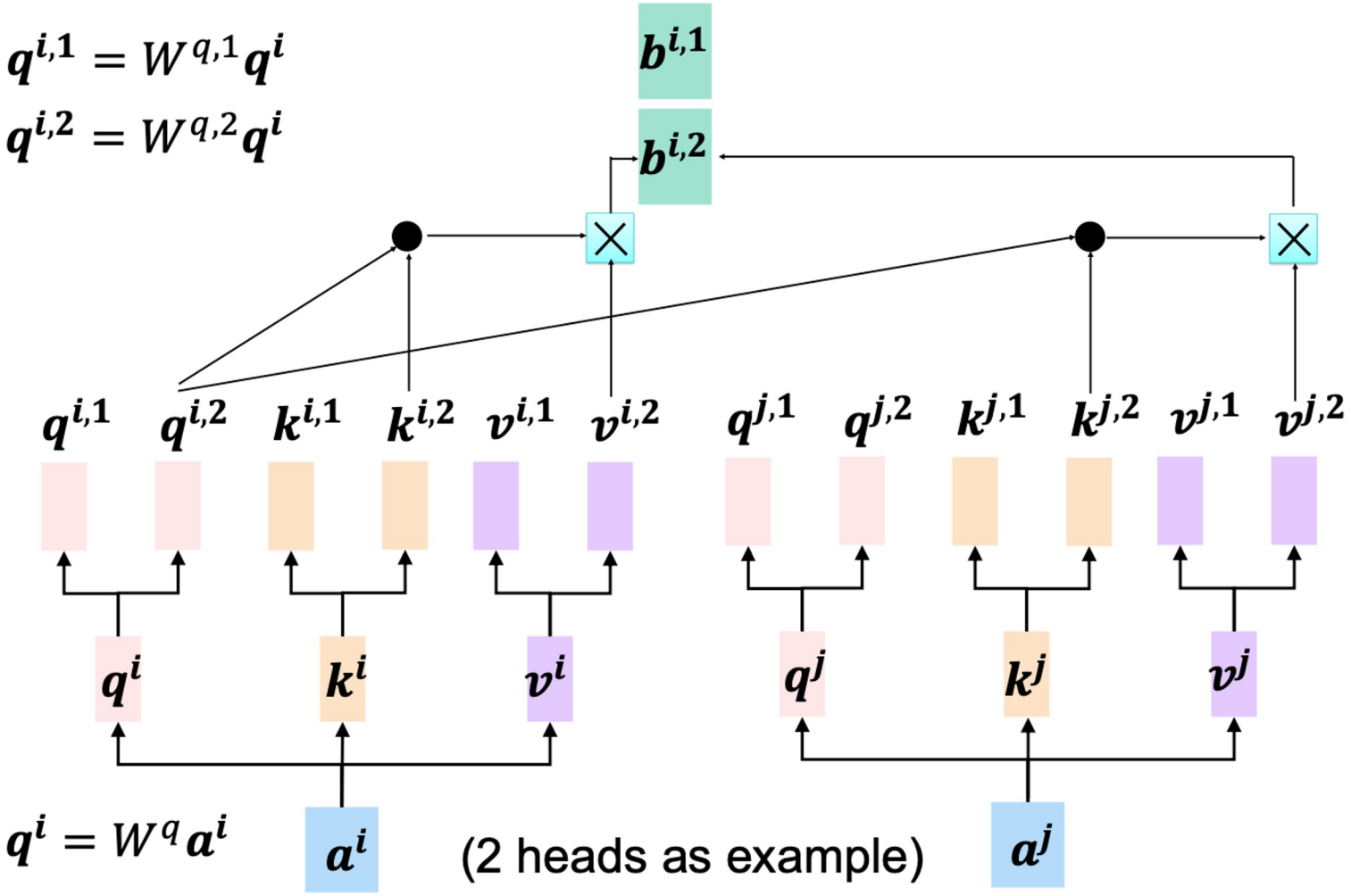
- Multi-Head Attention: parallel attention layers with different linear transformationson input/output



Multi-Head Attention

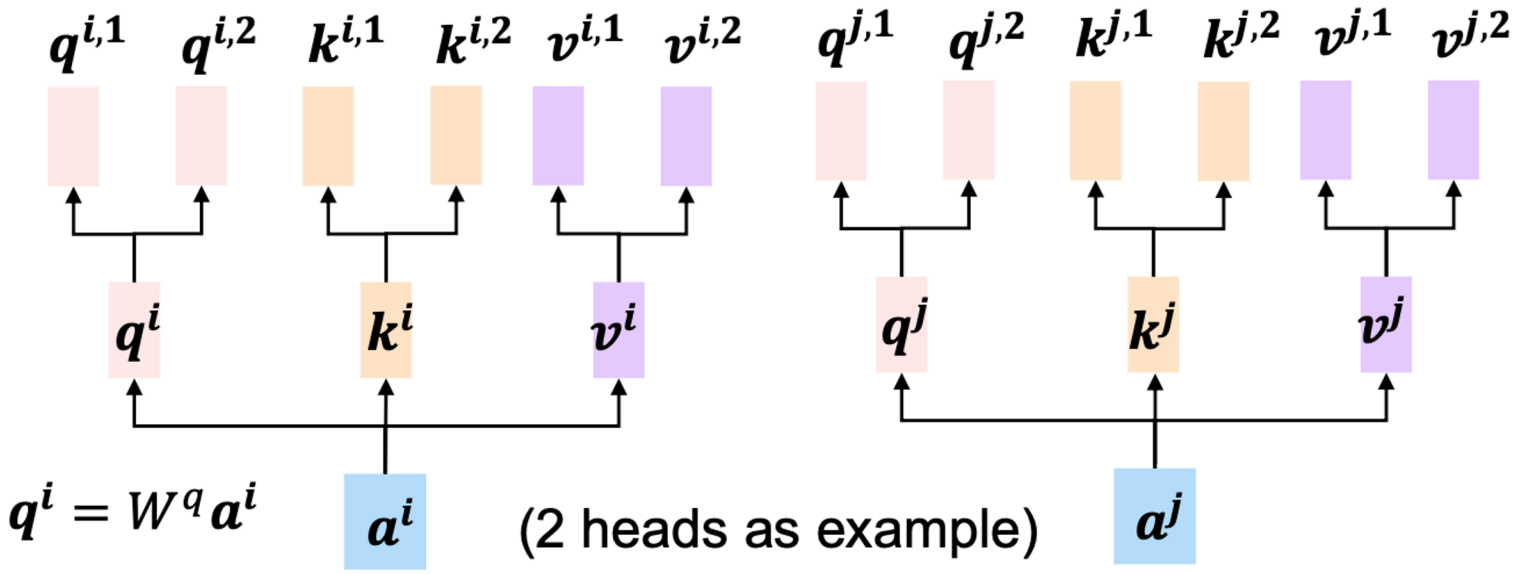


Multi-Head Attention



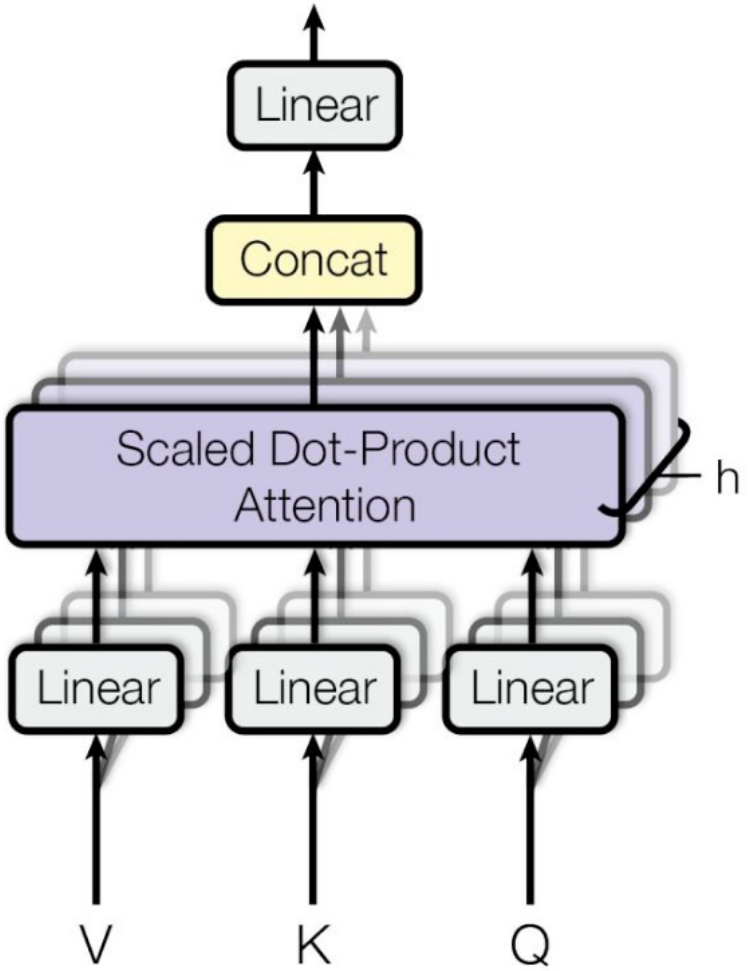
Multi-Head Attention

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



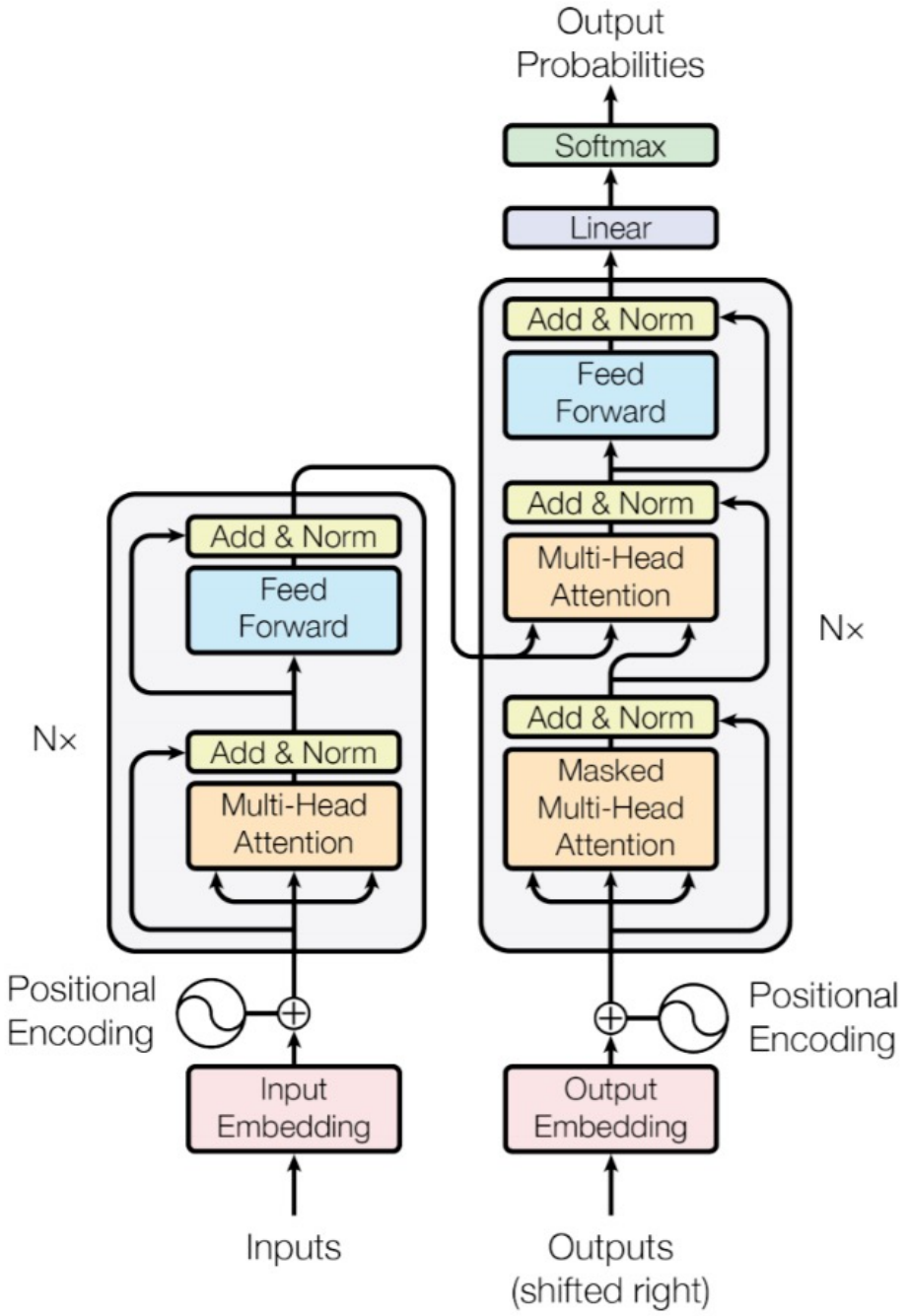
Multi-Head Attention

$$\begin{aligned}\text{MultiHead}(Q, K, V) \\ = \text{Concat}(\text{head}_1, \cdots, \text{head}_h)W^O \\ \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)\end{aligned}$$



Transformer Overview

$$A(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$



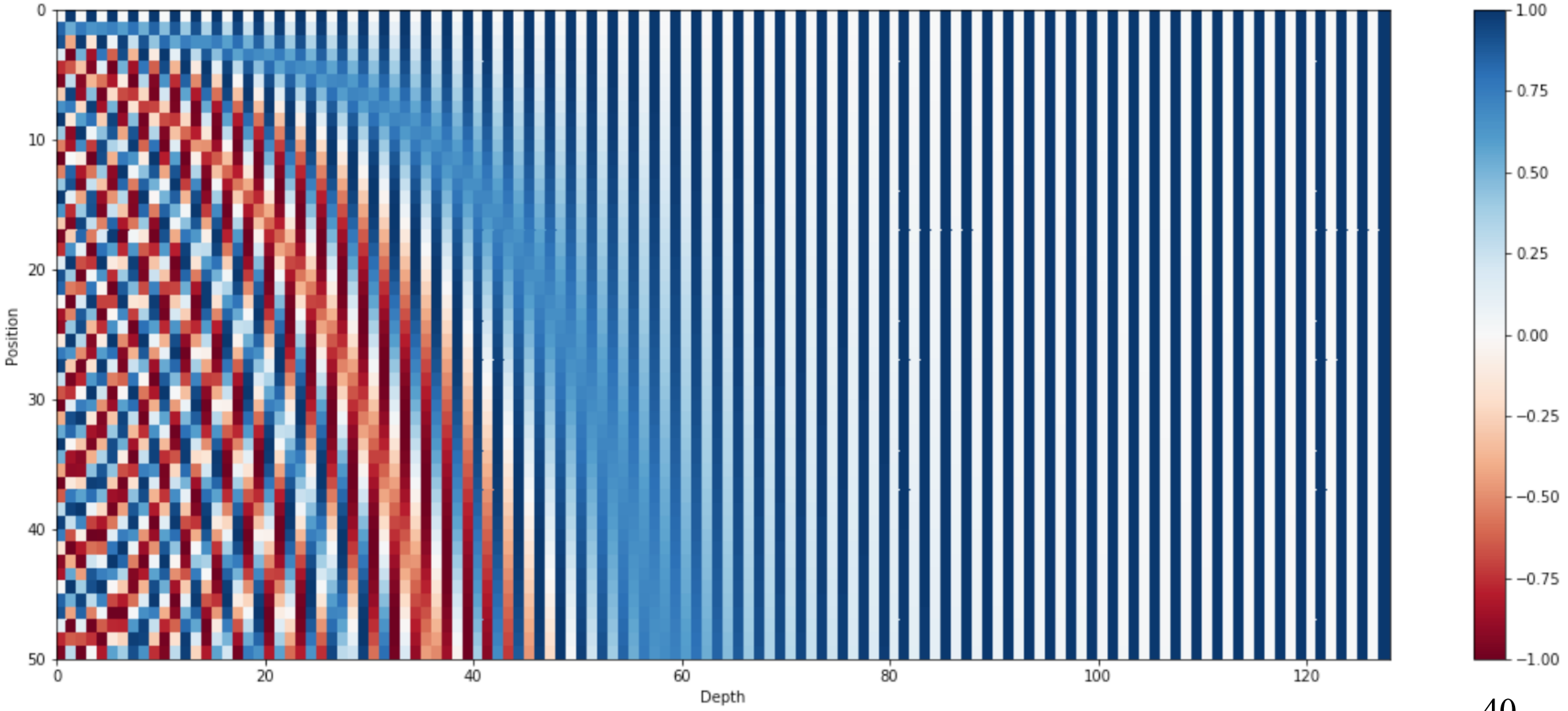
Sinusoidal Positional Encoding

- Criteria for positional embeddings
  - Unique encoding for each position
  - Deterministic
  - Distance between neighboring positions should be the same
  - Model can easily generalize to longer sentences

$$\mathrm{PE}(pos, 2i) = sin(\frac{pos}{100000^{2i/d}})$$

$$\mathrm{PE}(pos, 2i+1) = cos(\frac{pos}{100000^{2i/d}})$$

Sinusoidal Positional Encoding



Multi-Head Attention Details

encoder self attention

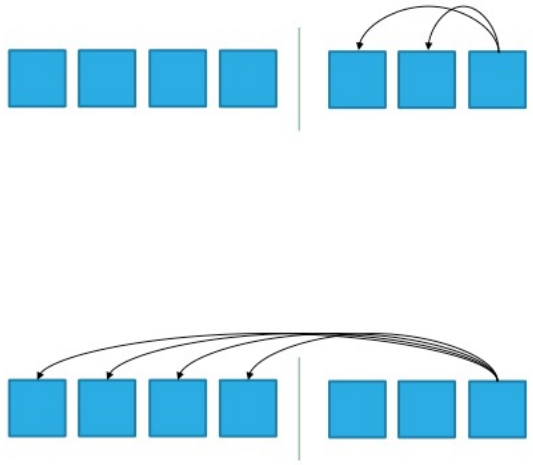
1. Multi-head Attention
2. Query=Key=Value

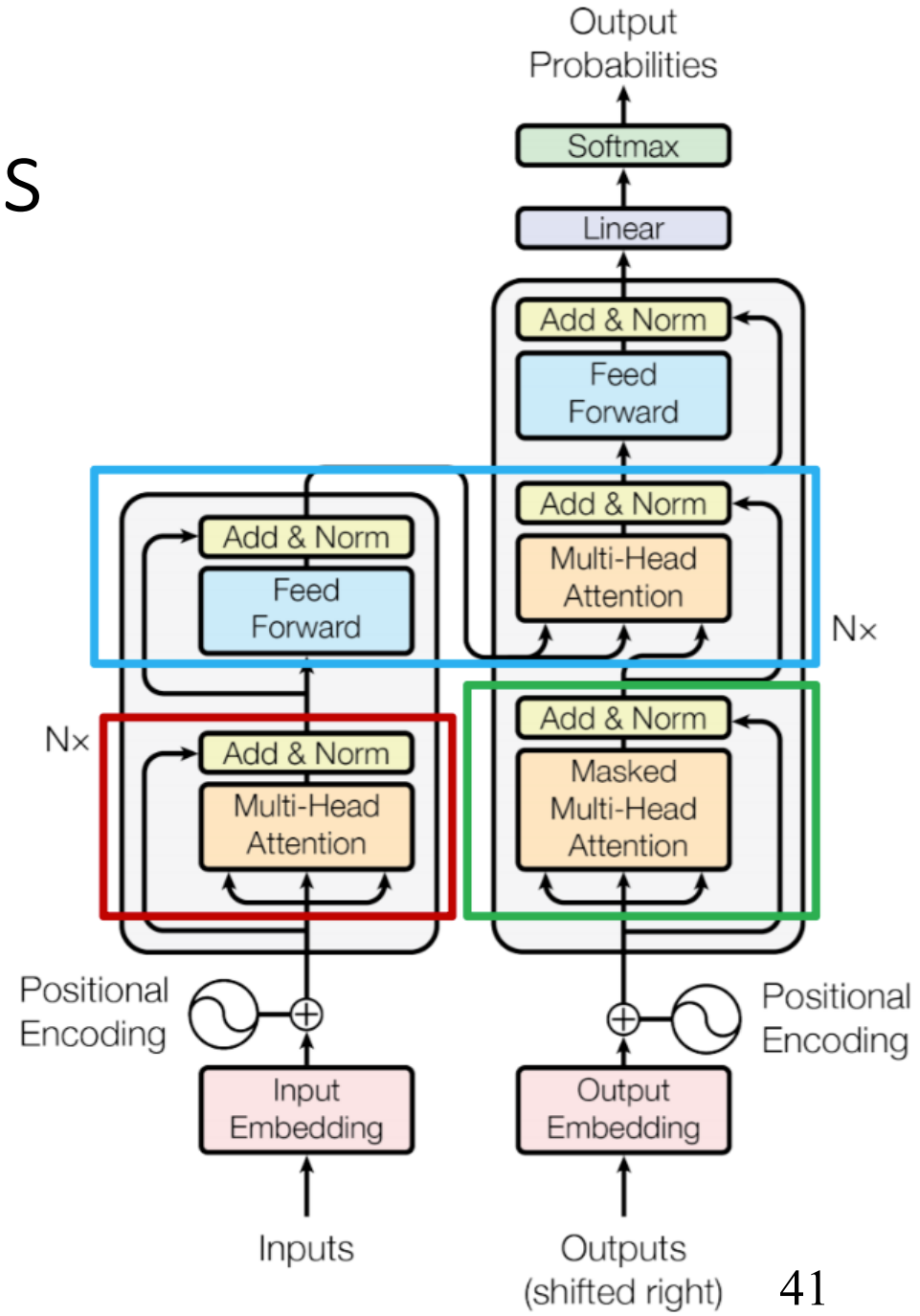
decoder self attention

1. Masked Multi-head Attention
2. Query=Key=Value

encoder-decoder attention

1. Multi-head Attention
2. Encoder Self attention=Key=Value
3. Decoder Self attention=Query





MT Experiments

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

Concluding Remarks

- **Non-recurrence** model is easy to parallelize
- **Multi-head attention** captures different aspects by interacting between words
- **Positional encoding** captures location information
- Each transformer block can be **applied to diverse tasks**

Outlines

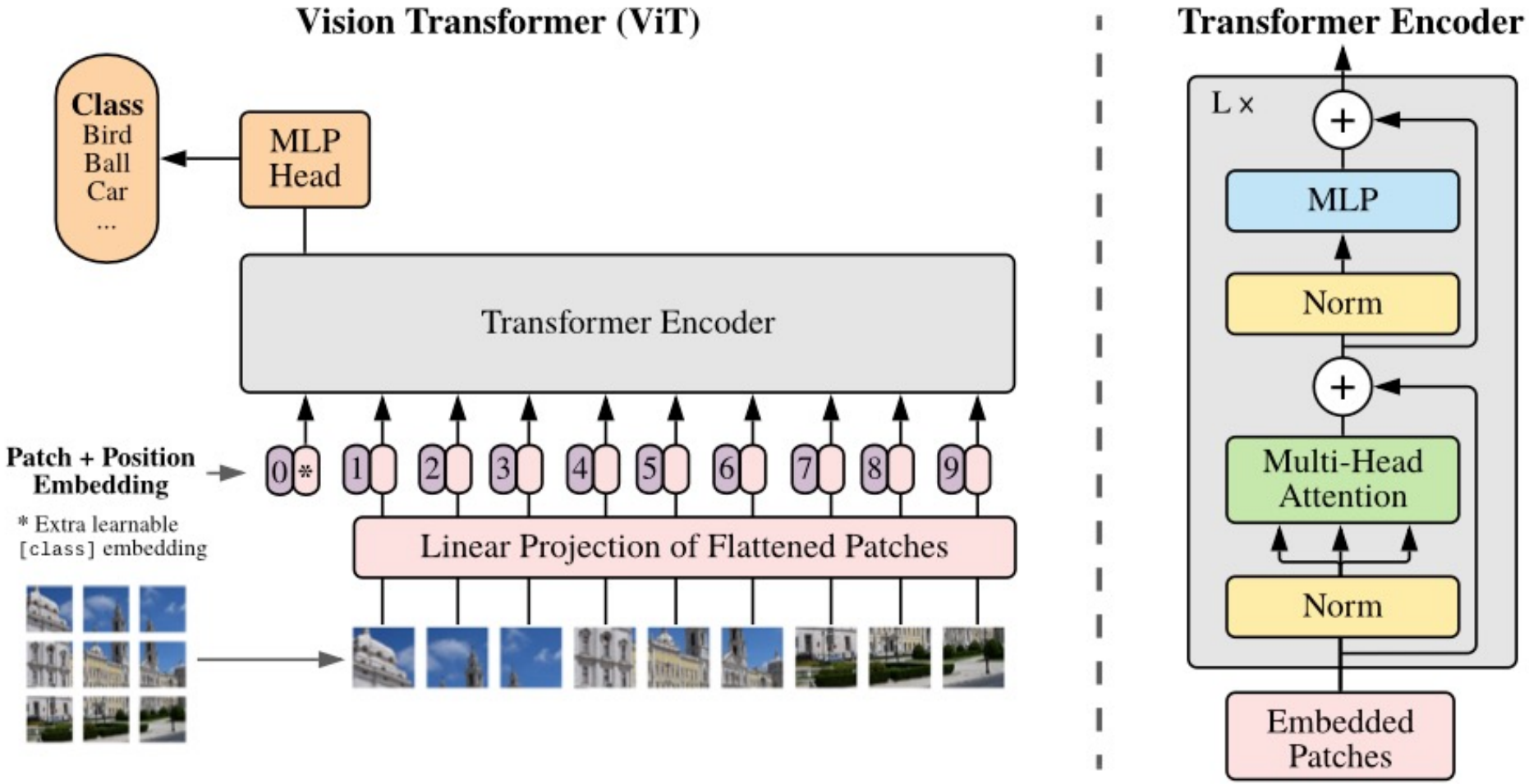
- Attention Mechanism
- Transformer
- Vision Transformer
- Swin Transformer

AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

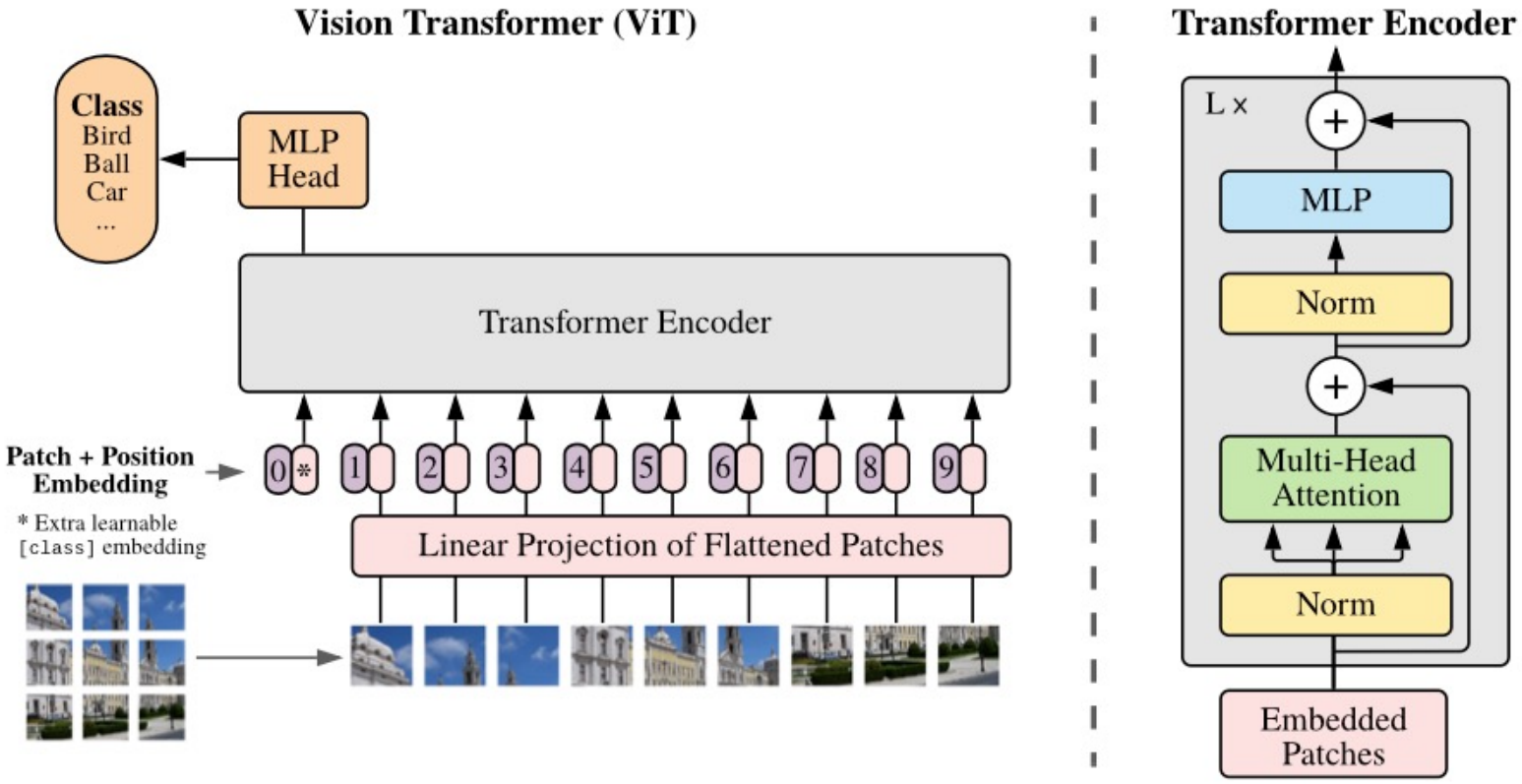
Alexey Dosovitskiy[*,†], **Lucas Beyer**[*], **Alexander Kolesnikov**[*], **Dirk Weissenborn**[*],
Xiaohua Zhai[*], **Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,**
Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby[*,†]
[*]equal technical contribution, [†]equal advising
Google Research, Brain Team
{adosovitskiy, neilhoulsby}@google.com

ICLR 2021

Vision Transformer (ViT)



Vision Transformer (ViT)



$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \qquad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D} \tag{1}$$

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \qquad \ell = 1 \ldots L \tag{2}$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \qquad \ell = 1 \ldots L \tag{3}$$

$$\mathbf{y} = \text{LN}(\mathbf{z}_L^0) \tag{4}$$

Vision Transformer (ViT)

- Reshape the image $x \in \mathbb{R}^{H*W*C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N*(P^2*C)}$, $N = HW/P^2$
- Maps to $D\ dimensions$
- Similar to BERT's $[class]$ token, prepend a learnable patch ($z_0^0 = x_{class}$)
- Add positional embedding
- Alternating layers of MSA and MLP blocks
- Layernorm (LN) is applied before every block, and residual connections after every block

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \qquad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D},\ \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D} \tag{1}$$

$$\mathbf{z'}_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \qquad \ell = 1 \ldots L \tag{2}$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z'}_\ell)) + \mathbf{z'}_\ell, \qquad \ell = 1 \ldots L \tag{3}$$

$$\mathbf{y} = \text{LN}(\mathbf{z}_L^0) \tag{4}$$

Details of Vision Transformer model variants

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Table 1: Details of Vision Transformer model variants.

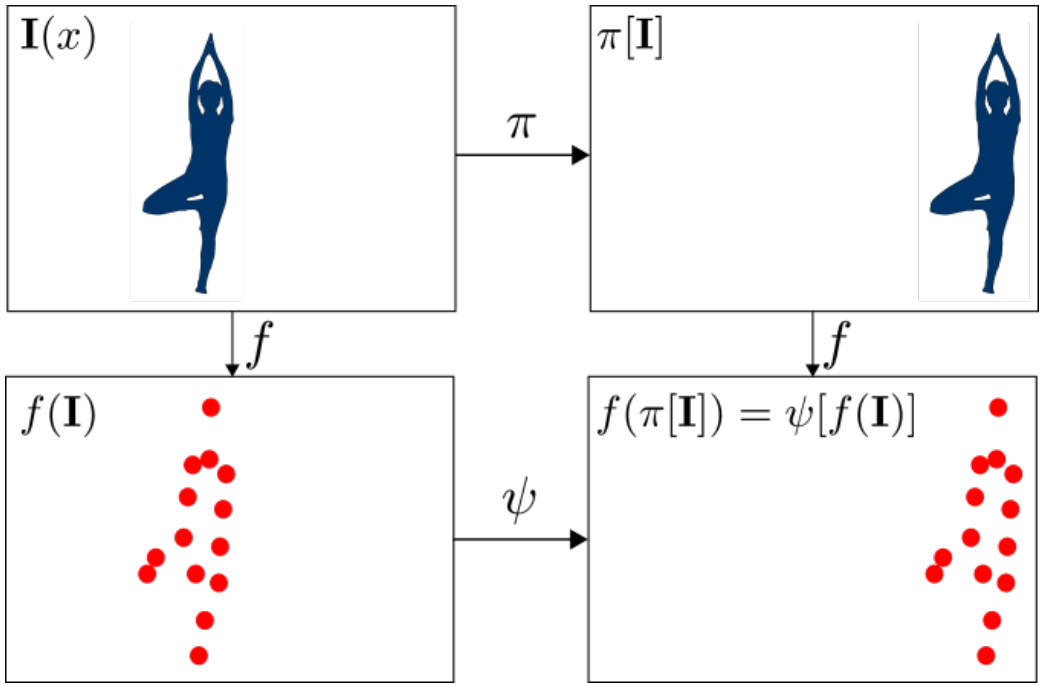
Scores

名稱	模型	pre-train dataset
Ous-JFT	ViT-H/14	JFT-300M
Ous-I21k	ViT-L/16	ImageNet-21k
BiT-L (Big Transfer)	ResNet	JFT-300M
Noisy Student	EfficientNet	semi-supervised learning on ImageNet and JFT-300M with the labels removed

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Comparison

- Why better than CNN?
  - Without inductive bias
  - But need sufficient data
- Inductive bias of CNN
  - translation equivariance
    - This means that if an object in an image is shifted, the same feature will be detected regardless of its position in the image
  - Locality bias
    - local receptive fields
    - weight sharing



Outlines

- Attention Mechanism
- Transformer
- Vision Transformer
- Swin Transformer

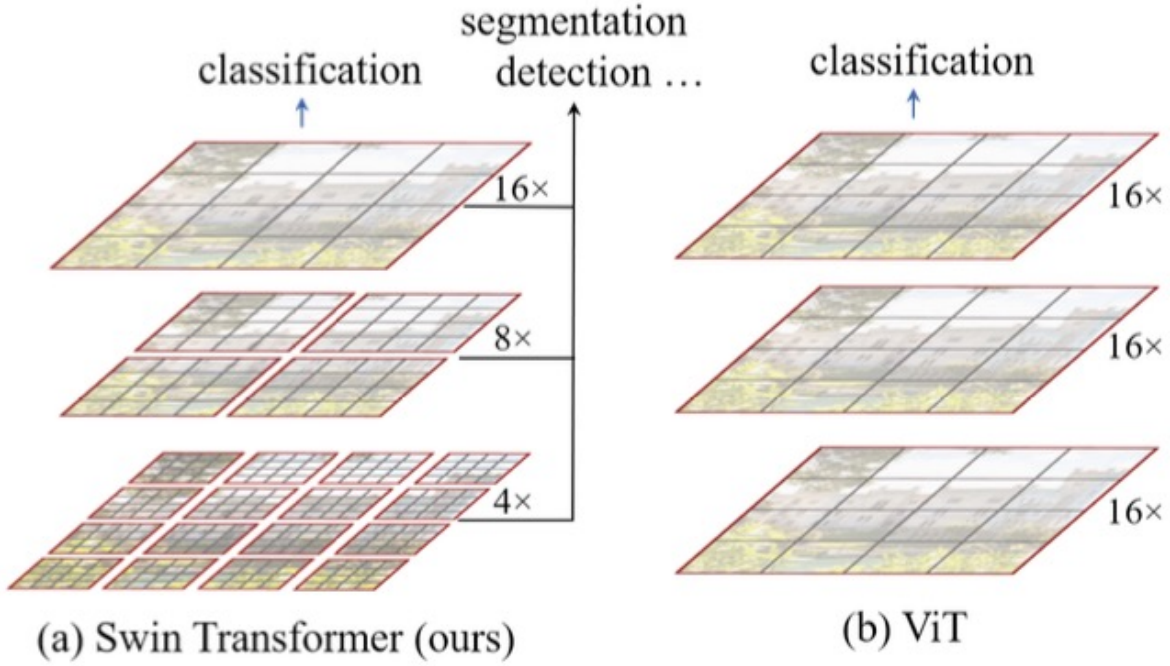
Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

Ze Liu[†*] Yutong Lin[†*] Yue Cao[*] Han Hu[*‡] Yixuan Wei[†]
Zheng Zhang Stephen Lin Baining Guo
Microsoft Research Asia
{v-zeliu1,v-yutlin,yuecao,hanhu,v-yixwe,zhez,stevelin,bainguo}@microsoft.com

ICCV 2021

Swin Transformer

- Vision Transformers produce feature maps of a single low resolution cannot be used for advanced task.
- Swin transformer complexity becomes linear to image size.



Swin Transformer

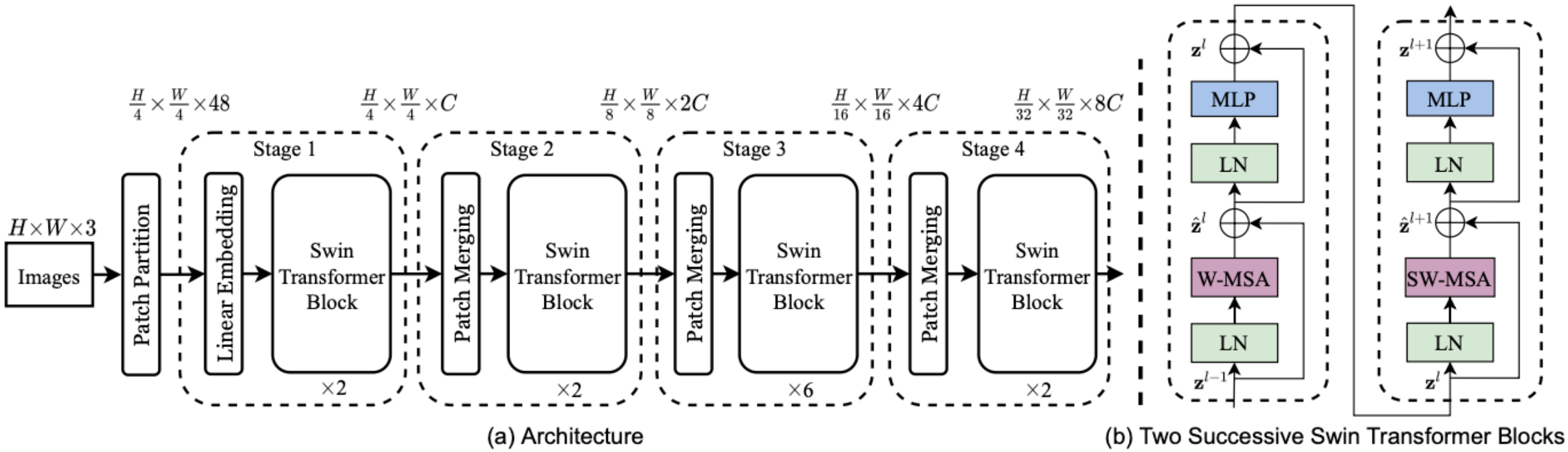
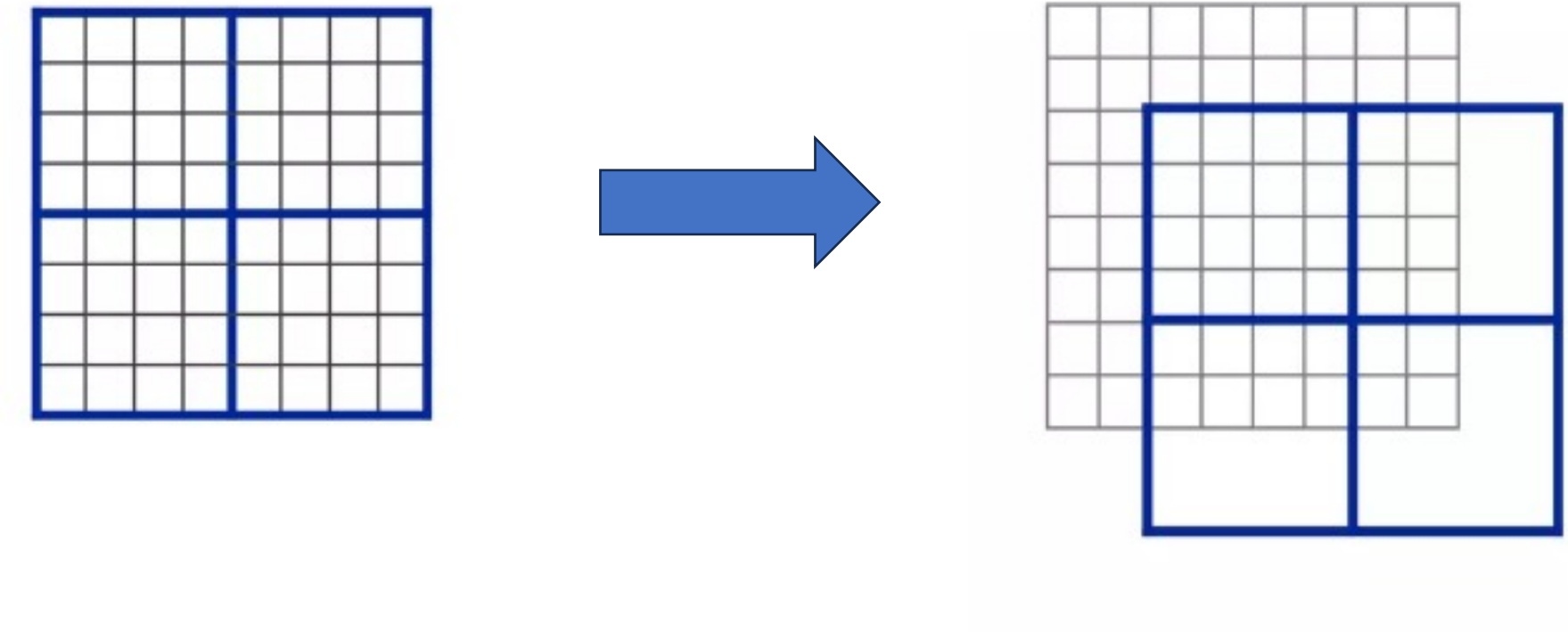


Figure 3. (a) The architecture of a Swin Transformer (Swin-T); (b) two successive Swin Transformer Blocks (notation presented with Eq. (3)). W-MSA and SW-MSA are multi-head self attention modules with regular and shifted windowing configurations, respectively.

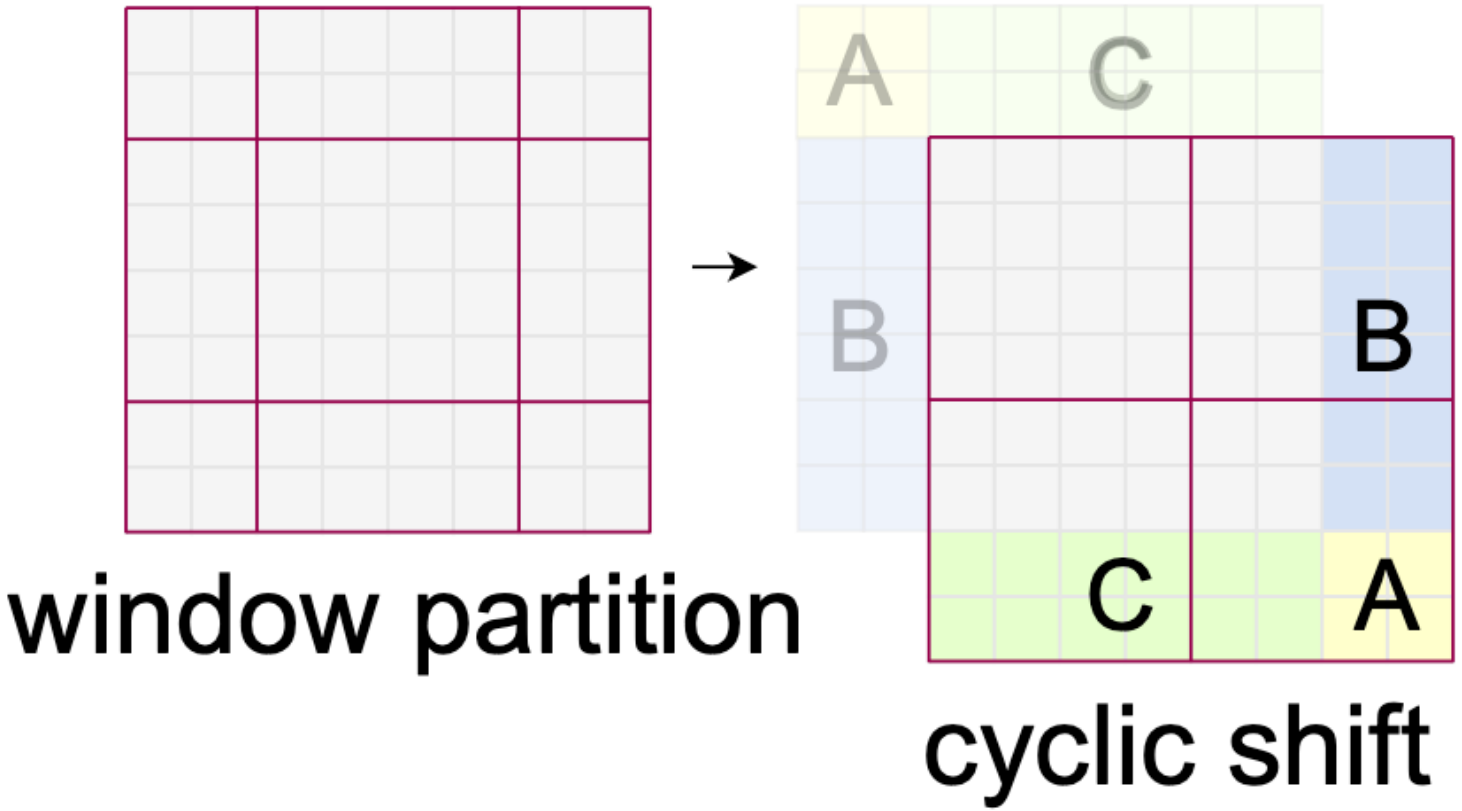
W-MSA

- Layer l has 4 windows, which means 4 attention mechanism

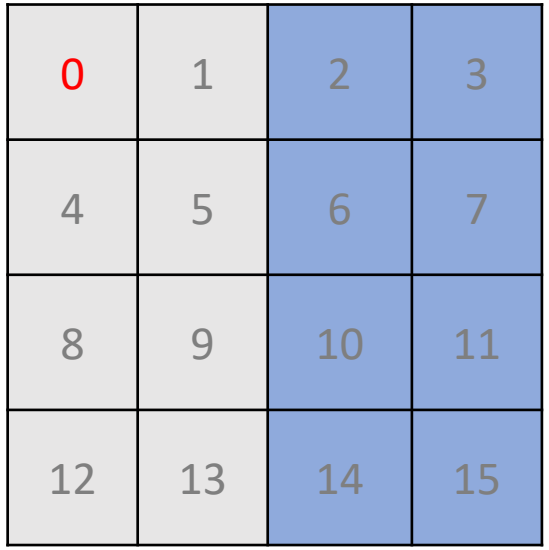
SW-MSA



SW-MSA



Masked MSA



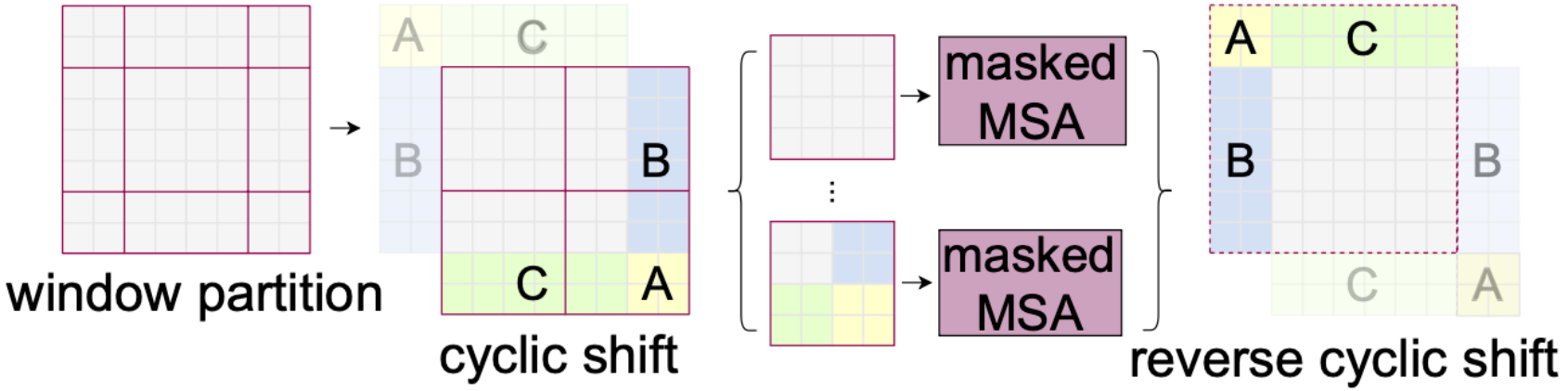
$\alpha_{0,0}$ $\alpha_{0,1}$ $\alpha_{0,2}$ $\alpha_{0,3}$ $\alpha_{0,4}$ $\alpha_{0,5}$ $\alpha_{0,6}$ $\alpha_{0,7}$ $\alpha_{0,8}$ $\alpha_{0,9}$ $\alpha_{0,10}$ $\alpha_{0,11}$ $\alpha_{0,12}$ $\alpha_{0,13}$ $\alpha_{0,14}$ $\alpha_{0,15}$

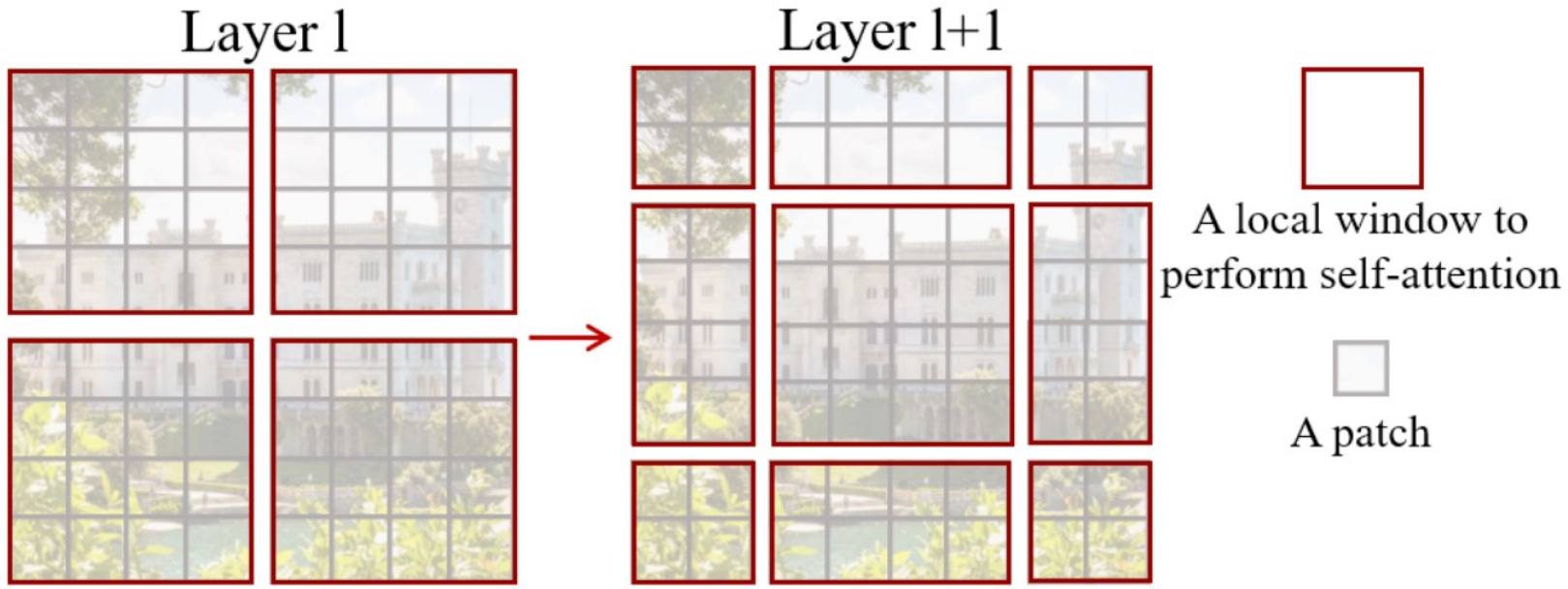
-100 -100 -100 -100

$softmax$

$\hat{\alpha}_{0,0}$ $\hat{\alpha}_{0,1}$ 0 0 $\hat{\alpha}_{0,4}$ $\hat{\alpha}_{0,5}$ 0 0 $\hat{\alpha}_{0,8}$ $\hat{\alpha}_{0,9}$ 0 0 $\hat{\alpha}_{0,12}$ $\hat{\alpha}_{0,13}$ 0 0

SW-MSA


A
C
B
B
C
A
window partition
cyclic shift
masked MSA
masked MSA
A
C
B
B
C
A
reverse cyclic shift

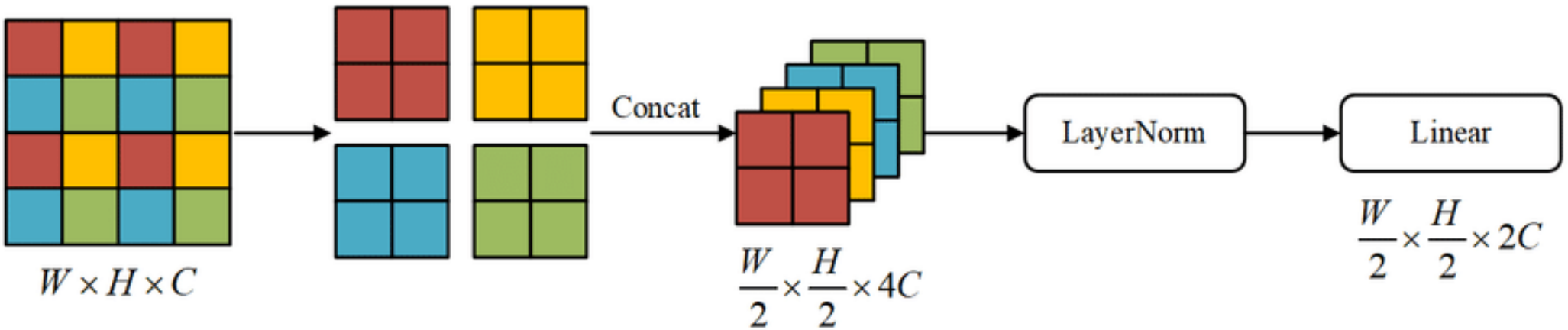

Layer l
Layer l+1
A local window to perform self-attention
A patch

Relative positive bias

$$\text{Attention}(Q, K, V) = \text{SoftMax}(QK^T/\sqrt{d} + B)V,$$

名稱	依據	學習	添加位置
Transformer	sin/cos 函數	不可以學習	concate 在開頭
Visual Transformer	絕對位置	可學習	加在開頭
Swin Transformer	相對位置	可學習	加在 attention matrix

Patch Merging



Detailed architecture specifications

	downsp. rate (output size)	Swin-T	Swin-S	Swin-B	Swin-L
stage 1	4× (56×56)	concat 4×4, 96-d, LN	concat 4×4, 96-d, LN	concat 4×4, 128-d, LN	concat 4×4, 192-d, LN
		[win. sz. 7×7, dim 96, head 3] × 2	[win. sz. 7×7, dim 96, head 3] × 2	[win. sz. 7×7, dim 128, head 4] × 2	[win. sz. 7×7, dim 192, head 6] × 2
stage 2	8× (28×28)	concat 2×2, 192-d , LN	concat 2×2, 192-d , LN	concat 2×2, 256-d , LN	concat 2×2, 384-d , LN
		[win. sz. 7×7, dim 192, head 6] × 2	[win. sz. 7×7, dim 192, head 6] × 2	[win. sz. 7×7, dim 256, head 8] × 2	[win. sz. 7×7, dim 384, head 12] × 2
stage 3	16× (14×14)	concat 2×2, 384-d , LN	concat 2×2, 384-d , LN	concat 2×2, 512-d , LN	concat 2×2, 768-d , LN
		[win. sz. 7×7, dim 384, head 12] × 6	[win. sz. 7×7, dim 384, head 12] × 18	[win. sz. 7×7, dim 512, head 16] × 18	[win. sz. 7×7, dim 768, head 24] × 18
stage 4	32× (7×7)	concat 2×2, 768-d , LN	concat 2×2, 768-d , LN	concat 2×2, 1024-d , LN	concat 2×2, 1536-d , LN
		[win. sz. 7×7, dim 768, head 24] × 2	[win. sz. 7×7, dim 768, head 24] × 2	[win. sz. 7×7, dim 1024, head 32] × 2	[win. sz. 7×7, dim 1536, head 48] × 2

Scores

(a) Regular ImageNet-1K trained models

method	image size	#param.	FLOPs	throughput (image / s)	ImageNet top-1 acc.
RegNetY-4G [48]	224^2	21M	4.0G	1156.7	80.0
RegNetY-8G [48]	224^2	39M	8.0G	591.6	81.7
RegNetY-16G [48]	224^2	84M	16.0G	334.7	82.9
EffNet-B3 [58]	300^2	12M	1.8G	732.1	81.6
EffNet-B4 [58]	380^2	19M	4.2G	349.4	82.9
EffNet-B5 [58]	456^2	30M	9.9G	169.1	83.6
EffNet-B6 [58]	528^2	43M	19.0G	96.9	84.0
EffNet-B7 [58]	600^2	66M	37.0G	55.1	84.3
ViT-B/16 [20]	384^2	86M	55.4G	85.9	77.9
ViT-L/16 [20]	384^2	307M	190.7G	27.3	76.5
DeiT-S [63]	224^2	22M	4.6G	940.4	79.8
DeiT-B [63]	224^2	86M	17.5G	292.3	81.8
DeiT-B [63]	384^2	86M	55.4G	85.9	83.1
Swin-T	224^2	29M	4.5G	755.2	81.3
Swin-S	224^2	50M	8.7G	436.9	83.0
Swin-B	224^2	88M	15.4G	278.1	83.5
Swin-B	384^2	88M	47.0G	84.7	84.5

(b) ImageNet-22K pre-trained models

method	image size	#param.	FLOPs	throughput (image / s)	ImageNet top-1 acc.
R-101x3 [38]	384^2	388M	204.6G	-	84.4
R-152x4 [38]	480^2	937M	840.5G	-	85.4
ViT-B/16 [20]	384^2	86M	55.4G	85.9	84.0
ViT-L/16 [20]	384^2	307M	190.7G	27.3	85.2
Swin-B	224^2	88M	15.4G	278.1	85.2
Swin-B	384^2	88M	47.0G	84.7	86.4
Swin-L	384^2	197M	103.9G	42.1	87.3

Ablation Study

- Window
- Shifted window
- Patch merging
- Relatively Position Embedding

Reference

- Attention Is All You Need(https://arxiv.org/abs/1706.03762)
- Applied Deep Learning 2022(https://www.csie.ntu.edu.tw/~miulab/f111-adl/)
- An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale(https://arxiv.org/abs/2010.11929)
- Swin Transformer: Hierarchical Vision Transformer using Shifted Windows(https://arxiv.org/abs/2103.14030)
- 210523 swin transformer v1.5(https://www.slideshare.net/taeseonryu/210523-swin-transformer-v15)