[Reminder] Midterm course feedback survey: 10/14 - 10/25
Feedback is welcome!

Deep Learning for Computer Vision

113-1/Fall 2024

https://cool.ntu.edu.tw/courses/41702 (NTU COOL)

http://vllab.ee.ntu.edu.tw/dlcv.html (Public website)

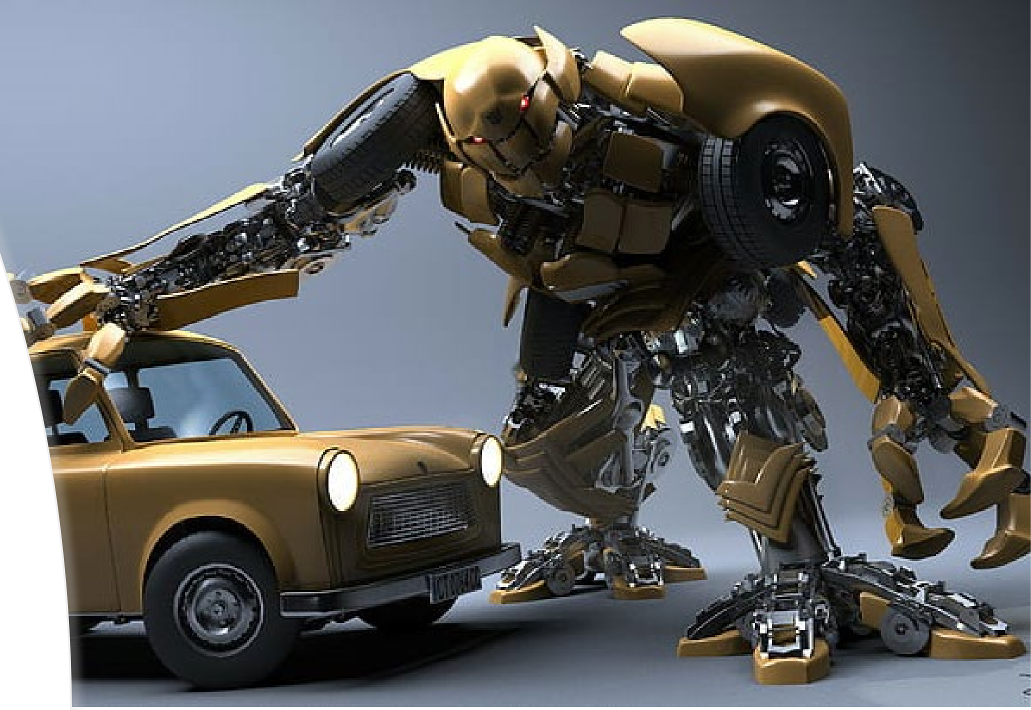
Yu-Chiang Frank Wang 王鈺強, Professor

Dept. Electrical Engineering, National Taiwan University

2024/10/22

What to Be Covered?

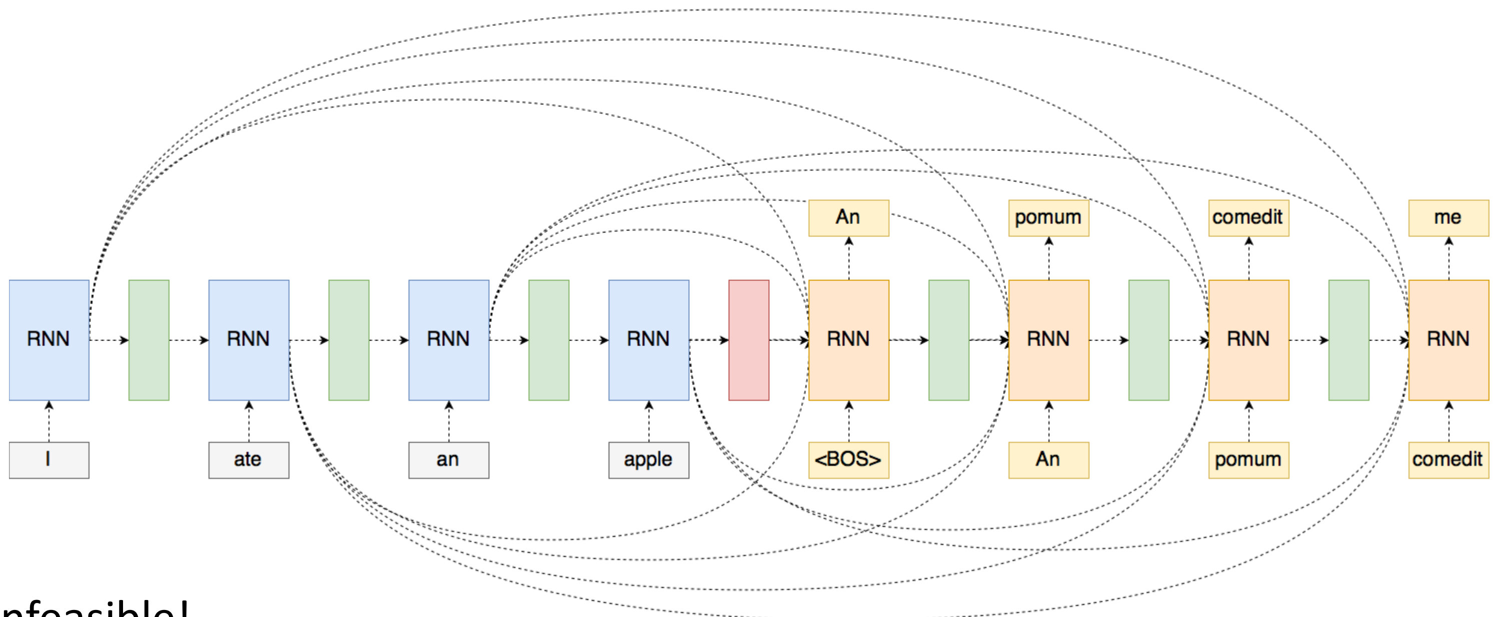
- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



https://medium.com/@navendubrajesh/vision-language-models-use-cases-ee6d54b2c557

RNN with Attention is Good, But..

- Attention in a pre-defined sequential order
- Information loss due to long sequences…
- Connecting every hidden state between encoder and decoder?



- Infeasible!
  - Both inputs and outputs are with varying sizes.
  - Overparameterized

Tokenization

Many words map to one token, but some don't: indivisible.

Many	words	map	to	one	token	,	but	some	don	't	:	indiv	isible	.
8607	4339	2472	311	832	4037	11	719	1063	1541	956	25	3687	23936	13

One-hot encoding

tokens

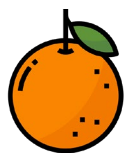
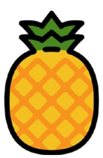
$$
\begin{array}{ccccc}
\text{cat} & \text{dog} & \text{bear} & \text{cow} & \text{indiv} \\
\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687th entry

Slide credit: Prof. J.-B. Huang @ U. Maryland

TOKEN EMBEDDING

One-hot encoding

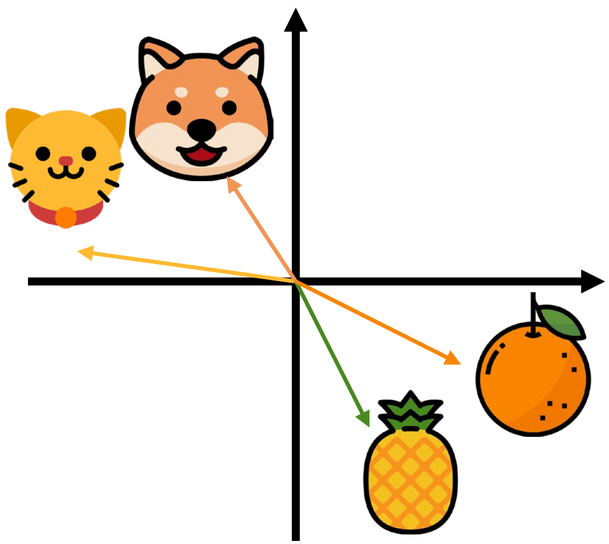





$$
\begin{array}{ccccc}
\text{cat} & \text{dog} & \text{bear} & \text{cow} & \text{indiv} \\
\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687^{th} entry

Slide credit: Prof. J.-B. Huang @ U. Maryland

TOKEN EMBEDDING



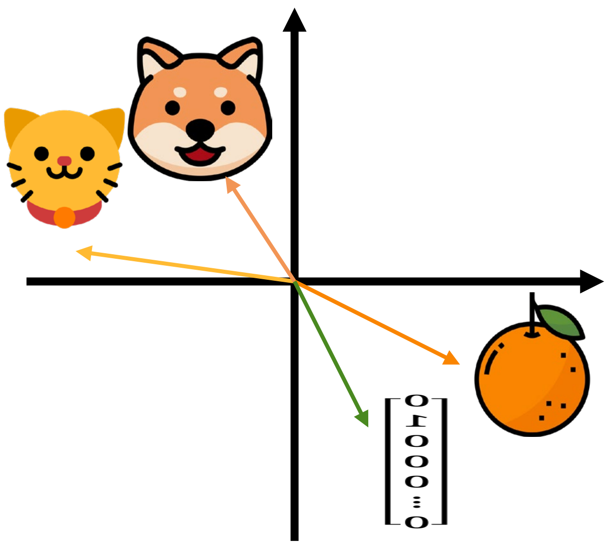
Embedding Space

$$
\begin{array}{ccccc}
\text{cat} & \text{dog} & \text{bear} & \text{cow} & \text{indiv} \\
\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687^{th} entry

Slide credit: Prof. J.-B. Huang @ U. Maryland

TOKEN EMBEDDING



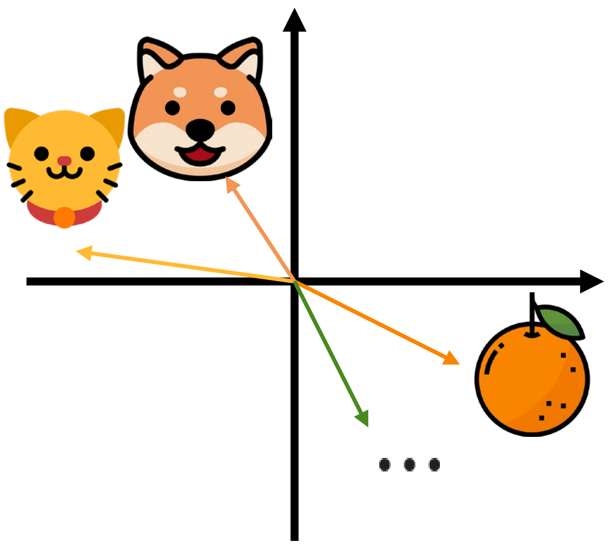
Embedding Space

$$\underbrace{\begin{bmatrix} 0.5 \\ 2.7 \\ 1.2 \\ \vdots \\ 0.2 \end{bmatrix}}_{\text{Embedded token}} d = d \underbrace{\overbrace{\begin{bmatrix} & W_E & \end{bmatrix}}^{\#\text{ tokens}}}_{\text{Embedding Matrix}} \begin{array}{c} \text{dog} \\ \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \end{array}$$

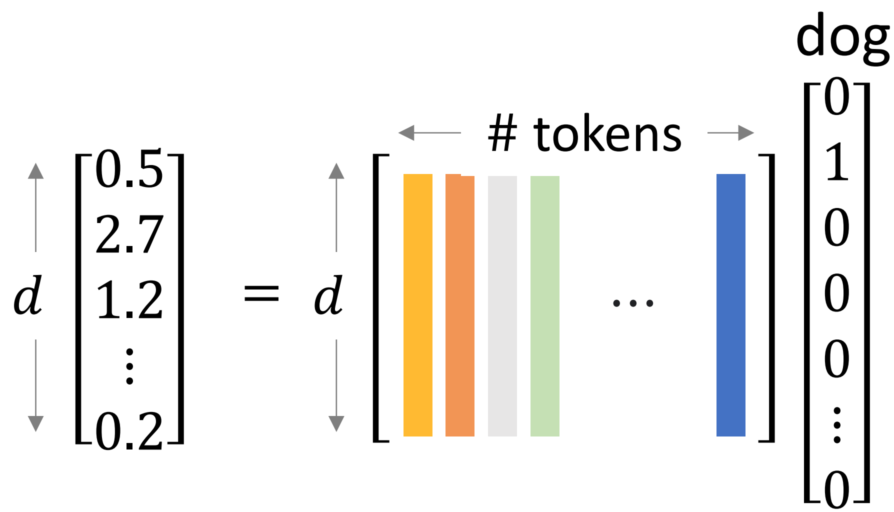
Embedded token

Embedding Matrix

TOKEN EMBEDDING



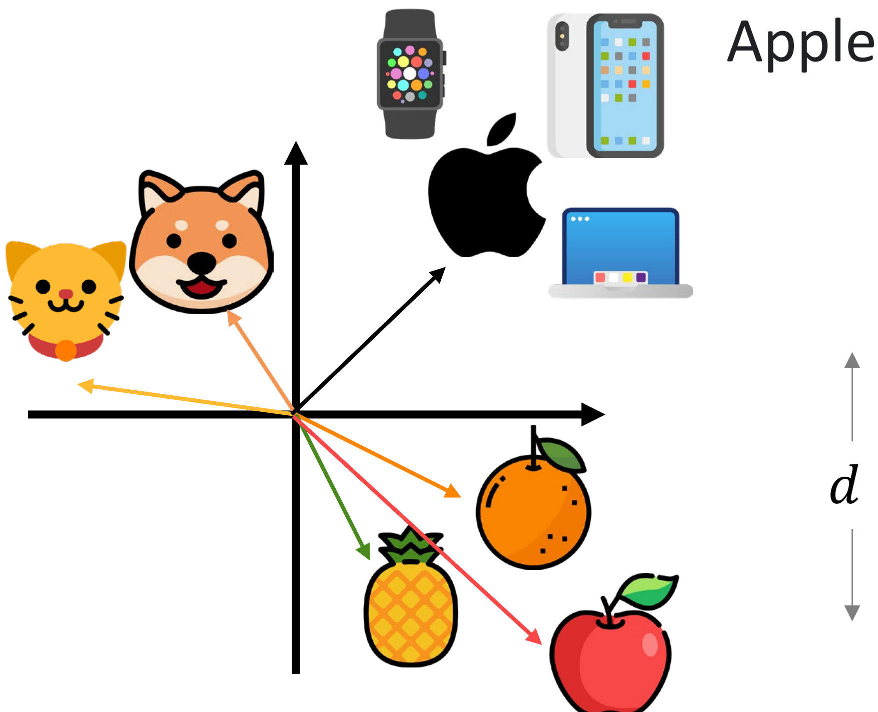
Embedding Space



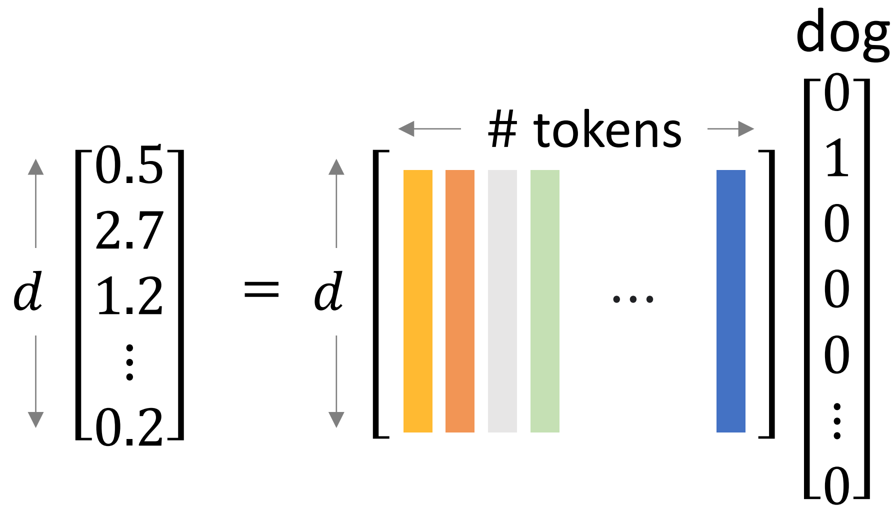
Embedded token

Embedding Matrix

TOKEN EMBEDDING



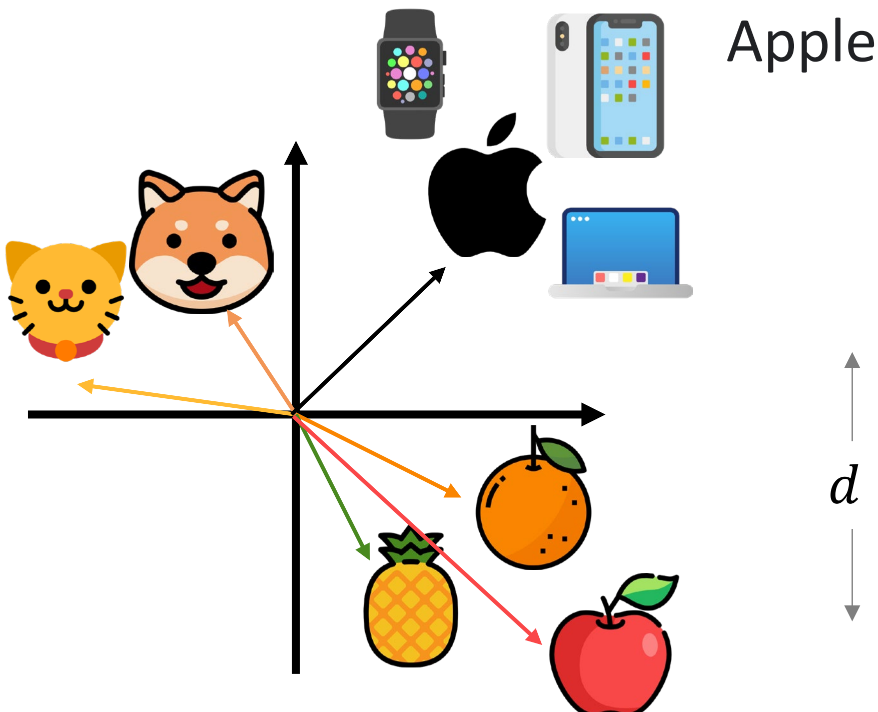
Embedding Space



Embedded token

Embedding Matrix

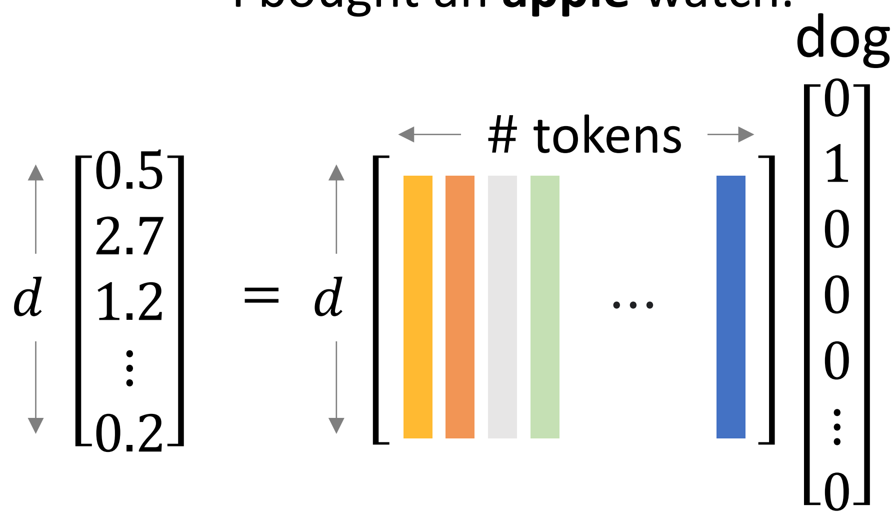
TOKEN EMBEDDING



Embedding Space

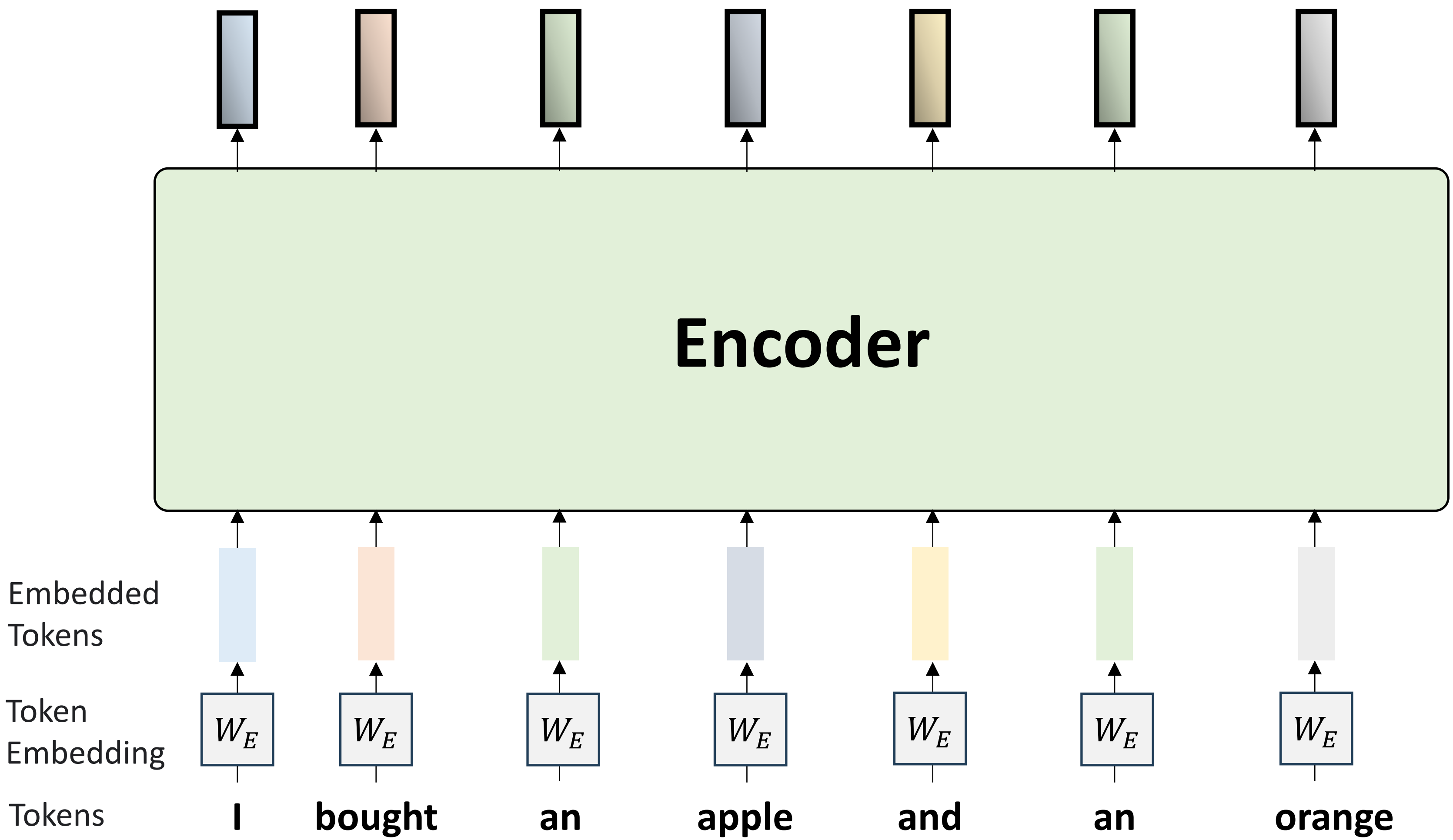
I bought an **apple** and an orange.

I bought an **apple** watch.

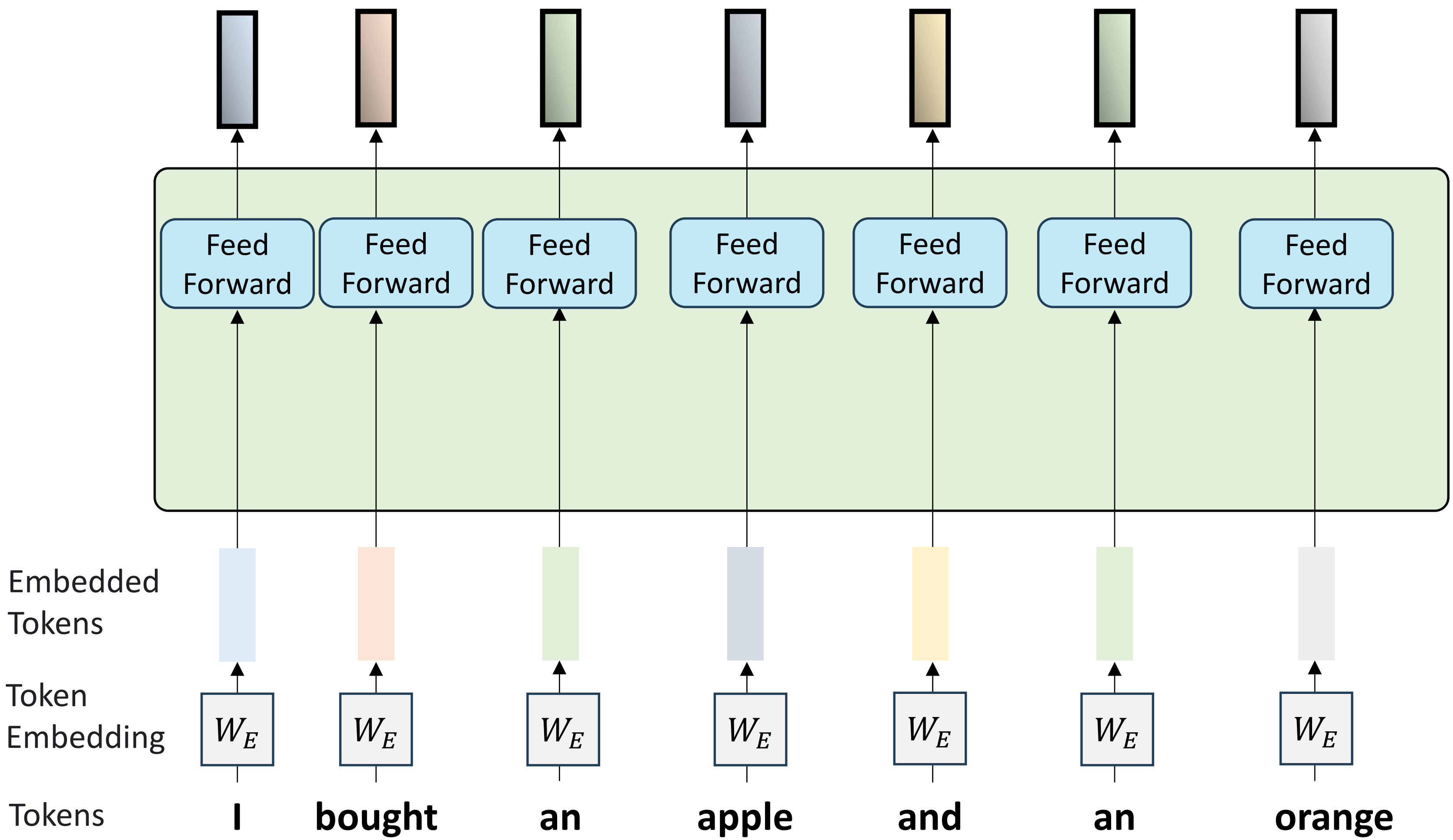


Embedded token

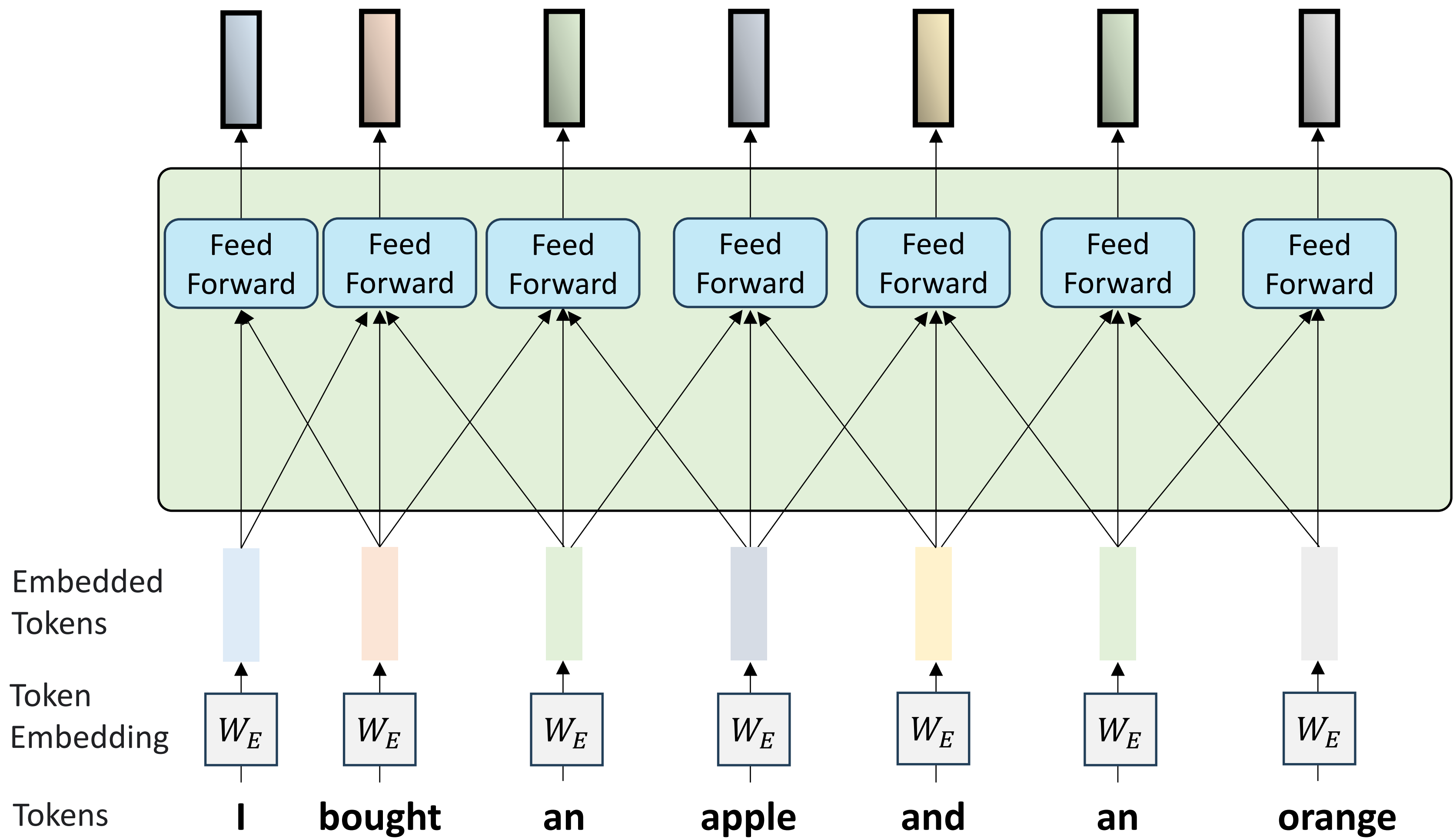
Embedding Matrix



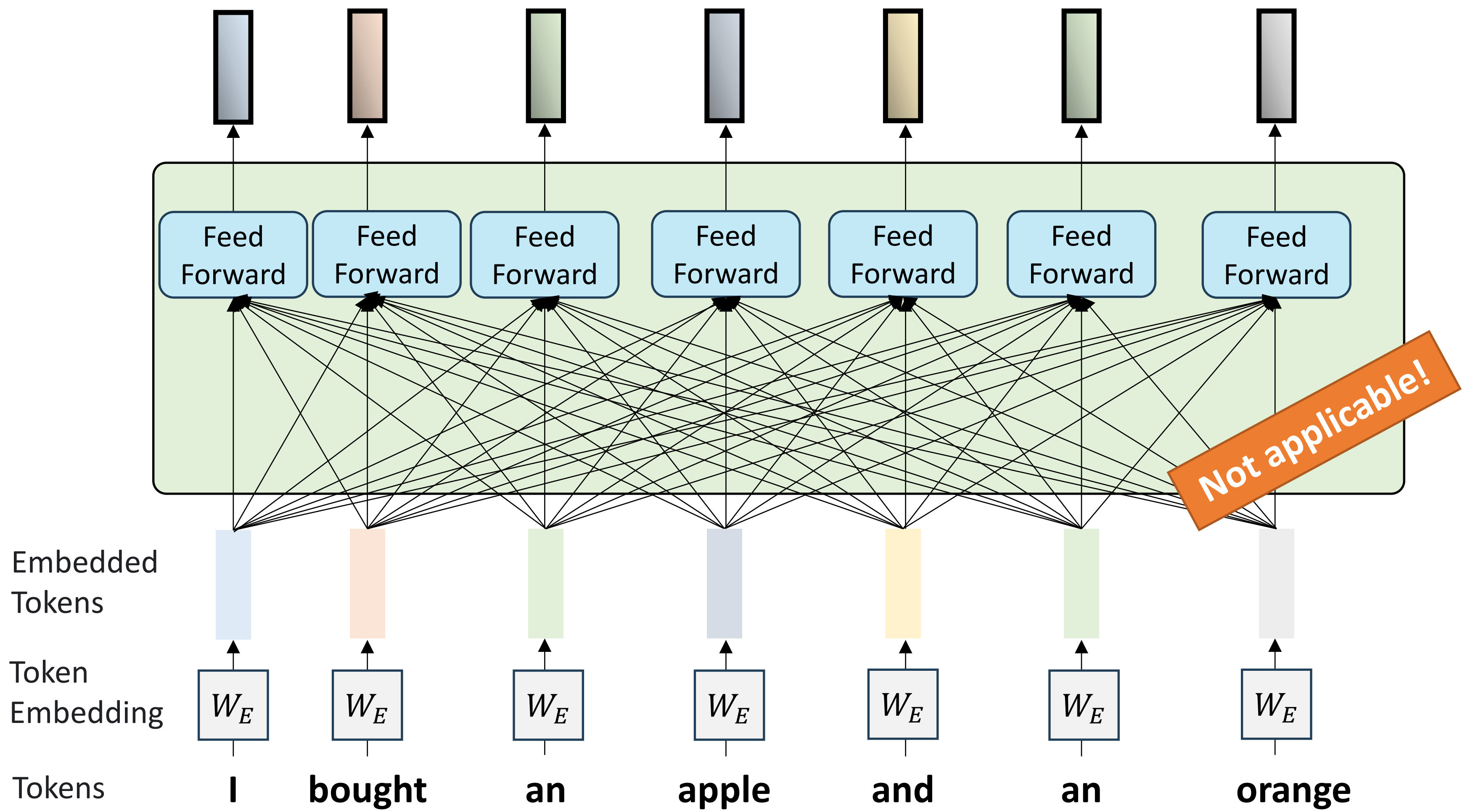
Slide credit: Prof. J.-B. Huang @ U. Maryland



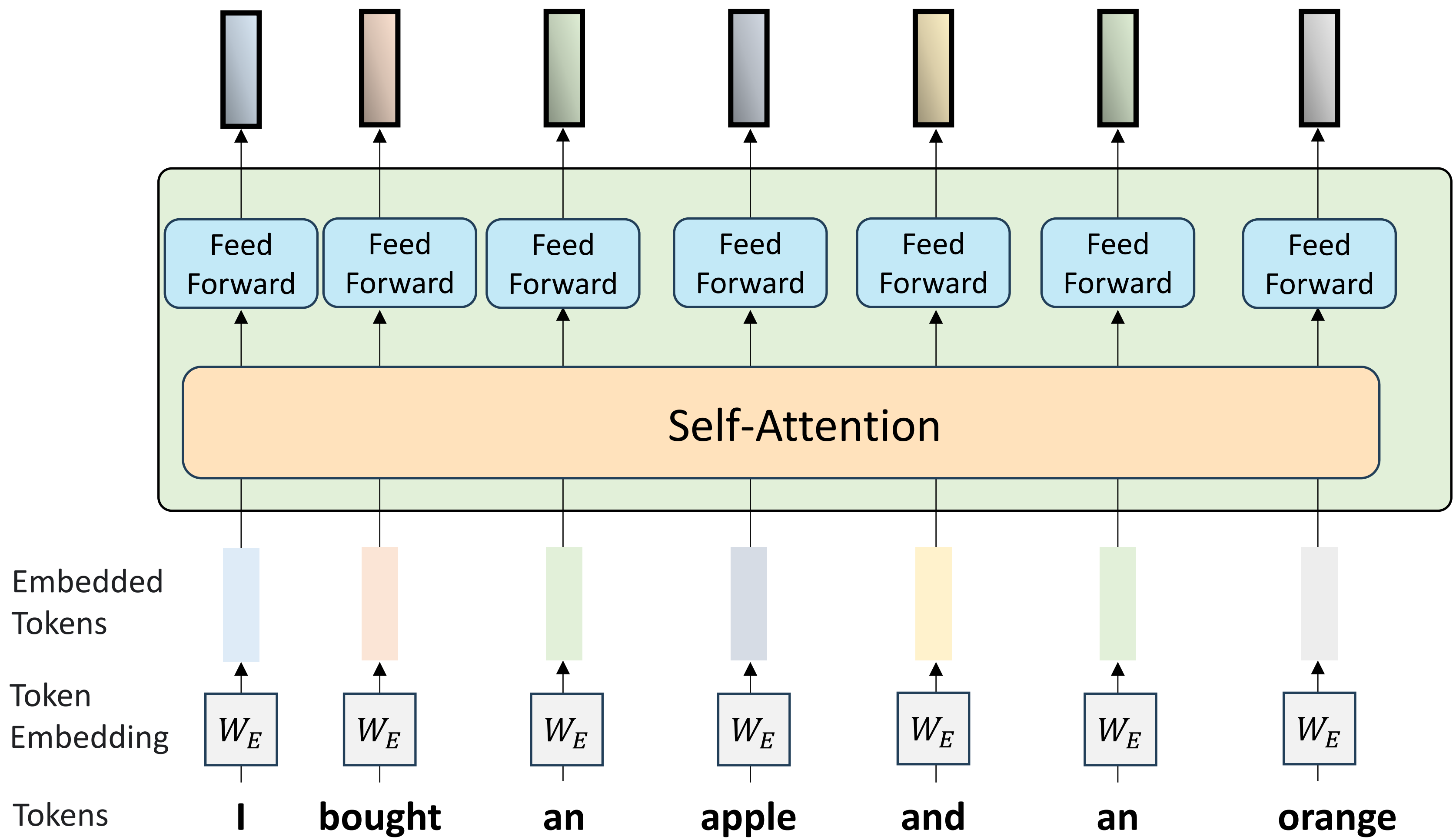
Slide credit: Prof. J.-B. Huang @ U. Maryland



Slide credit: Prof. J.-B. Huang @ U. Maryland



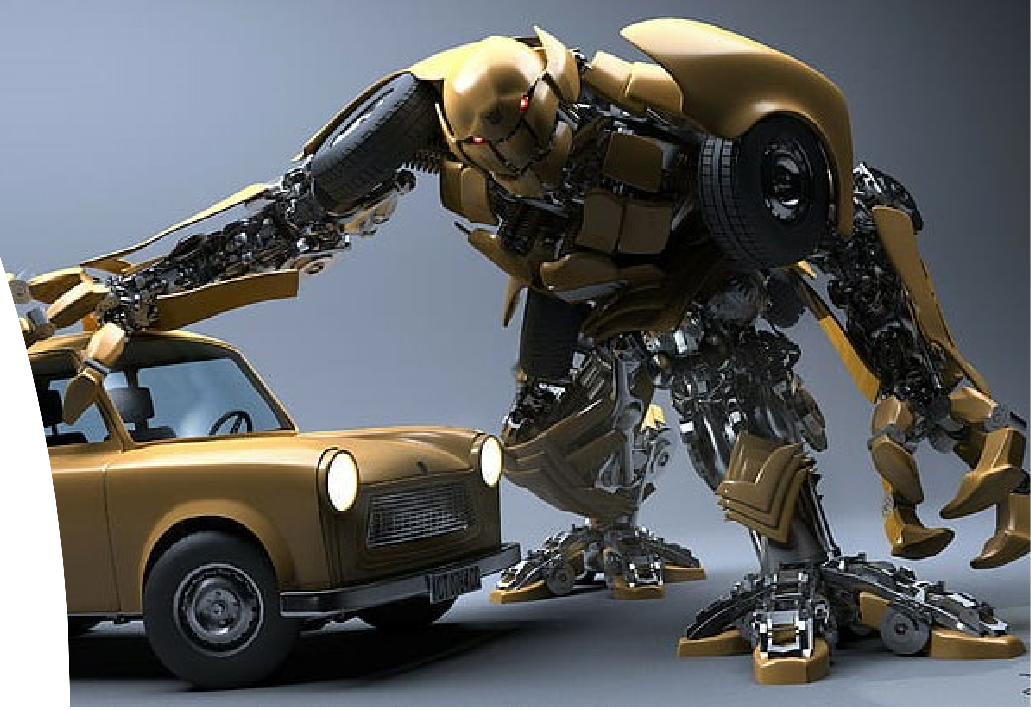
Slide credit: Prof. J.-B. Huang @ U. Maryland



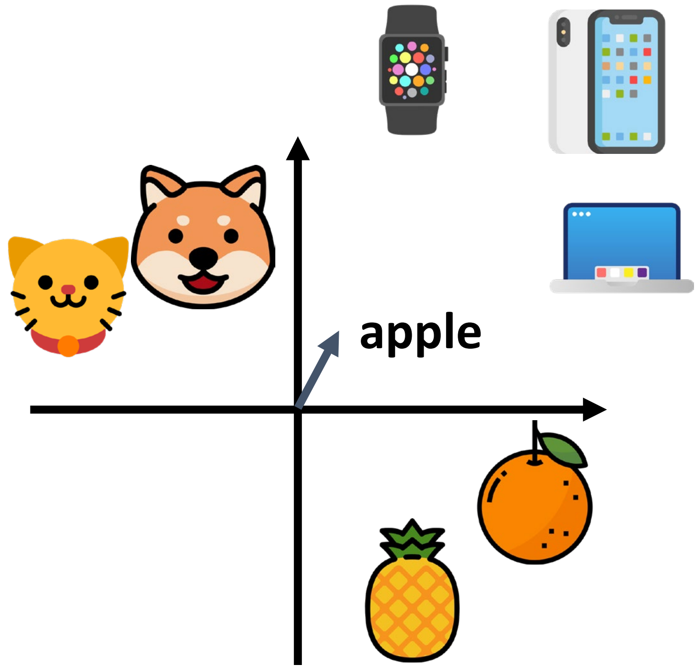
Slide credit: Prof. J.-B. Huang @ U. Maryland

What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (v)
  - Image-text models



Self-Attention



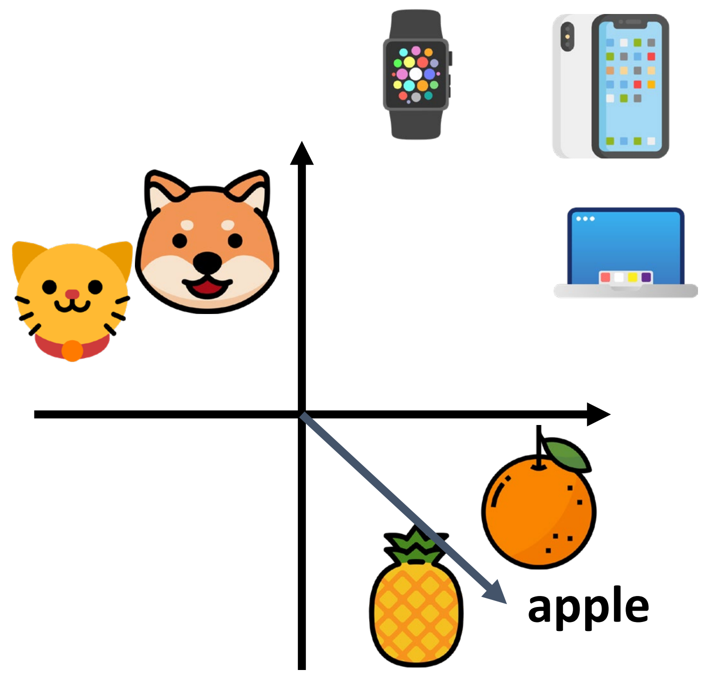
Embedding Space

Embedded Tokens

Tokens: **I** **bought** **an** **apple** **and** **an** **orange**

Slide credit: Prof. J.-B. Huang @ U. Maryland

Self-Attention



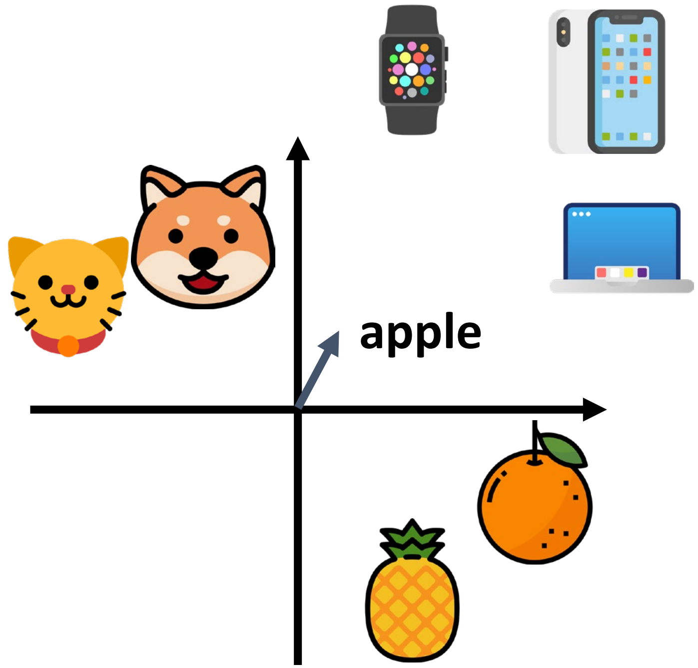
Embedding Space

Embedded Tokens

Tokens **I** **bought** **an** **apple** **and** **an** **orange**

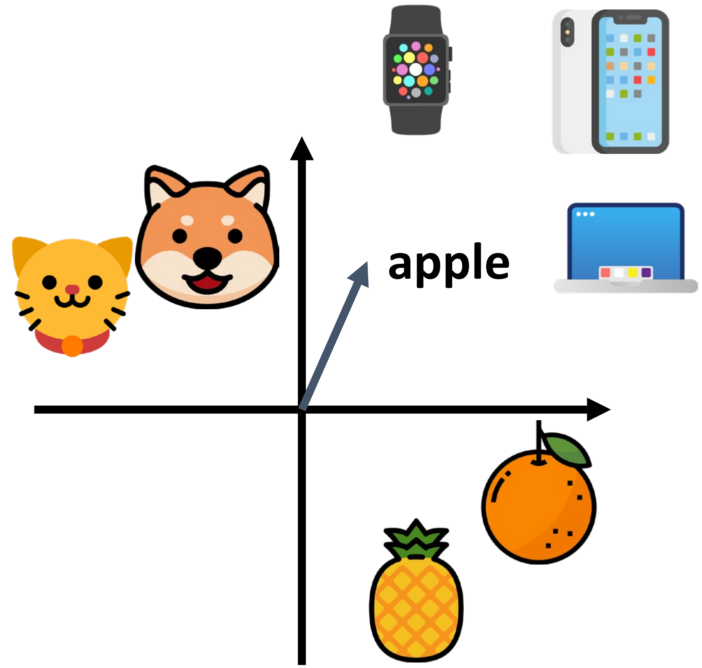
Slide credit: Prof. J.-B. Huang @ U. Maryland

Self-Attention



Embedding Space

Self-Attention

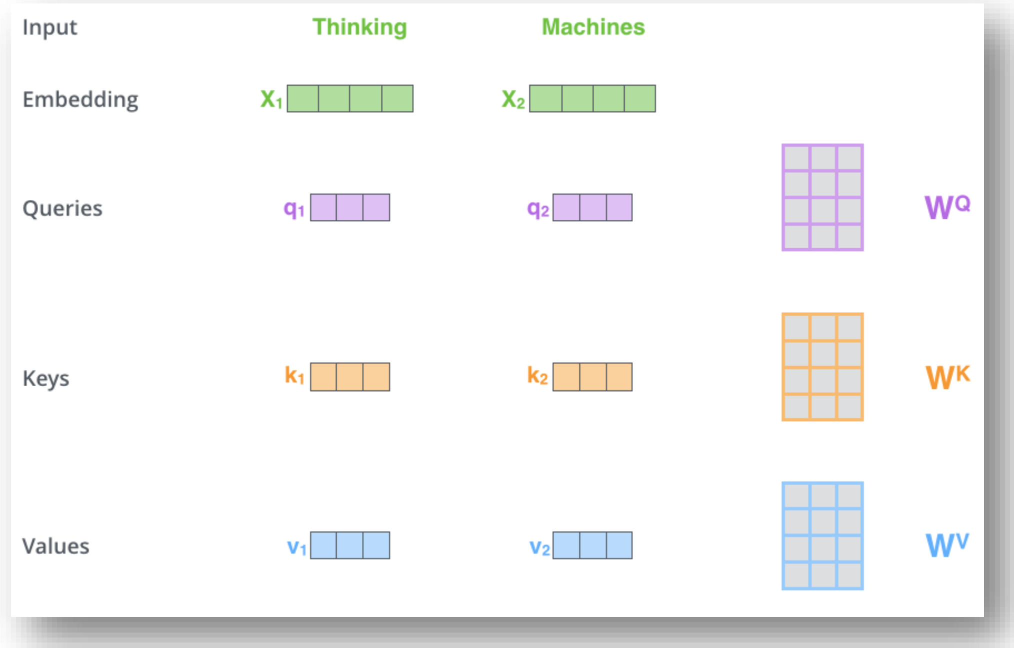


Embedding Space

Embedded Tokens

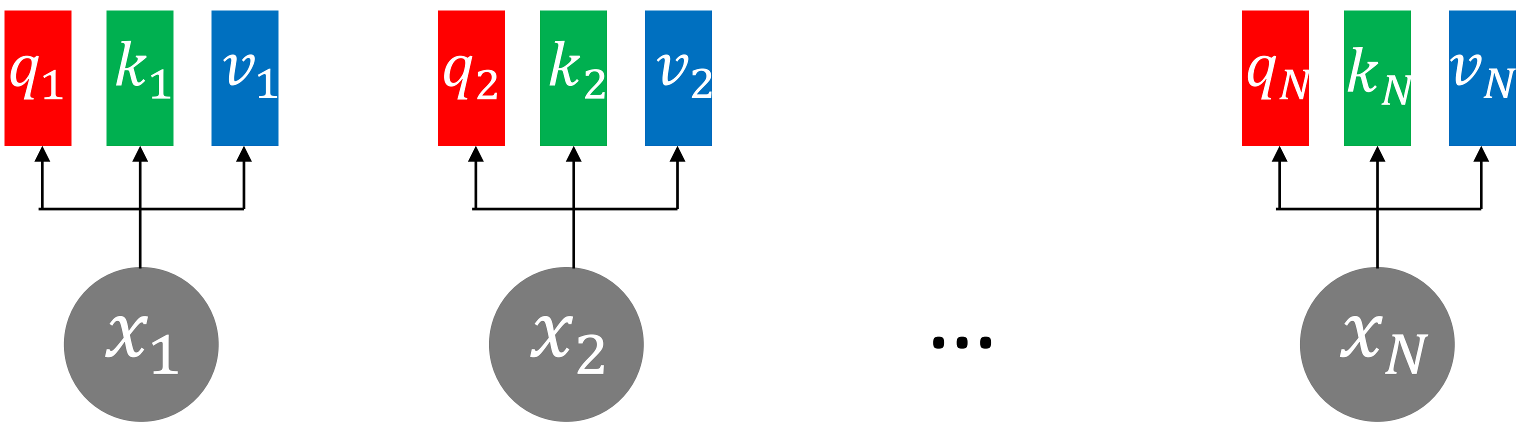
Tokens **I** **bought** **an** **apple** **watch**

Self-Attention (1/5)



- Query ***q***, key ***k***, value ***v*** vectors are learned from each input ***x***

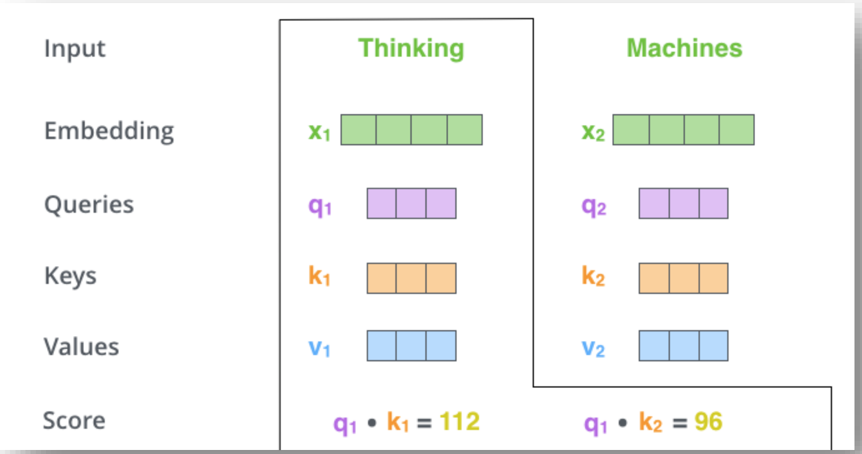
$$q_i = W^Q x_i$$
$$k_i = W^K x_i$$
$$v_i = W^V x_i$$

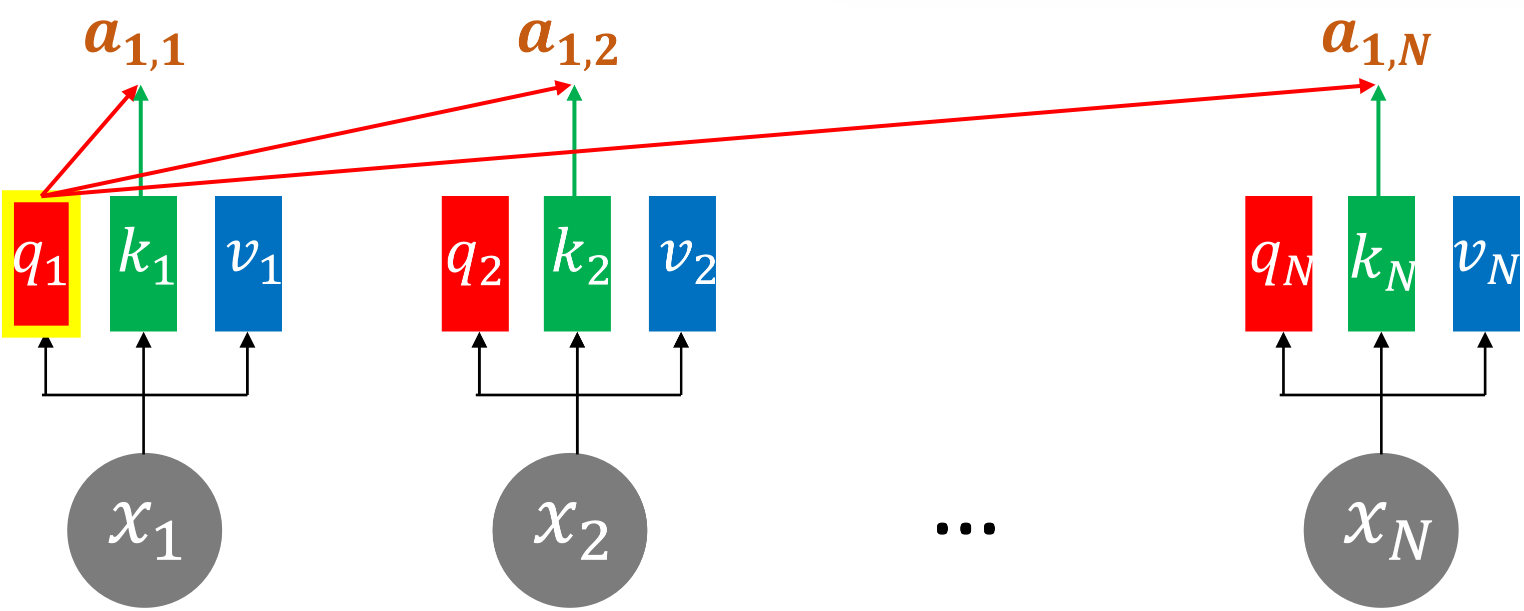


Self-Attention (2/5)

- Relation between each input is modeled by inner-product of query $\boldsymbol{q}$ and key $\boldsymbol{k}$.

$$a_{1,i} = \frac{q_1 \cdot k_i}{\sqrt{d}}, \text{ where } a \in R, q, k \in R^d$$

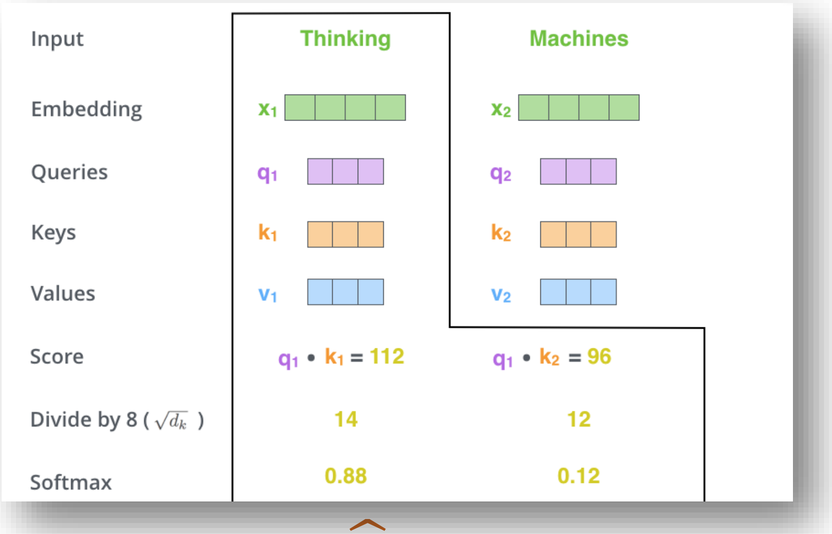


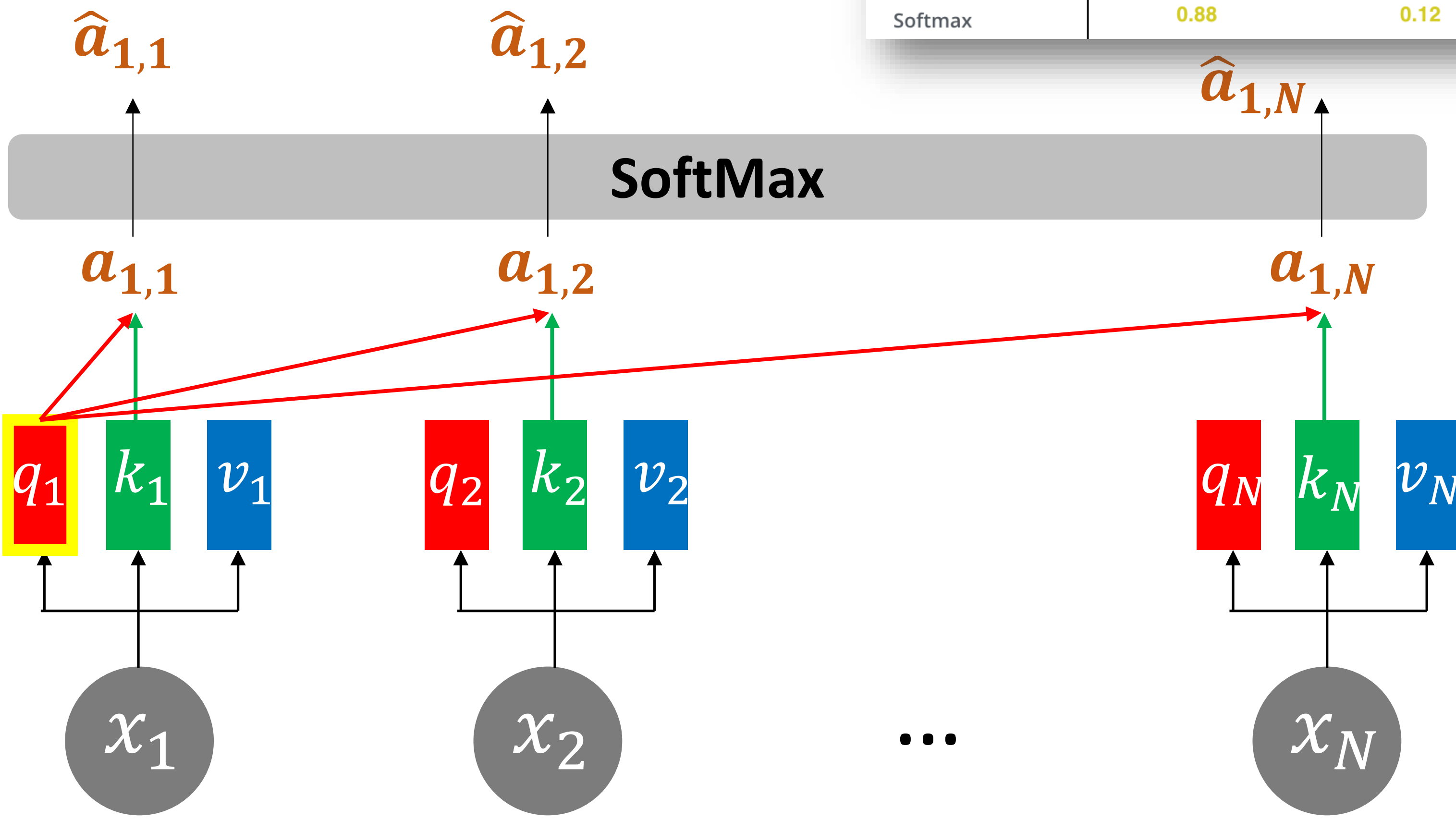


Self-Attention (3/5)

- SoftMax is applied:

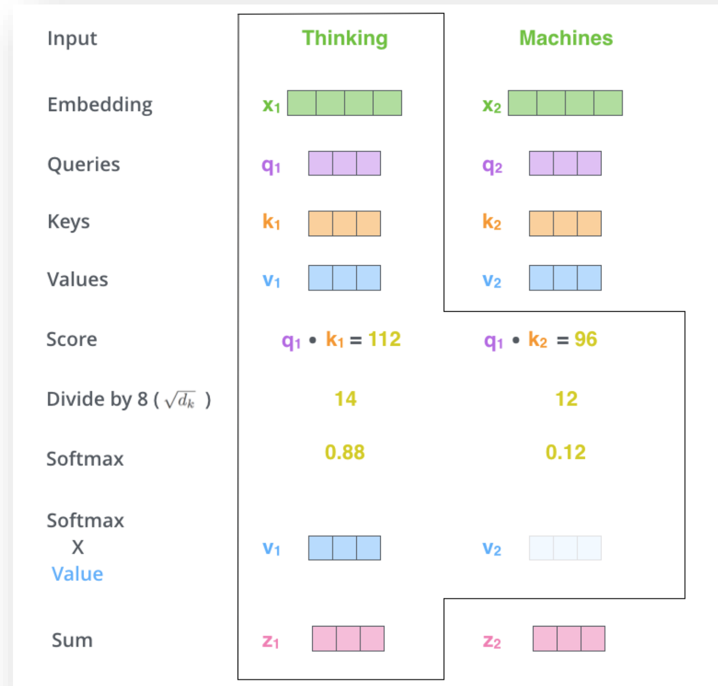
  $0 \leq \hat{a}_i = e^{a_i} / \sum_j^N e^{a_j} \leq 1$, for I =1, …, N

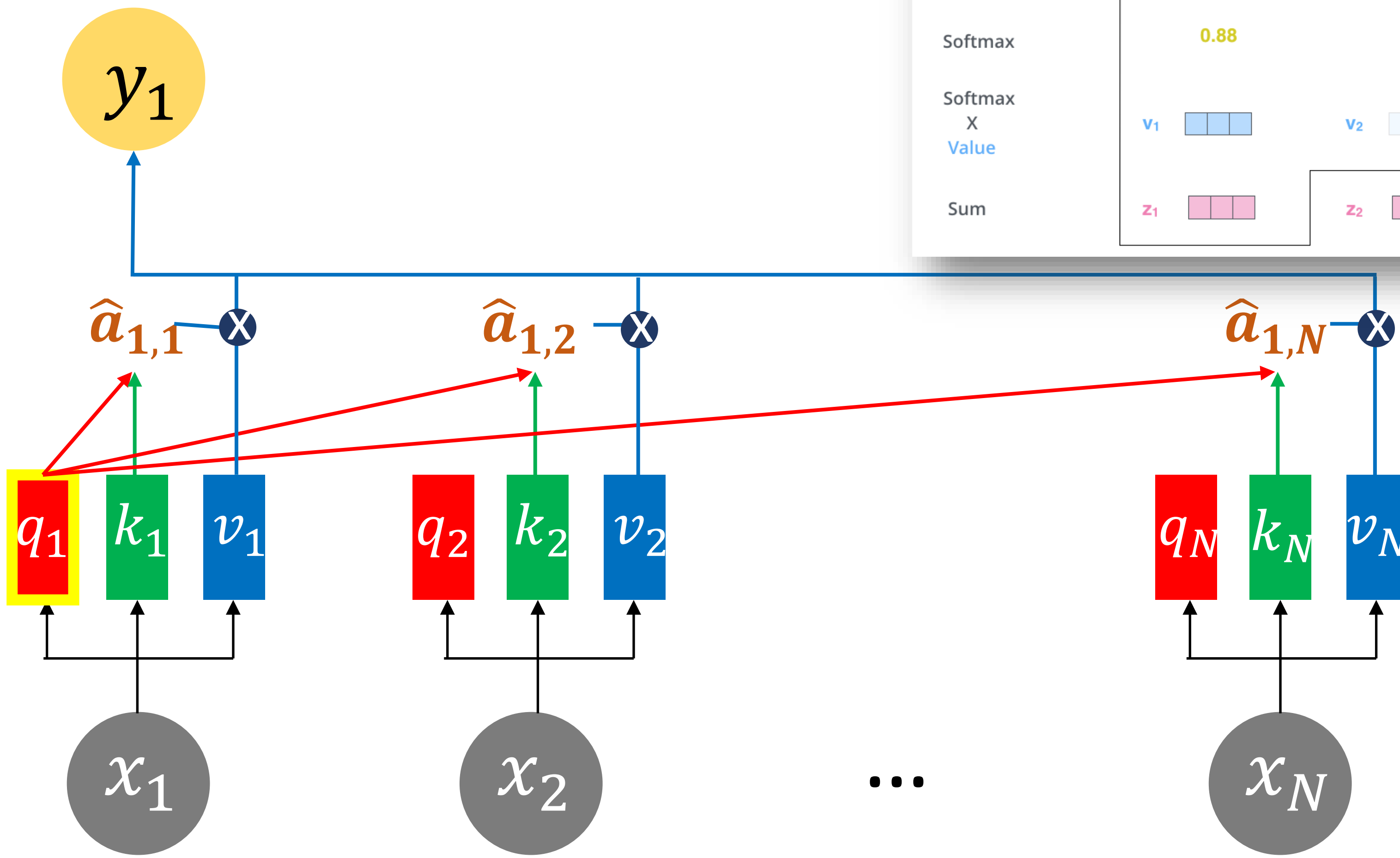




Self-Attention (4/5)

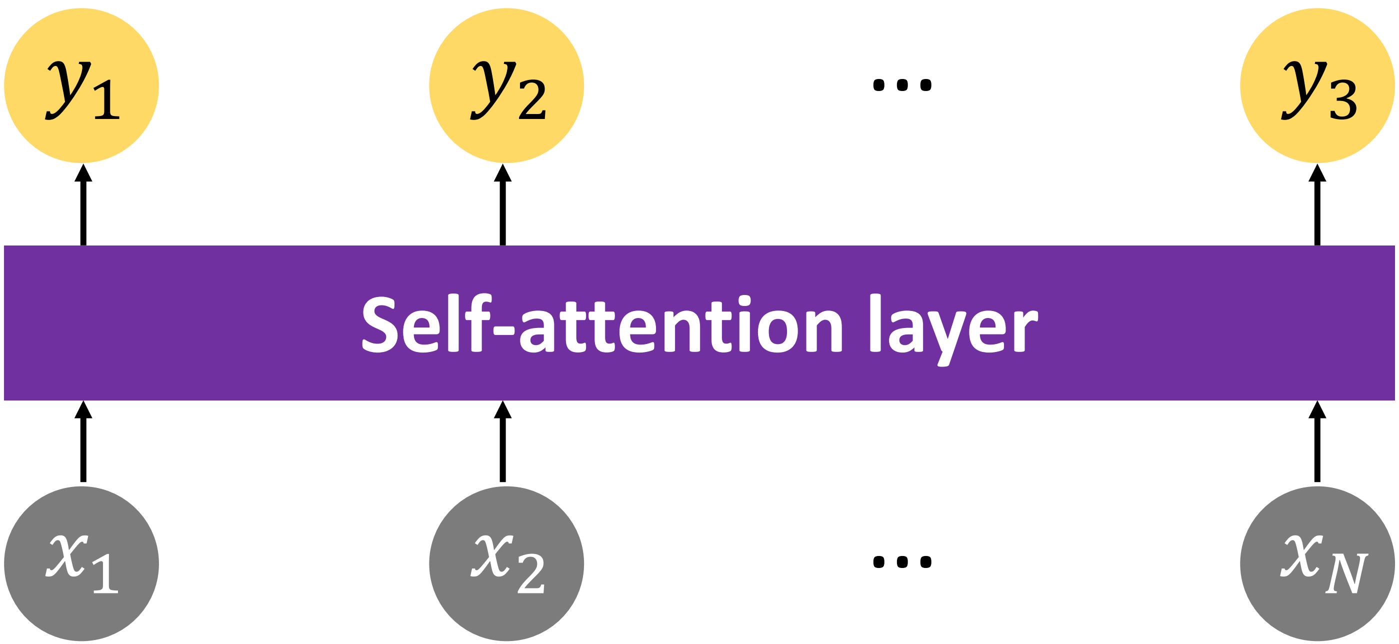
- Value vectors $\boldsymbol{v}$ are aggregated with attention weight $\hat{a}$, i.e., $y_1 = \sum_i^N \hat{a}_i \cdot v_i$





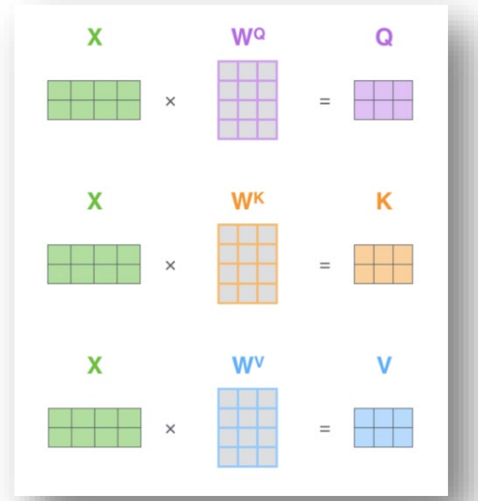
Self-Attention (5/5)

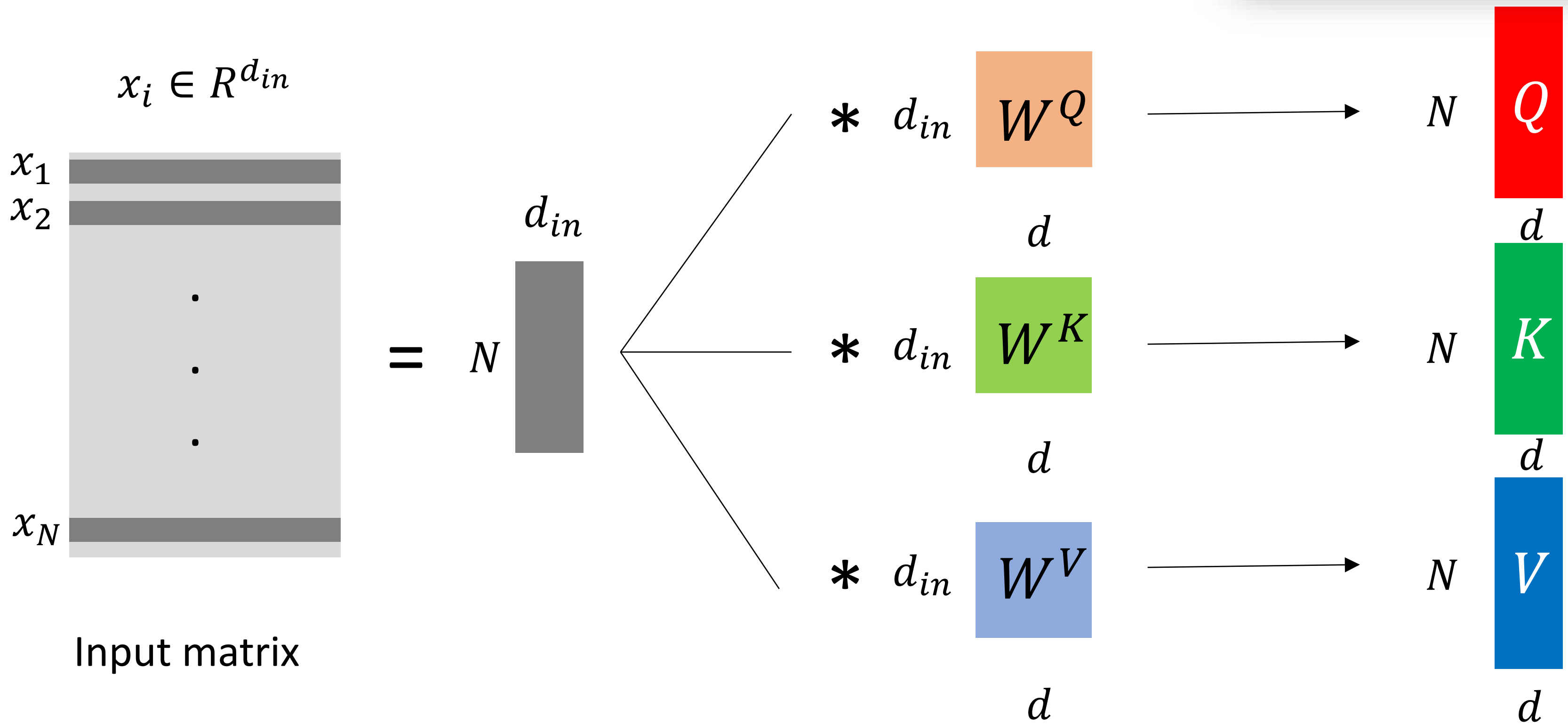
- All y_i can be computed **in parallel**
- Each y_i considers $x_1 \sim x_N$, modeling their **long-distance dependencies**.
- Global feature can be obtained by **average-pooling** over $y_1 \sim y_N$



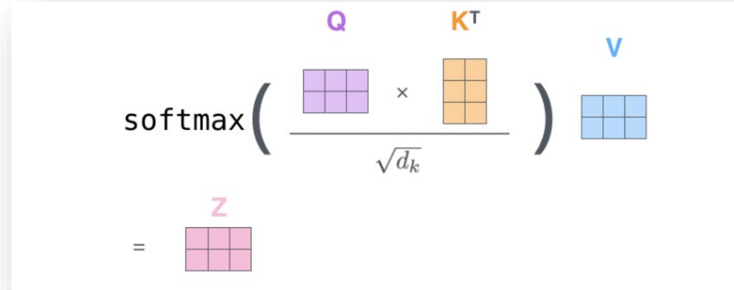
Self-Attention: Implementation

- Input sequence can be represented as a N x d_{in} matrix
- * denotes matrix multiplication

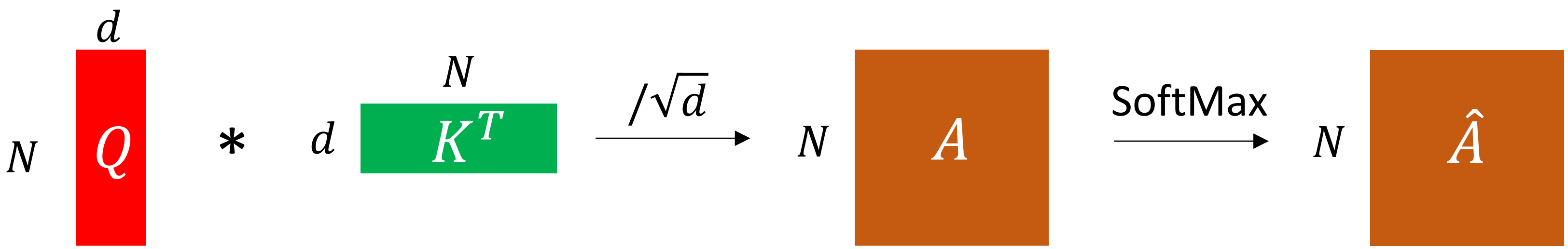


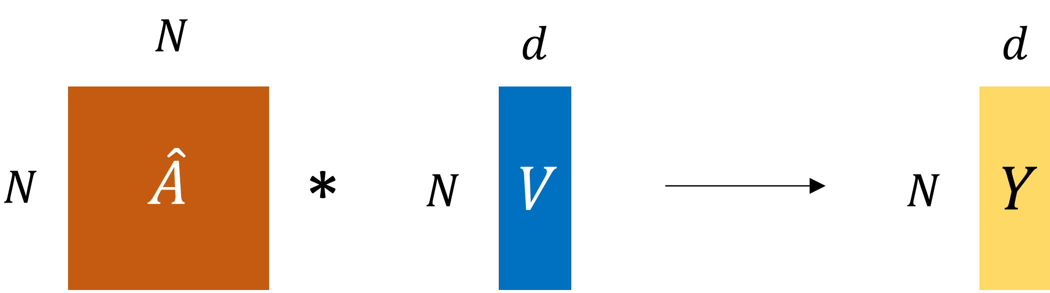


Self-Attention: Implementation



- Output matrix **Y**
- All operations are **matrix multiplication**, can be parallelized on GPU.

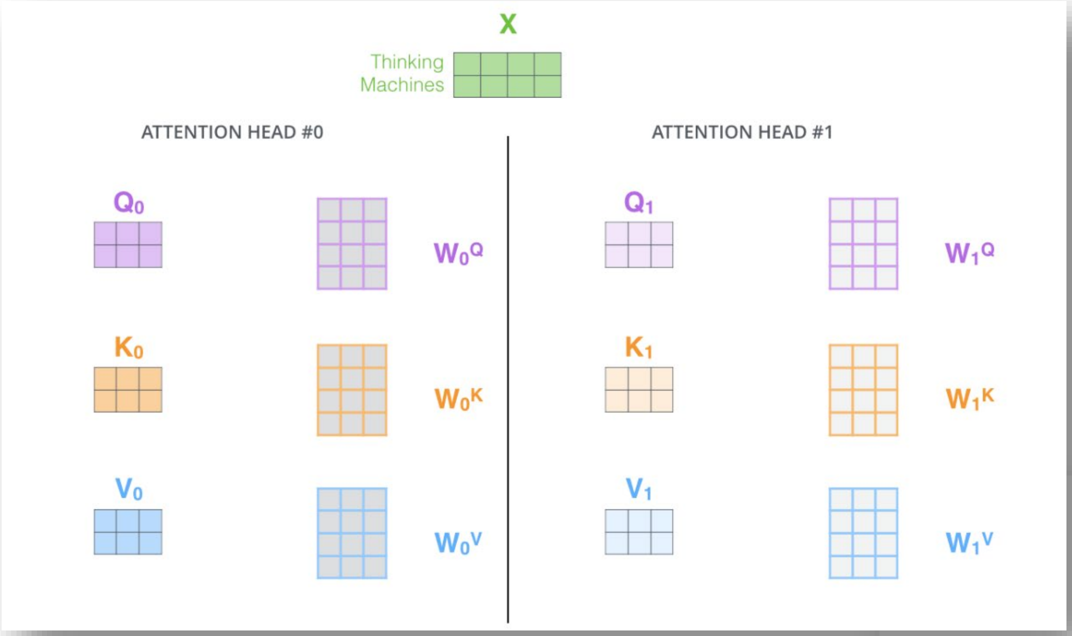


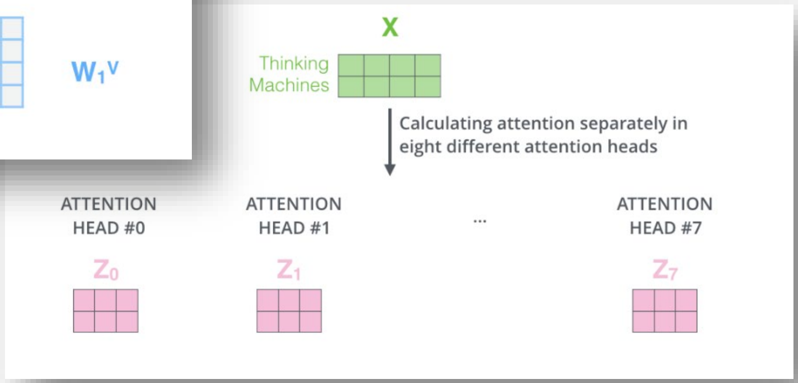


$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

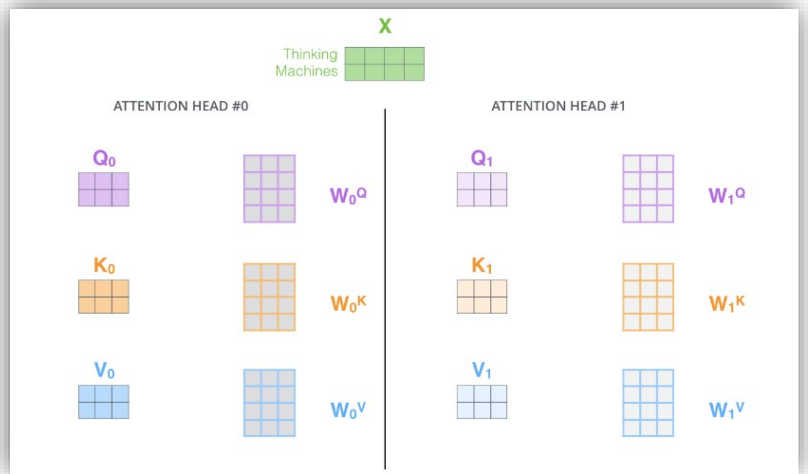
Multi-Head Self-Attention (1/4)

- Perform self-attention at different subspaces, implying performing attention over different input feature types (e.g., representations, modalities, positions, etc.)

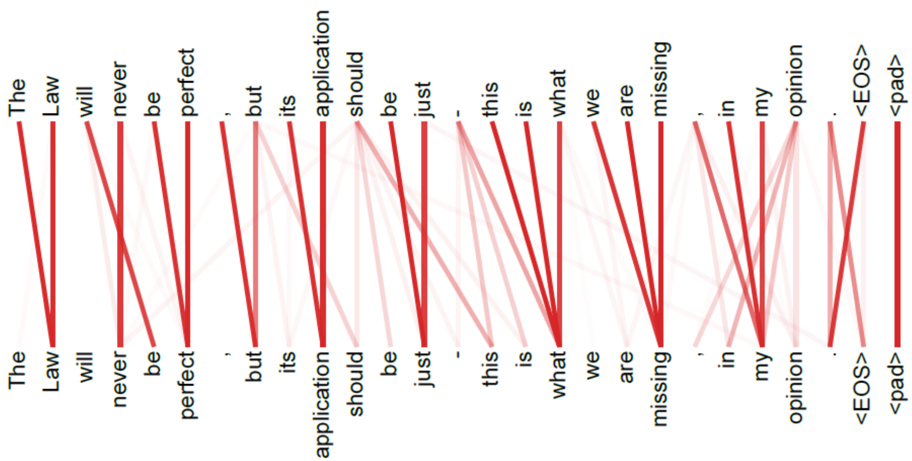




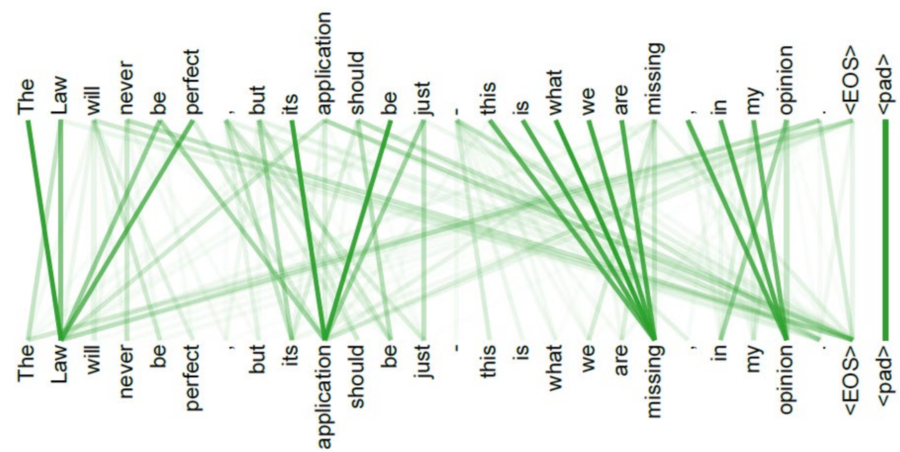
Multi-Head Self-Attention (2/4)



- Perform self-attention at different subspaces, implying performing attention over different input feature types
- See example below



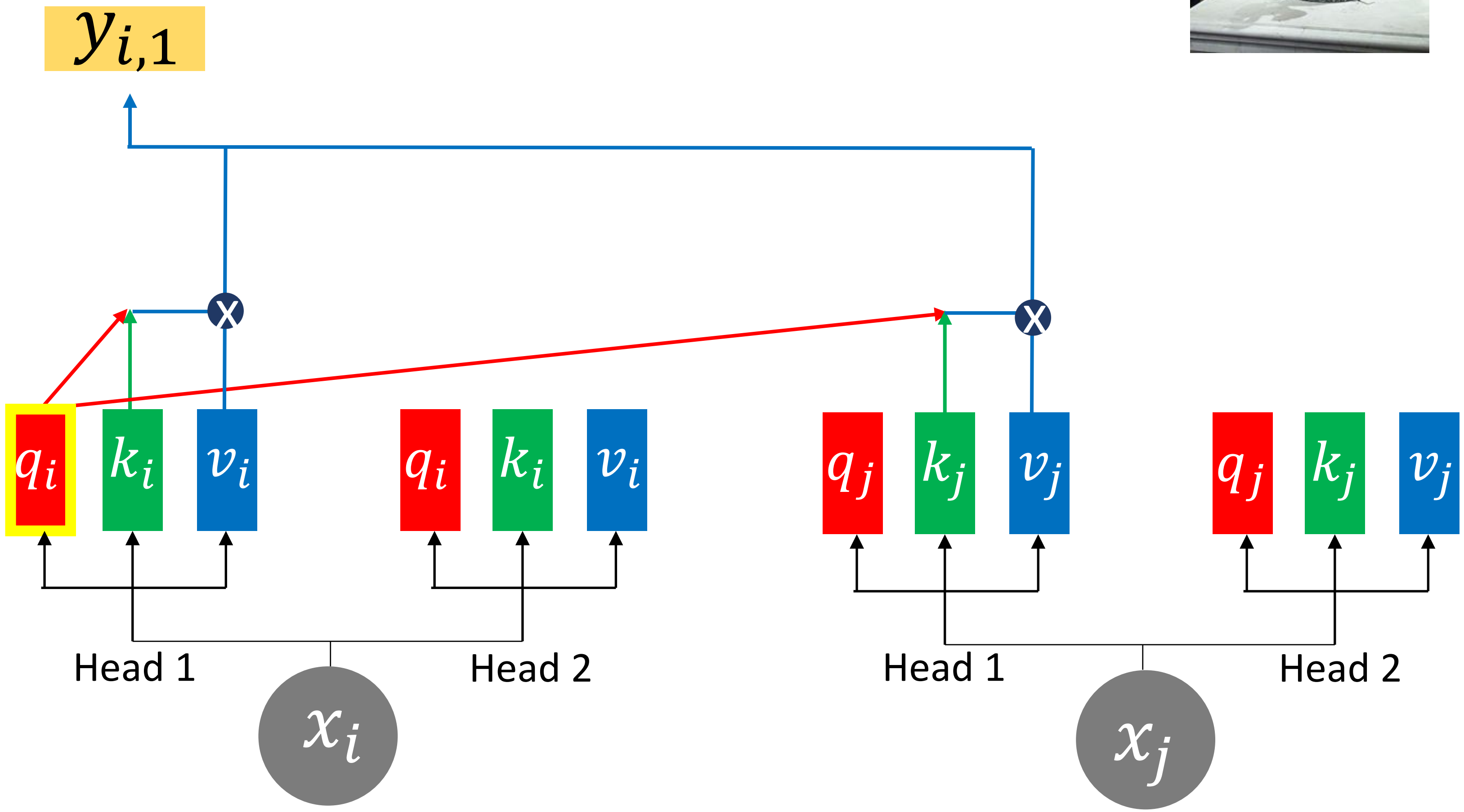
Attention weights
of Head 1



Attention weights
of Head 2

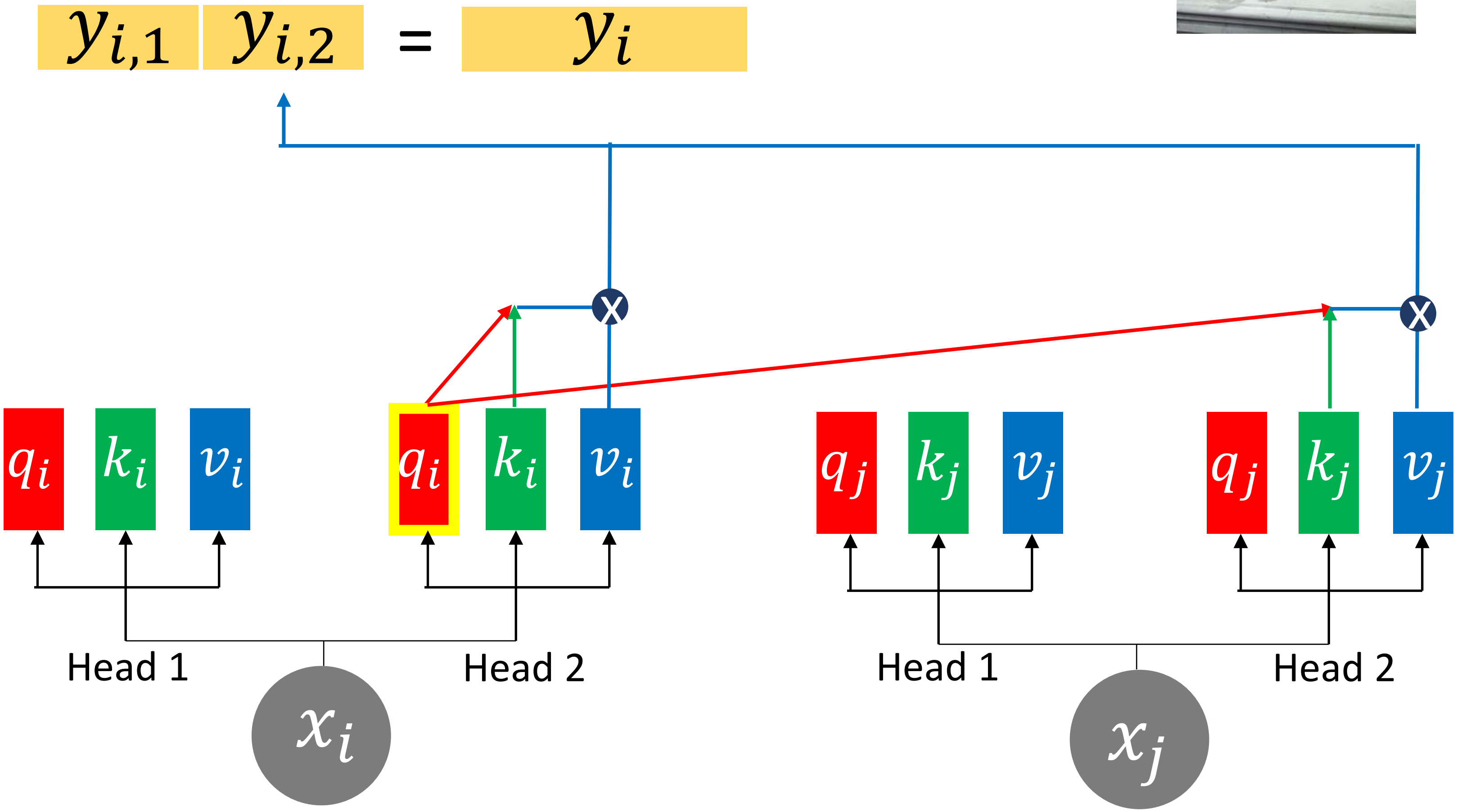
Multi-Head Self-Attention (3/4)

- A two-head example:
  output of two heads are concatenated as the output embedding

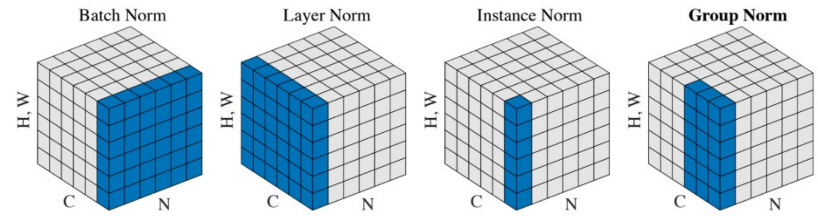


Multi-Head Self-Attention (4/4)

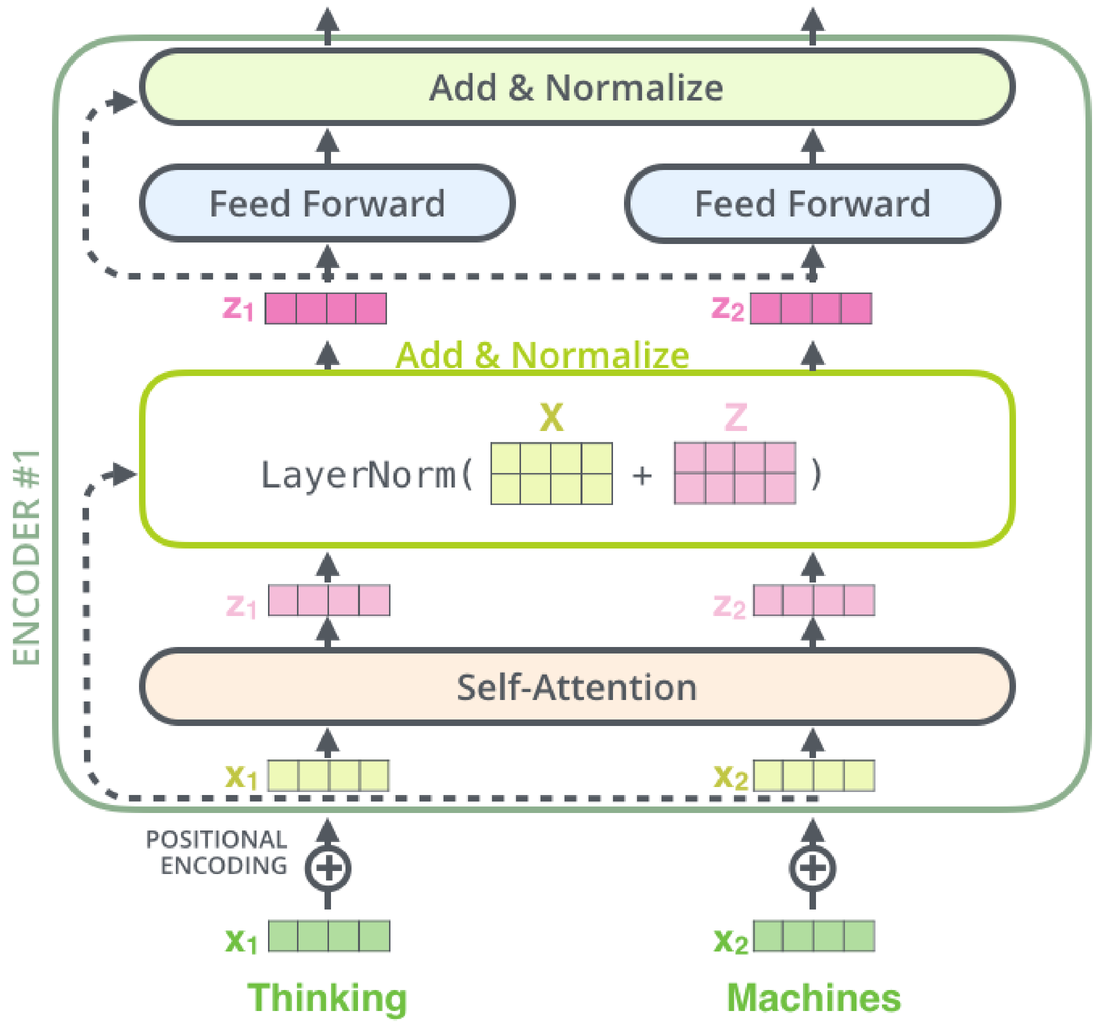
- A two-head example:
  output of two heads are concatenated as the output embedding



The Residuals

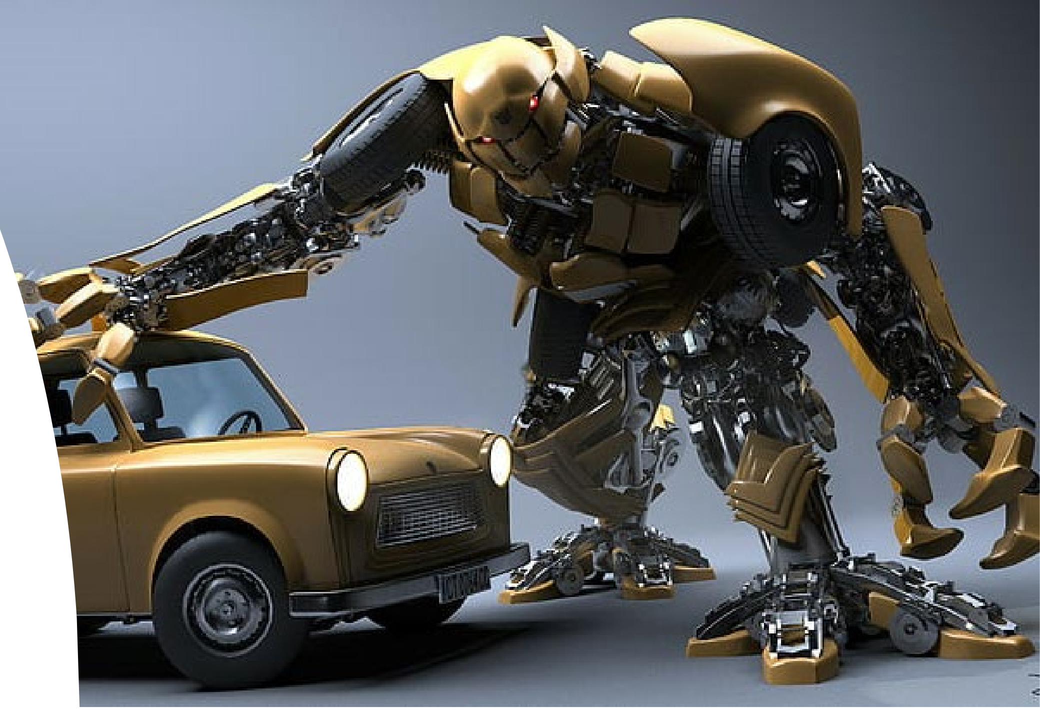


- A **residual connection** followed by layer normalization



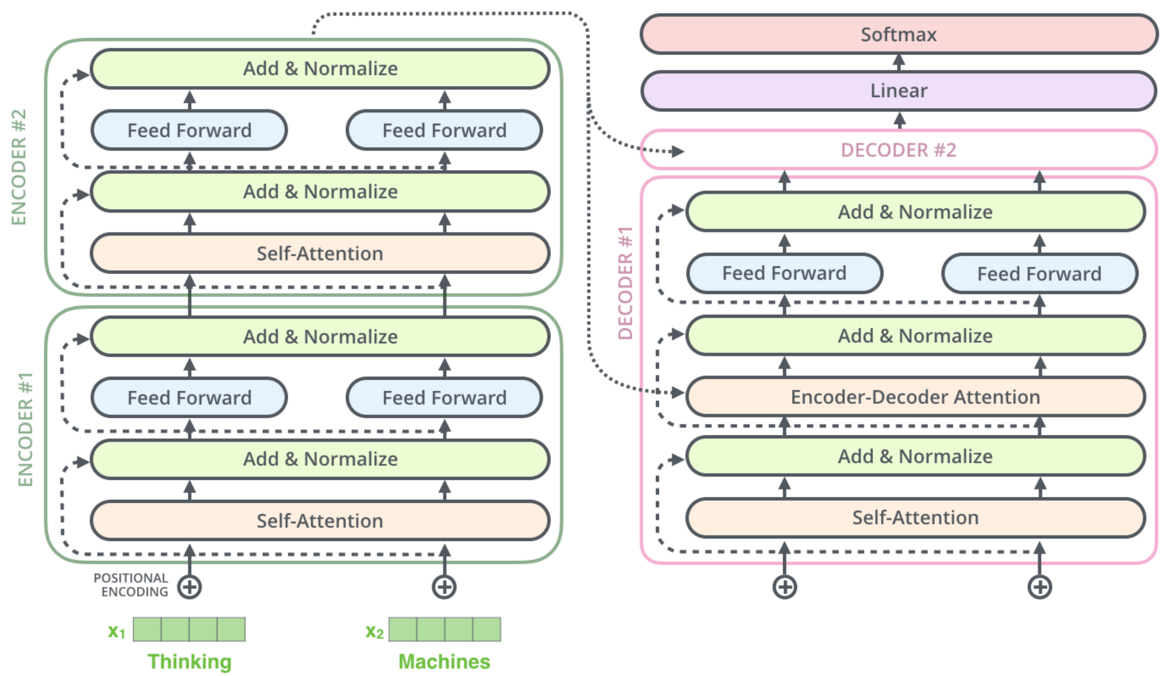
What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (v)
  - Image-text models



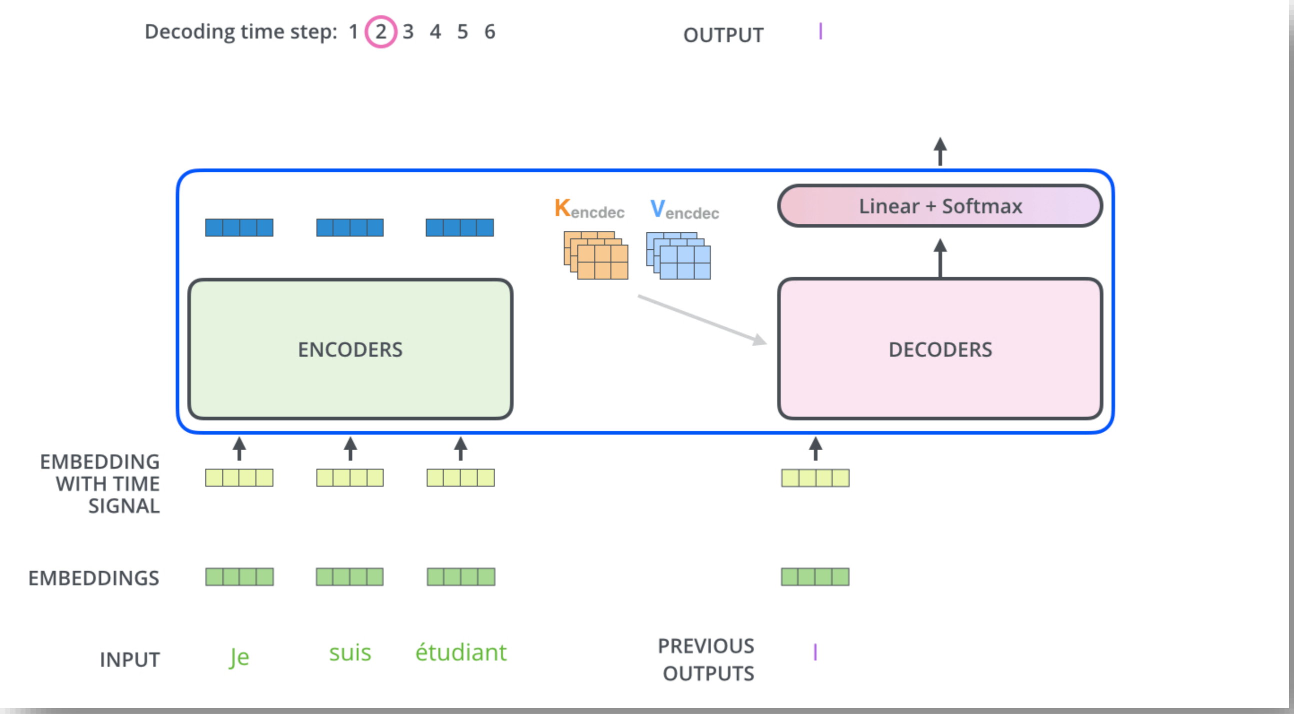
Training The Decoder of Transformer

- Encoder-decoder attention
  - Q from self-attn output in decoder, K & V from encoder outputs
- Masked multi-head attention
  - Design similar to that of encoder, except for decoder #1 which takes additional inputs (of GT/predicted word embeddings).
  - Mask unpredicted tokens during softmax: what does this mean & why?



Training The Decoder of Transformer (cont'd)

- Encoder-decoder attention
  - Q from self-attn in decoder, K & V from encoder outputs
- Masked multi-head attention
  - Design similar to that of encoder, except for decoder #1 which takes additional inputs (of GT/predicted word embeddings).
  - Mask unpredicted tokens during softmax: what does this mean & why?



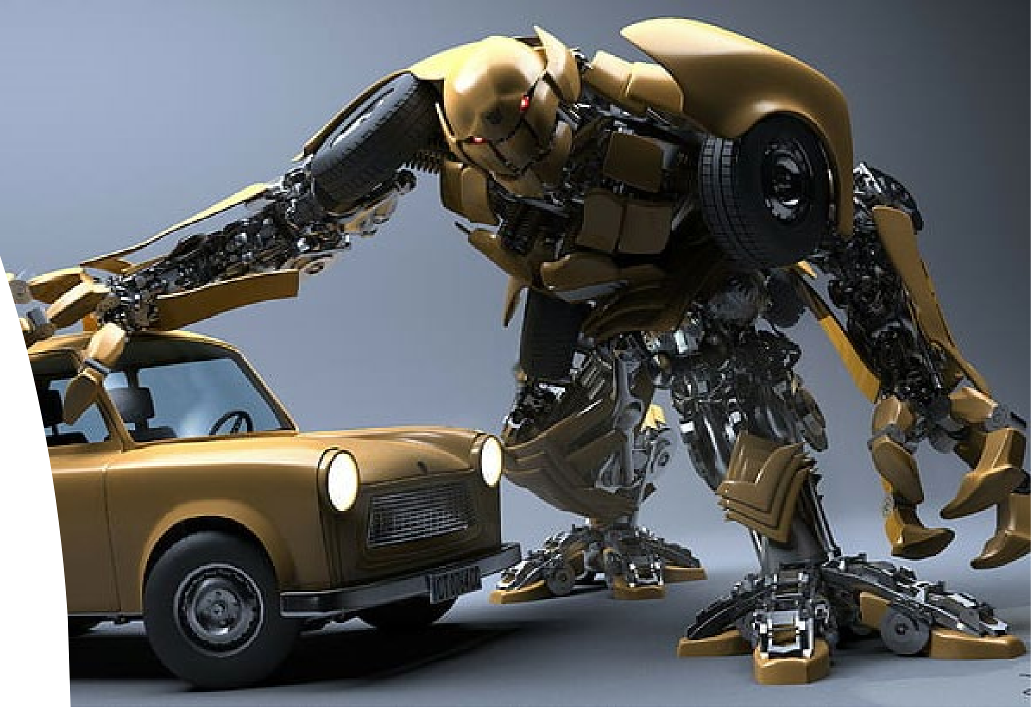
Overview of Encoding & Decoding in Transformer

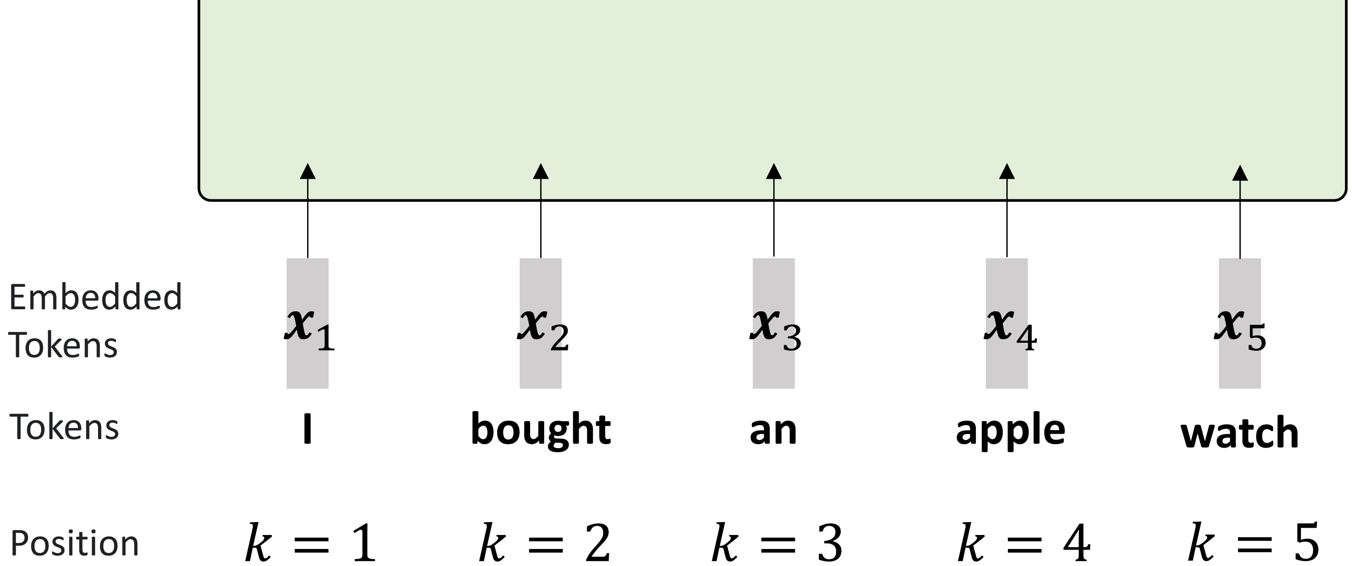
- Encoder/Decoder Self & Cross-Attention

https://medium.com/ml-note/autoencoder-%E4%B8%89-self-attention-transformer-c37f719d222

What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT
  - Swin Transformer
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models





Position k

Angular frequency

$w_i = N^{-2i/d}$

$N = 100{,}000$

$$d \begin{bmatrix} \sin(w_0 k) \\ \cos(w_0 k) \\ \sin(w_1 k) \\ \cos(w_1 k) \\ \vdots \\ \vdots \\ \sin\left(w_{\frac{d}{2}-1} k\right) \\ \cos\left(w_{\frac{d}{2}-1} k\right) \end{bmatrix}$$

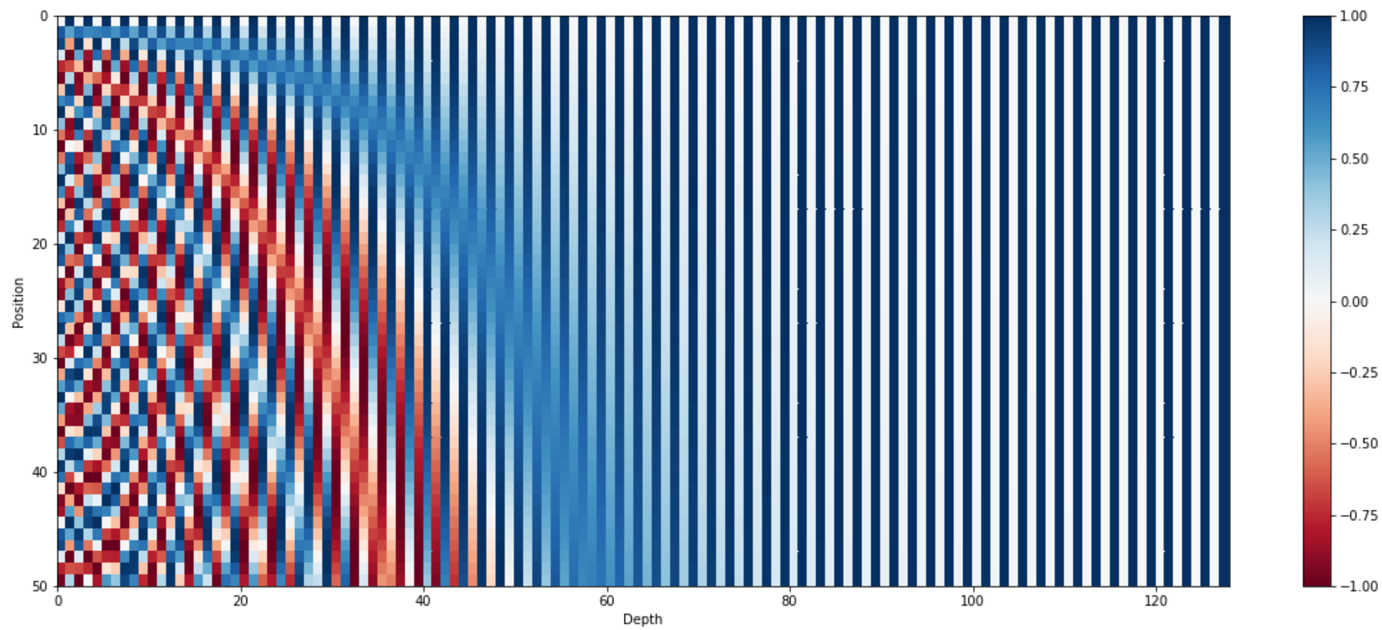
← Fast oscillating

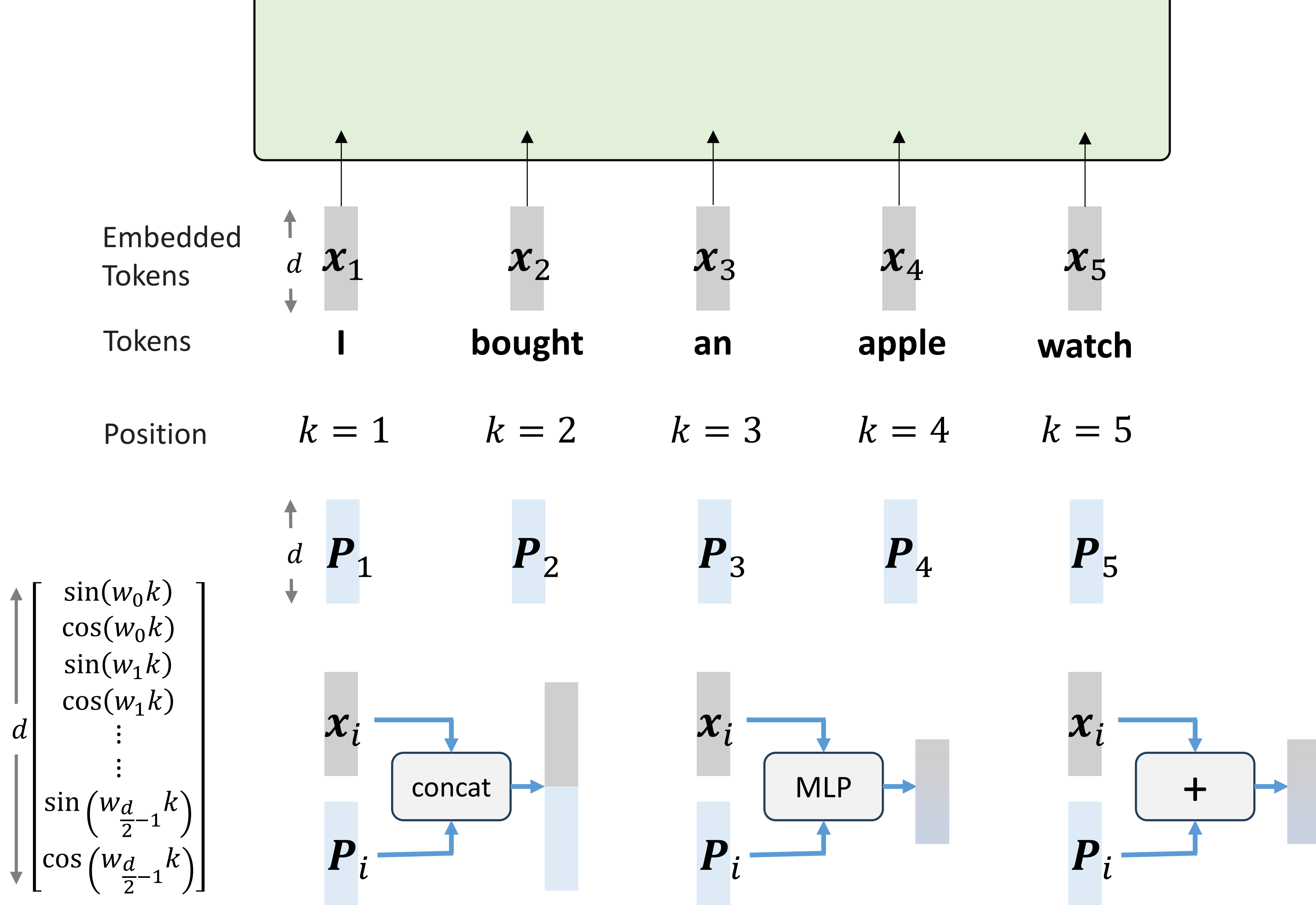
← Slow oscillating

Slide credit: Prof. J.-B. Huang @ U. Maryland

Embedded Tokens	$\boldsymbol{x}_1$	$\boldsymbol{x}_2$	$\boldsymbol{x}_3$	$\boldsymbol{x}_4$	$\boldsymbol{x}_5$
Tokens	**I**	**bought**	**an**	**apple**	**watch**
Position	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$

$$d \begin{bmatrix} \sin(w_0 k) \\ \cos(w_0 k) \\ \sin(w_1 k) \\ \cos(w_1 k) \\ \vdots \\ \vdots \\ \sin\left(w_{\frac{d}{2}-1} k\right) \\ \cos\left(w_{\frac{d}{2}-1} k\right) \end{bmatrix}$$





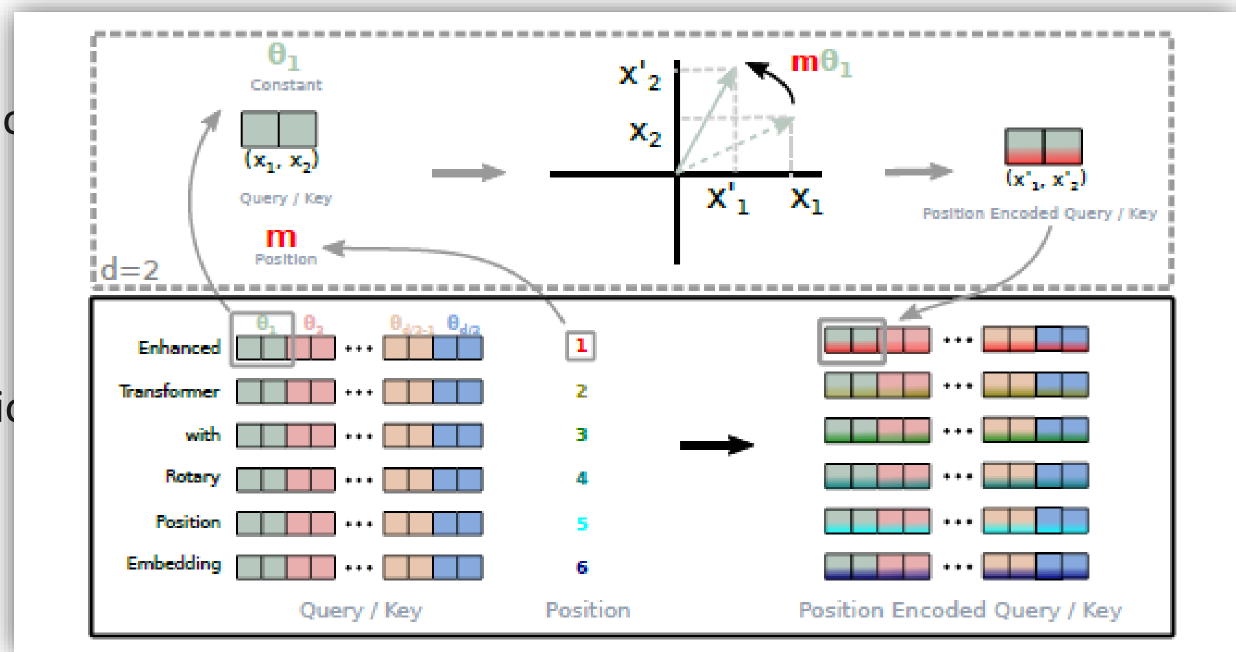
Embedded Tokens
d
$\boldsymbol{x}_1$
$\boldsymbol{x}_2$
$\boldsymbol{x}_3$
$\boldsymbol{x}_4$
$\boldsymbol{x}_5$
Tokens
I
bought
an
apple
watch
Position
$k = 1$
$k = 2$
$k = 3$
$k = 4$
$k = 5$
d
$\boldsymbol{P}_1$
$\boldsymbol{P}_2$
$\boldsymbol{P}_3$
$\boldsymbol{P}_4$
$\boldsymbol{P}_5$
d
$\begin{bmatrix} \sin(w_0 k) \\ \cos(w_0 k) \\ \sin(w_1 k) \\ \cos(w_1 k) \\ \vdots \\ \vdots \\ \sin\left(w_{\frac{d}{2}-1} k\right) \\ \cos\left(w_{\frac{d}{2}-1} k\right) \end{bmatrix}$
$\boldsymbol{x}_i$
$\boldsymbol{P}_i$
concat
MLP
+

Position	1	2	3	4	5	6
	I	**walk**	**my**	**dog**	**every**	**day**

Position	1	2	3	4	5	6
	every	**day**	**I**	**walk**	**my**	**dog**

Slide credit: Prof. J.-B. Huang @ U. Maryland

Positi



Positi

ay

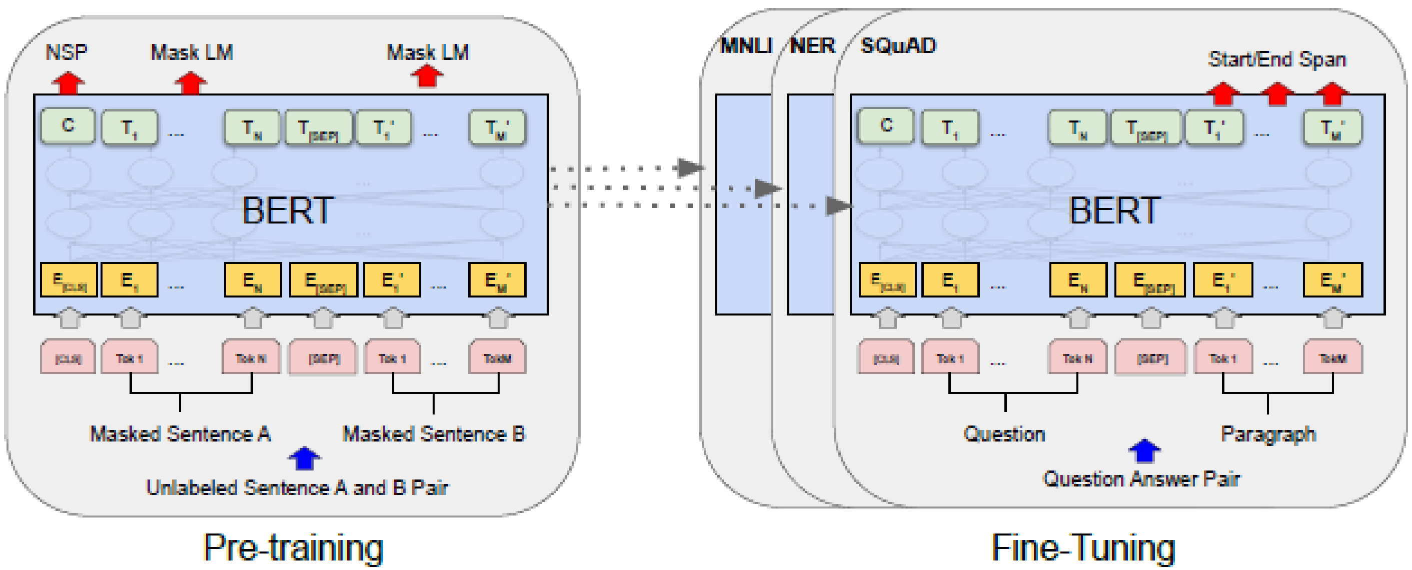
og

From *absolute* to *relative* positional embedding

"RoFormer: Enhanced Transformer with Rotary Position Embedding", arxiv 2021

Extension:
BERT - Bidirectional Encoder Representation from Transformers

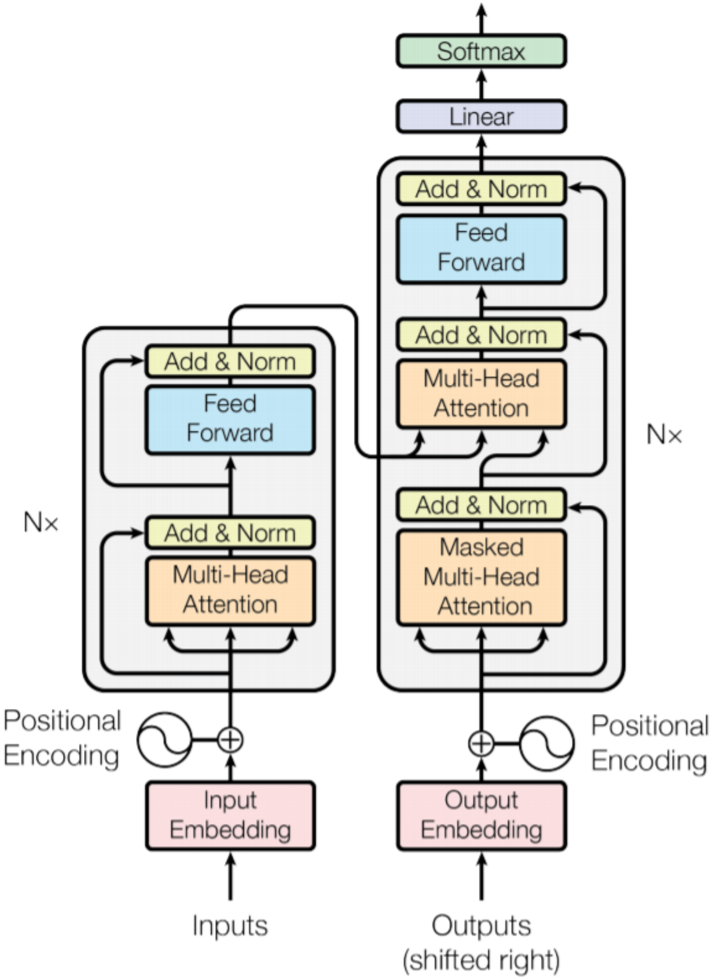
- Proposed by Google AI Language
- Two additional objectives
  - Masked language model (MLM)
  - Next sentence prediction (NSP)



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arxiv. 2018

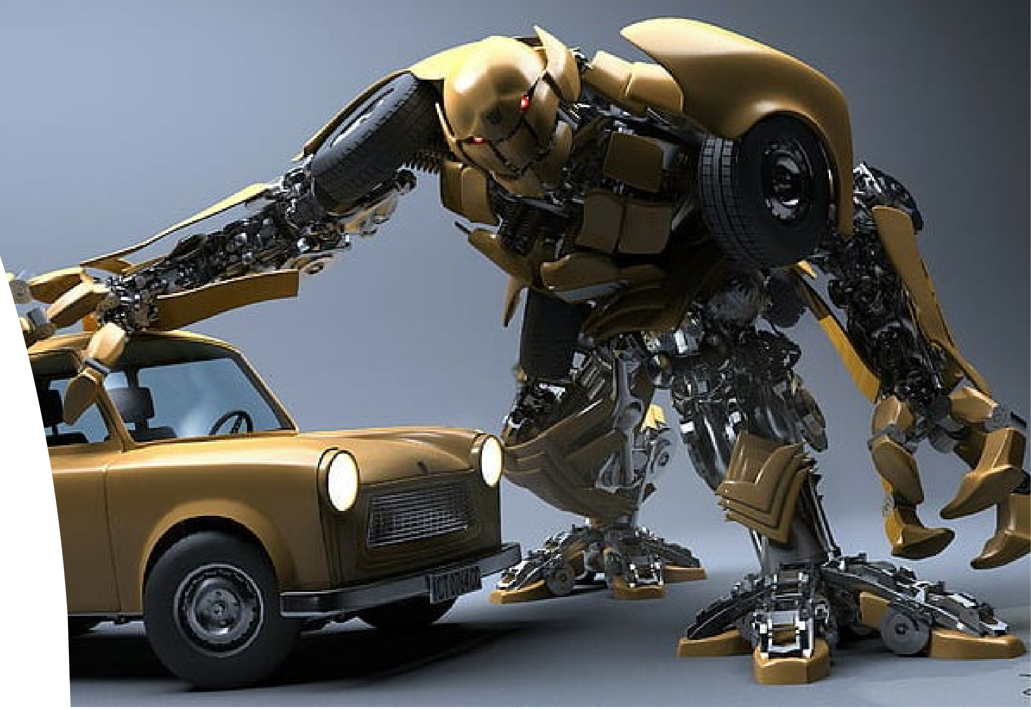
Transformer is promising, but…

- Concerns of Transformer?
    - Computation
    - Space/memory
- Potential solutions
    - Sparse Transformer
    - Linformer
    - Linearized attention, etc.
    - Ever heard of *Mamba*?



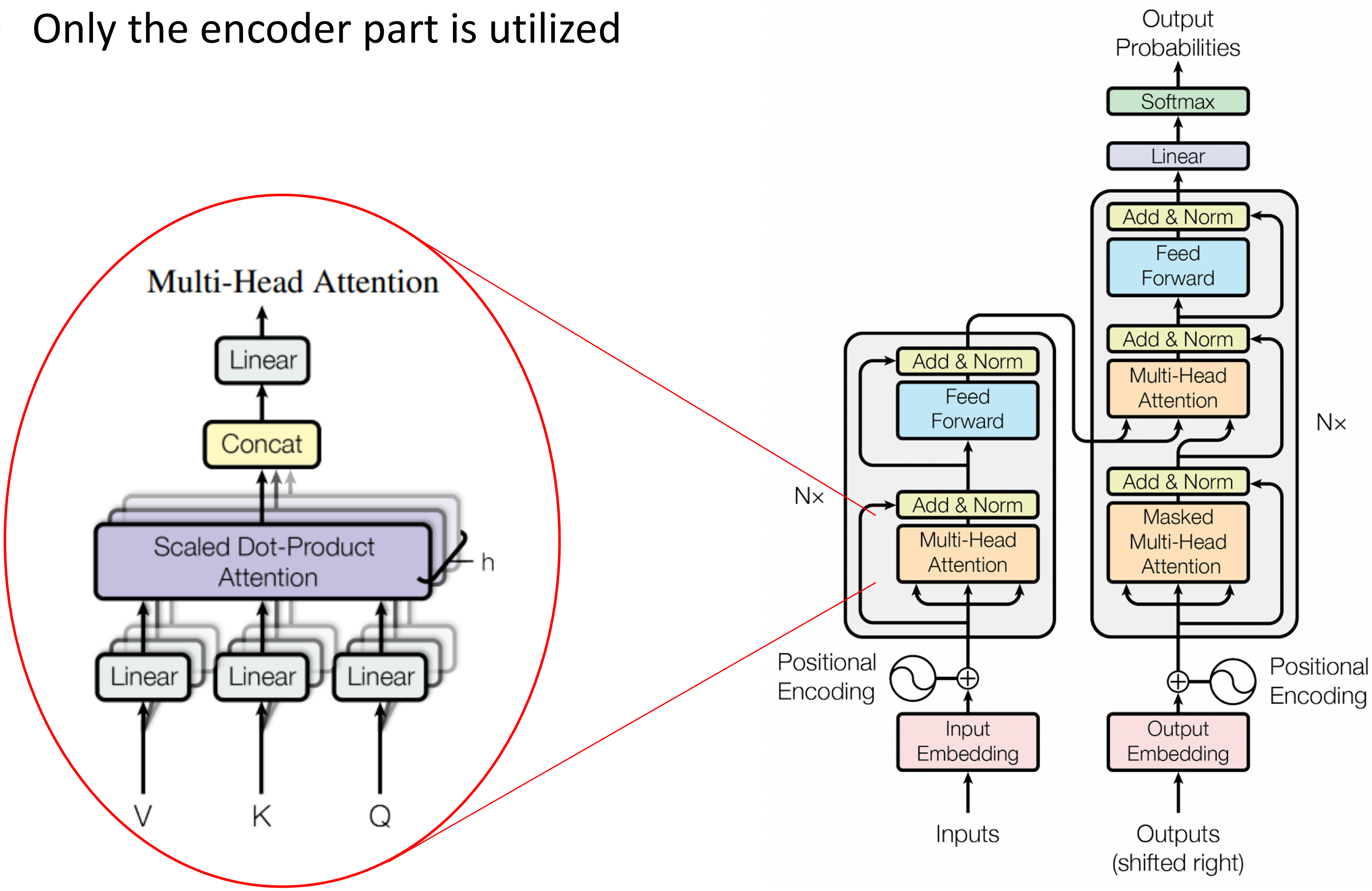
What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



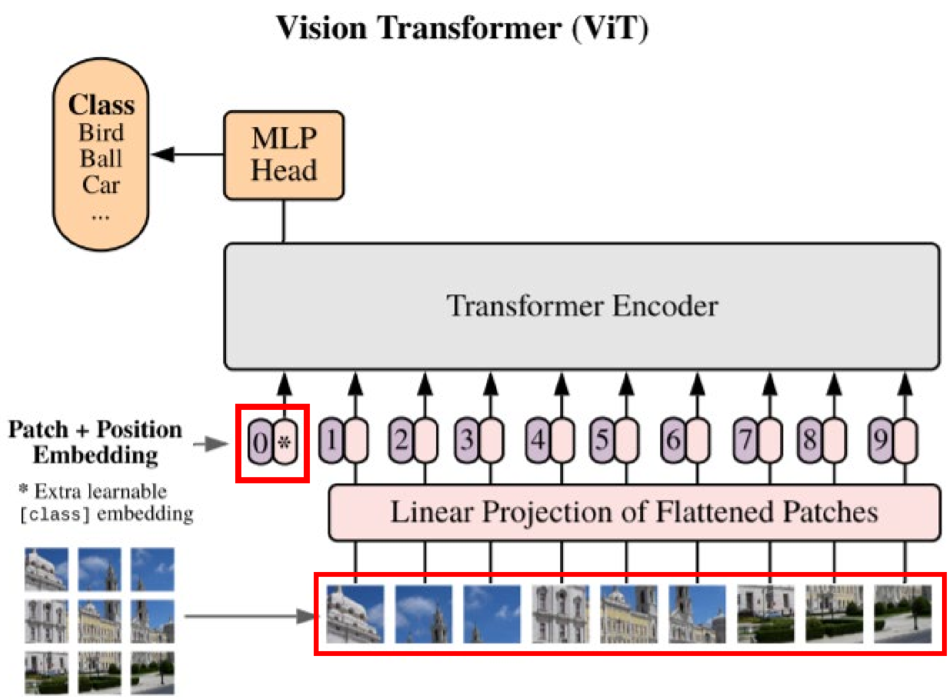
Vision Transformer

- “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”, ICLR, 2021. (Google Research)
- Only the encoder part is utilized



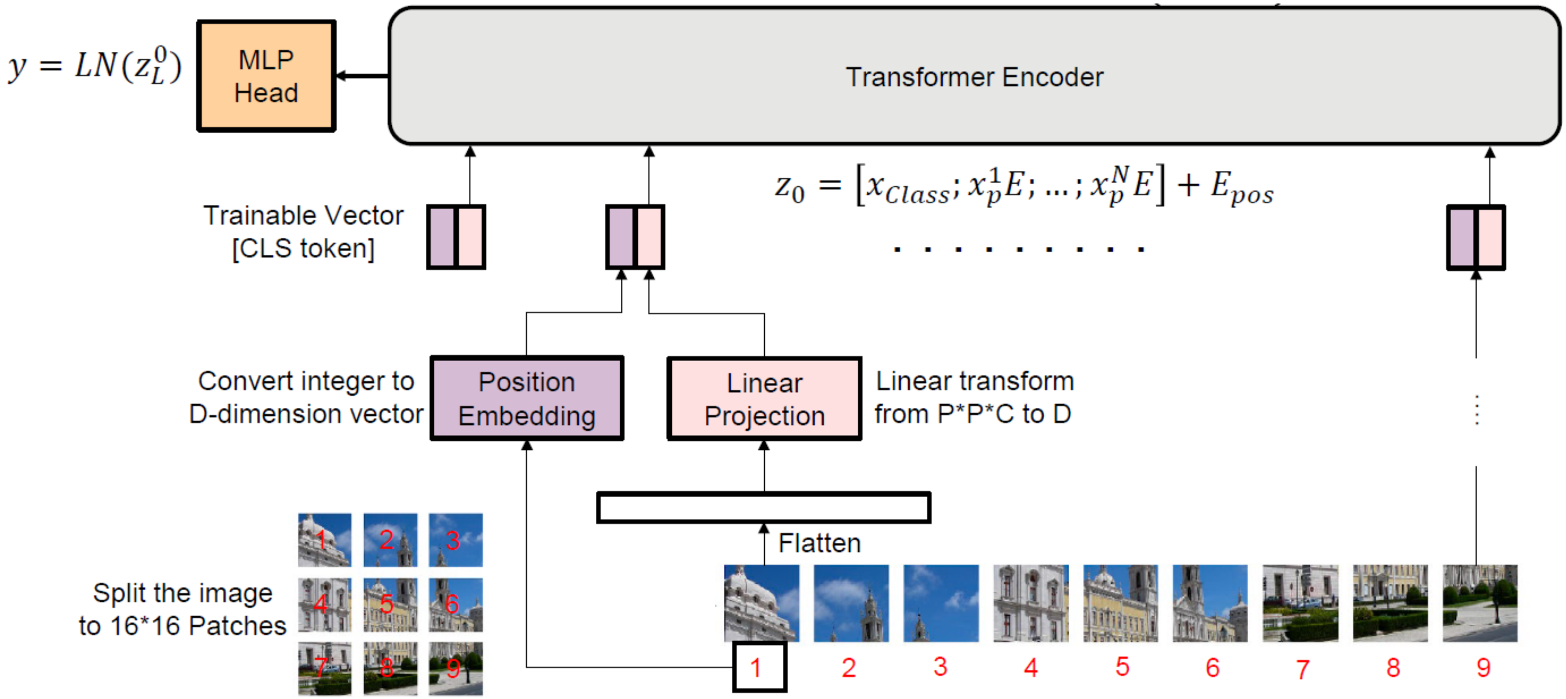
Vision Transformer (cont'd)

- Partition the input image into a **patch sequence**
- An additional **token** (*) is appended to perform attention on patches
- Both the "*" token and positional embeddings (denoted by 0, 1, 2 …) are **trainable vectors**.



Vision Transformer (cont'd)

- "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", ICLR, 2021. (Google Research)



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

$$\mathbf{z'}_\ell = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \qquad \ell = 1 \dots L$$ Multiheaded self-attention (MSA)

$$\mathbf{z}_\ell = \mathrm{MLP}(\mathrm{LN}(\mathbf{z'}_\ell)) + \mathbf{z'}_\ell, \qquad \ell = 1 \dots L$$ Layer norm (LN)

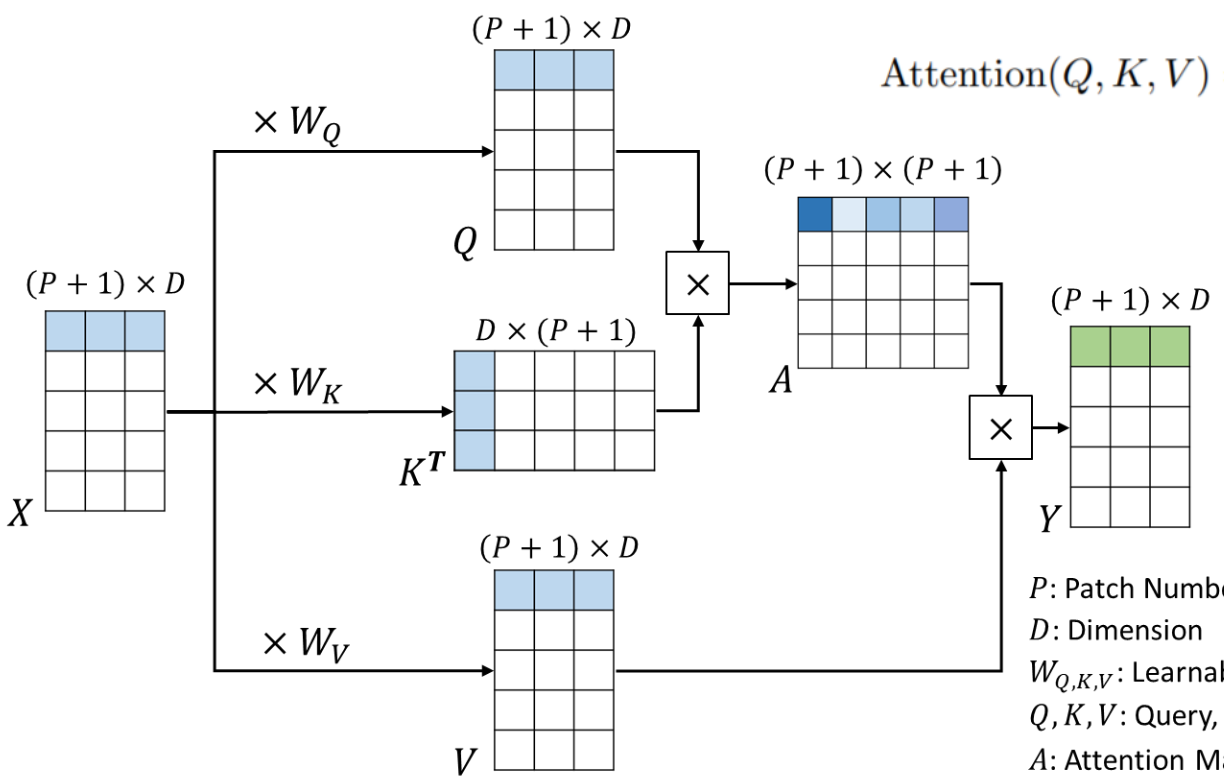
$$\mathbf{y} = \mathrm{LN}(\mathbf{z}_L^0)$$

Figure credit: CS886 Univ. Waterloo

Query-Key-Value Attention in ViT

- E.g., An input image is partitioned into 4 patches, with feature dimension = 3 (i.e., P=4 and D=3).
- Note that there are (P+1) rows since we have an additional token “*”.

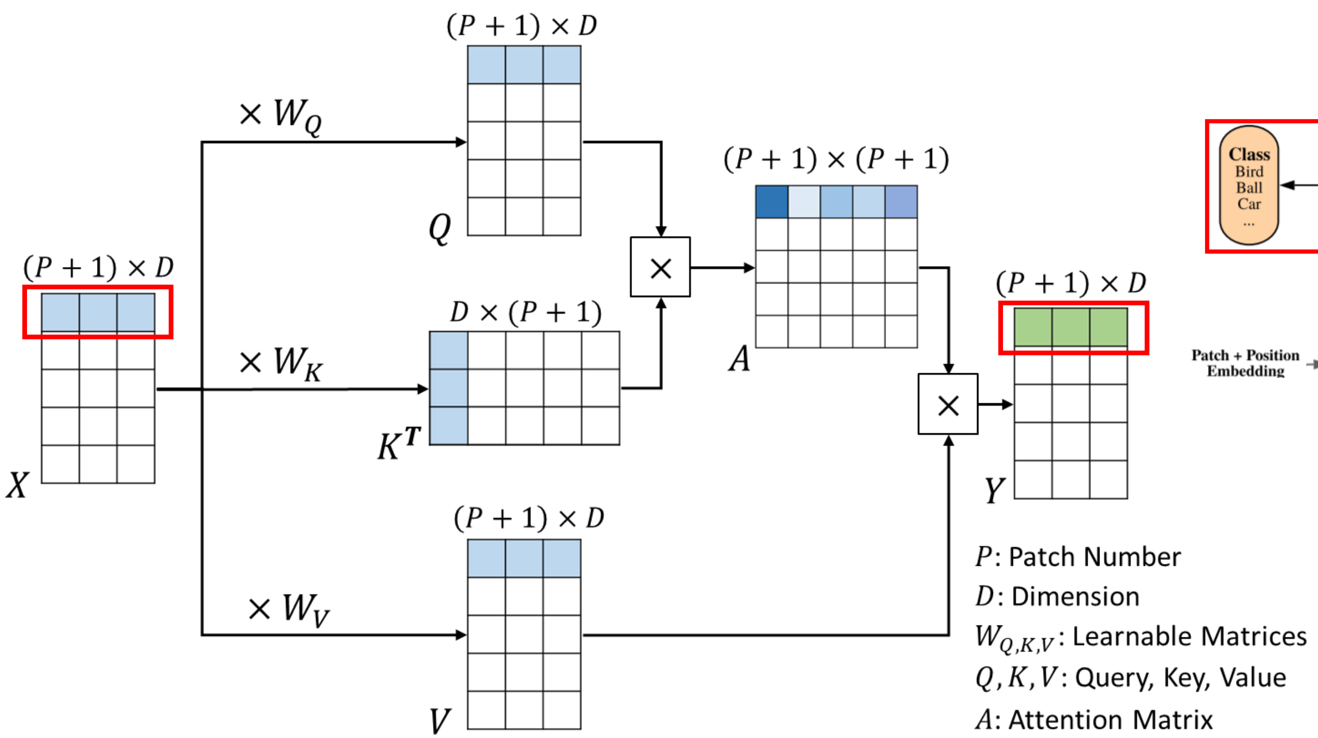
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

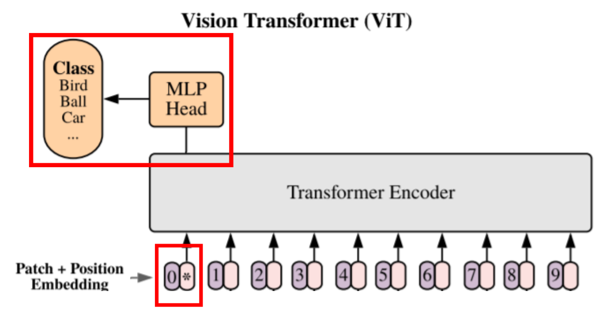


P: Patch Number
D: Dimension
$W_{Q,K,V}$: Learnable Matrices
Q, K, V: Query, Key, Value
A: Attention Matrix

Query-Key-Value Attention in ViT (cont'd)

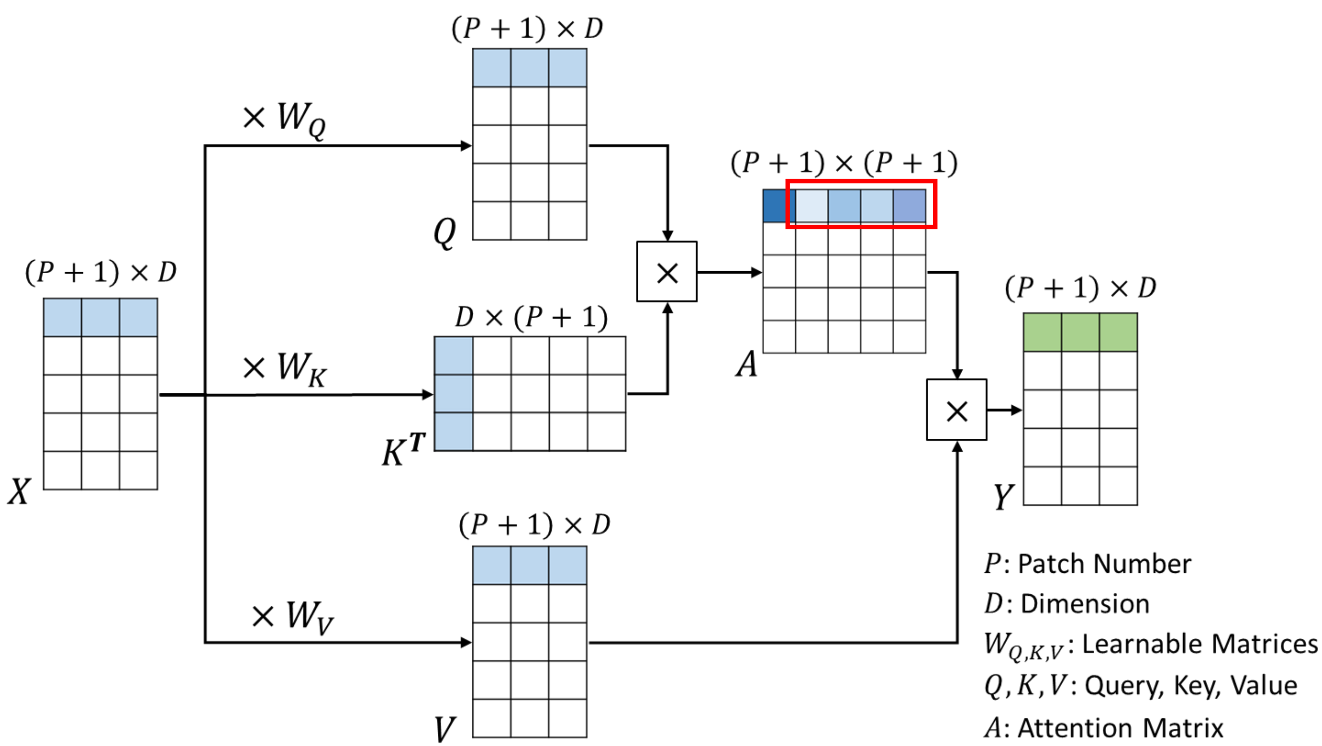
- In the standard vision transformer, we only take the **first output token** of the output sequence (the **first row** of Y) for classification purposes
- This corresponds to the output when **token "0"** serves as query





Visualization of ViT

- To visualize the **attention maps**, we take the attention scores from the **first row** of A (when token "0" serves as query)
- Note the first element is excluded, and thus there are **P scores** corresponding to the P image patches

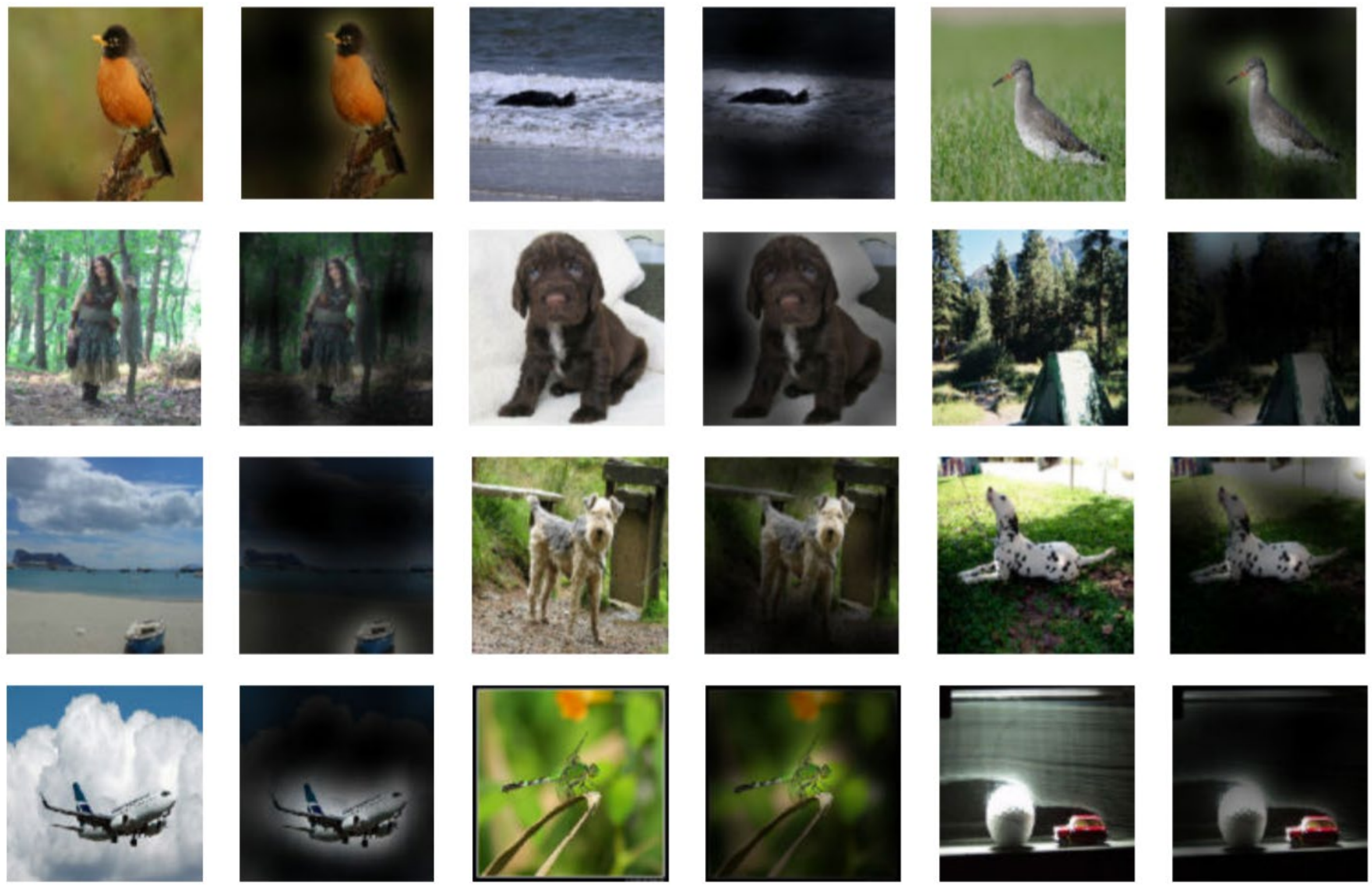


ViT Results

- ViT outperforms CNN-based models
  - Pretrained on JFT & ImageNet
  - Can be trained using TPUv3 w/ 8 cores in 30 days; faster than CNN
  - ViT for visualization/interpretability?

	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** $\pm$ 0.04	87.76 $\pm$ 0.03	85.30 $\pm$ 0.02	87.54 $\pm$ 0.02	88.4/88.5*
ImageNet ReaL	**90.72** $\pm$ 0.05	90.54 $\pm$ 0.03	88.62 $\pm$ 0.05	90.54	90.55
CIFAR-10	**99.50** $\pm$ 0.06	99.42 $\pm$ 0.03	99.15 $\pm$ 0.03	99.37 $\pm$ 0.06	–
CIFAR-100	**94.55** $\pm$ 0.04	93.90 $\pm$ 0.05	93.25 $\pm$ 0.05	93.51 $\pm$ 0.08	–
Oxford-IIIT Pets	**97.56** $\pm$ 0.03	97.32 $\pm$ 0.11	94.67 $\pm$ 0.15	96.62 $\pm$ 0.23	–
Oxford Flowers-102	99.68 $\pm$ 0.02	**99.74** $\pm$ 0.00	99.61 $\pm$ 0.02	99.63 $\pm$ 0.03	–
VTAB (19 tasks)	**77.63** $\pm$ 0.23	76.28 $\pm$ 0.46	72.72 $\pm$ 0.21	76.29 $\pm$ 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Example Visualization for Object Recognition



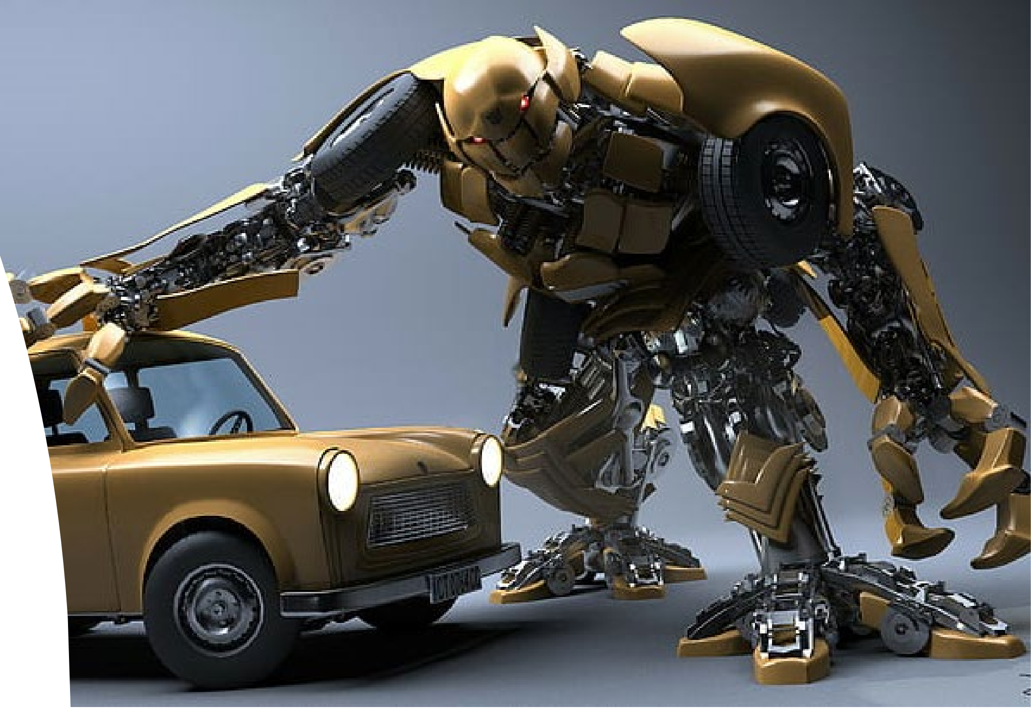
ViT Results

- ViT outperforms CNN-based models
  - Pretrained on JFT & ImageNet
  - Can be trained using TPUv3 w/ 8 cores in 30 days; faster than CNN
  - However, JFT is not publicly available…

	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** $\pm$ 0.04	87.76 $\pm$ 0.03	85.30 $\pm$ 0.02	87.54 $\pm$ 0.02	88.4/88.5*
ImageNet ReaL	**90.72** $\pm$ 0.05	90.54 $\pm$ 0.03	88.62 $\pm$ 0.05	90.54	90.55
CIFAR-10	**99.50** $\pm$ 0.06	99.42 $\pm$ 0.03	99.15 $\pm$ 0.03	99.37 $\pm$ 0.06	–
CIFAR-100	**94.55** $\pm$ 0.04	93.90 $\pm$ 0.05	93.25 $\pm$ 0.05	93.51 $\pm$ 0.08	–
Oxford-IIIT Pets	**97.56** $\pm$ 0.03	97.32 $\pm$ 0.11	94.67 $\pm$ 0.15	96.62 $\pm$ 0.23	–
Oxford Flowers-102	99.68 $\pm$ 0.02	**99.74** $\pm$ 0.00	99.61 $\pm$ 0.02	99.63 $\pm$ 0.03	–
VTAB (19 tasks)	**77.63** $\pm$ 0.23	76.28 $\pm$ 0.46	72.72 $\pm$ 0.21	76.29 $\pm$ 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

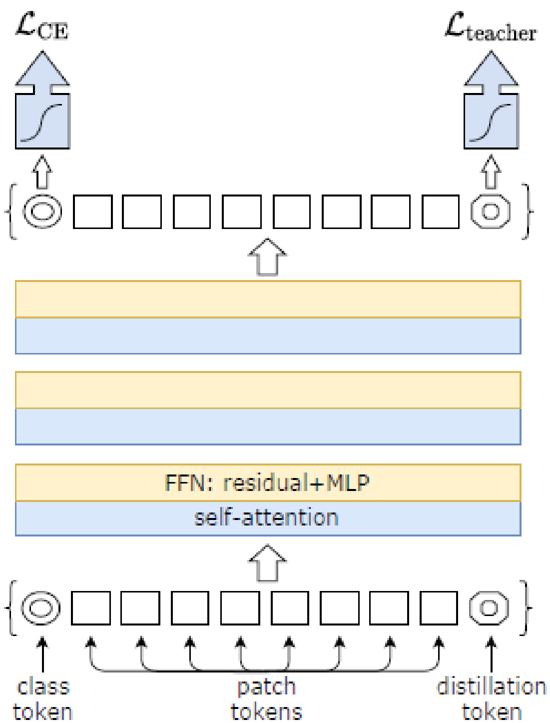
What to Be Covered?

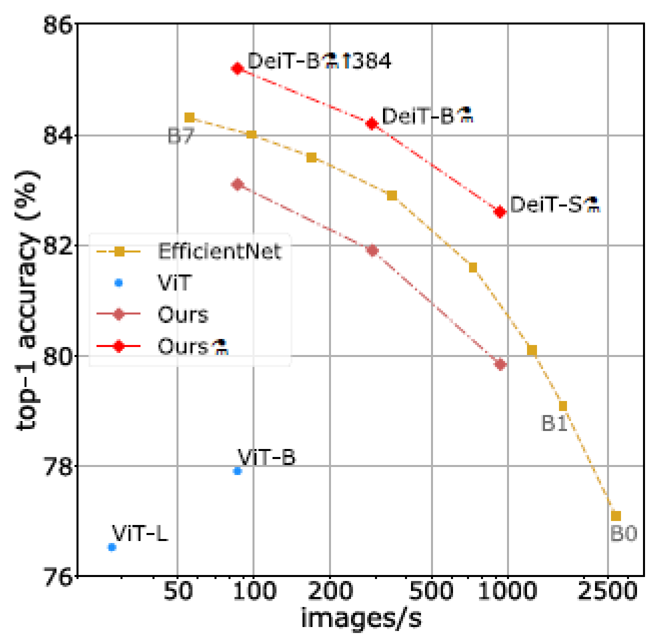
- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



DeiT

- "Training data-efficient image transformers & distillation through attention", ICML 2021. (Facebook AI)
- ViT outperforms CNN, but the dataset is not publicly available.
- By distillation, DeiT only requires ImageNet for pretraining (10 times smaller).

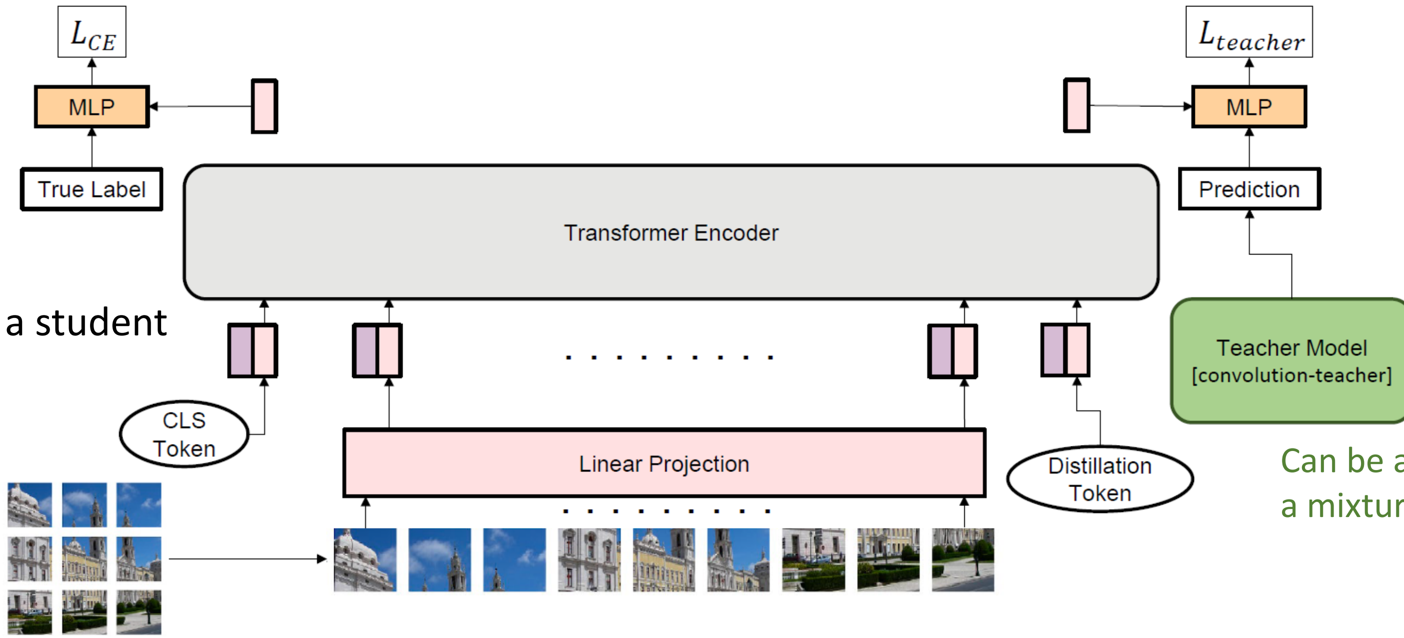




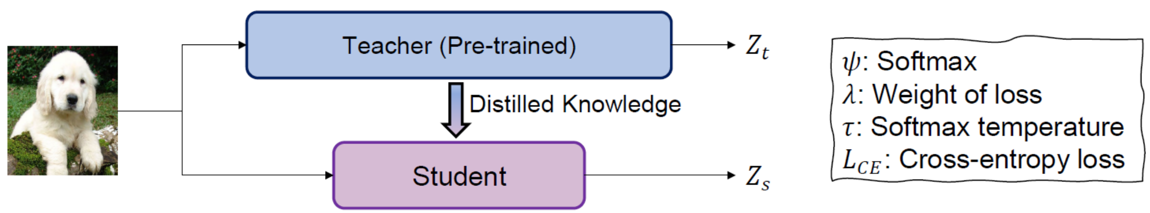
Accuracy vs. throughput on ImageNet (Ours = DeiT)

DeiT: Distill through Attention

viewed as a student



Can be a convnet, or a mixture of classifier



Soft Distillation: $L_t = (1-\lambda)\boxed{L_{CE}(\psi(Z_s), y)} + \lambda\tau^2\boxed{KL(\psi(Z_s/\tau), \psi(Z_t/\tau))}$

Hard Distillation: $L_t = \frac{1}{2}\boxed{L_{CE}(\psi(Z_s), y)} + \frac{1}{2}\boxed{L_{CE}(\psi(Z_s), y_t)}$

Figure credit: CS886 Univ. Waterloo

DeiT Results

- Variants of DeiT architectures (adopted from the ViT backbone)

Model	embedding dimension	#heads	#layers	#params	training resolution	throughput (im/sec)
DeiT-Ti	192	3	12	5M	224	2536
DeiT-S	384	6	12	22M	224	940
DeiT-B	768	12	12	86M	224	292

Same as the original ViT model

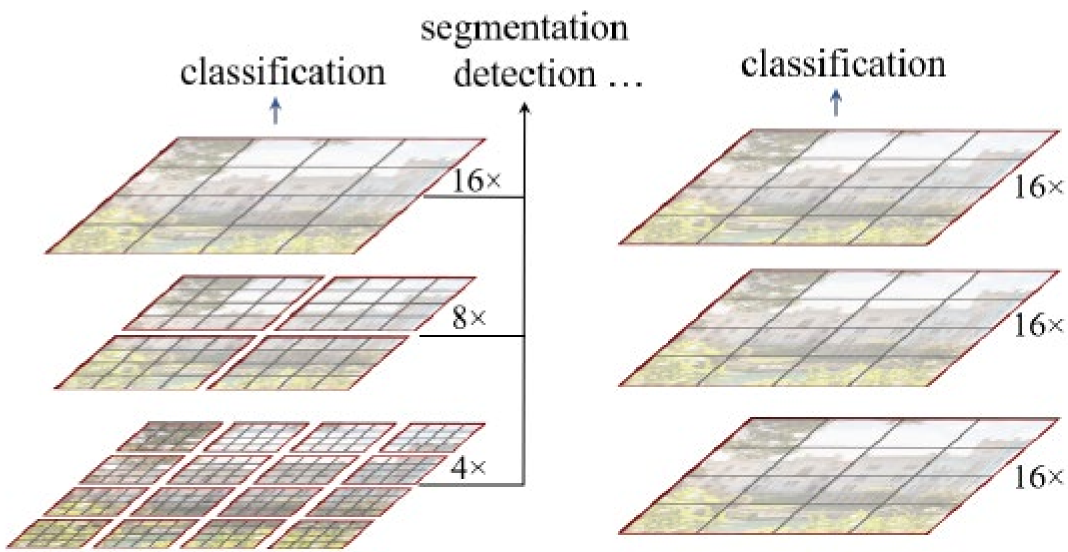
- Choices of different teacher models & ablation studies

Teacher Models	acc.	Student: DeiT-B pretrain	↑384
DeiT-B	81.8	81.9	83.1
RegNetY-4GF	80.0	82.7	83.6
RegNetY-8GF	81.7	82.7	83.8
RegNetY-12GF	82.4	83.0	83.9
RegNetY-16GF	82.9	83.0	84.0

DeiT: method ↓	supervision: label	supervision: teacher	ImageNet top-1 (%): Ti 224	S 224	B 224	B↑384
no distillation	✓	✗	72.2	79.8	81.8	83.1
usual distillation	✗	soft	72.2	79.8	81.8	83.2
hard distillation	✗	hard	74.3	80.9	83.0	84.0
class embedding	✓	hard	73.9	80.9	83.0	84.2
distil. embedding	✓	hard	74.6	81.1	83.1	84.4
DeiT⚗: class+distil.	✓	hard	74.5	81.2	83.4	84.5

Swin Transformer

- “Swin Transformer: Hierarchical Vision Transformer Shifted Windows”, ICCV 2021. (MSRA)
- ViT’s computation complexity vs. image size ->
- Propose to perform patch merging & Swin Transformer architecture



(a) Swin Transformer (ours)

(b) ViT

Swin Architecture (1/6)

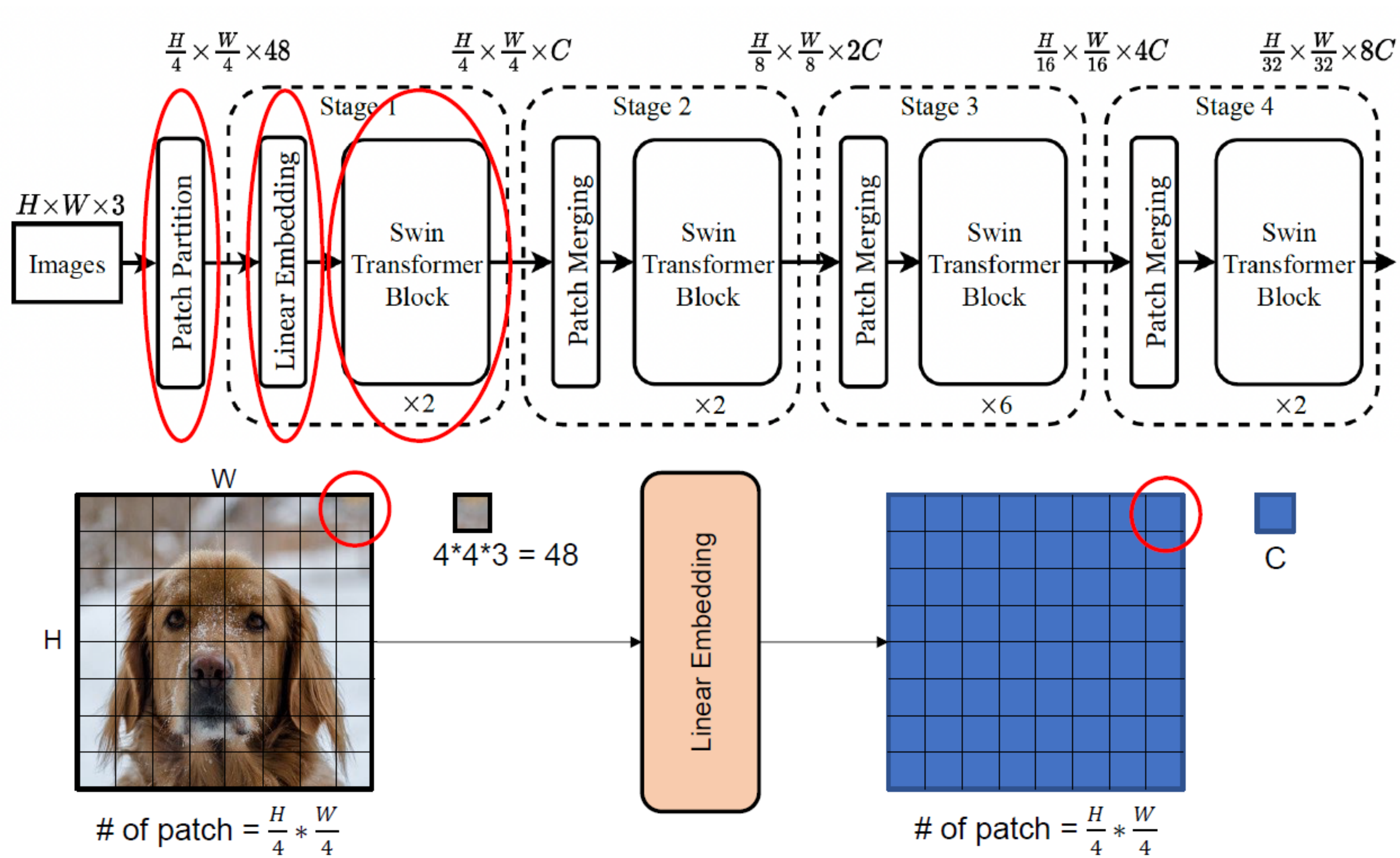


Figure credit: CS886 Univ. Waterloo

Swin Architecture (2/6)

Identical to Transformer but replaced the standard multi-head self-attention (MSA) with:

- ***Window MSA (W-MSA)***
- ***Shifted Window MSA (SW-MSA)***

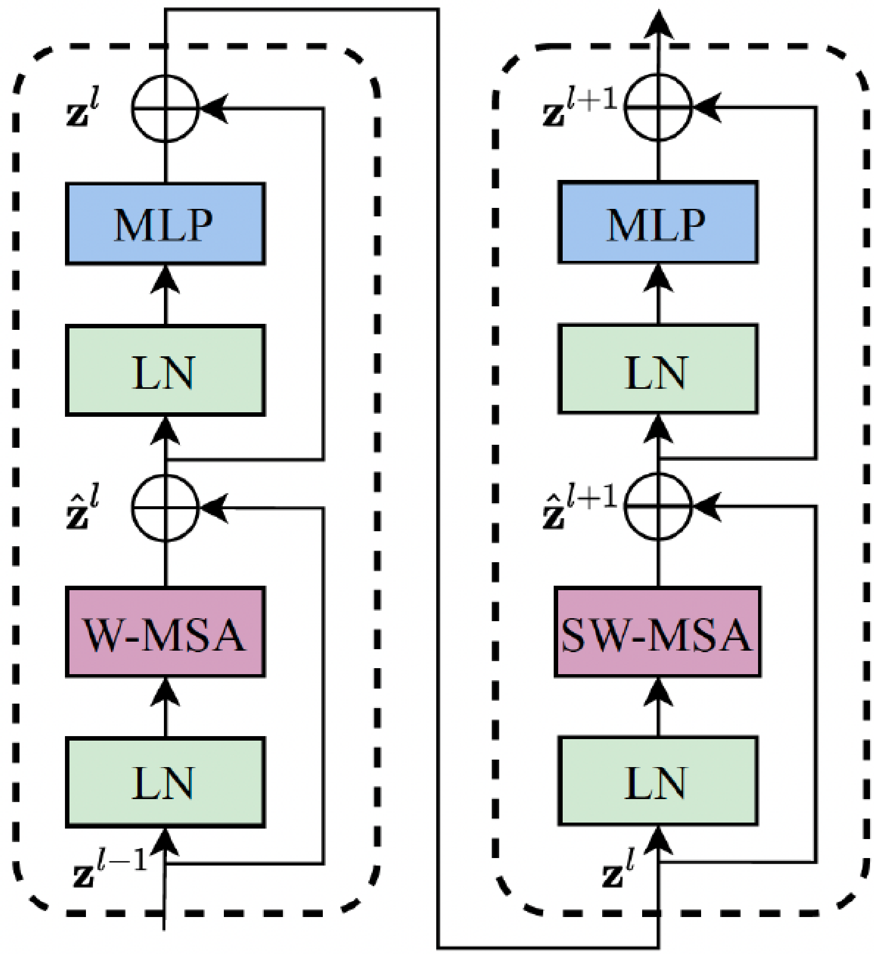


Figure credit: CS886 Univ. Waterloo

Swin Architecture (3/6)

- Window MSA (W-MSA)
  - Compute attention only within each window
  - Linear complexity (wrt the # of patches) due to the fixed window size
  - What about attention across different windows?

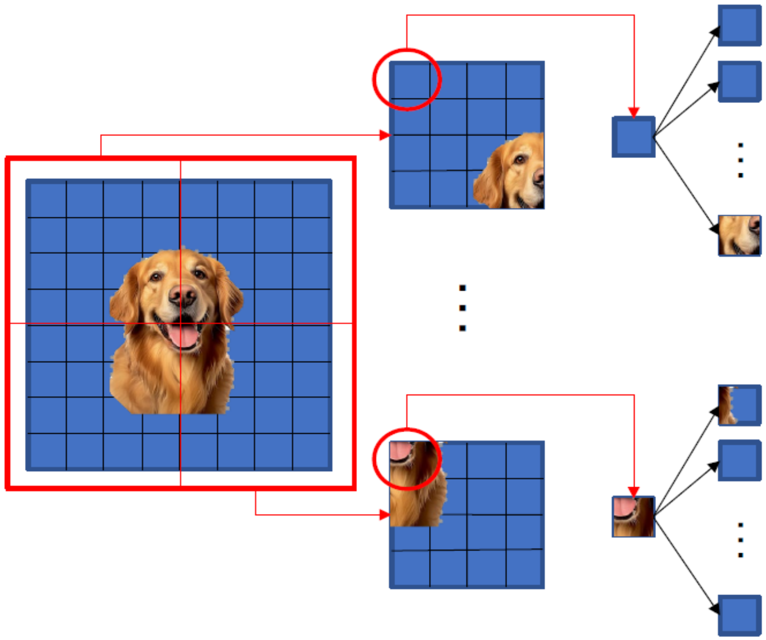


Figure credit: CS886 Univ. Waterloo

Swin Architecture (4/6)

- Shifted Window MSA (SW-MSA)
  - How to perform across different windows?
  - Shift the window by half the window size (M/2)
  - Additional problem?
    9 instead of 4 windows, plus padding?

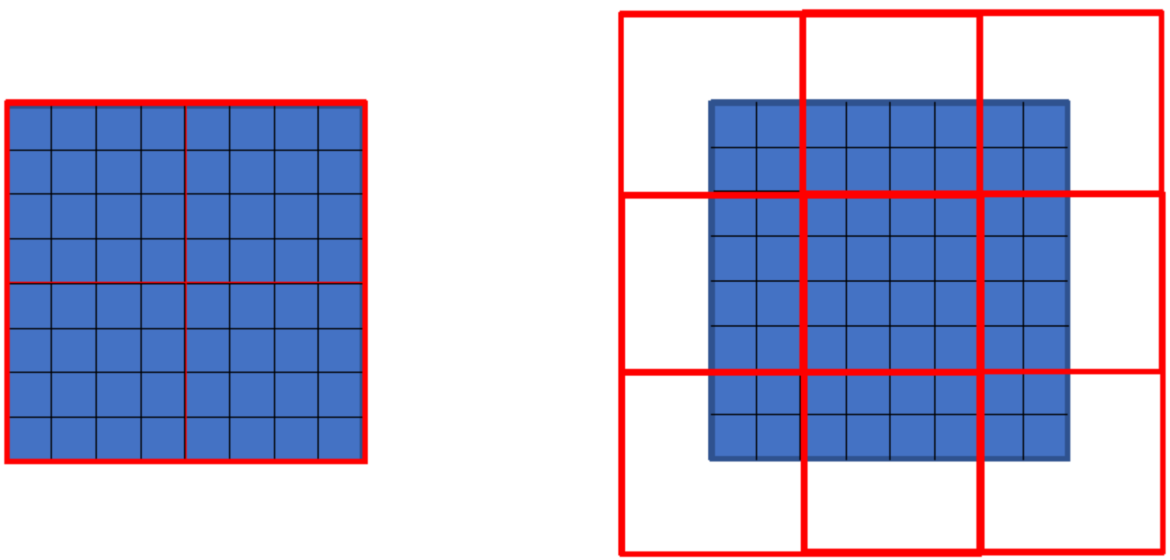


Figure credit: CS886 Univ. Waterloo

Swin Architecture (5/6)

- Shifted Window MSA (SW-MSA)
  - How to perform across different windows?
  - Shift the window by half the window size (M/2)
  - Additional problem?
    9 instead of 4 windows, plus padding?
  - Introduce *cycle shift*

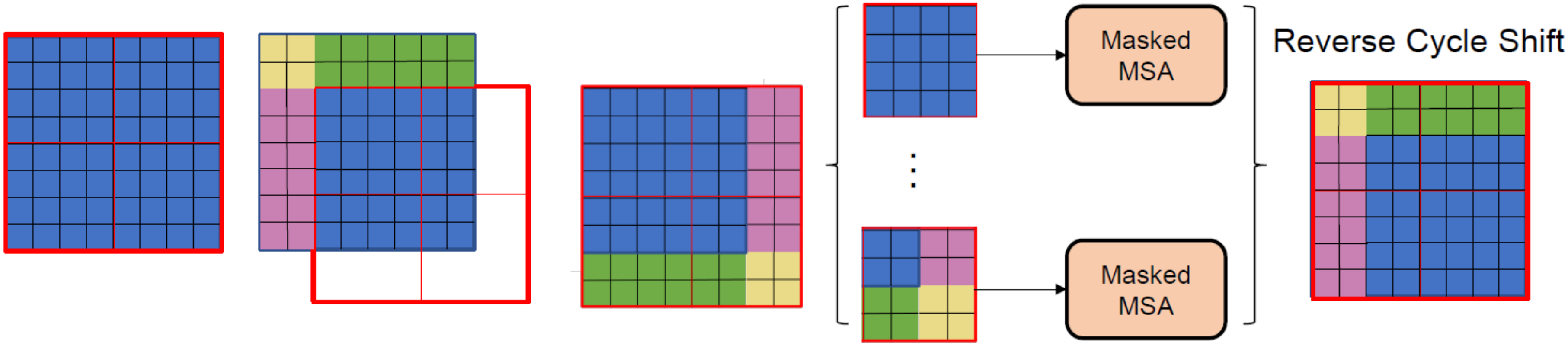


Figure credit: CS886 Univ. Waterloo

Swin Architecture (6/6)

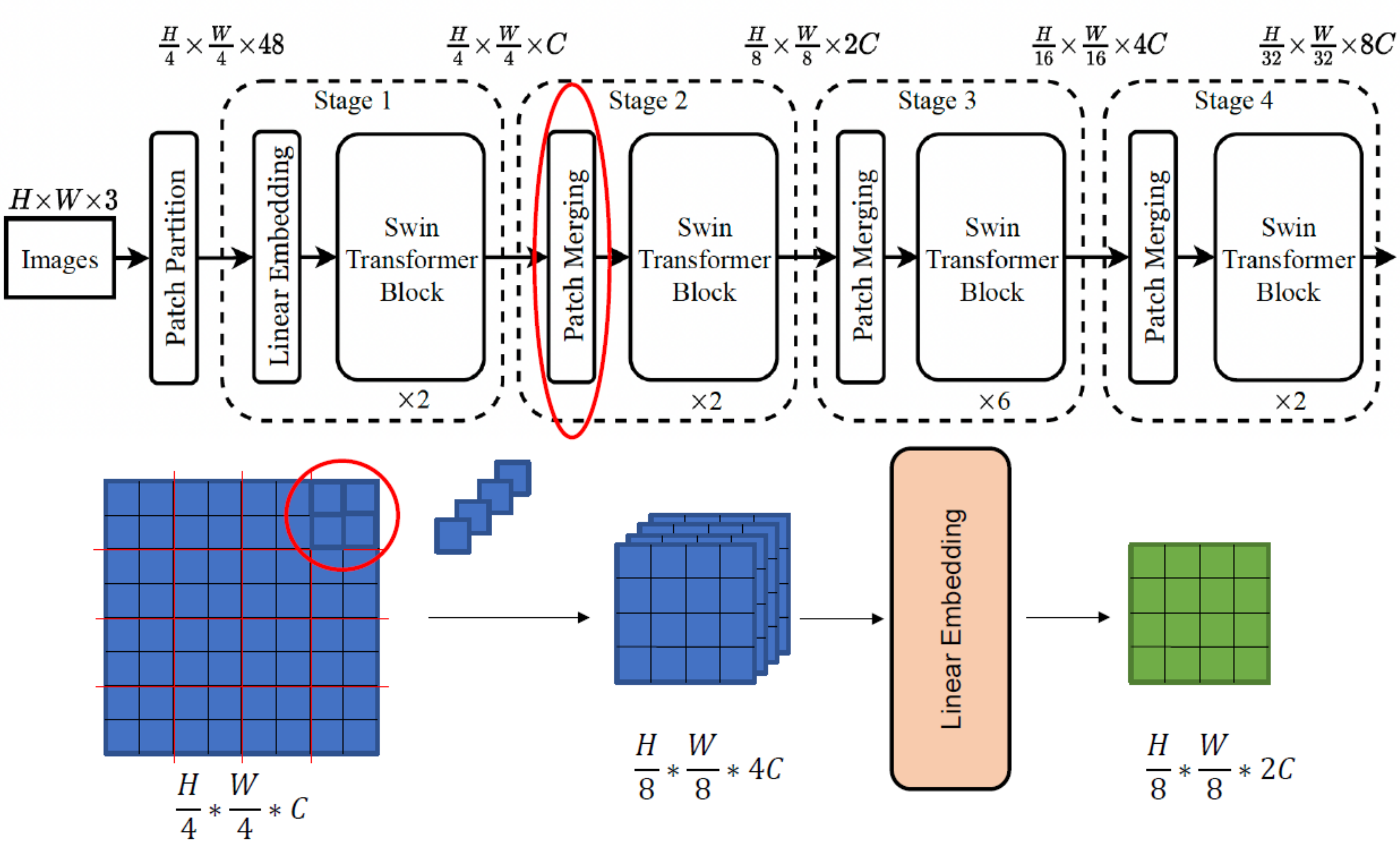
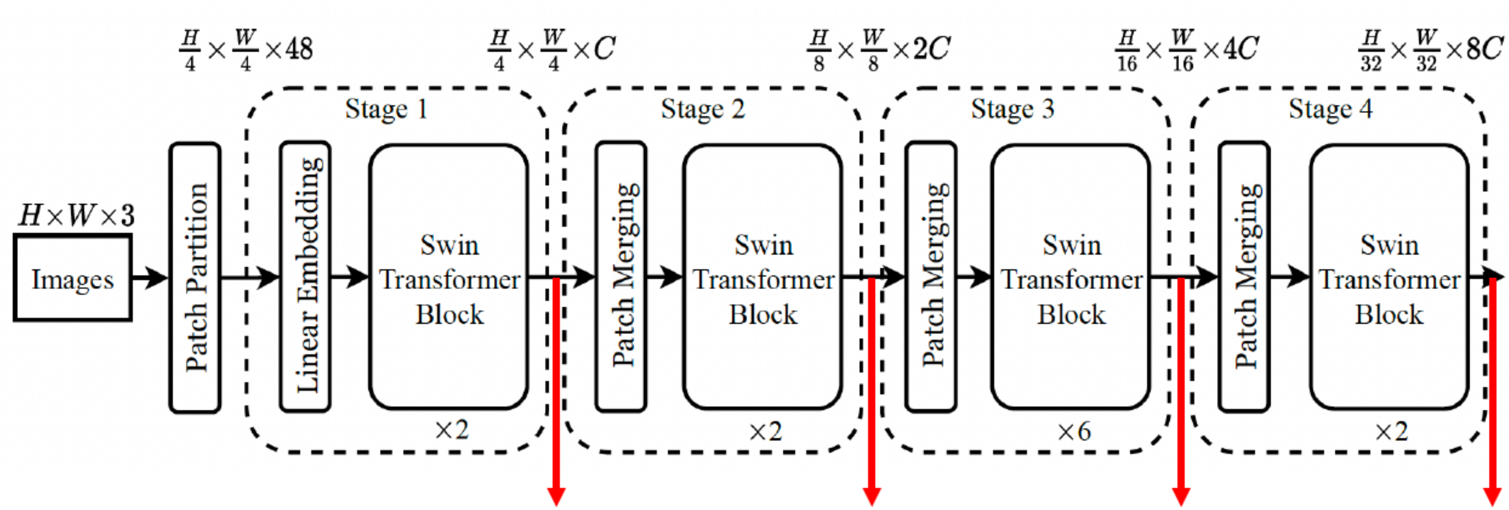


Figure credit: CS886 Univ. Waterloo

Swin Transformer Output



- **Image classification:** Use the last output
- **Object detection and Image segmentation:** Use the output of all the stages

Figure credit: CS886 Univ. Waterloo

Swin Transformer Results

- Image classification

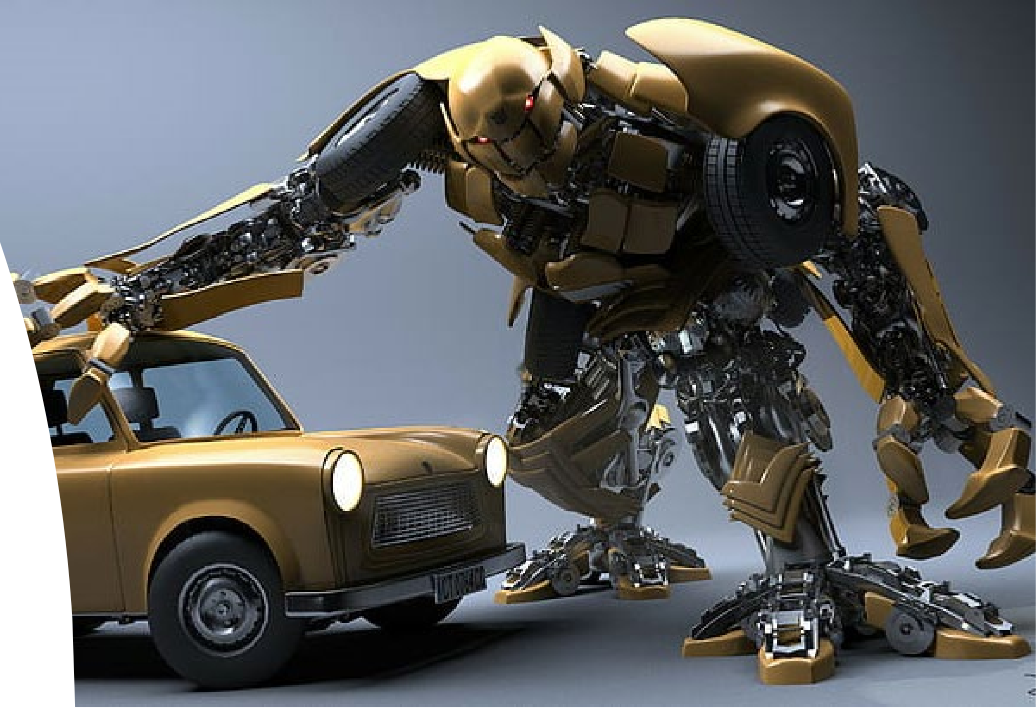
(a) Regular ImageNet-1K trained models					
method	image size	#param.	FLOPs	throughput (image / s)	ImageNet top-1 acc.
RegNetY-4G [48]	224^2	21M	4.0G	1156.7	80.0
RegNetY-8G [48]	224^2	39M	8.0G	591.6	81.7
RegNetY-16G [48]	224^2	84M	16.0G	334.7	82.9
EffNet-B3 [58]	300^2	12M	1.8G	732.1	81.6
EffNet-B4 [58]	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

Figure credit: CS886 Univ. Waterloo

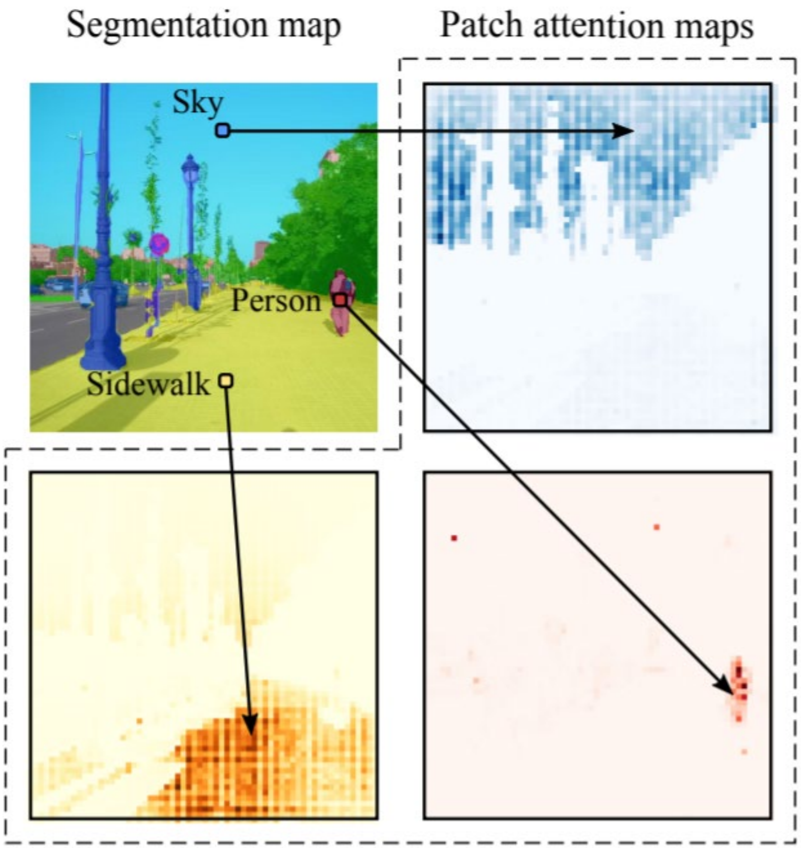
What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



Transformer for Semantic Segmentation

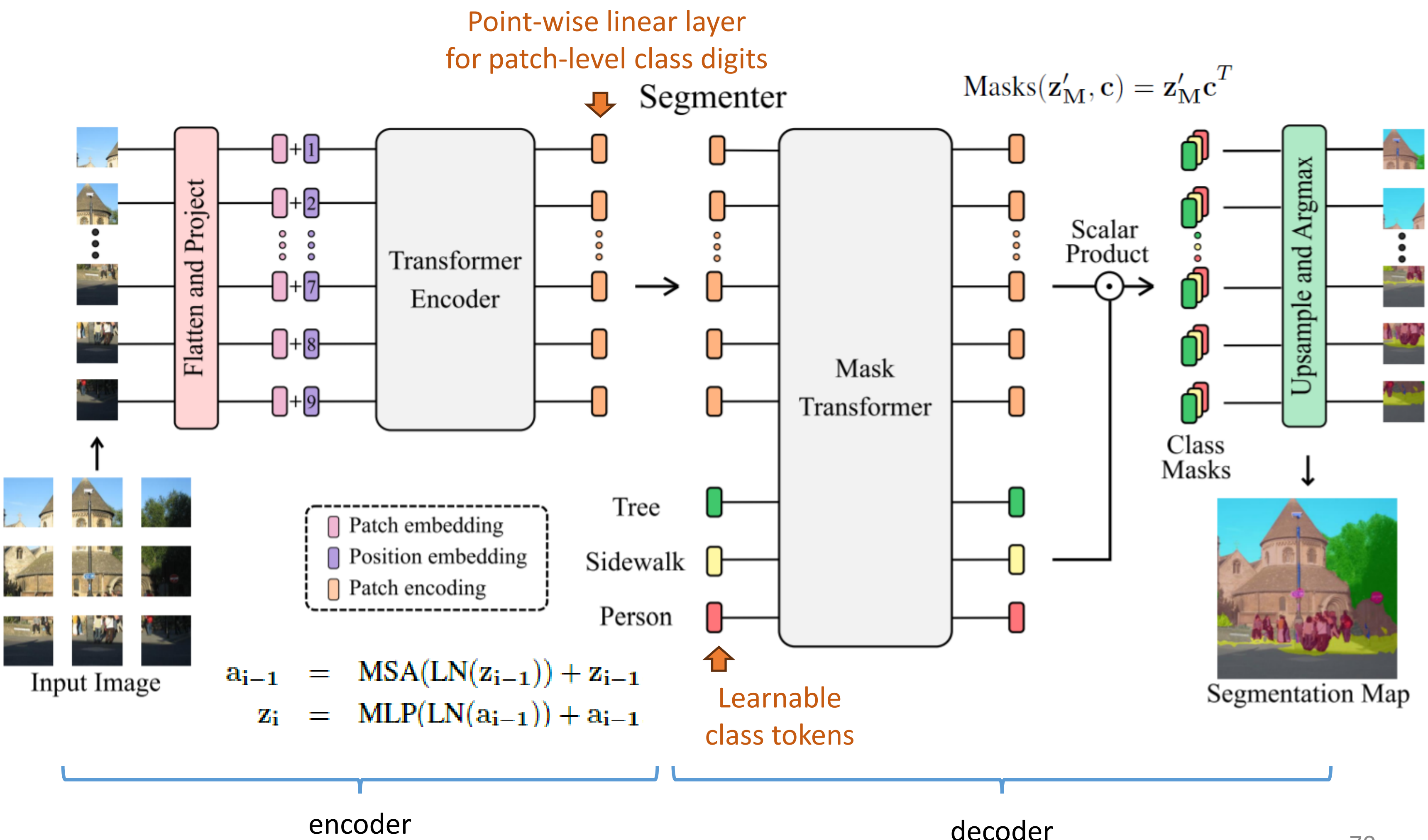
- Segmentation via attention



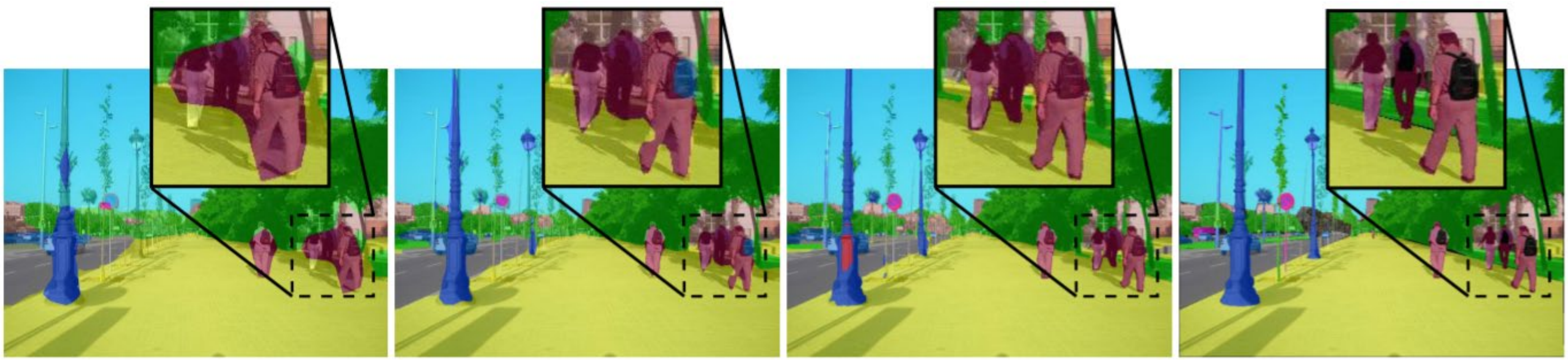
Strudel et al. "Segmenter: Transformer for Semantic Segmentation." ICCV 2021

Transformer for Semantic Segmentation (cont'd)

- Inspired by object detection models of DETR (ECCV'20), etc.



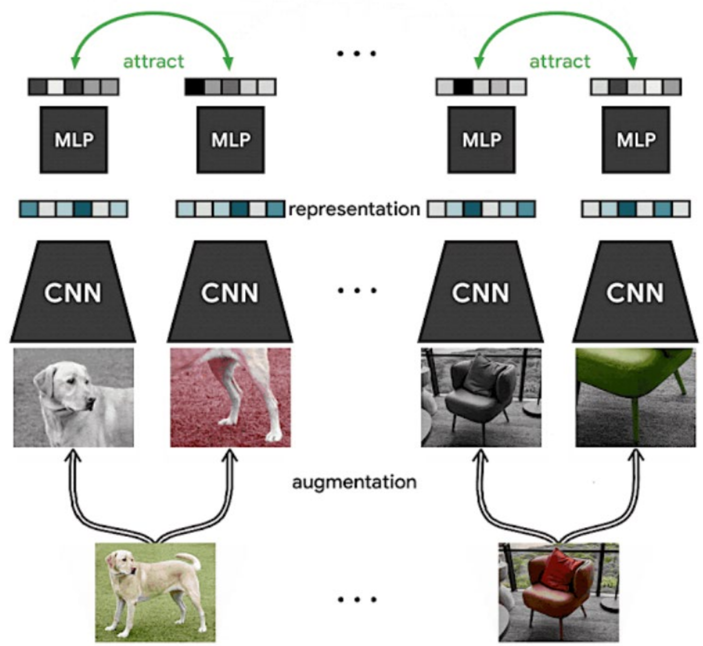
Example Visualization



(a) Patch size 32×32 (b) Patch size 16×16 (c) Patch size 8×8 (d) Ground Truth

Self-Supervised Learning (SSL)

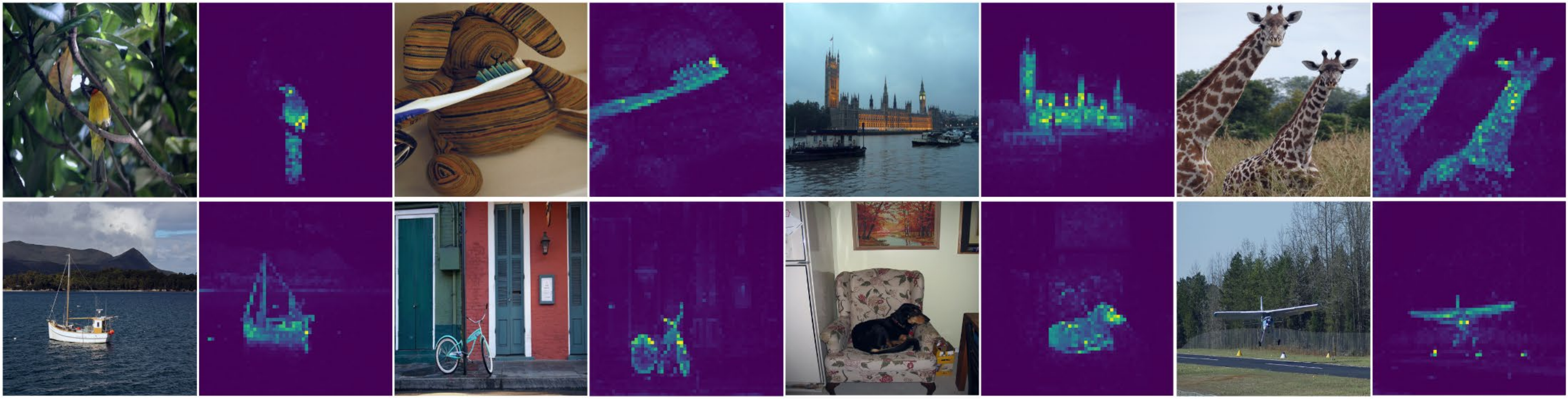
- Learning (somewhat) discriminative representations from **unlabeled** data
- Create self-supervised tasks via **data augmentation**
- Recall: SSL for CNN using image data:



Chen et al. "A simple framework for contrastive learning of visual representations." ICML 2020

Self-Supervised Learning (SSL) for Transformer (cont’d)

- SSL ViT/DeiT features contain info about semantic segmentation
- The above features are excellent k-NN classifiers
- By visualizing self-attention of the CLS token on different heads of the last layer:



Facebook AI Research & INRIA, "Emerging Properties in Self-Supervised Vision Transformers." ICCV 2021

Self-Supervised Learning (SSL) for Transformer (cont'd)

- Illustration of the proposed idea:

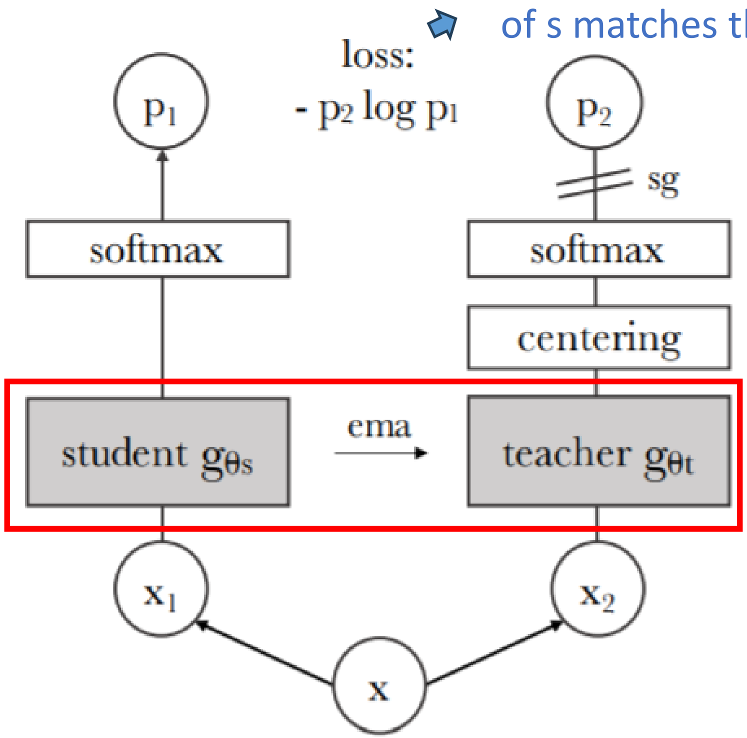
Facebook AI Research & INRIA, "Emerging Properties in Self-Supervised Vision Transformers." ICCV 2021

Self-Distillation with No Labels (DINO)

- Vision Transformer + **SSL**
- Maximize the prediction similarity btw input & its augmented version
- Idea: a **teacher-student** network (i.e., knowledge distillation) & **EMA** (exponential moving average)

CE loss measures how well prediction of s matches that of t.

$$\theta_T \leftarrow \tau\theta_T + (1-\tau)\theta_S$$



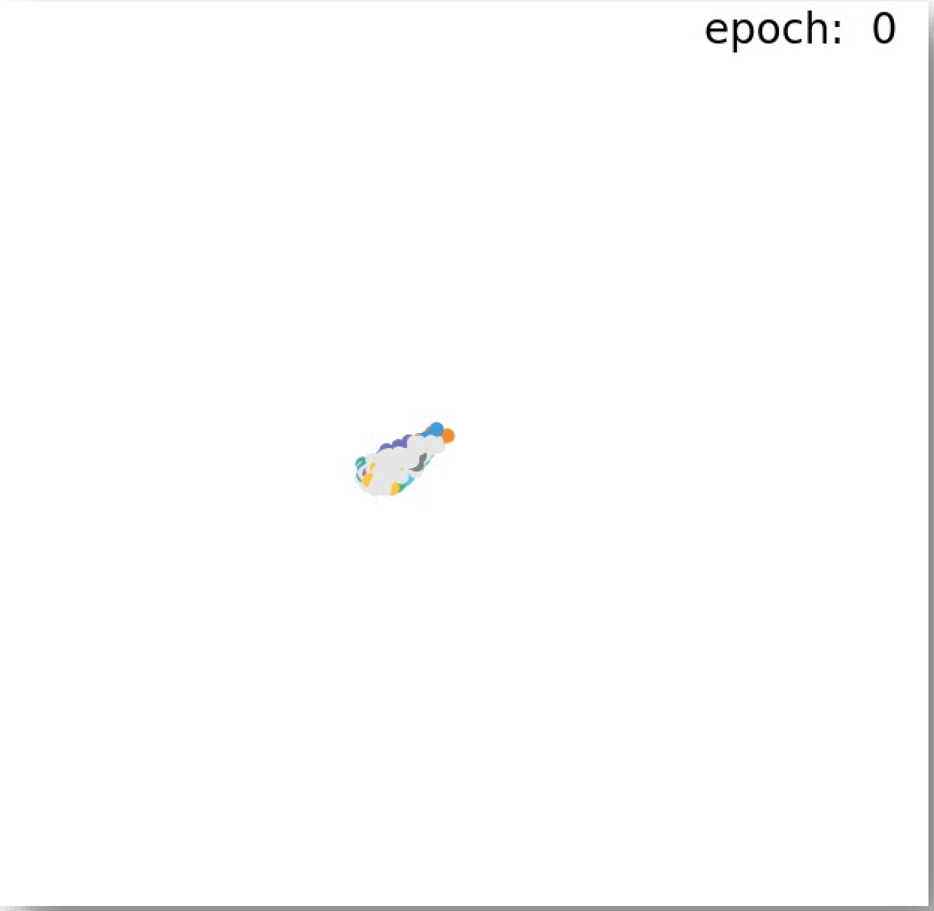
Avoid model collapse

Vision Transformer

Caron et al. “Emerging properties in self-supervised vision transformers.” ICCV 2021

Self-Supervised Learning (SSL) for Transformer (cont'd)

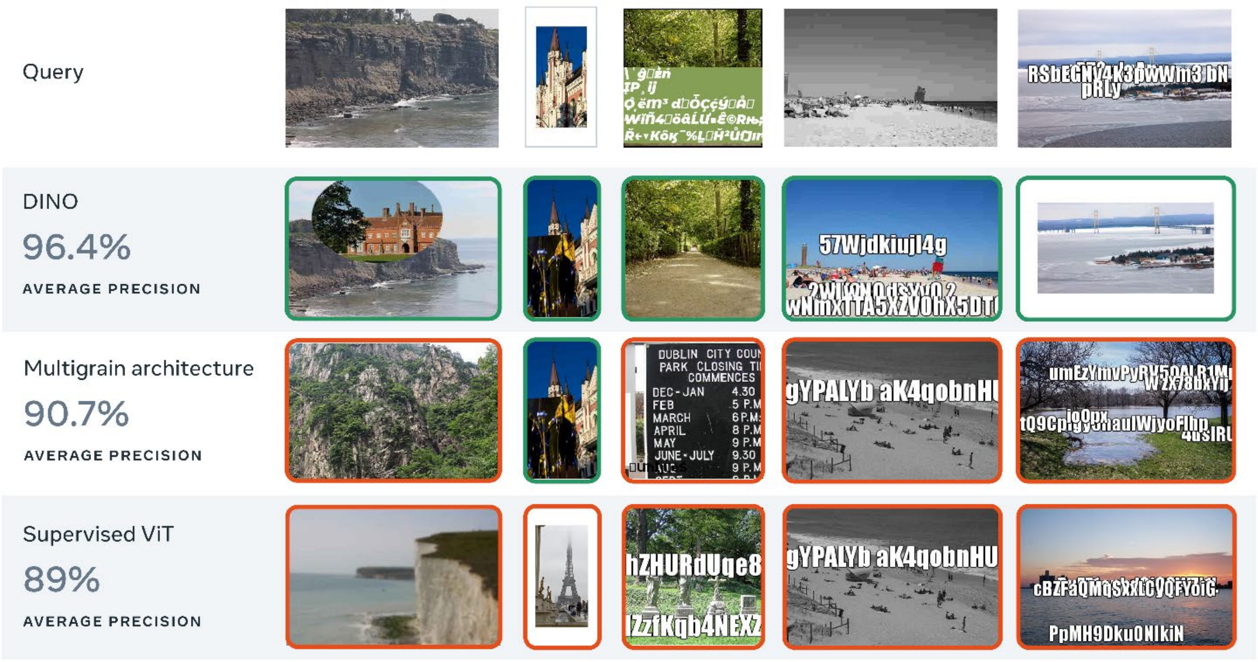
- Illustration of the learned features:



Facebook AI Research & INRIA, "Emerging Properties in Self-Supervised Vision Transformers." ICCV 2021

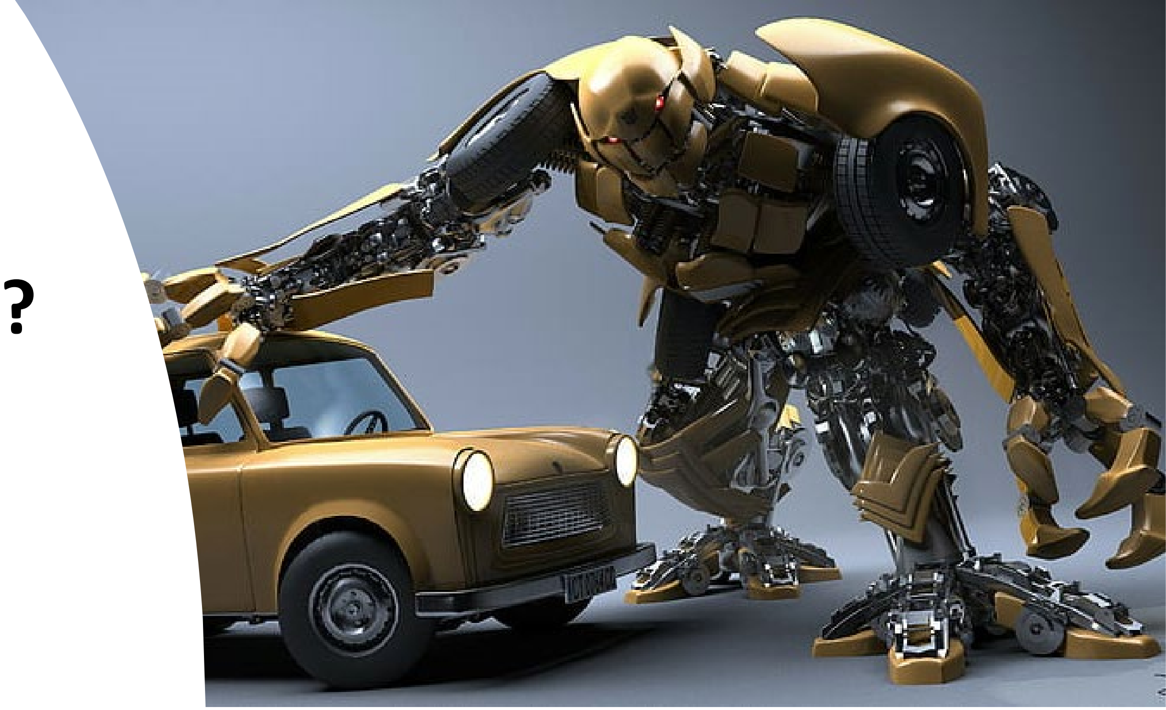
Highlights & Example Results of DINO

- Learning discriminative representations from **unlabeled** data
- Create self-supervised tasks via **data augmentation**



Facebook AI Research & INRIA, "Emerging Properties in Self-Supervised Vision Transformers." ICCV 2021

What to Be Covered?



- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models

A picture is worth a thousand words…

Vision and Language

- Image-to-Text: Image Captioning
- Text-to-Image: Image Manipulation
- Composed Image Retrieval
- Visual Question Answering (VQA)
- VQA + Natural Language Explanation
  and many more…

- E.g., Question: Is the xiaolongbao fresh at Din Tai Fung?
  Answer: Yes.
  Explanation: Because the xiaolongbao is made to order at the restaurant.

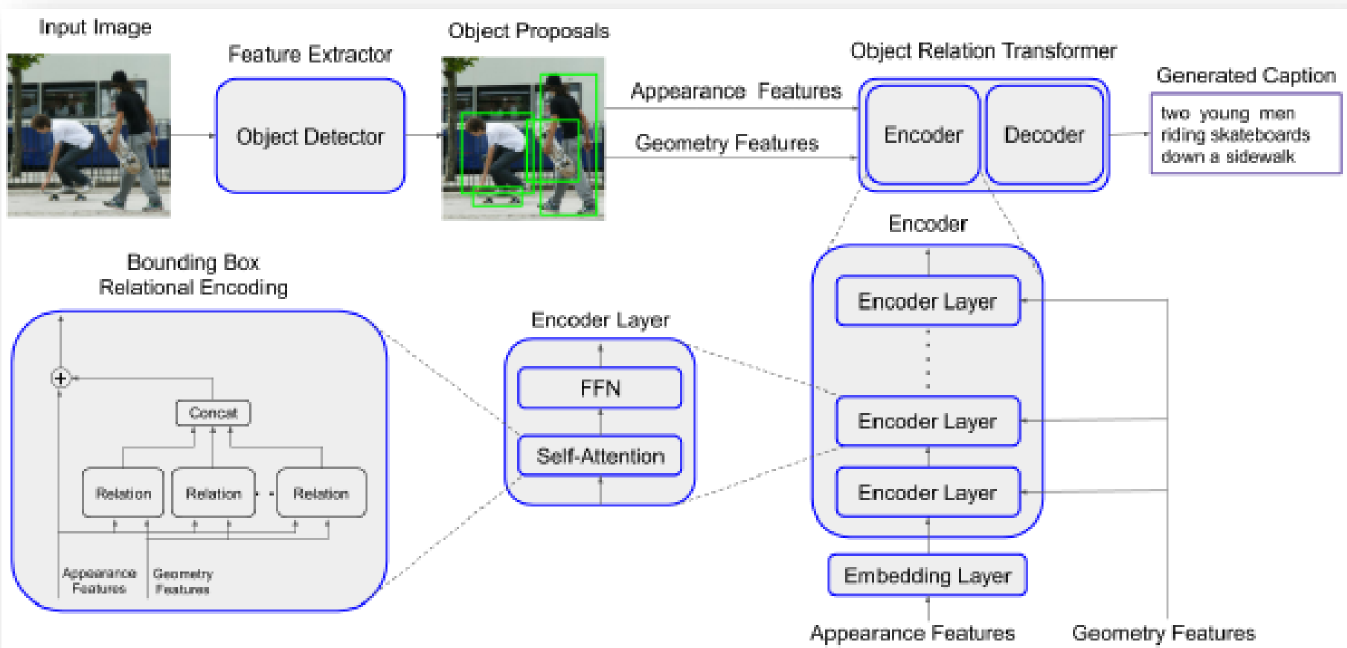
Visual Input

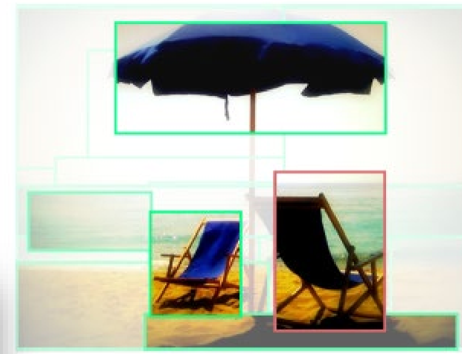
Image Captioning

Applications: semantics understanding, image-text retrieval, medical AI, etc.
How does it help GenAI? (e.g., text-to-image generation)

Image Captioning (cont'd)

- *Image Captioning: Transforming Objects into Words*, Yahoo Research, NeurIPS 2019
- Motivation: mid-level image understanding for captioning
- Framework:

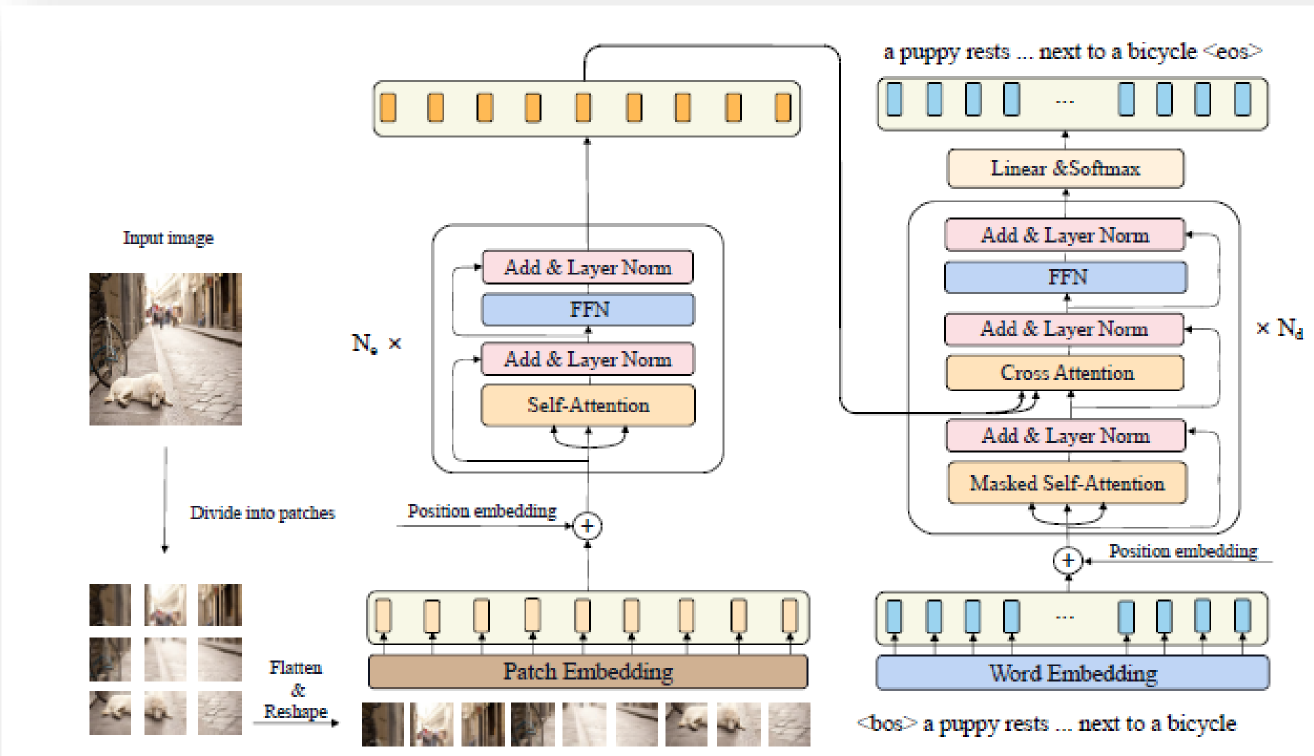




Generated Caption: *two beach chairs under an umbrella on the beach*

Image Captioning (cont'd)

- **Cap**tion **T**ransformer (CPTR) -
  CPTR: Full Transformer Network for Image Captioning, CAS, arxiv 2021
- Motivation: patch translation for image captioning



Remarks & Extension:

- Training a captioning model requires a large amount of image-caption data pairs
- Image captioning in the wild:
  - Describing images with novel content during inference
  - For example, COCO dataset has 80 object categories.
    How to generalize captioning models to Open Image (w/ 600 classes)?
- Domain-specific image captioning:
  - From general-purpose captioning to task-oriented captioning **-> finetuning?**

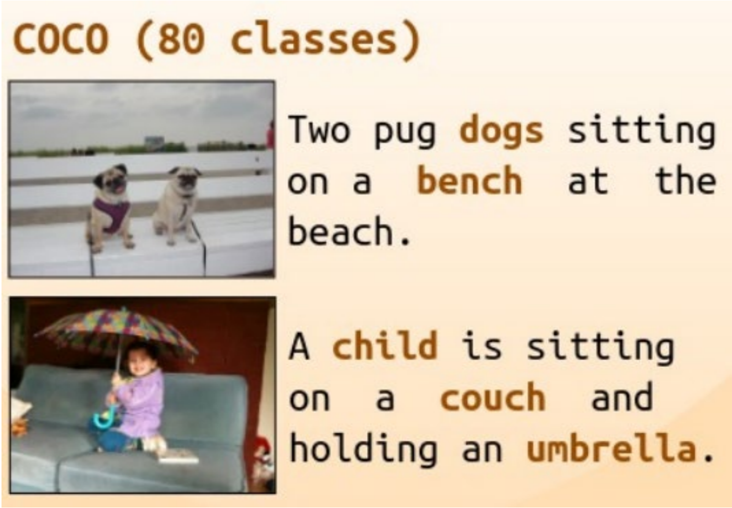
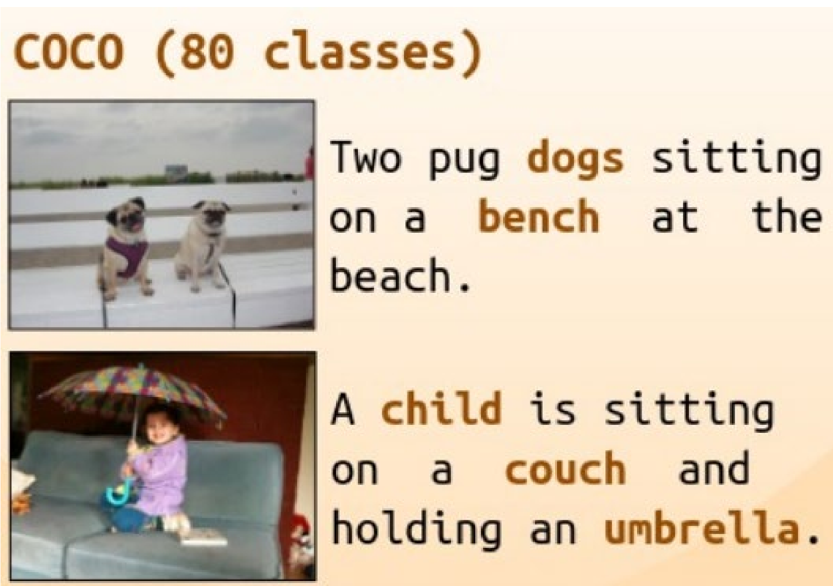


Image Captioning *in the Wild*

- **Novel Object Captioning (NOC)**
  - Training with captioned and uncaptioned data
    captioned data: labeled image data with captions (e.g., COCO)
    uncaptioned data: only labels of novel classes available (e.g., Open Images)

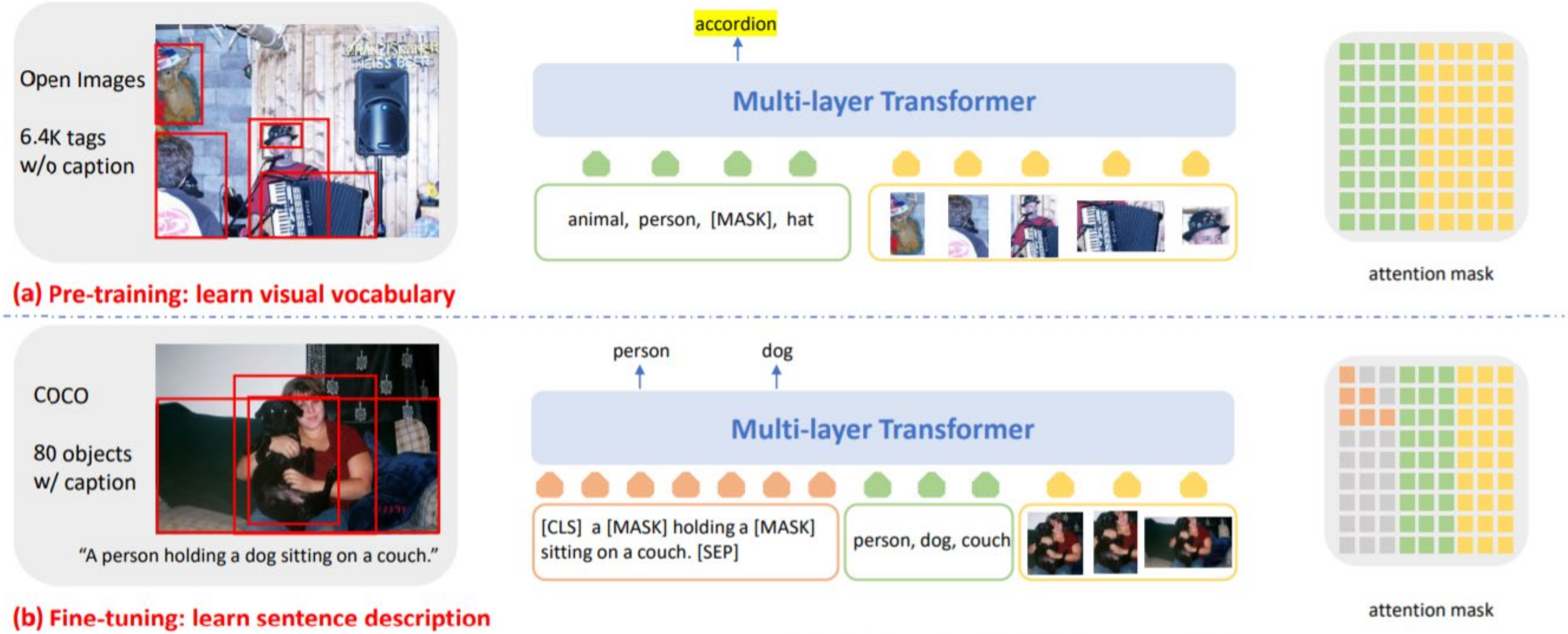


We have image-caption data

Data with labels for novel objects but w/o captions

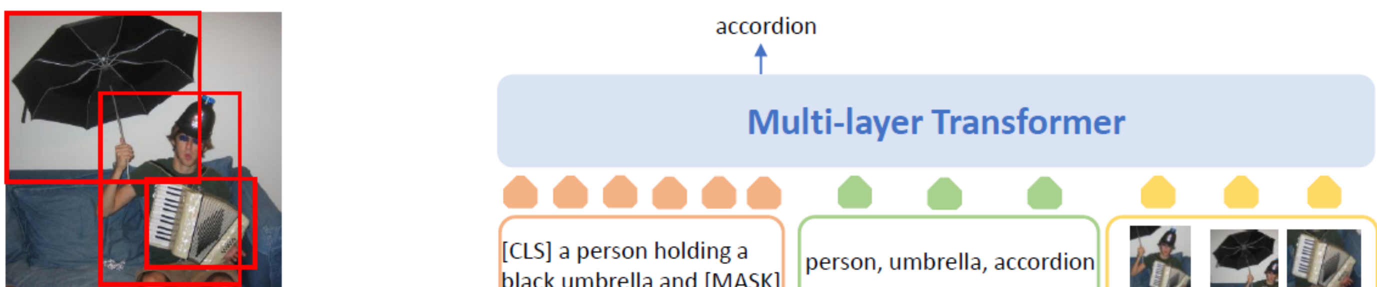
Novel Object Captioning (cont'd)

- "**Paraphrasing Is All You Need for Novel Object Captioning**", NTU, NeurIPS'22
- **VIVO**: **Vi**sual **Vo**cabulary Pre-Training for Novel Object Caption Captioning, Microsoft, AAAI'21
  - Pre-training: uncaptioned image data containing novel class labels
  - Fine-tuning: (a limited amount of) image data with class labels & descriptions



Novel Object Captioning (cont'd)

- **VIVO**: **Vi**sual **Vo**cabulary Pre-Training for Novel Object Caption Captioning (AAAI'21)
  - Pre-training: uncaptioned image data containing novel class labels
  - Fine-tuning: (a limited amount of) image data with class labels & descriptions
  - Inference:
    - Inputs: image (with region features & tags) & [CLS]
    - Output: caption

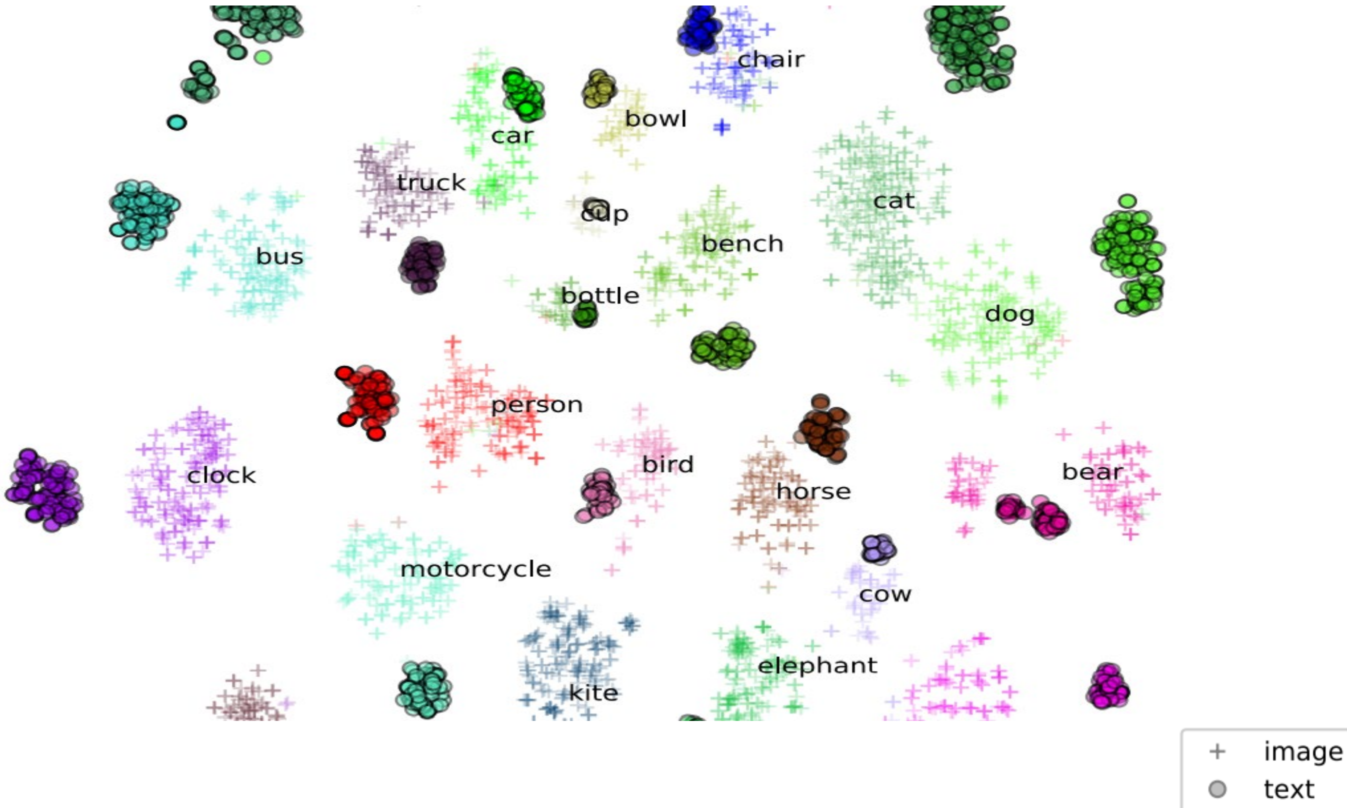


(c) Inference: novel object captioning

A person holding a black umbrella and accordion.

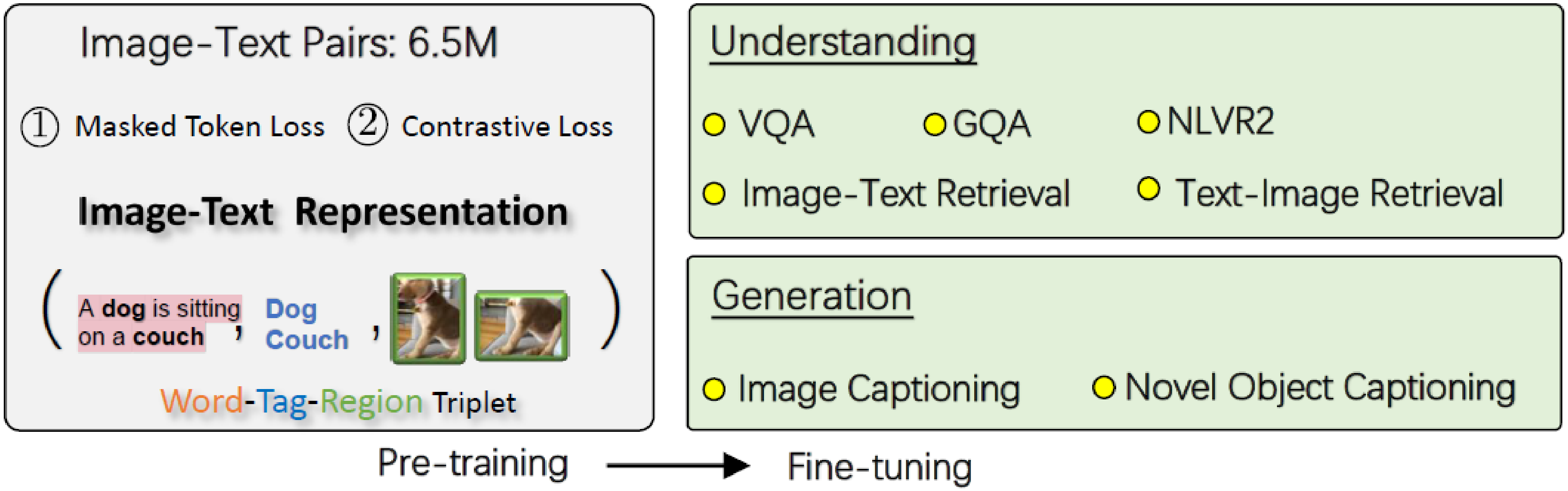
Novel Object Captioning (cont'd)

- VIVO: Visual Vocabulary Pre-Training for Novel Object Caption Captioning
    - Visualization image-text alignment



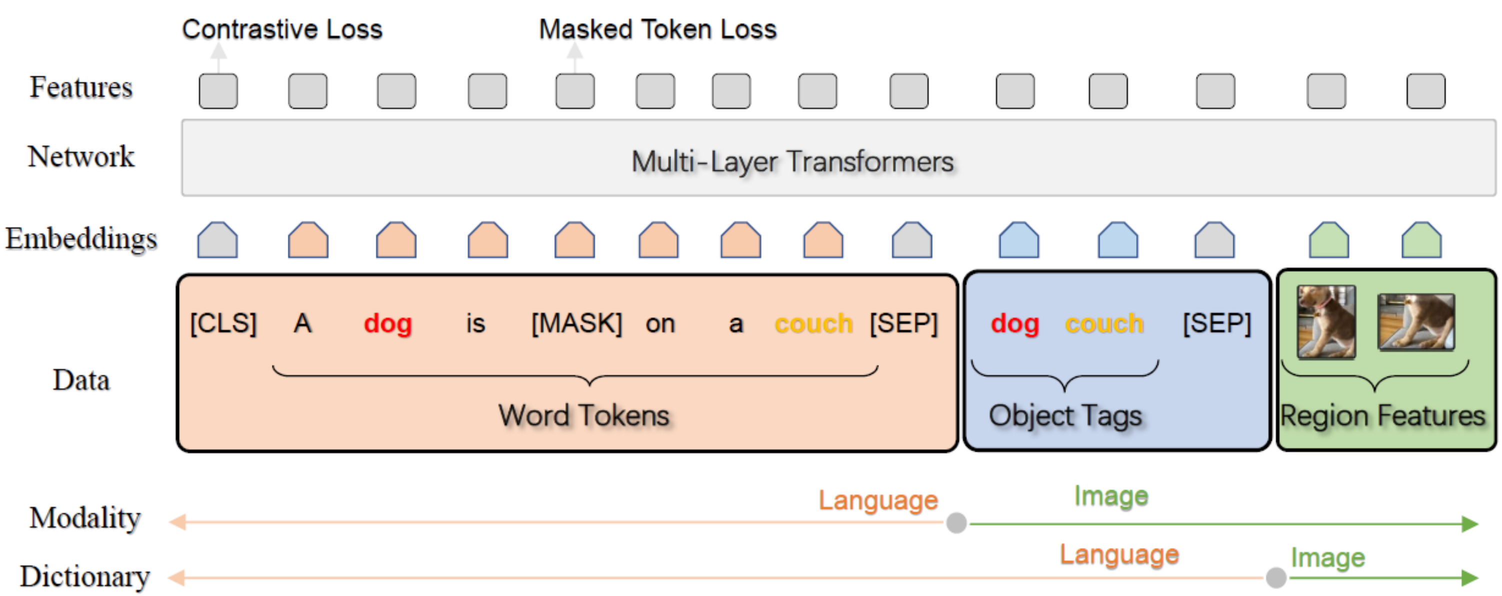
Beyond Image Captioning:
Unified Vision & Language Model

- **Oscar**: **O**bject-**S**emanti**c**s **A**ligned P**r**e-training for Vision-Language Tasks, Microsoft, ECCV'20
  - Training data:
    triplets of caption-tag-region
  - Objectives:
    1. Masked token loss for words & tags
    2. Contrastive loss tags and others
  - Fine-tuning:
    5 vision & language tasks (VQA, image-text retrieval, image captioning, NOC, etc.)



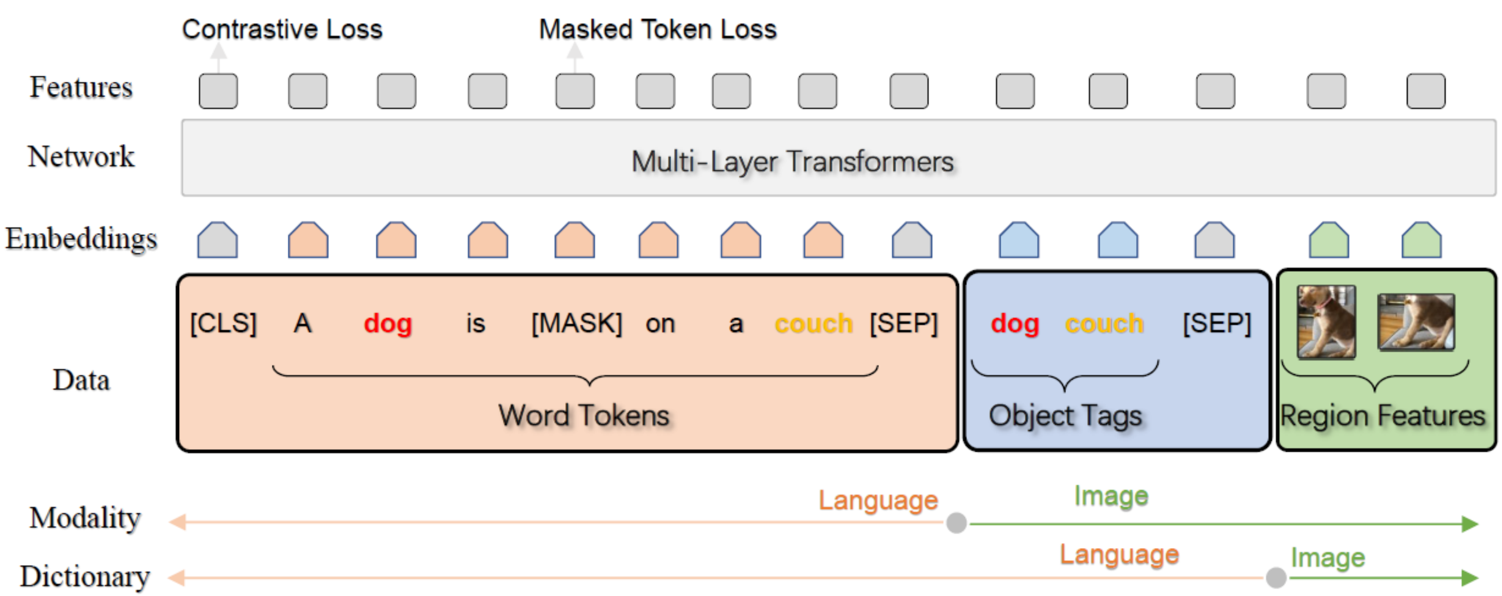
Semantics-Aligned Pre-training for V+L Tasks

- **Oscar**: **O**bject-**S**emanti**c**s **A**ligned P**r**e-training for Vision-Language Tasks
  - Training:
    - Inputs: triplets of caption-tag-region
    - Objectives: Masked token loss for words & tags + Contrastive loss tags and others
  - Fine-tuning:
    5 vision & language tasks (image captioning, NOC, VQA, image-text retrieval, etc.)



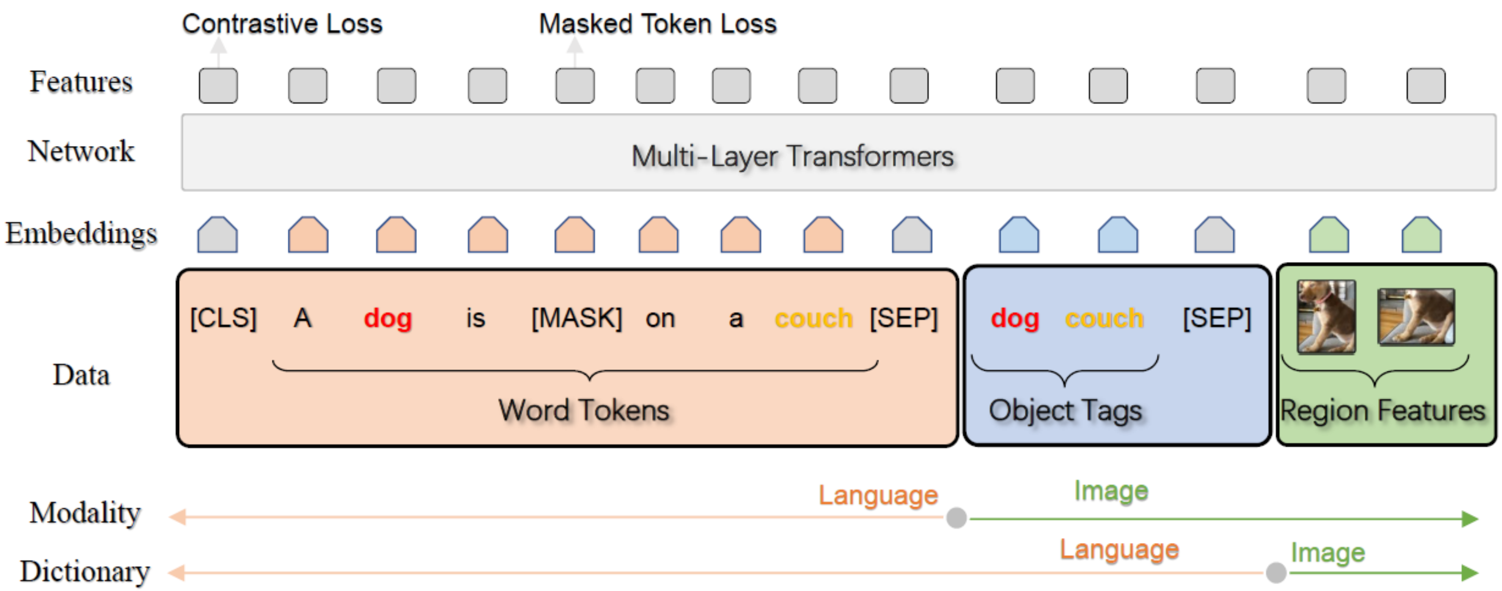
Semantics-Aligned Pre-training for V+L Tasks (cont'd)

- **Oscar**: **O**bject-**S**emanti**c**s **A**ligned P**r**e-training for Vision-Language Tasks (ECCV'20)
  - Training:
    - Inputs: triplets of word-tag-region
    - Objectives: Masked token loss for words & tags + Contrastive loss tags and others
  - Fine-tuning:
    - 5 vision & language tasks (image captioning, NOC, VQA, image-text retrieval, etc.)



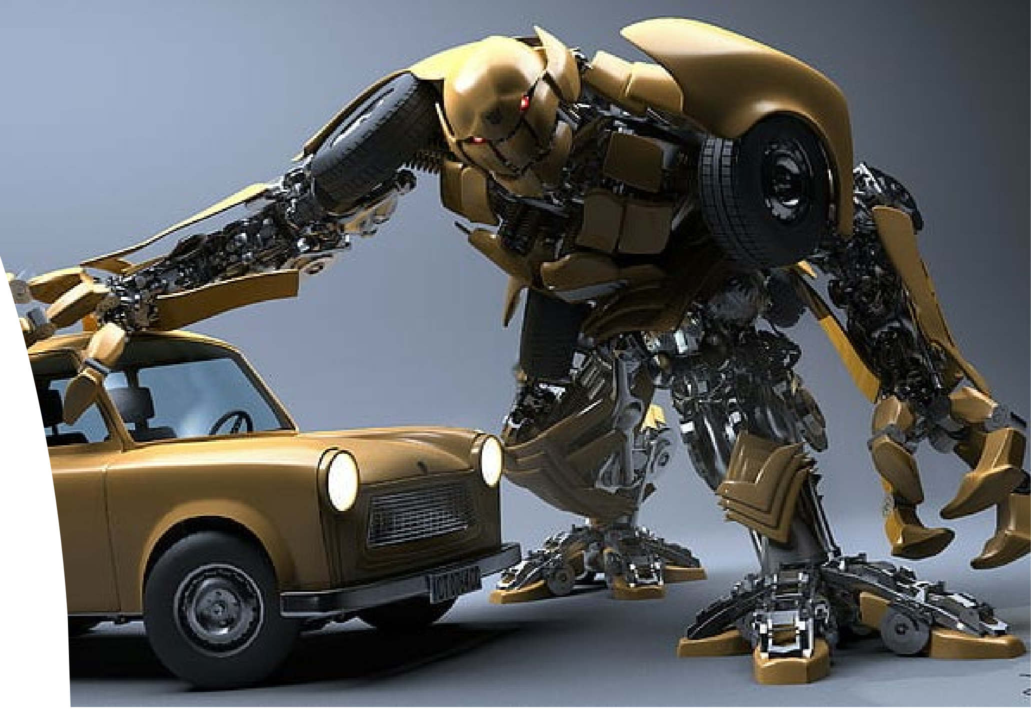
Holding an apple ↔  or 

- **Oscar** (cont'd)
  - Fine-tuning:
    5 vision & language tasks (image captioning, NOC, VQA, image-text retrieval, etc.)
  - Take **image-text retrieval** as an example
    - Training: aligned/mis-aligned image-text pairs as positive/negative input pairs, with **[CLS]** for binary classification (1/0)
    - Inference: for either image or text retrieval, calculate classification score of **[CLS]** for the input query



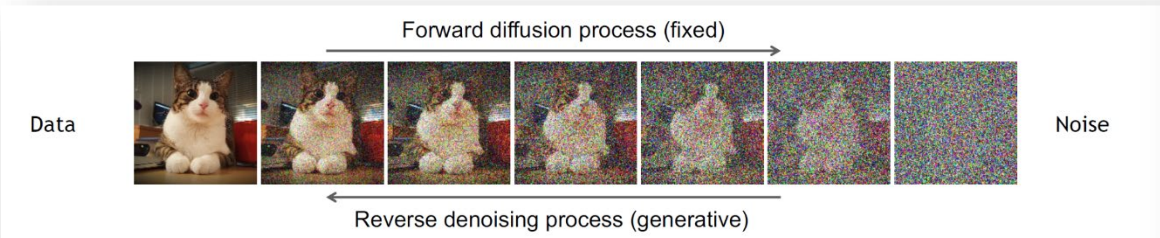
What to Be Covered?

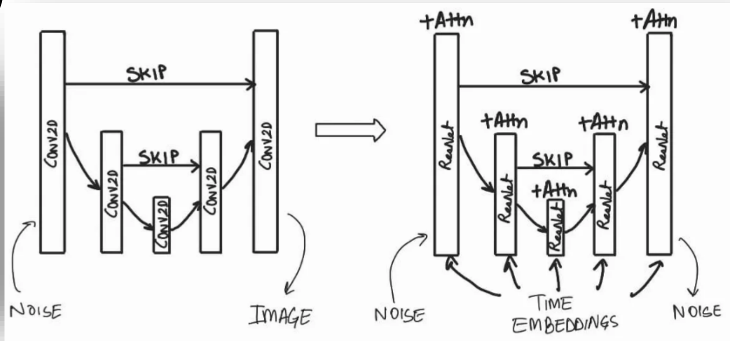
- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



Recap: Denoising Diffusion Probabilistic Models (DDPM)

- Training:
  - Forward/reverse diffusion & denoising process
    - learns to generate/restore data by denoising
    - typically implemented via a **conditional U-net**)





- Pseudo Code for Training/Inference (Sampling):

Algorithm 1 Training

repeat
$\mathbf{x}_0 \sim q(\mathbf{x}_0)$
$t \sim \mathrm{Uniform}(\{1, \dots, T\})$
$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boxed{\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}}, t) \right\|^2$$
until converged

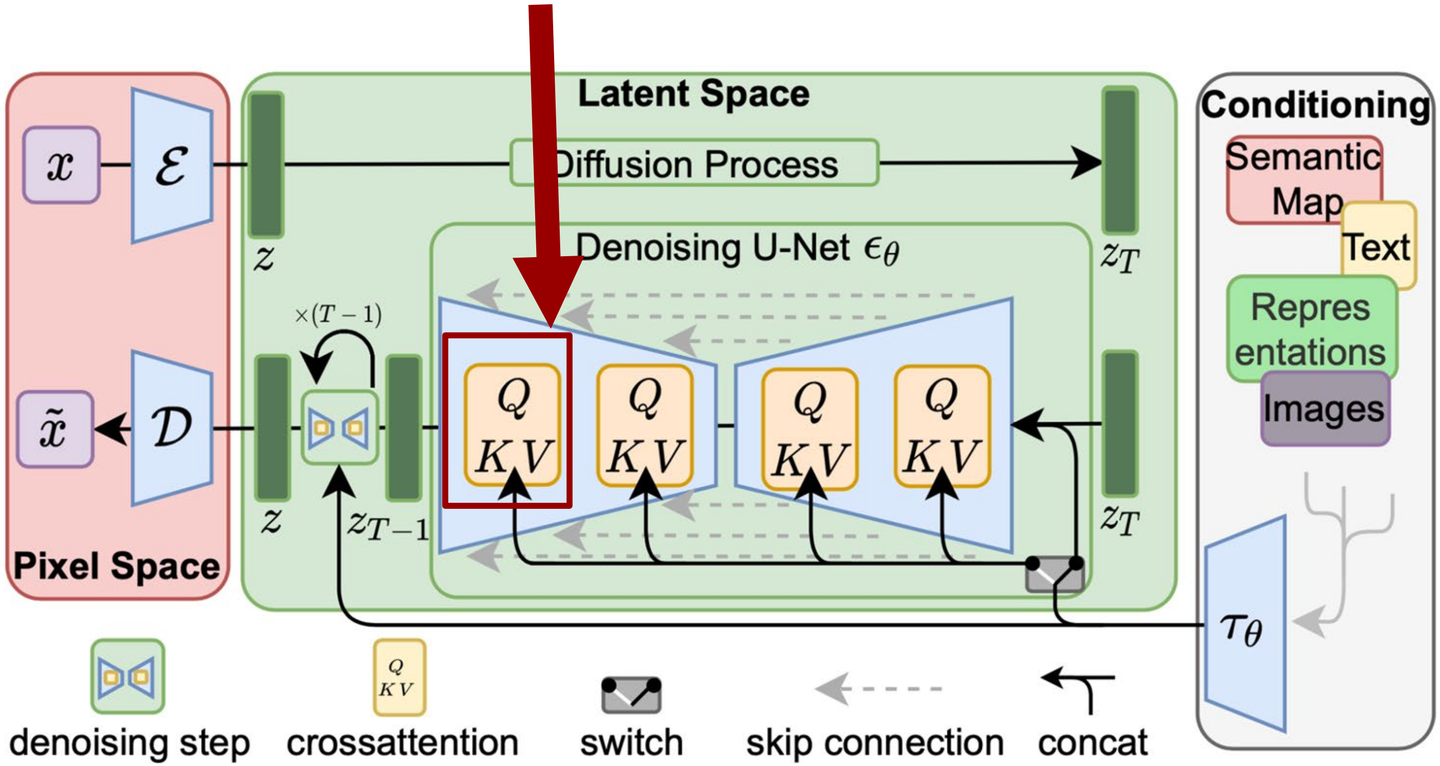
Algorithm 2 Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ **do**
$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
$$\mathbf{x}_{t-1} = \boxed{\frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right)} + \sigma_t \mathbf{z}$$
end for
return $\mathbf{x}_0$

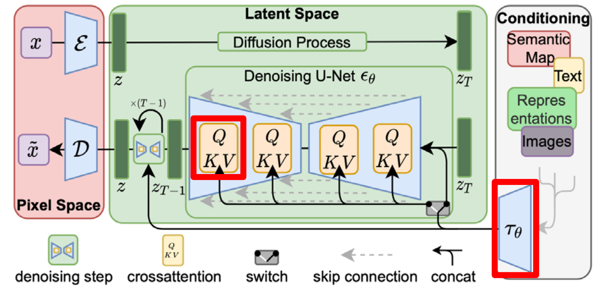
Ho et al., Denoising Diffusion Probabilistic Models, NeurIPS 2020
Song et al., Score-Based Generative Modeling through Stochastic Differential Equations, ICLR 2021

From Unconditional to Conditional Latent Diffusion Model

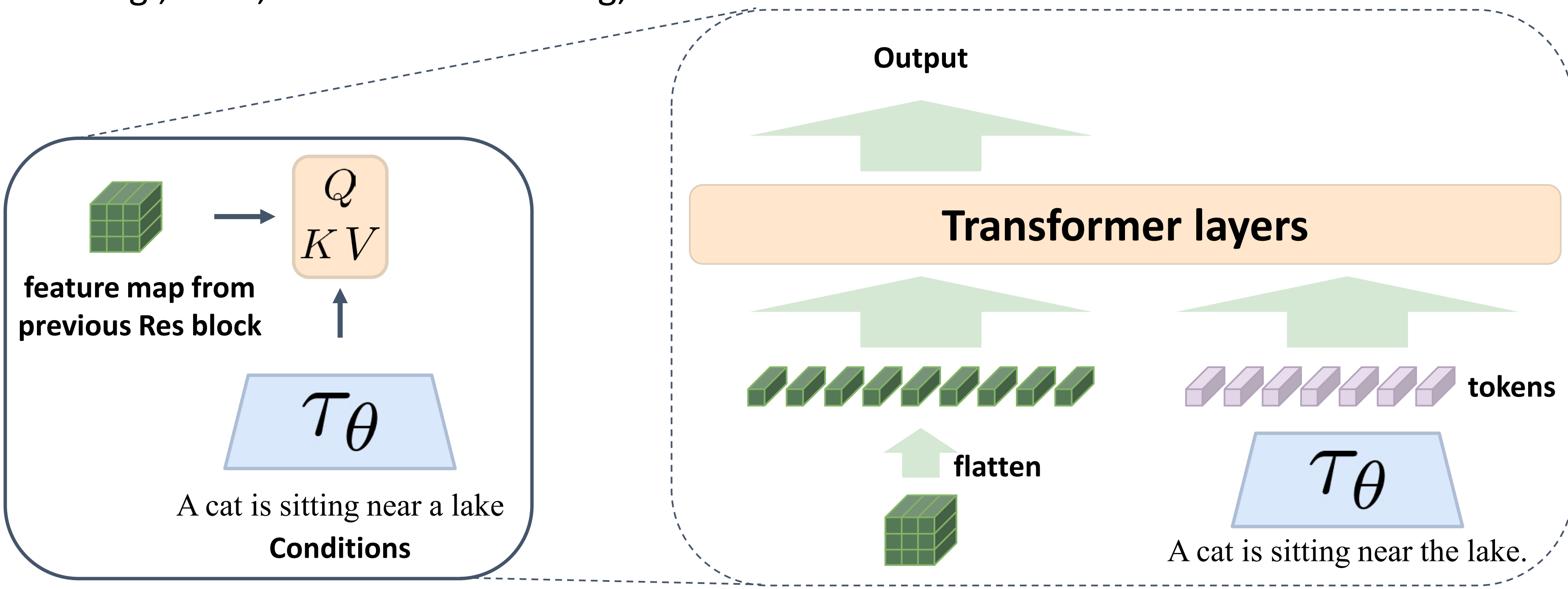
- Latent diffusion model (LDM), CVPR'22: DDPM in latent space
- Condition mechanism: cross-attention at transformer layers



From Unconditional to Conditional Latent Diffusion Model (cont'd)

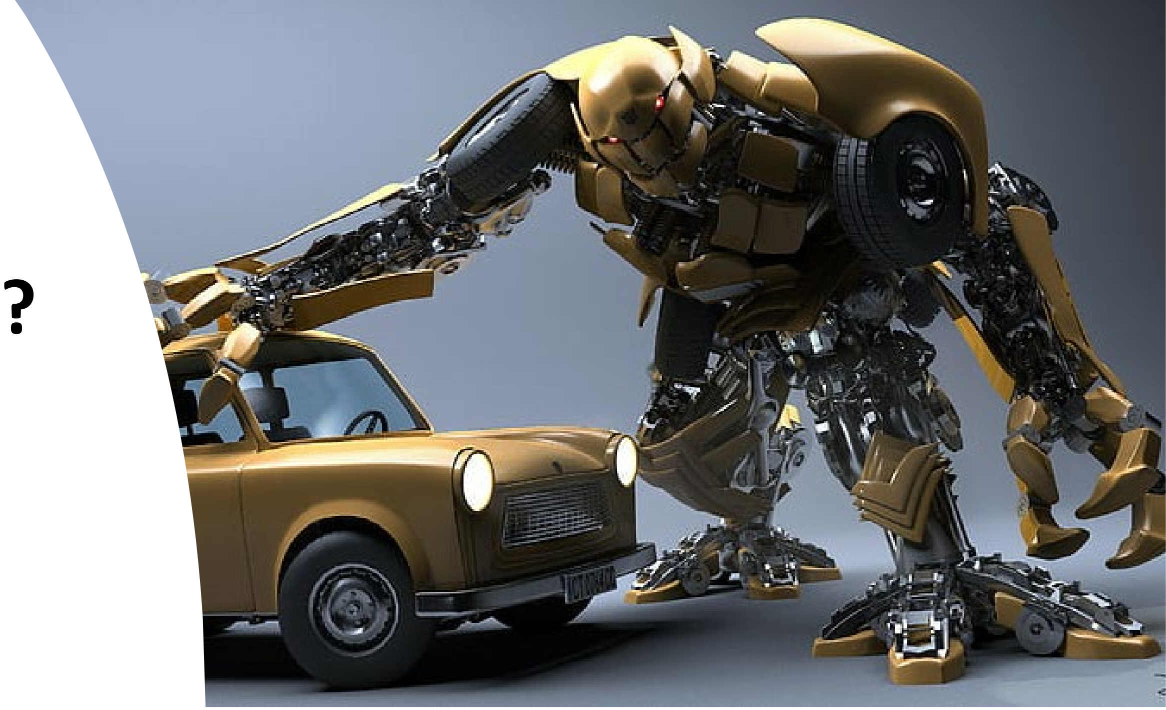


- **Condition mechanism:** using transformer layers
- τ_θ is the embedding module for conditions e.g., BERT, CLIP text embedding, etc.



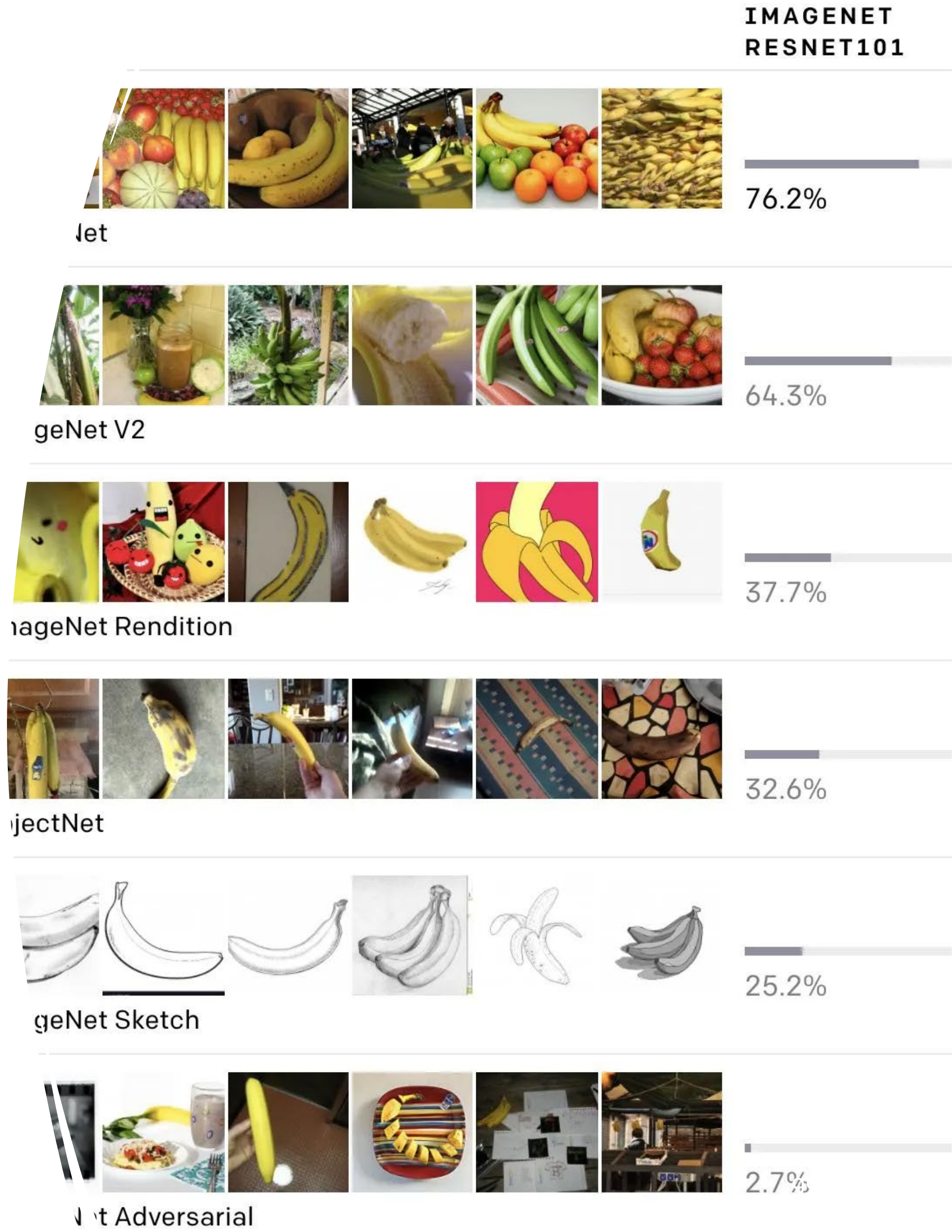
What to Be Covered?

- Transformer
  - Self-Attention
  - Cross-Attention
  - Positional Embedding
- Transformer for Visual Analysis
  - Vision Transformer (ViT)
  - DeiT & Swin Transformer
  - SSL & Beyond
- Vision-Language Model
  - Image2Text
  - Text2Image (√)
  - Image-text models



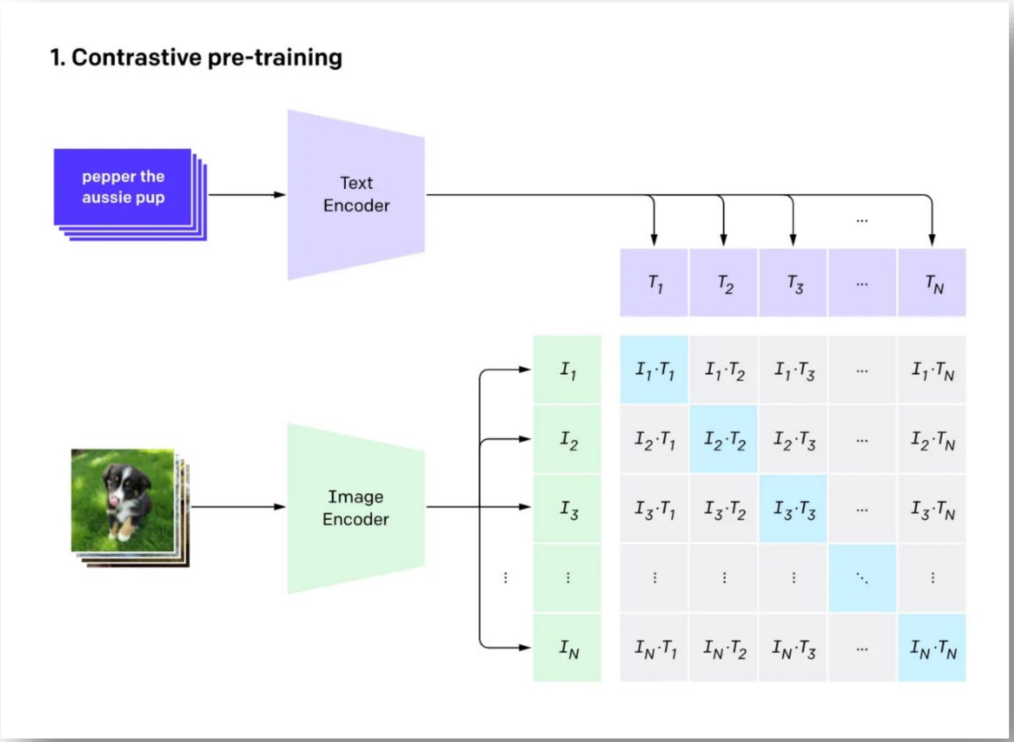
CLIP: Contrastive Language-Image Pretraining

- OpenAI, *Learning Transferable Visual Models From Natural Language Supervision*, NeurIPS WS 2021 (w/ 9000+ citations)
- Why DL/CNN not good enough?
  - Require annotated data for training image classification
  - Domain gap between closed-world and open-world domain data
  - Lack of ability for zero-shot classification



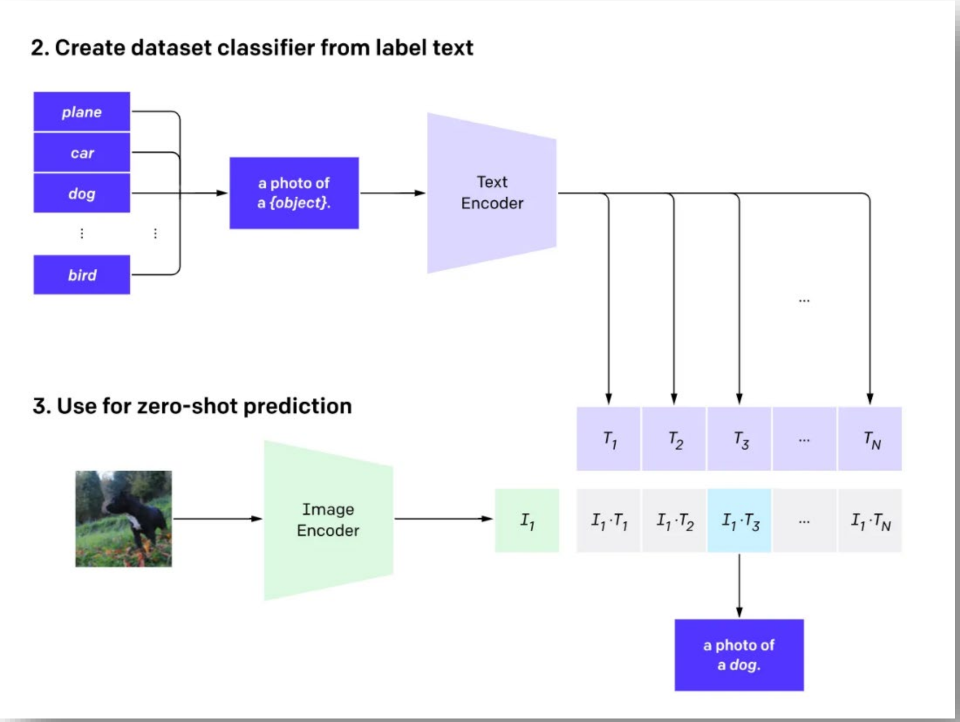
CLIP (cont'd)

- Why DL/CNN not good enough?
  - Require annotated data for training image classification
  - Domain gap between closed-world and open-world domain data
  - Lack of ability for zero-shot classification
- Motivation/Objectives
  - Cross-domain contrastive learning from large-scale image-language data



CLIP (cont'd)

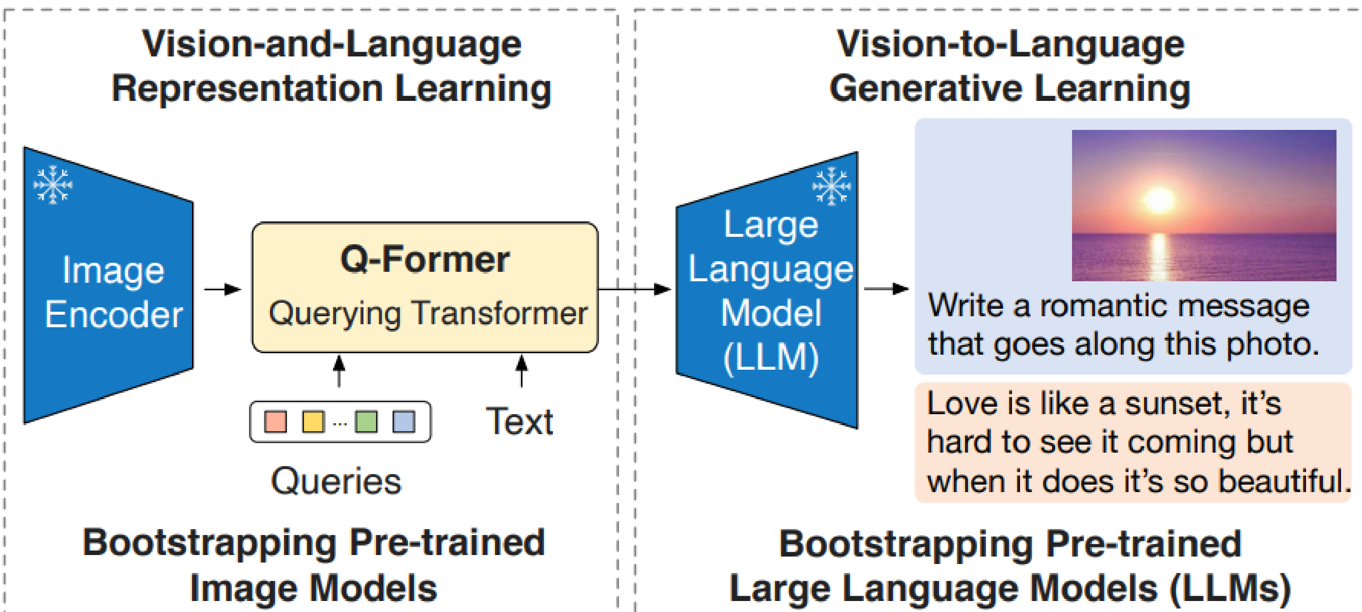
- (Zero-shot) Inference:



- Limitation
  - Fine-grained description ability
  - Any examples?

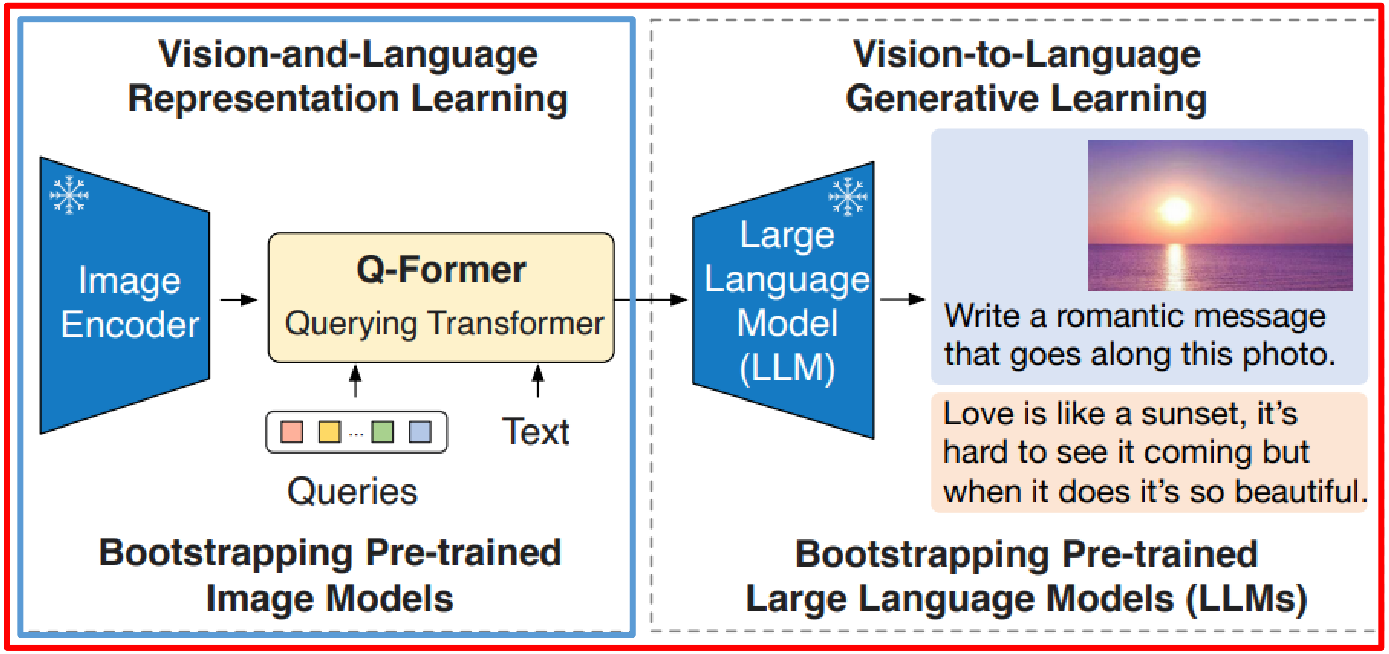
BLIP-2 (ICML'23)

- **BLIP:**
  Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, Salesforce Research, NeurIPS 2021
- **Goal:**
  Bridge the modality gap between off-the-shelf **frozen pre-trained image encoders** and **frozen large language models** with a lightweight Querying Transformer (Q-Former).
- **Advantages:**
  1. No need to train from scratch
  2. Avoid catastrophic forgetting (w/ fixed VLM & LLM)



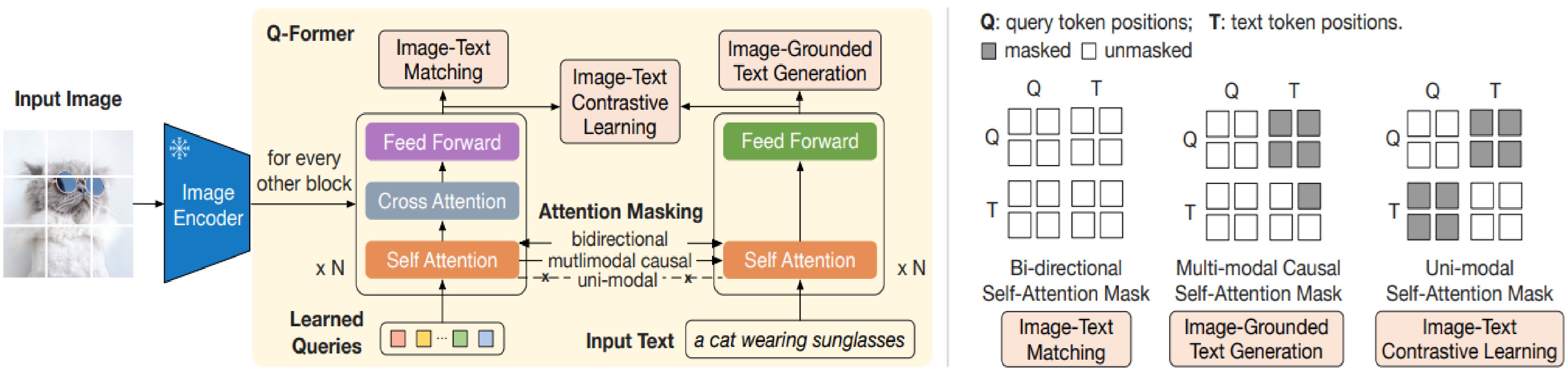
Pre-training

- A **two-stage** pre-training strategy
  - Stage 1: Representation Learning
    - enforce **Q-Former** to learn **visual representation** that is most relevant to the text description
  - Stage 2: Generative Learning
    - make the output representation of **Q-Former** to be understood by **LLM**



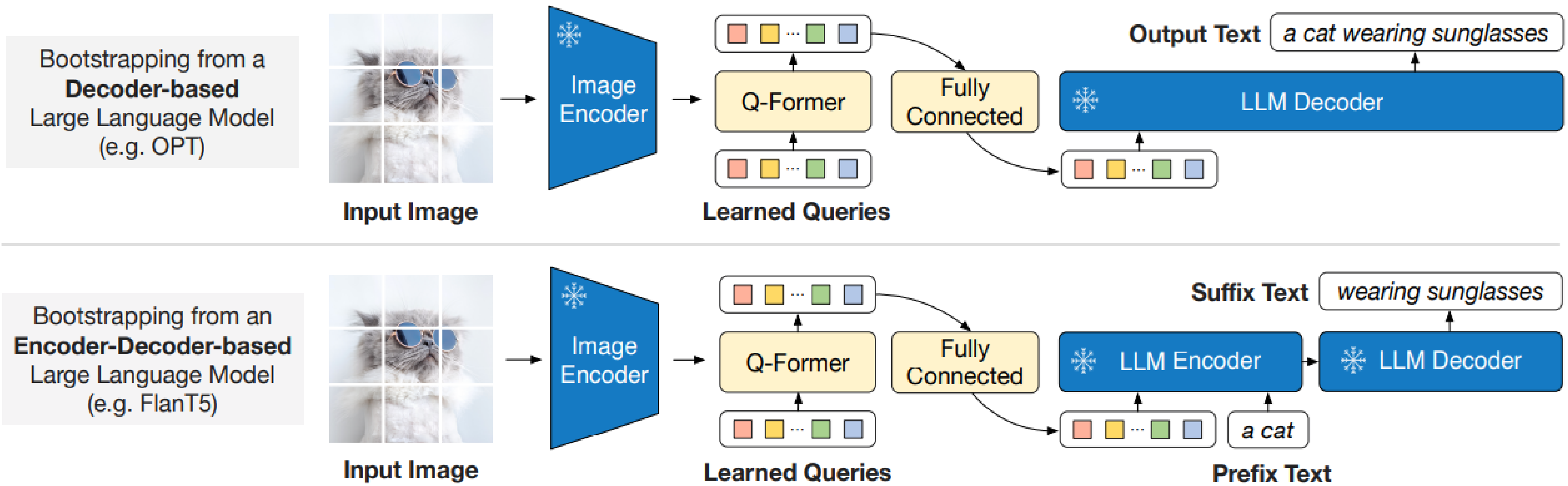
Pre-training Stage 1 - VL Representation Learning

- **Goal:** enforce **Q-Former** to extract visual representation relevant to text
- **Method:** three pre-training tasks
  - **Image-Text Contrastive Learning (ITC)**:
    self-attn in Q/T, followed by max (sim(Q, T)) -> can be viewed as CLIP training
  - **Image-Text Matching (ITM)**:
    for each learnable query -> linear classifier for binary decision
  - **Image-grounded Text Generation (ITG)**:
    self-attn in Q for encoder training; T->Q for image-to-text generation



Pre-training Stage 2 - VL Generative Learning

- **Goal:**
  Learning with LLM guidance
  i.e., make the output representation of **Q-Former** to be understood by **LLMs**.
- **Method:**
  pre-training with Image-grounded Text Generation (ITG)



Quantitative Results

- Comparison on zero-shot visual question answering (VQA)

Models	#Trainable Params	#Total Params	VQAv2 val	VQAv2 test-dev	OK-VQA test	GQA test-dev
VL-T5$_{\text{no-vqa}}$	224M	269M	13.5	-	5.8	6.3
FewVLM (Jin et al., 2022)	740M	785M	47.7	-	16.5	29.3
Frozen (Tsimpoukelli et al., 2021)	40M	7.1B	29.6	-	5.9	-
VLKD (Dai et al., 2022)	406M	832M	42.6	44.5	13.3	-
Flamingo3B (Alayrac et al., 2022)	1.4B	3.2B	-	49.2	41.2	-
Flamingo9B (Alayrac et al., 2022)	1.8B	9.3B	-	51.8	44.7	-
Flamingo80B (Alayrac et al., 2022)	10.2B	80B	-	56.3	**50.6**	-
BLIP-2 ViT-L OPT$_{2.7B}$	104M	3.1B	50.1	49.7	30.2	33.9
BLIP-2 ViT-g OPT$_{2.7B}$	107M	3.8B	53.5	52.3	31.7	34.6
BLIP-2 ViT-g OPT$_{6.7B}$	108M	7.8B	54.3	52.6	36.4	36.4
BLIP-2 ViT-L FlanT5$_{XL}$	103M	3.4B	62.6	62.3	39.4	<u>44.4</u>
BLIP-2 ViT-g FlanT5$_{XL}$	107M	4.1B	<u>63.1</u>	<u>63.0</u>	40.7	44.2
BLIP-2 ViT-g FlanT5$_{XXL}$	108M	12.1B	**65.2**	**65.0**	<u>45.9</u>	**44.7**

Quantitative Results (cont'd)

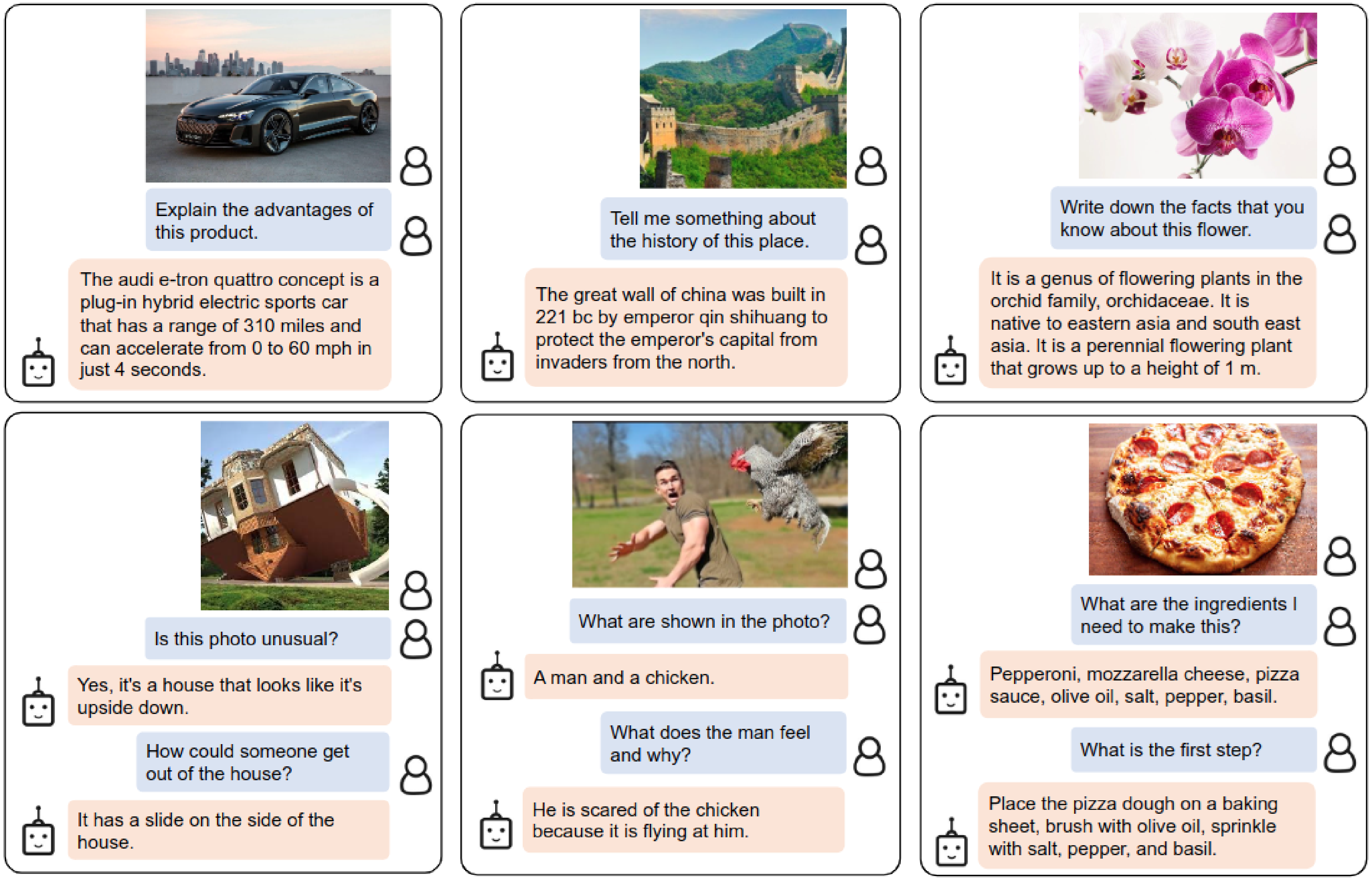
- Comparison on image captioning

Models	#Trainable Params	NoCaps Zero-shot (validation set) in-domain C	in-domain S	near-domain C	near-domain S	out-domain C	out-domain S	overall C	overall S	COCO Fine-tuned Karpathy test B@4	C
OSCAR (Li et al., 2020)	345M	-	-	-	-	-	-	80.9	11.3	37.4	127.8
VinVL (Zhang et al., 2021)	345M	103.1	14.2	96.1	13.8	88.3	12.1	95.5	13.5	38.2	129.3
BLIP (Li et al., 2022)	446M	114.9	15.2	112.1	14.9	115.3	14.4	113.2	14.8	40.4	136.7
OFA (Wang et al., 2022a)	930M	-	-	-	-	-	-	-	-	**43.9**	145.3
Flamingo (Alayrac et al., 2022)	10.6B	-	-	-	-	-	-	-	-	-	138.1
SimVLM (Wang et al., 2021b)	~1.4B	113.7	-	110.9	-	115.2	-	112.2	-	40.6	143.3
BLIP-2 ViT-g $OPT_{2.7B}$	1.1B	123.0	15.8	117.8	15.4	123.4	**15.1**	119.7	15.4	43.7	**145.8**
BLIP-2 ViT-g $OPT_{6.7B}$	1.1B	**123.7**	15.8	119.2	15.3	124.4	14.8	121.0	15.3	43.5	145.2
BLIP-2 ViT-g $FlanT5_{XL}$	1.1B	**123.7**	**16.3**	**120.2**	**15.9**	**124.8**	**15.1**	**121.6**	**15.8**	42.4	144.5

C: CIDEr S: SPICE B@4:BLEU@4

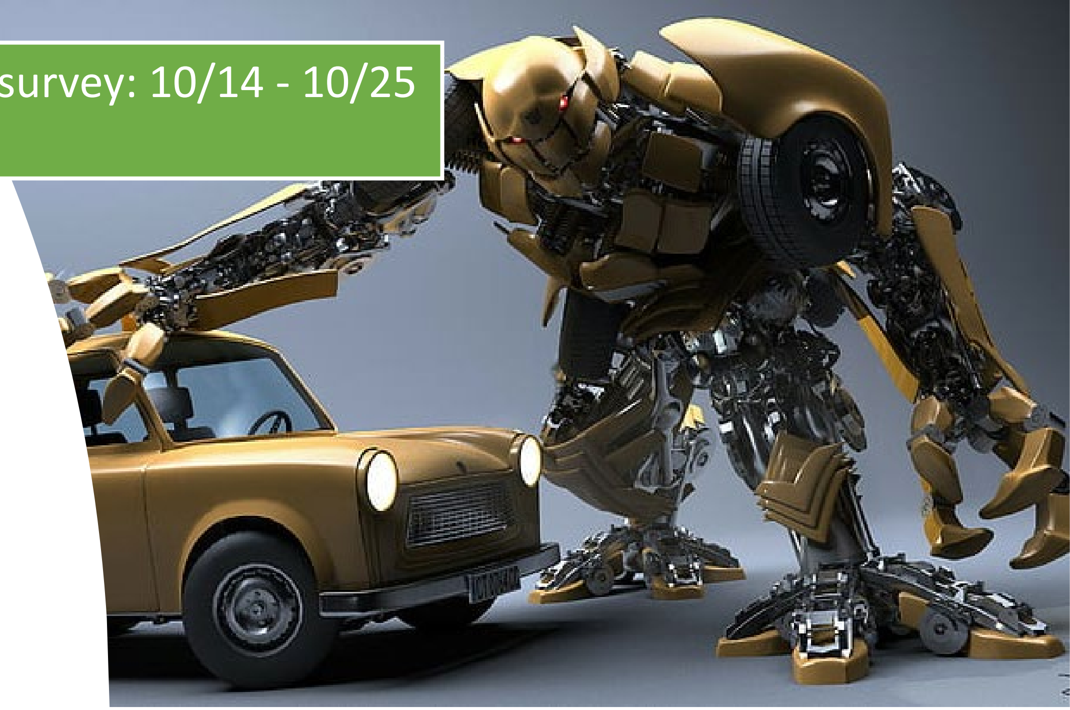
Visualization

- Instructed zero-shot image-to-text generation examples



[Reminder] Midterm course feedback survey: 10/14 - 10/25
Feedback is welcome!

What We have Covered Today



- Transformer
    - Self-Attention
    - Cross-Attention
    - Positional Embedding
- Transformer for Visual Analysis
    - Vision Transformer (ViT)
    - DeiT & Swin Transformer
    - SSL & Beyond
- Vision-Language Model
    - Image2Text
    - Text2Image (√)
    - Image-text models