CTU

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Deep Learning Essentials

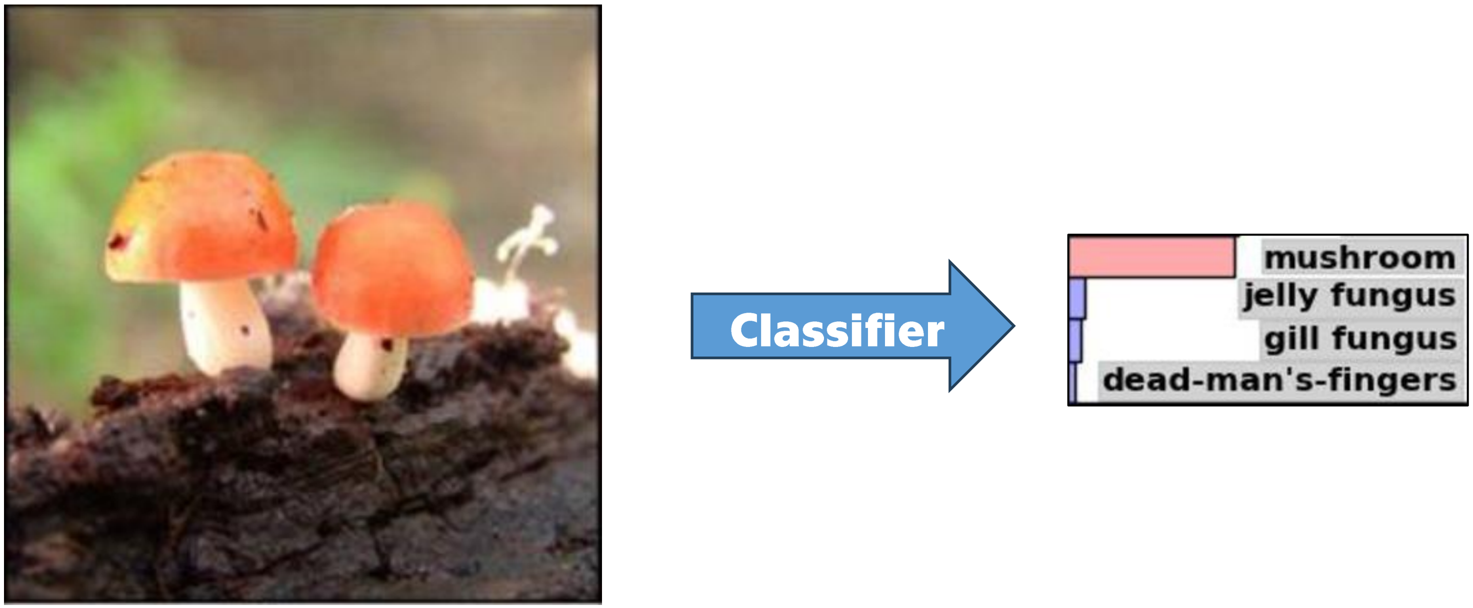
8. Backbone Architectures

ResNet, EfficientNet, Self-attention, Transformers

Lukáš Neumann

Adapted from B3B33UROB slides of Karel Zimmerman

Image Classification

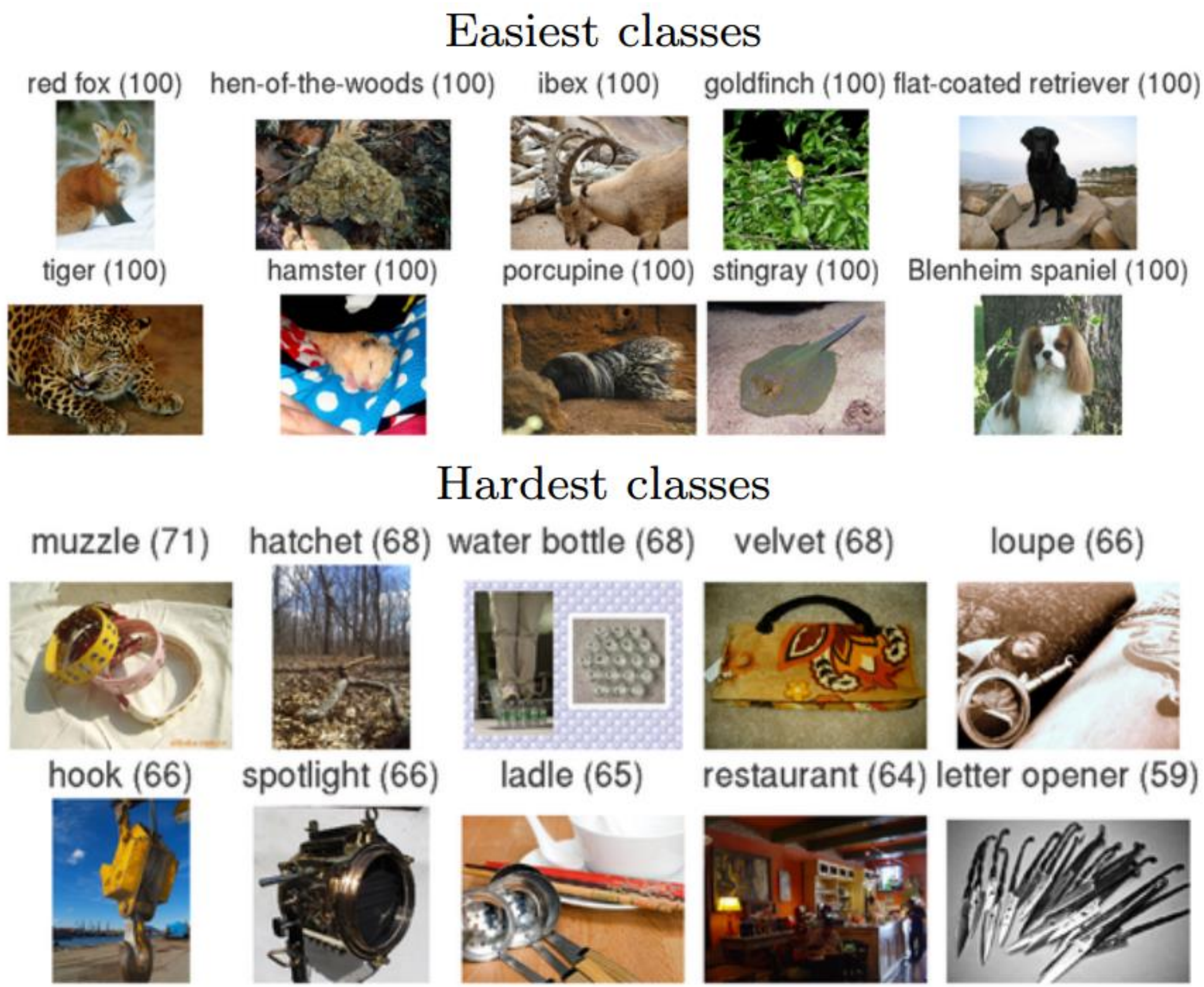


ImageNet

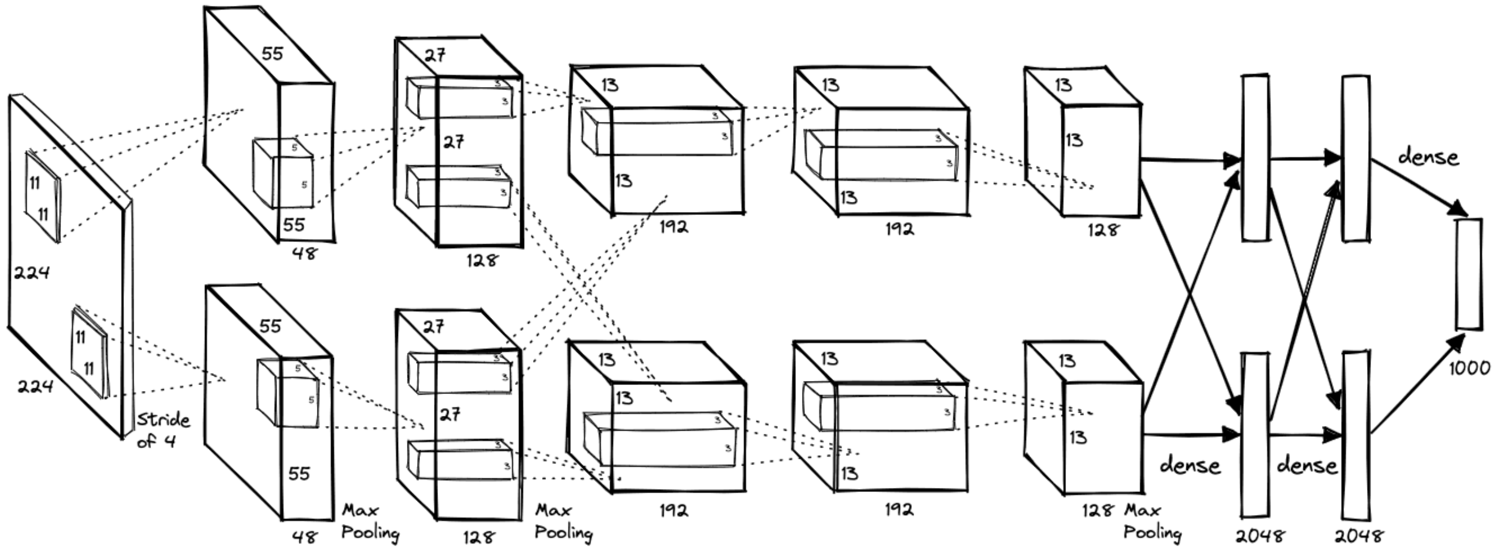
Deng, Jia, et al. "ImageNet: A large-scale hierarchical image database." CVPR 2009

- 1000 image classes
- 1.2M training images, 100k validation

ImageNet


Easiest classes
red fox (100)
hen-of-the-woods (100)
ibex (100)
goldfinch (100)
flat-coated retriever (100)
tiger (100)
hamster (100)
porcupine (100)
stingray (100)
Blenheim spaniel (100)
Hardest classes
muzzle (71)
hatchet (68)
water bottle (68)
velvet (68)
loupe (66)
hook (66)
spotlight (66)
ladle (65)
restaurant (64)
letter opener (59)

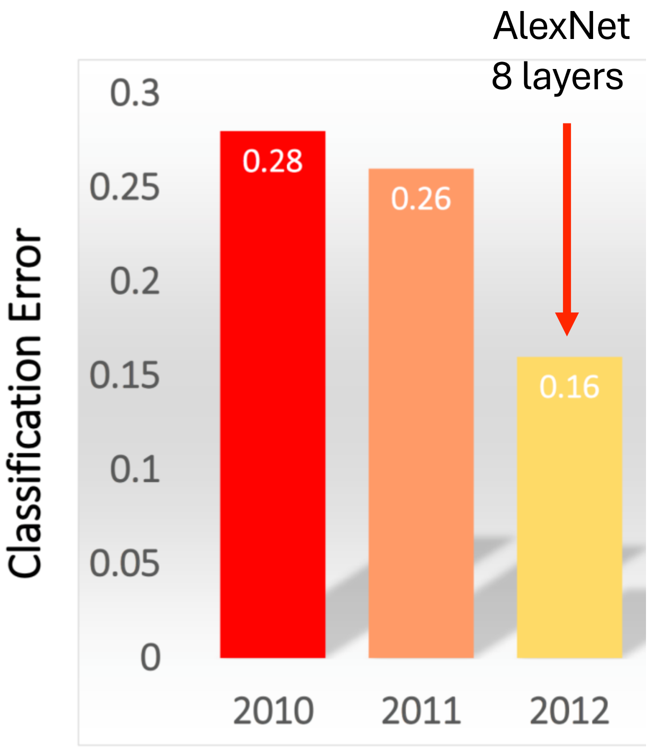
AlexNet [2012]



- Input = 224x224
- First Layer = 11x11 Conv
- 8 layers
- 60M parameters
- Used 2 Nvidia GTX 580 3GB

Alex Krizhevsky et al, ImageNet classification with deep convolutional neural networks, NIPS, 2012

ImageNet



AlexNet
8 layers
Classification Error
0.3
0.25
0.2
0.15
0.1
0.05
0
0.28
0.26
0.16
2010
2011
2012

VGGNet [2013]

large filters
shallow (8 layers)

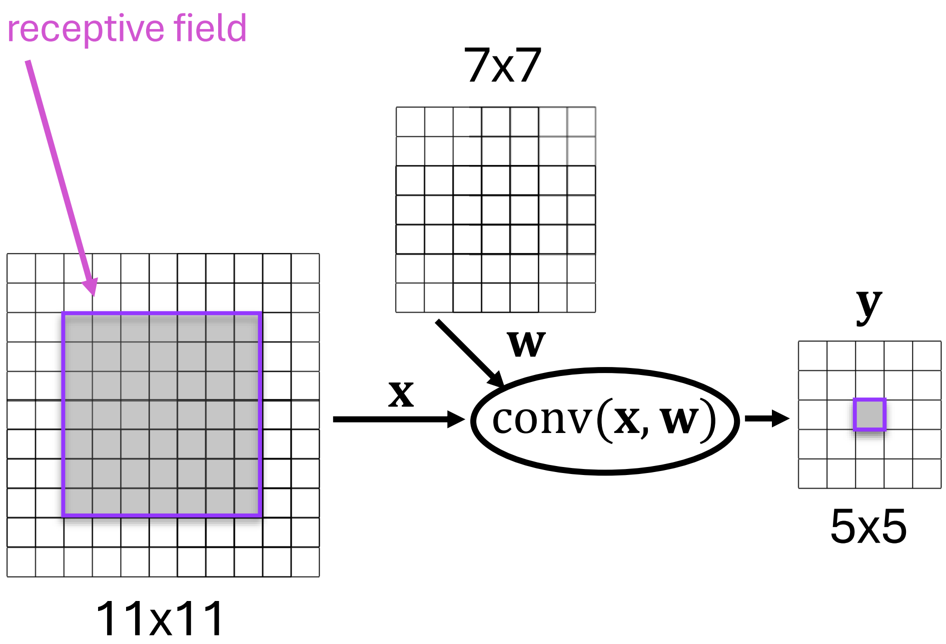
small filters
deeper (19 layers)

- 19 layers
- 138M parameters

Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition.", arXiv 2014

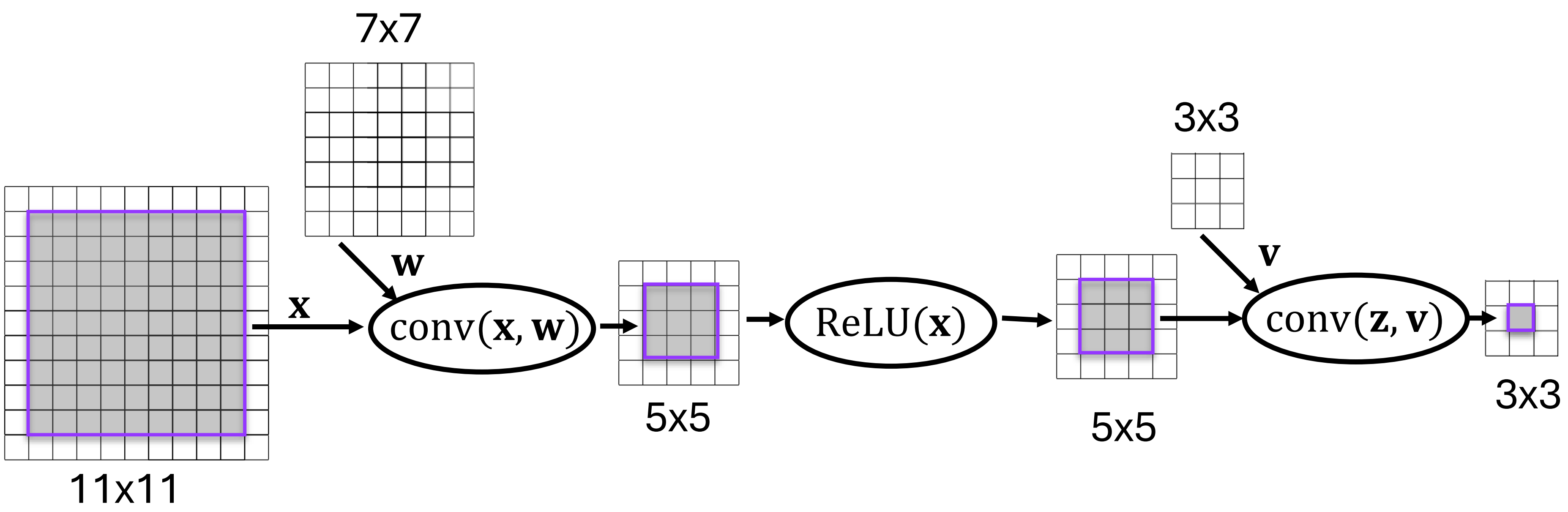
Receptive field

- Receptive field = area in the image whose values affect given cell (neuron)



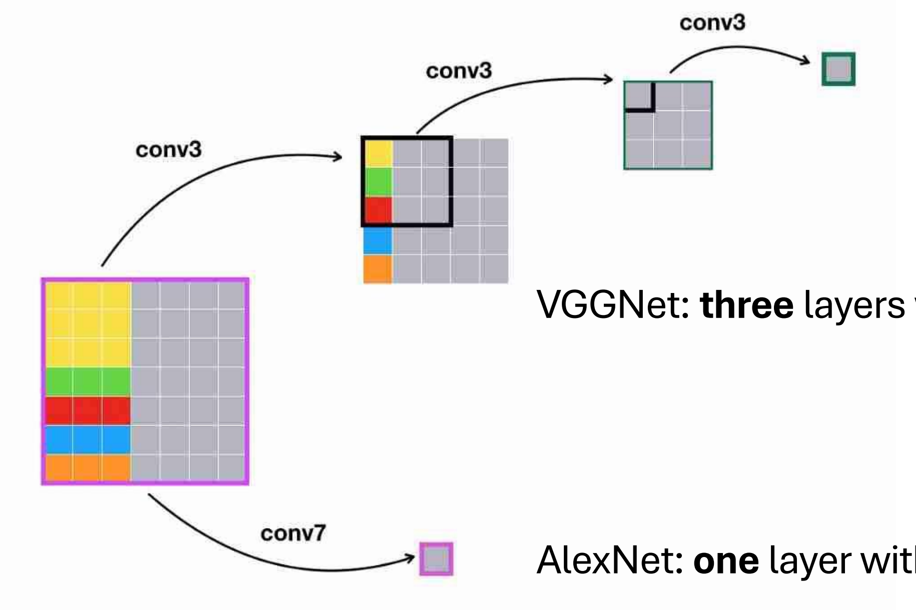
Receptive field

- Receptive field = area in the image whose values affected selected cell (neuron)



VGGNet [2013]

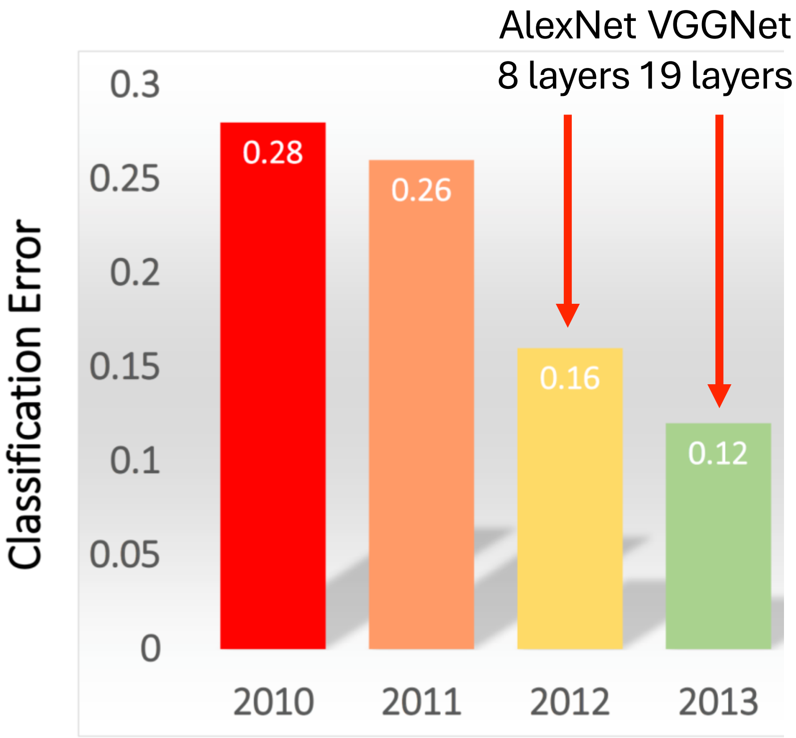
- VGGNet has **the same receptive field** with **less parameters**



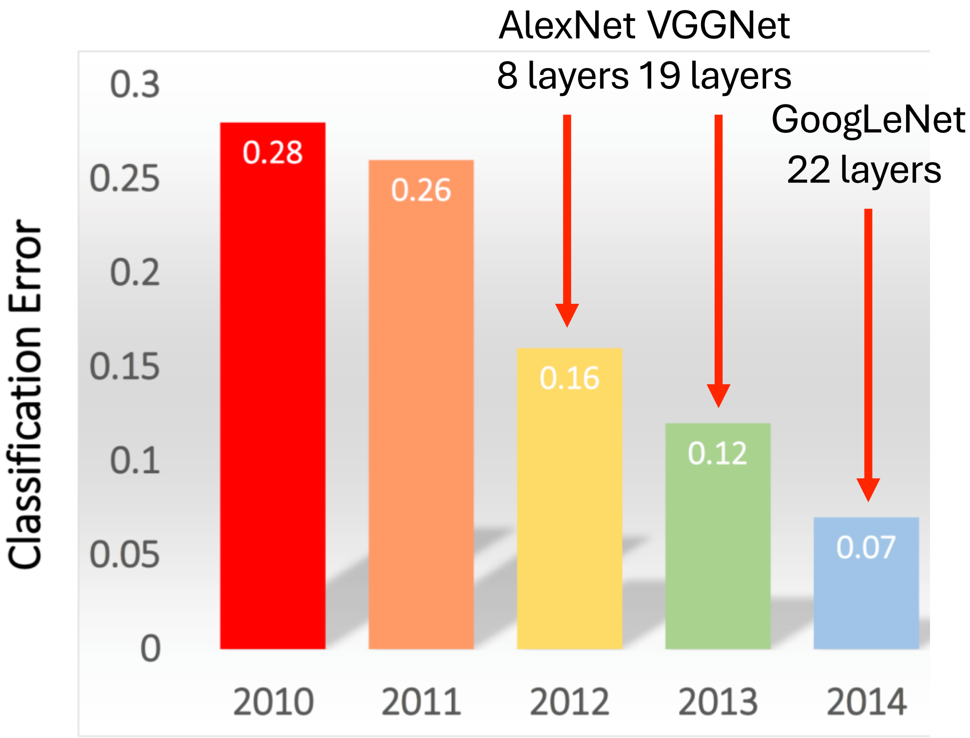
VGGNet: **three** layers with **3x3** filters (3x9+3 params)

AlexNet: **one** layer with **7x7** filter (49+1 params)

ImageNet

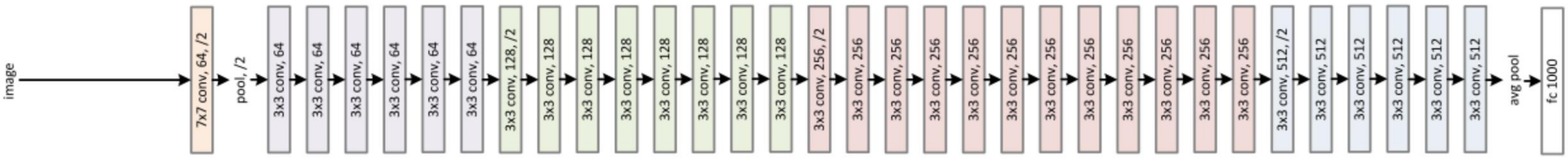

AlexNet VGGNet
8 layers 19 layers
Classification Error
0.3
0.25
0.2
0.15
0.1
0.05
0
0.28
0.26
0.16
0.12
2010
2011
2012
2013

ImageNet



AlexNet VGGNet
8 layers 19 layers
GoogLeNet
22 layers
Classification Error
0.3
0.25
0.2
0.15
0.1
0.05
0
0.28
0.26
0.16
0.12
0.07
2010
2011
2012
2013
2014

ResNet [2015]



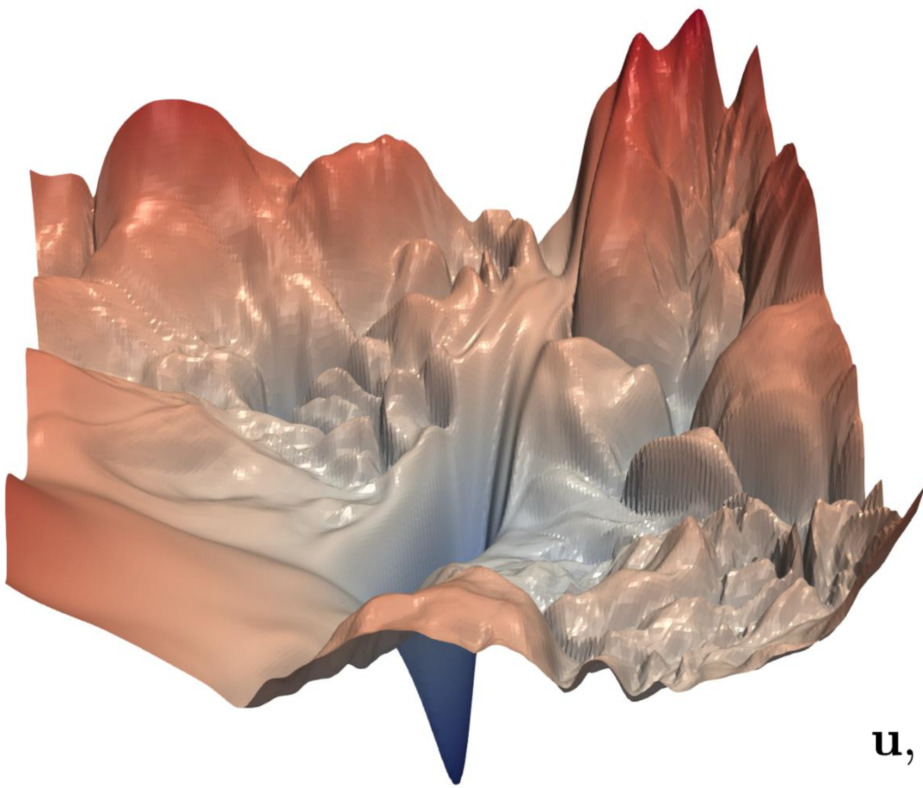
- Deeper architectures had higher training errors

- Is it overfitting?
- No overfitting, but vanishing gradients !

He et al. "Going Deeper with Convolutions", CVPR 2015

ResNet [2015]

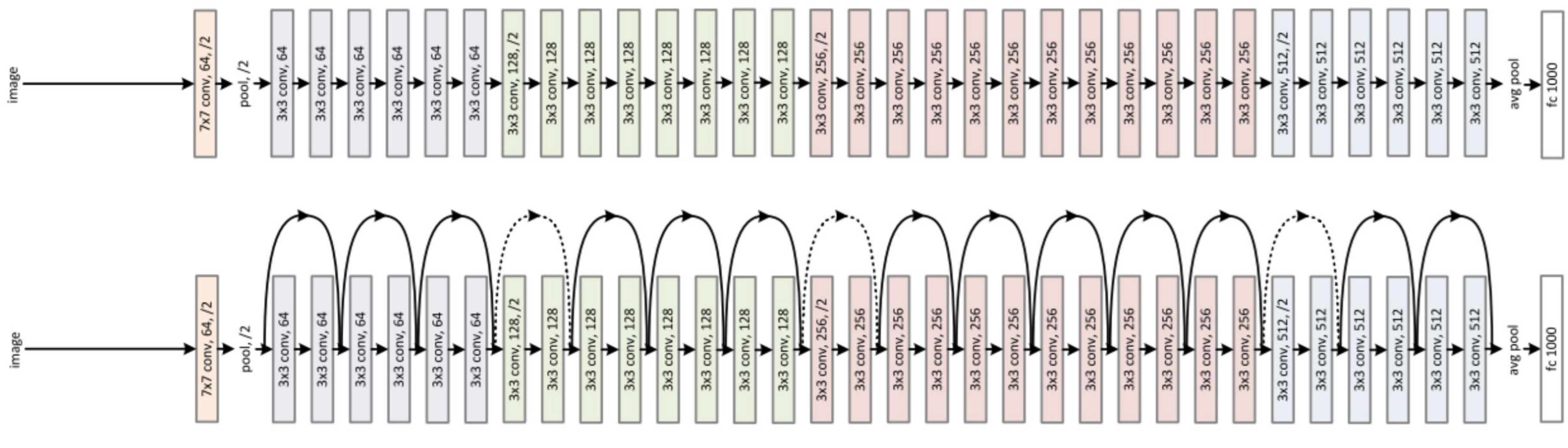
$f(\alpha, \beta) = \mathcal{L}(\mathbf{w}^* + \alpha \mathbf{u} + \beta \mathbf{v})$ for randomly chosen (and normalized) directions



$\mathbf{u}, \mathbf{v}$

Li, Hao, et al. "Visualizing the loss landscape of neural nets.", NIPS 2018

ResNet [2015]

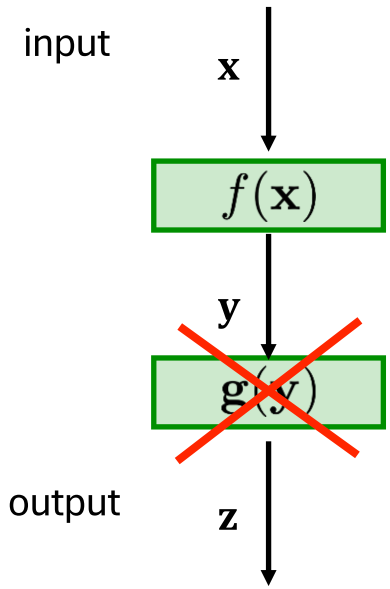


- ResNet **adds skip connections** to prevent vanishing gradients
- Allows training of very deep networks (e.g. ResNet-152)

He et al. "Going Deeper with Convolutions", CVPR 2015

ResNet [2015]

forward pass

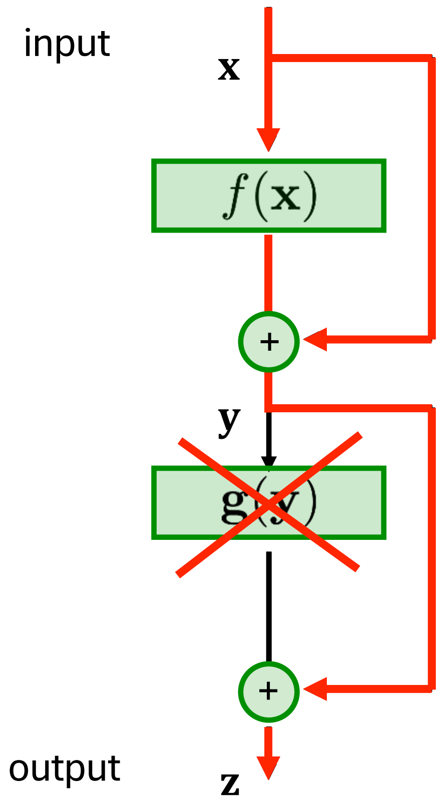


backward pass

gradient $$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \boxed{\underbrace{\frac{\partial \mathbf{z}}{\partial \mathbf{y}}}_{\approx \mathbf{0}} \underbrace{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}}_{\approx \mathbf{0}}} \approx \mathbf{0}$$

if any local gradient is zero

ResNet [2015]



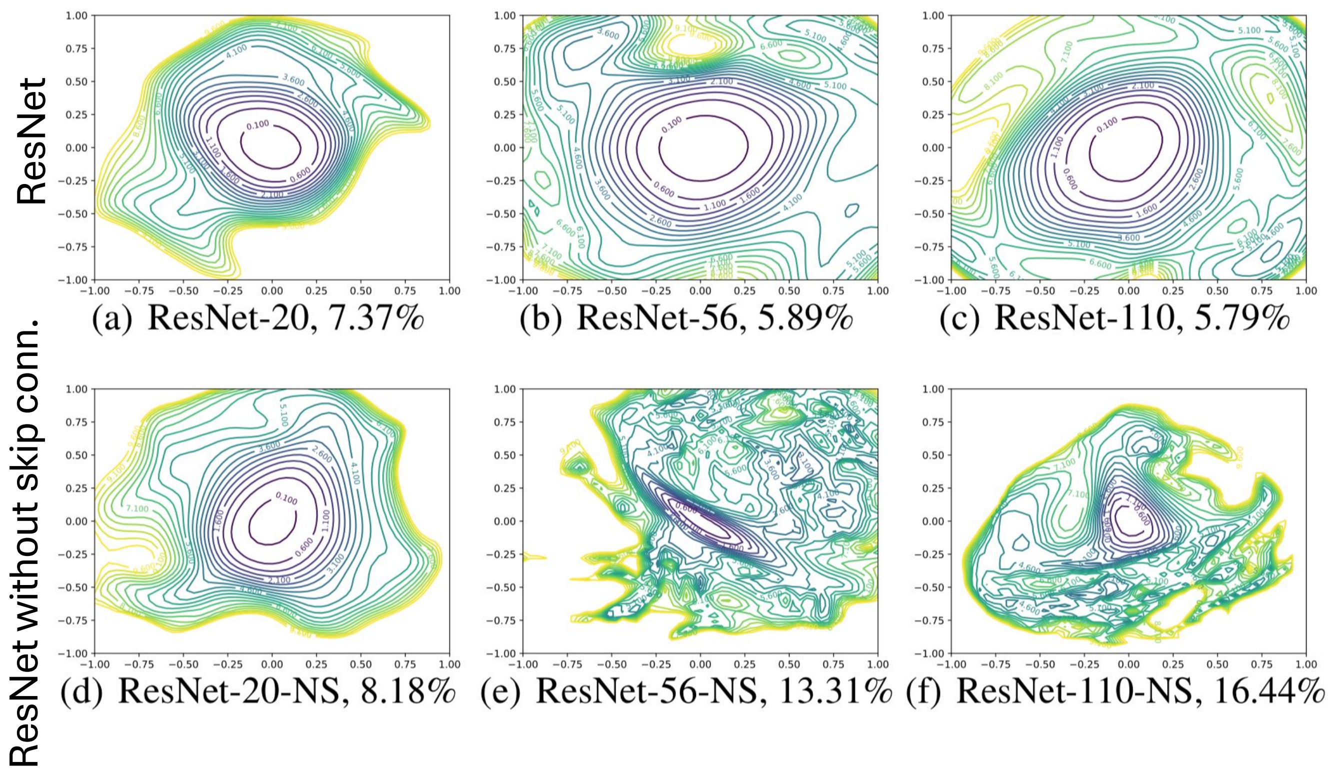
gradient $\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \left(\boxed{\frac{\partial \mathbf{z}}{\partial \mathbf{y}}} + 1\right)\left(\boxed{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}} + 1\right) \neq \mathbf{0}$

$\approx \mathbf{0}$ $\approx \mathbf{0}$

if any local gradient is zero

the gradient can still flow through another path

ResNet [2015]



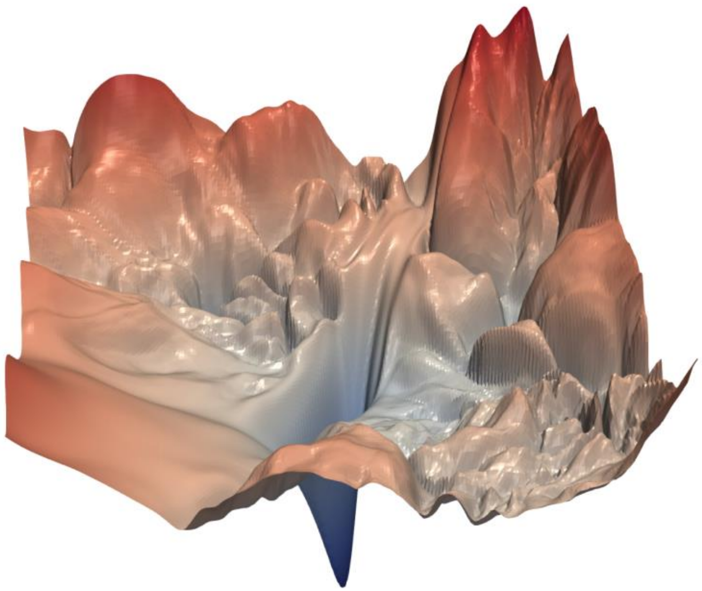
(a) ResNet-20, 7.37% (b) ResNet-56, 5.89% (c) ResNet-110, 5.79%

(d) ResNet-20-NS, 8.18% (e) ResNet-56-NS, 13.31% (f) ResNet-110-NS, 16.44%

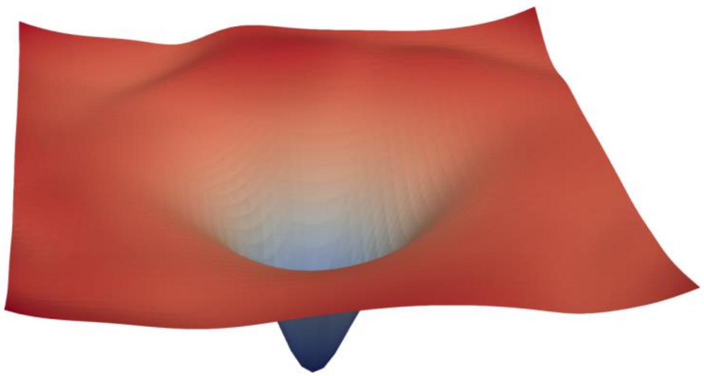
He et al. “Going Deeper with Convolutions”, CVPR 2015

ResNet [2015]

$f(\alpha, \beta) = \mathcal{L}(\mathbf{w}^* + \alpha\mathbf{u} + \beta\mathbf{v})$ for randomly chosen (and normalized) directions



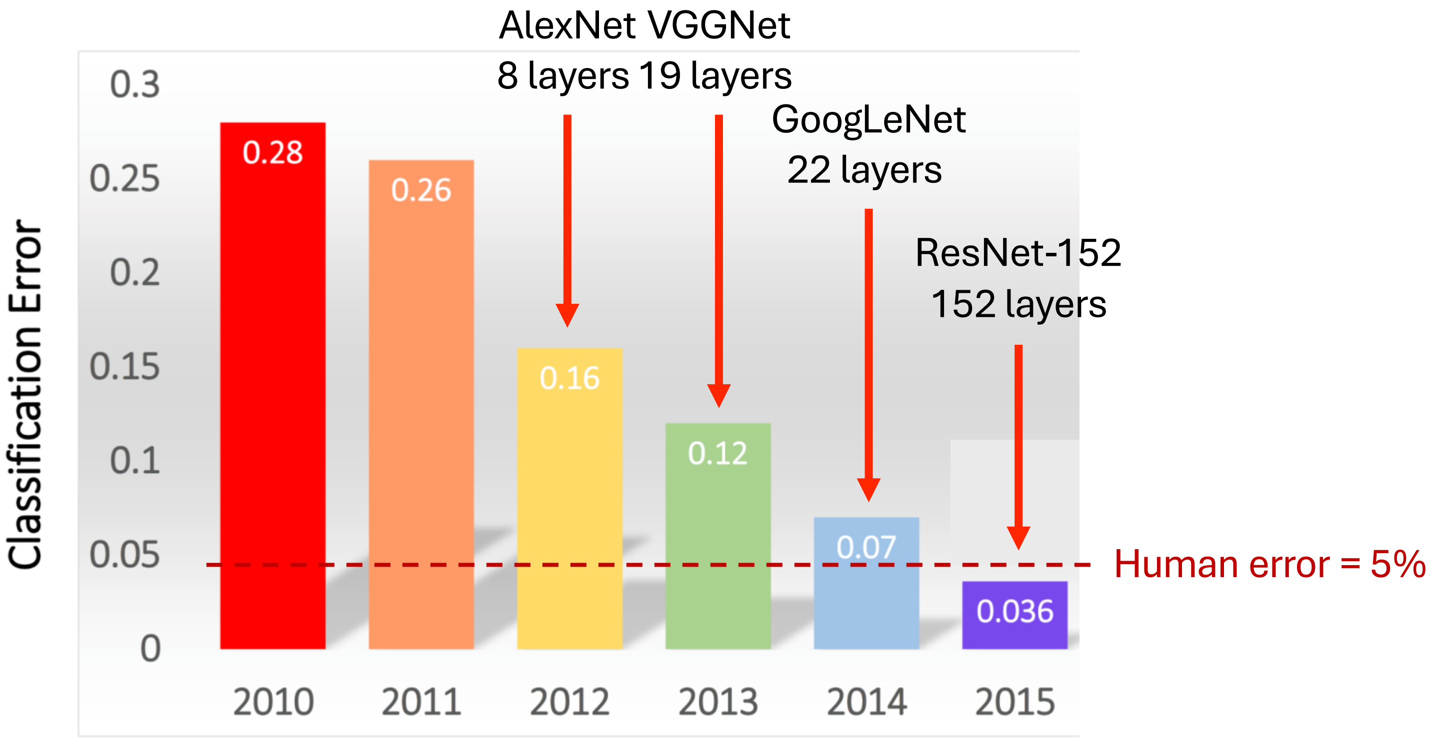
(a) without skip connections



(b) with skip connections

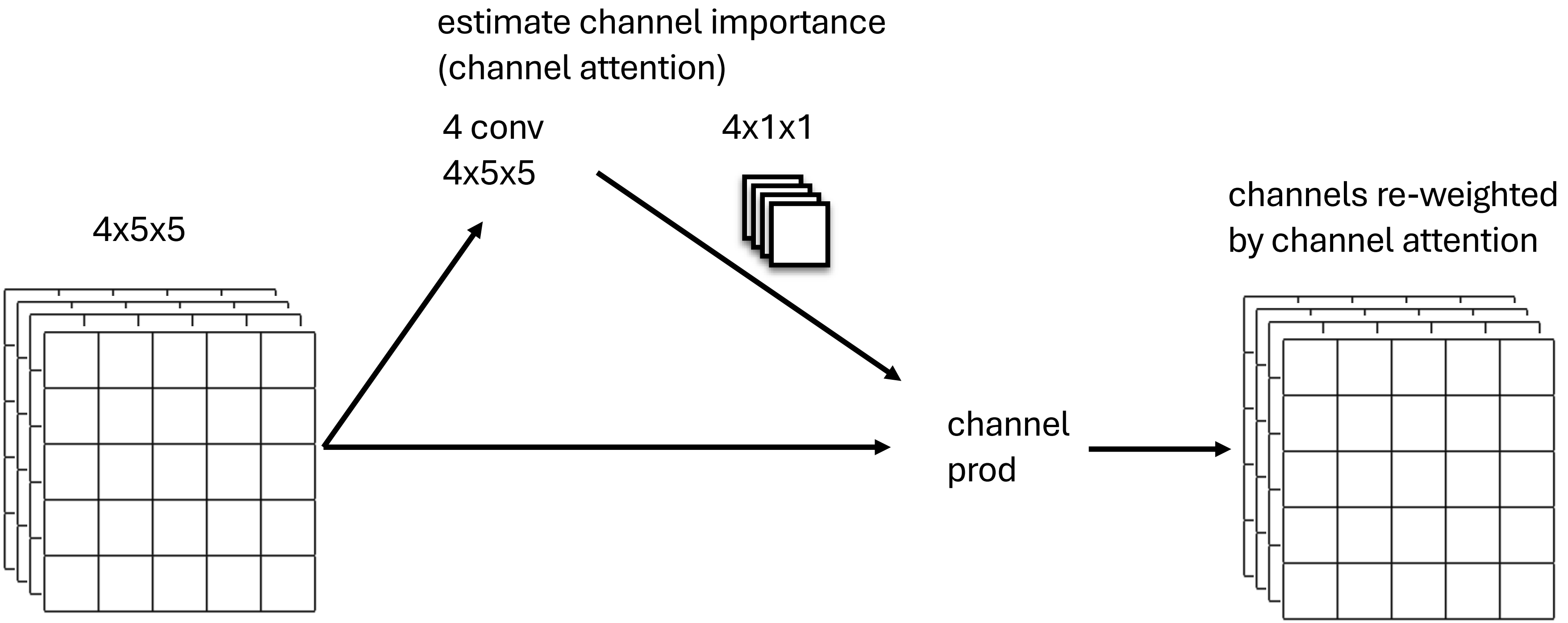
Li, Hao, et al. "Visualizing the loss landscape of neural nets.", NIPS 2018

ImageNet


AlexNet
8 layers
VGGNet
19 layers
GoogLeNet
22 layers
ResNet-152
152 layers
Classification Error
0.3
0.25
0.2
0.15
0.1
0.05
0
0.28
0.26
0.16
0.12
0.07
0.036
2010
2011
2012
2013
2014
2015
Human error = 5%

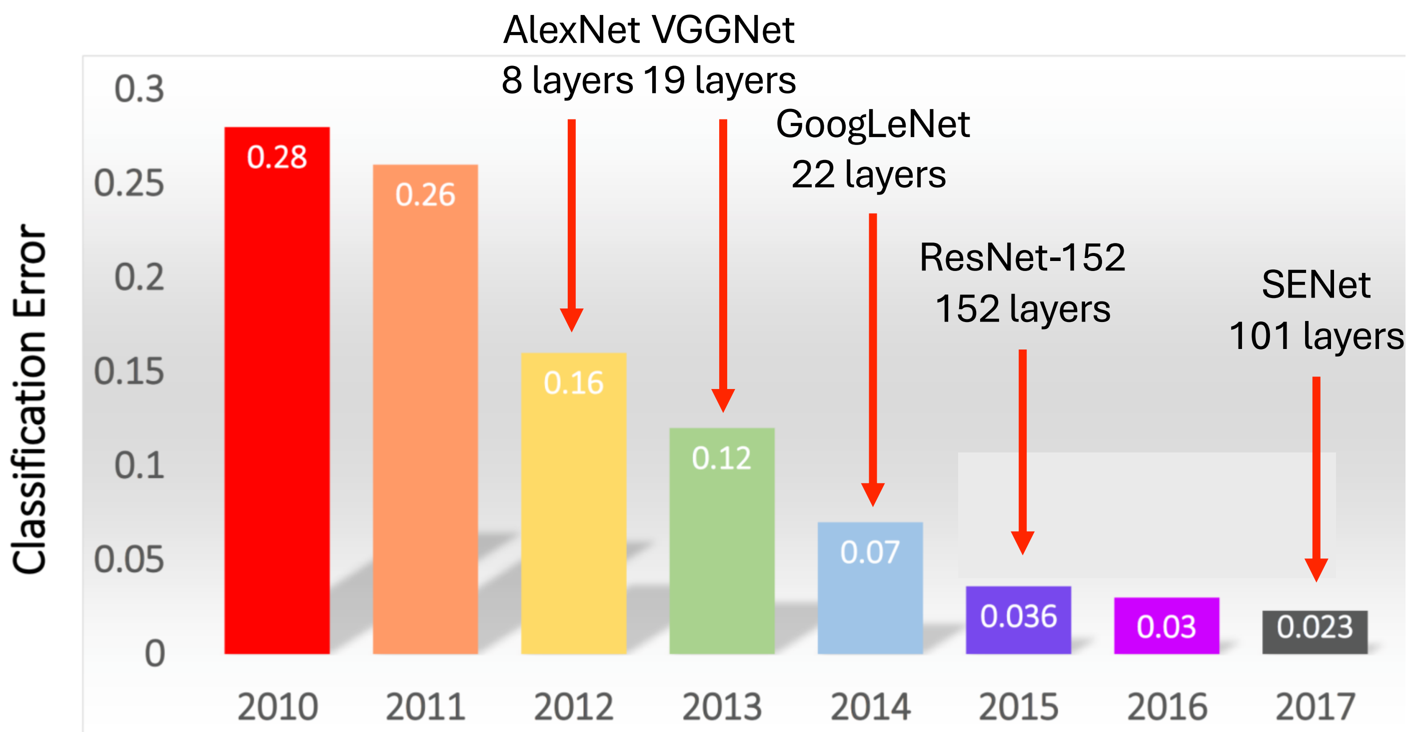
Squeeze and Excitation Networks (SEN)

- Channel attention



Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks.", CVPR 2018

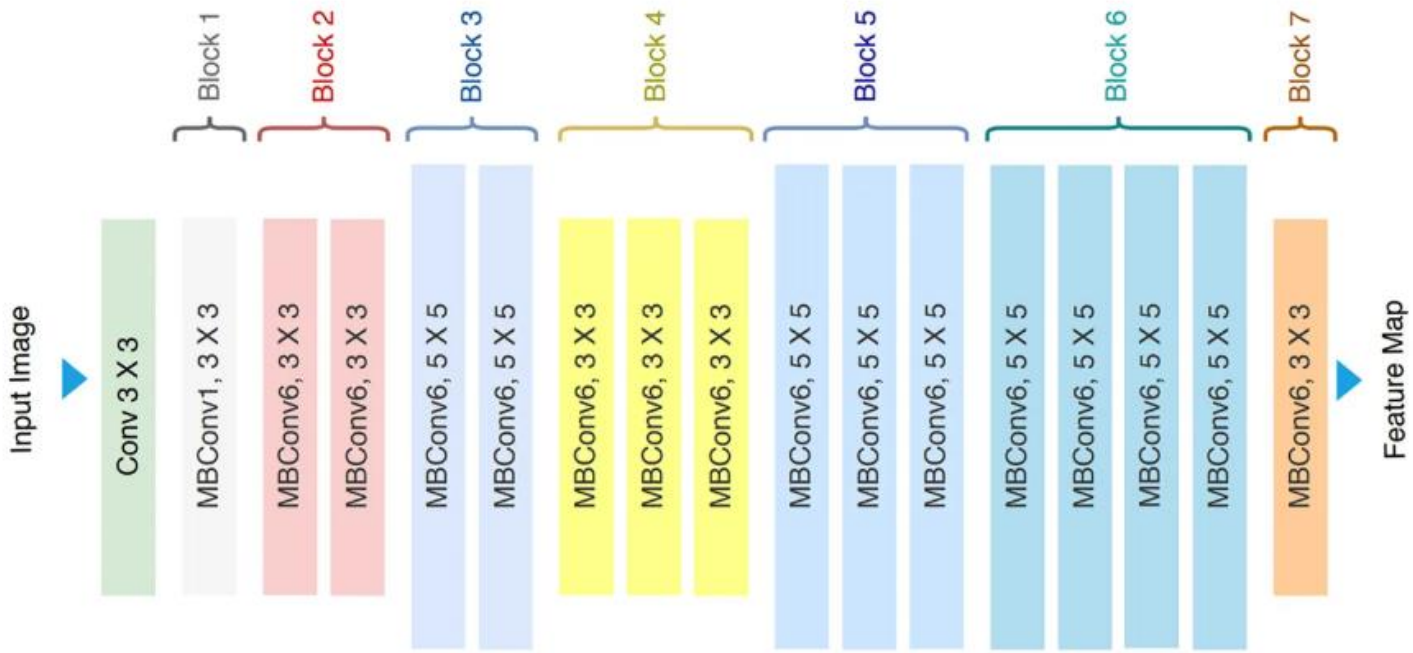
ImageNet



AlexNet VGGNet
8 layers 19 layers
GoogLeNet
22 layers
ResNet-152
152 layers
SENet
101 layers
Classification Error
0.3
0.25
0.2
0.15
0.1
0.05
0
0.28
0.26
0.16
0.12
0.07
0.036
0.03
0.023
2010
2011
2012
2013
2014
2015
2016
2017

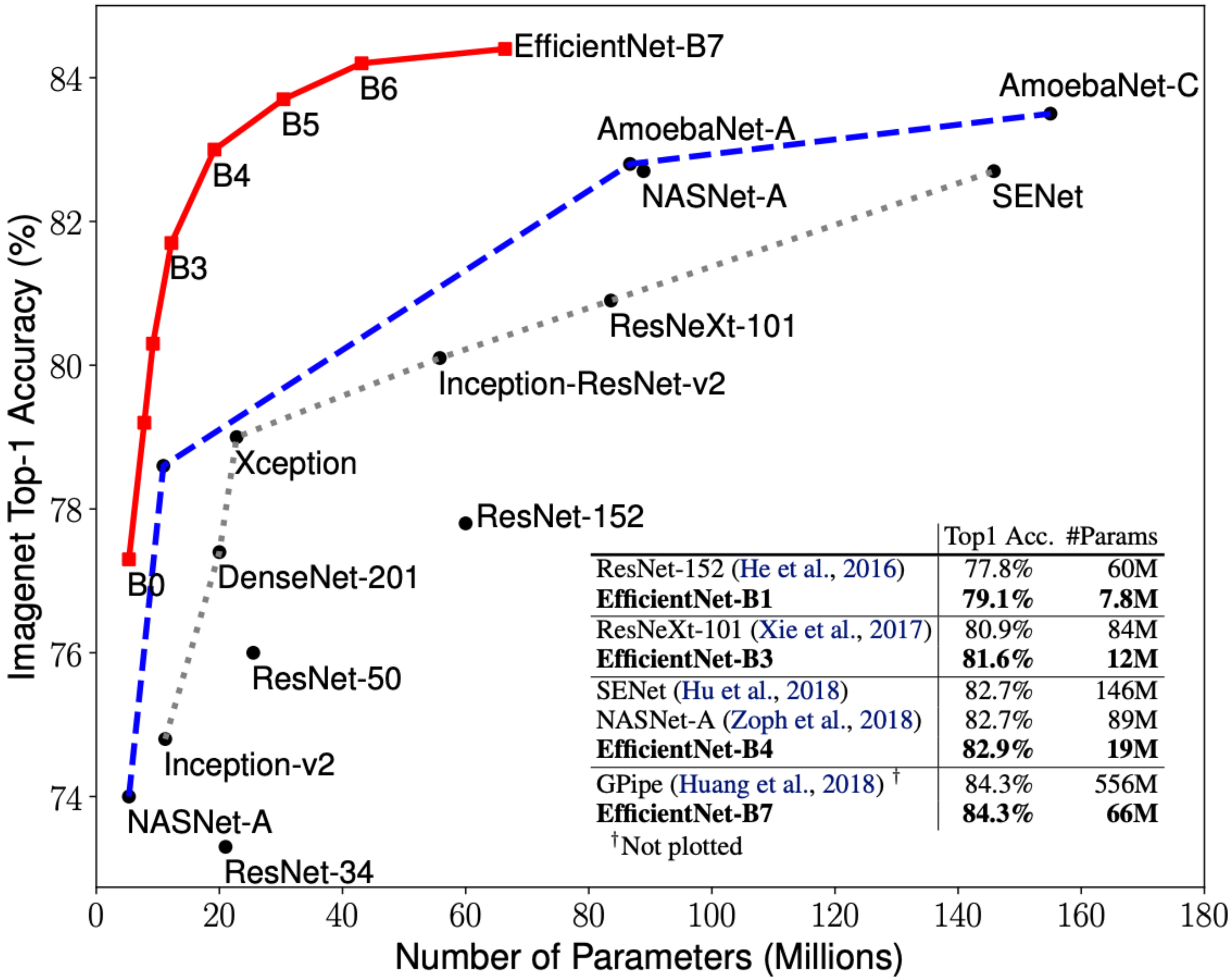
EfficientNet [2019]

- Good balance between number of channels, depth and resolution
- Good accuracy with fewer parameters → faster to train, less overfitting
- **In practice, good first choice as a backbone (EfficientNetV2 [2021])**



Tan, Mingxing, and Q. Efficientnet Le. "Rethinking model scaling for convolutional neural networks." ICML 2019

EfficientNet [2019]



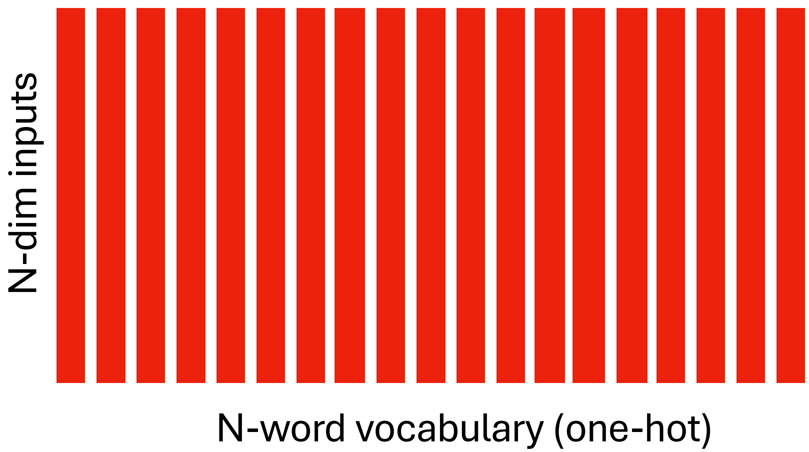
	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**79.1%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.6%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.9%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.3%**	**66M**

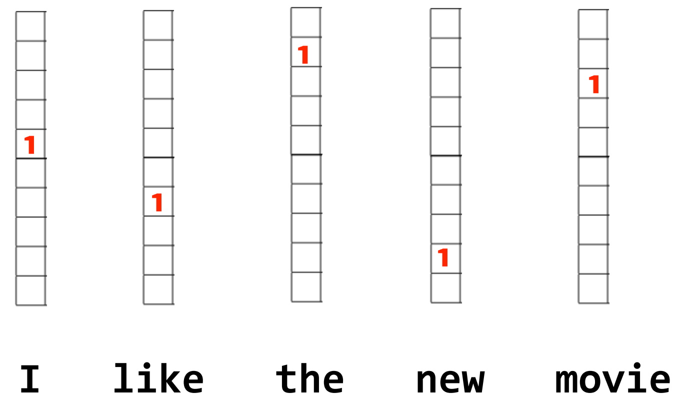
†Not plotted

Tan, Mingxing, and Q. Efficientnet Le. "Rethinking model scaling for convolutional neural networks." ICML 2019

Word embedding

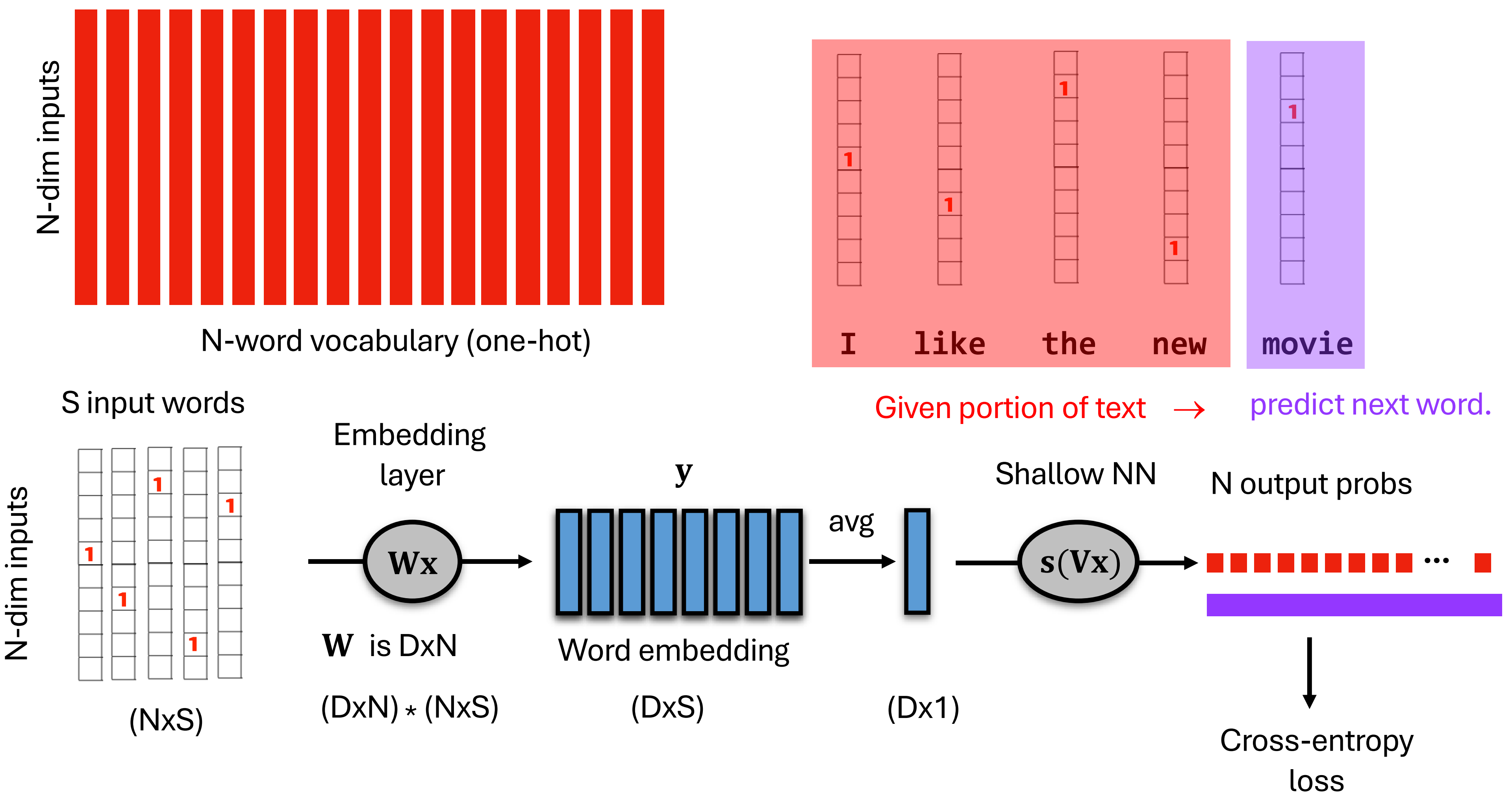
- Word2vec represents words in low-dimensional continuous space





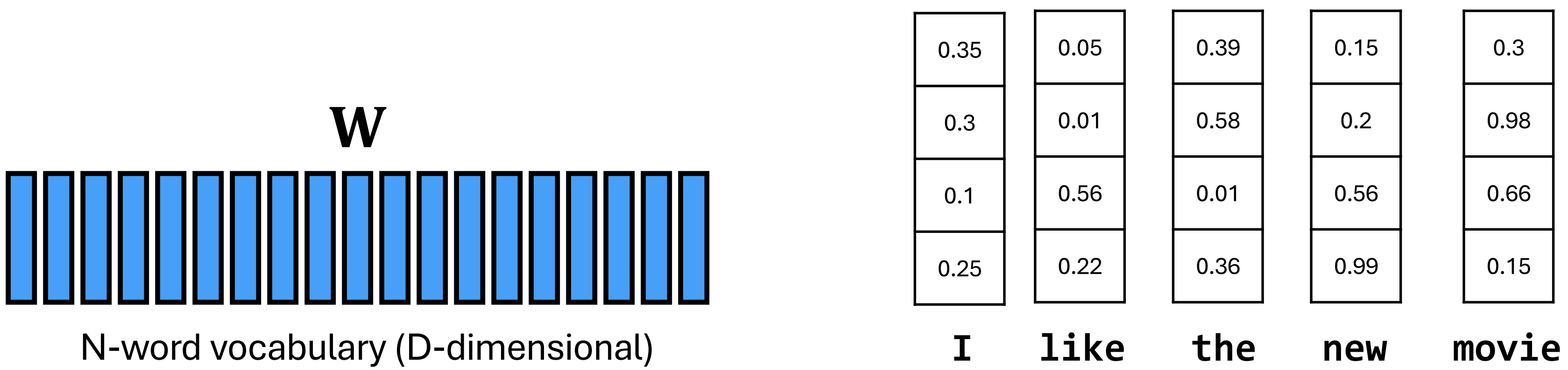
Word embedding

- Word2vec represents words in low-dimensional continuous space



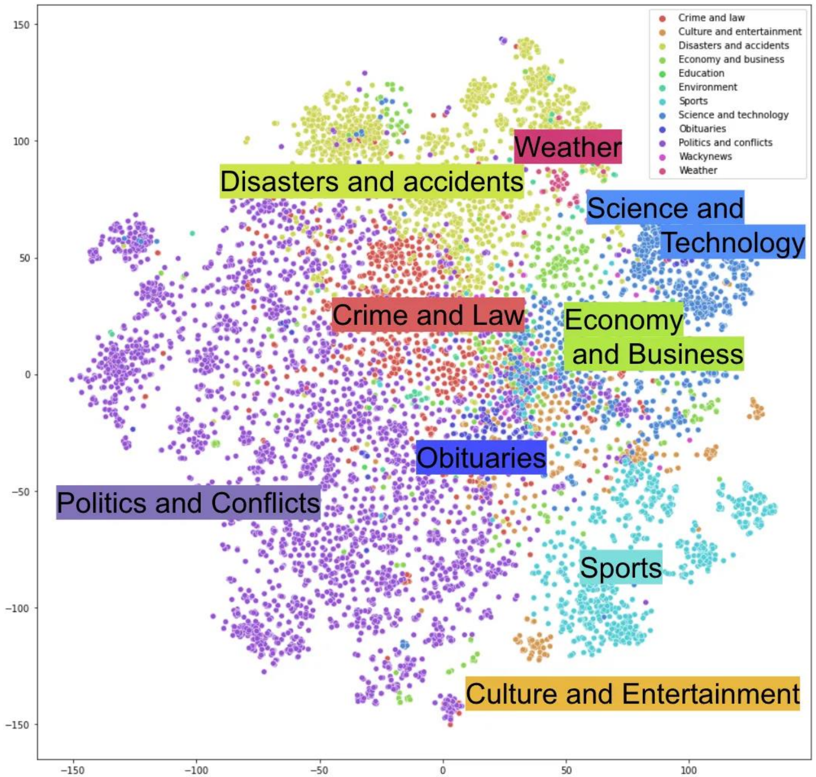
Word embedding

- Word2vec represents words in low-dimensional continuous space



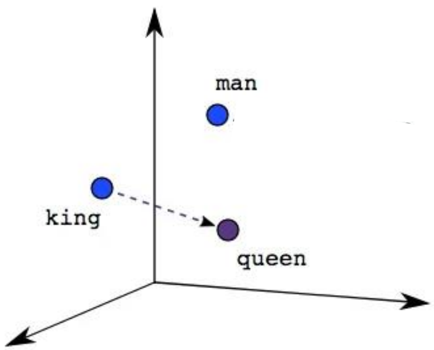
Word embedding

- Word2vec represents words in low-dimensional continuous space



Word embedding

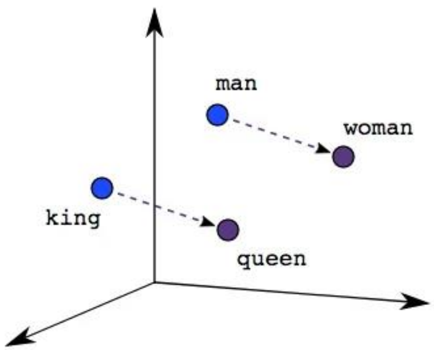
- Word2vec represents words in low-dimensional continuous space



Male-Female

Word embedding

- Word2vec represents words in low-dimensional continuous space

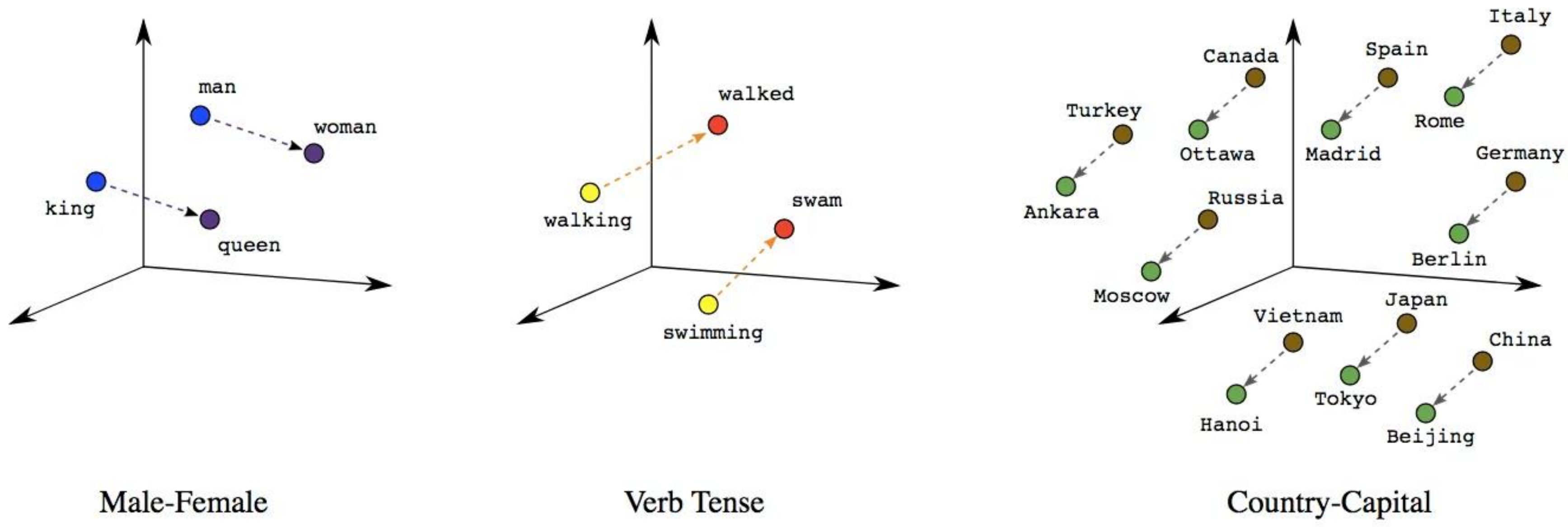


Male-Female

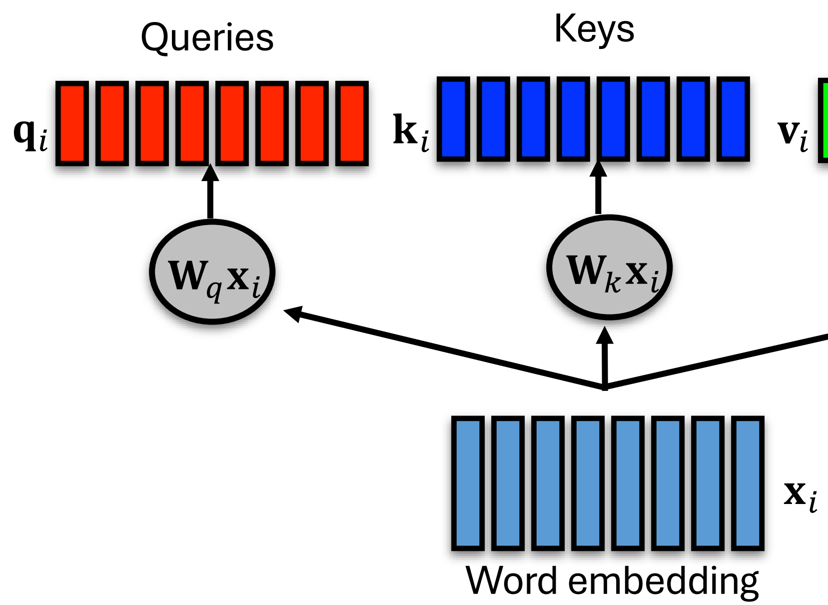
“Word algebra”: `king - man + woman = queen`

Word embedding

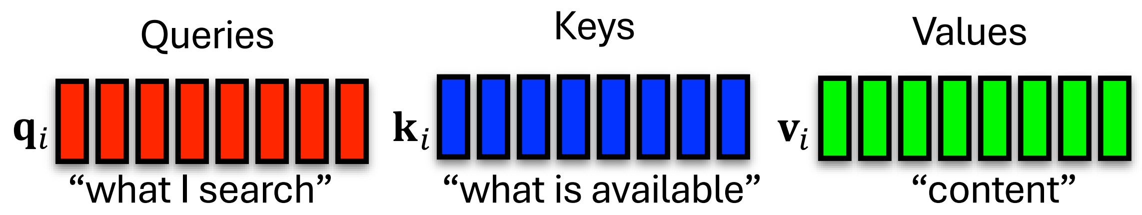
- Word2vec represents words in low-dimensional continuous space



Self-attention

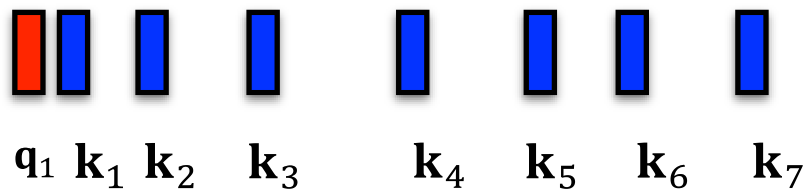


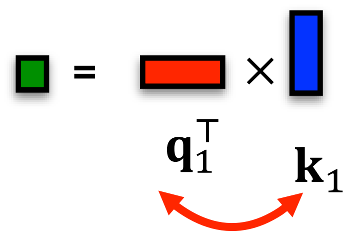
Self-attention

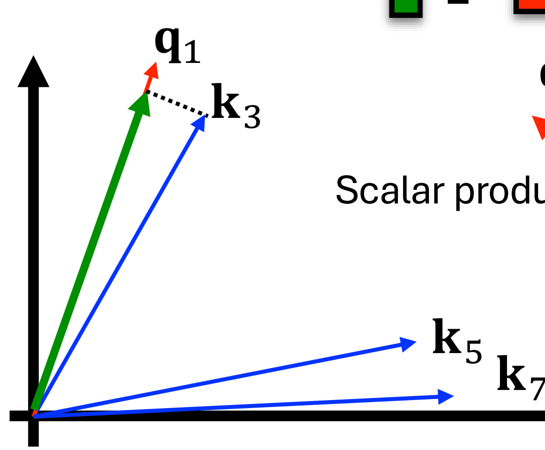


`Karel is teacher and Mario is plumber.`

Which words contribute to meaning of Karel?

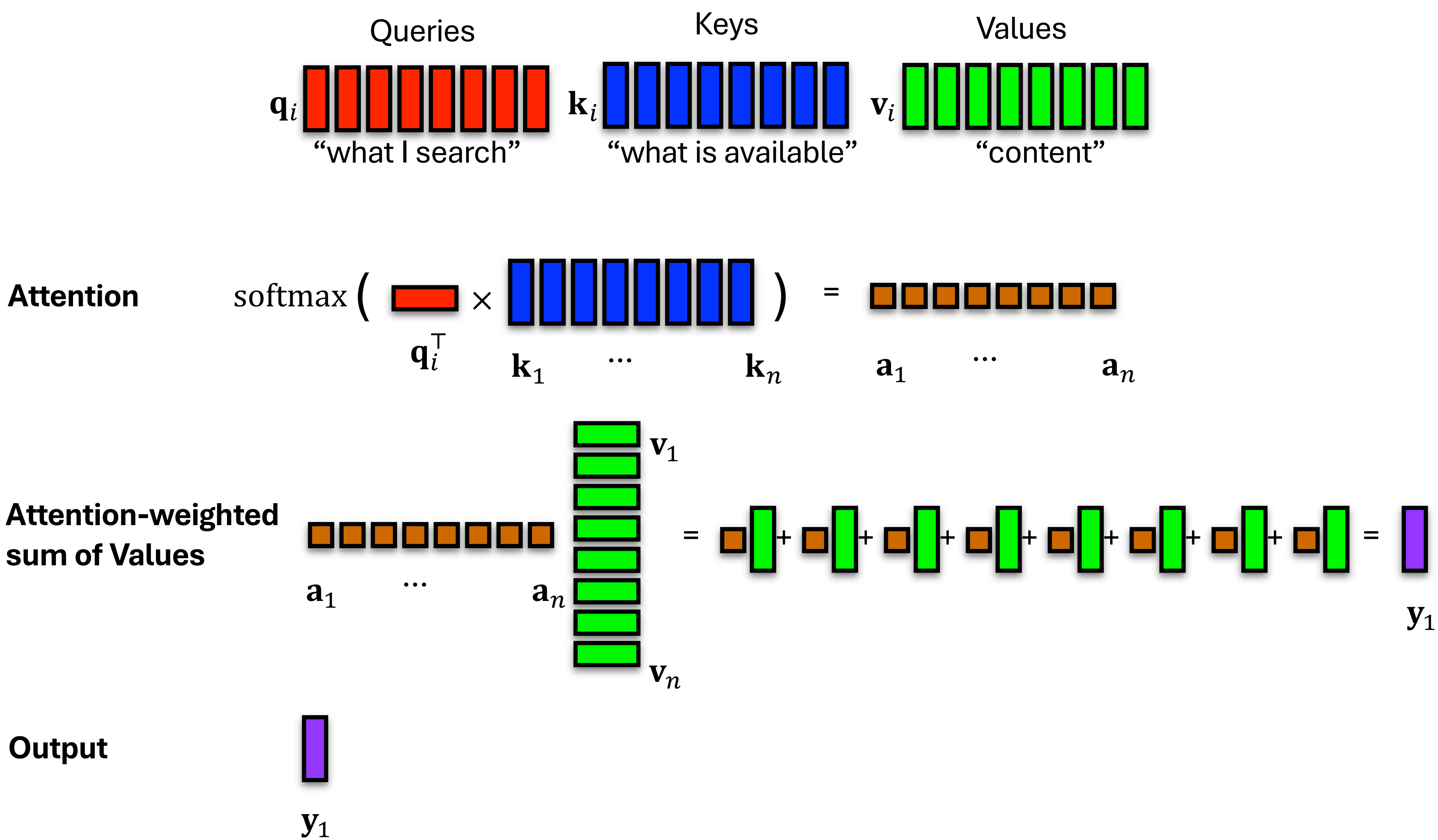




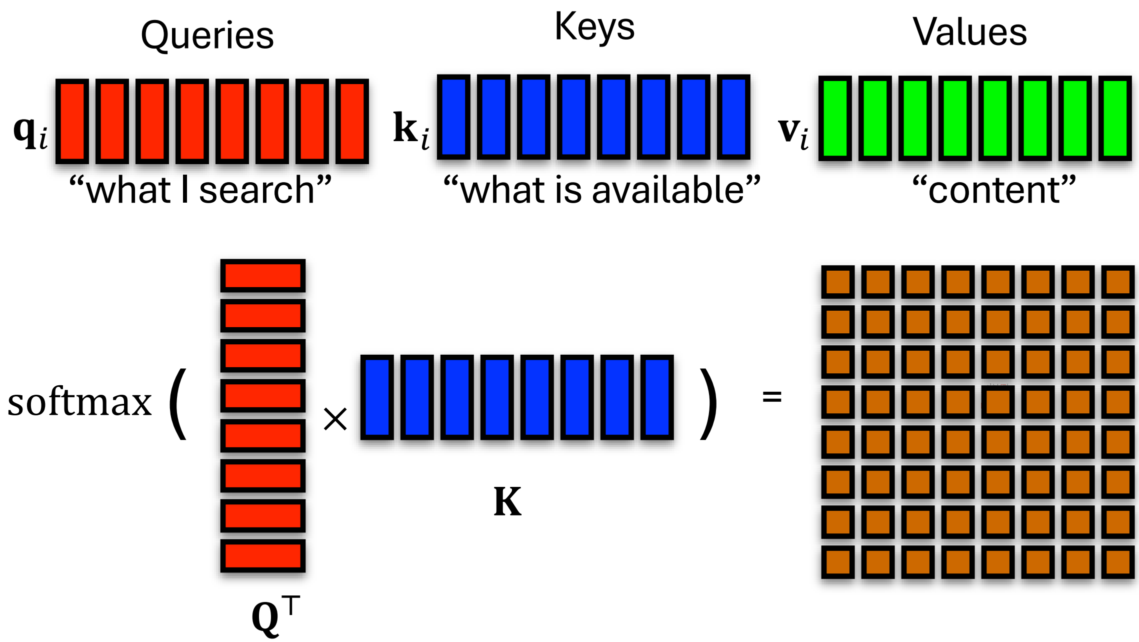


Scalar product measures similarity between vectors.

Self-attention

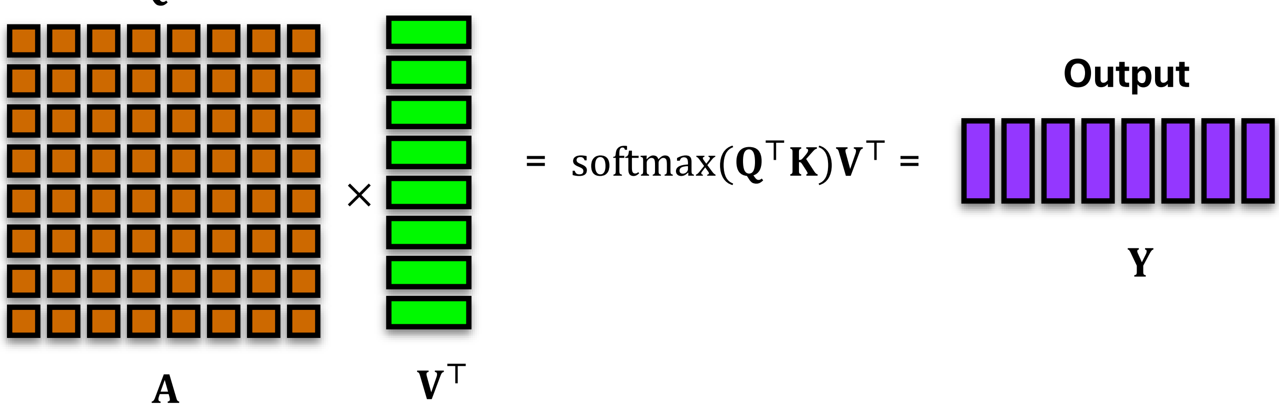


Self-attention

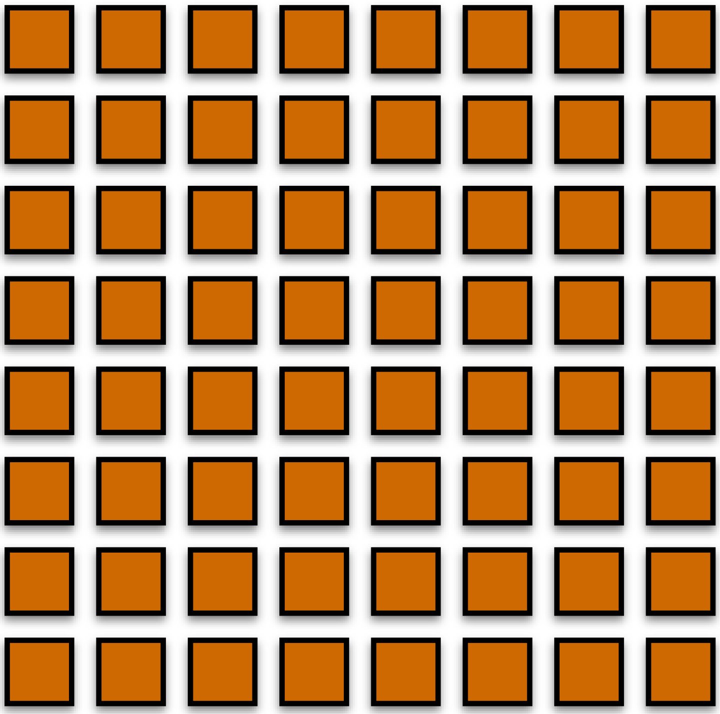


Attention

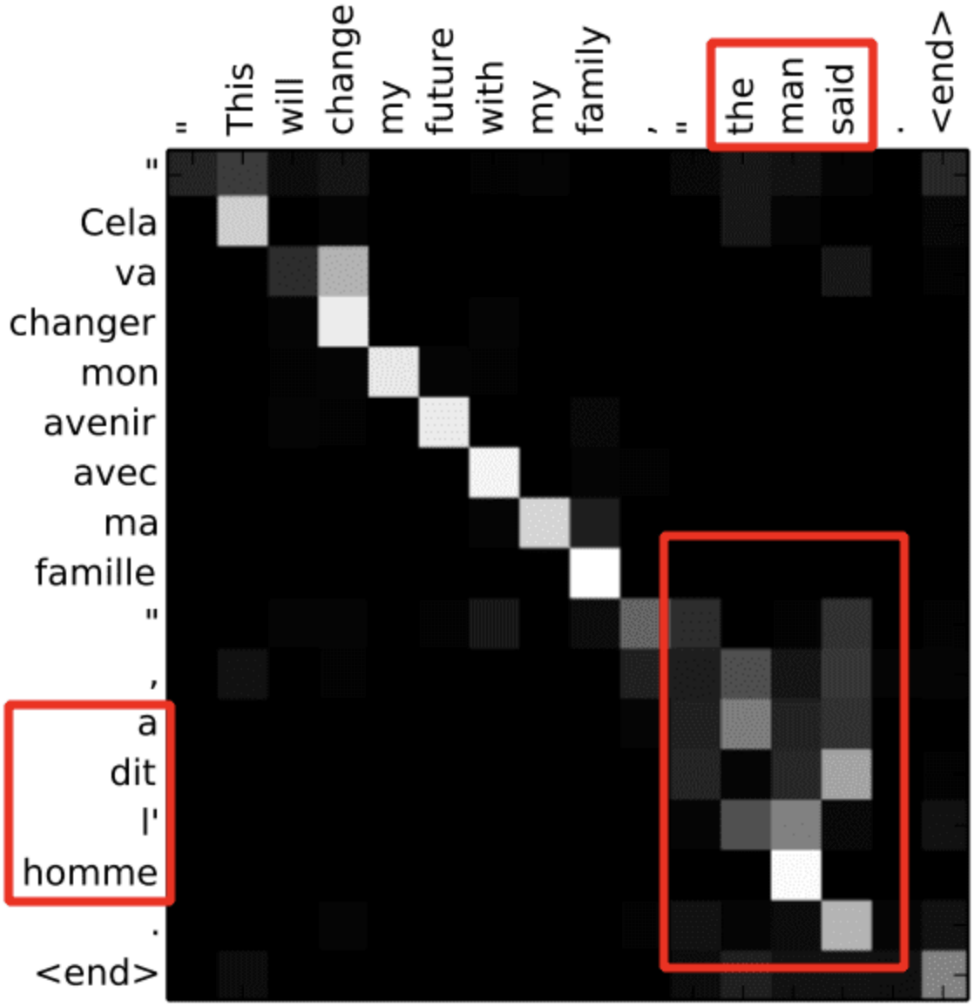
Attention-weighted sum of values



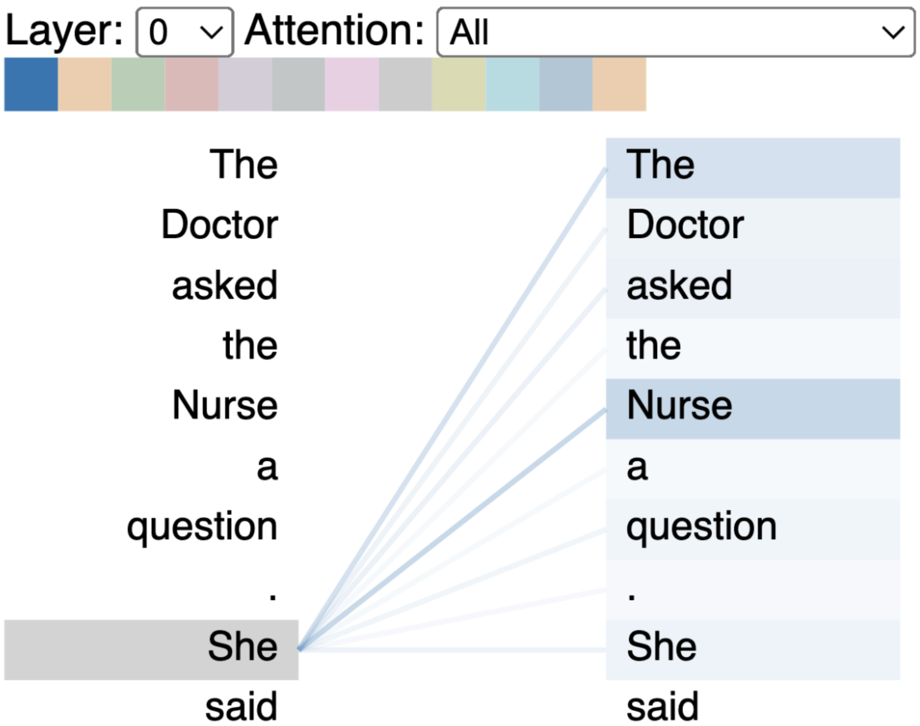
Self-attention



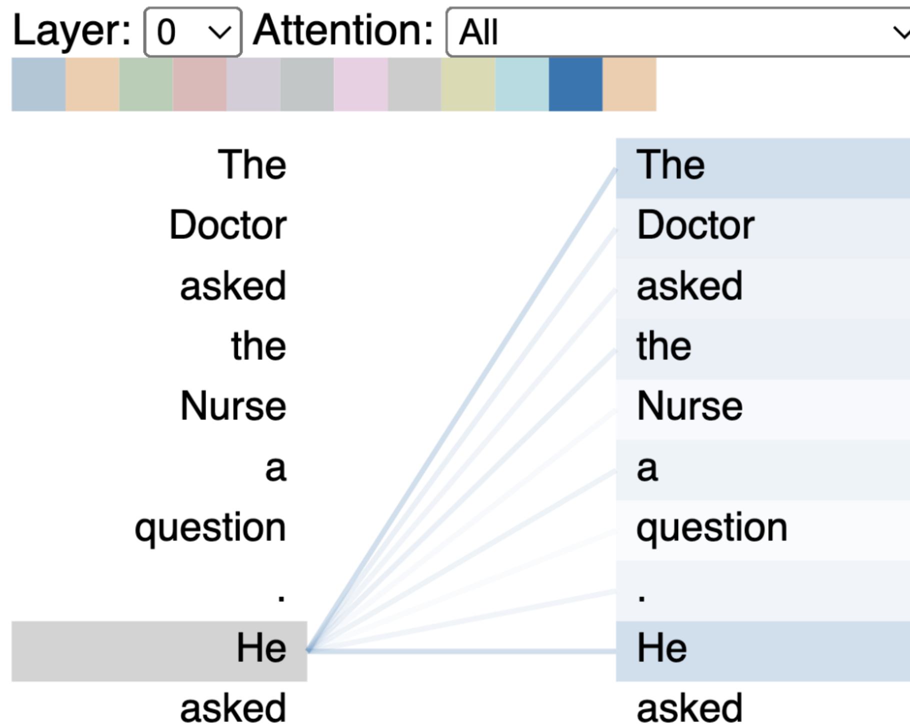
=



Self-attention



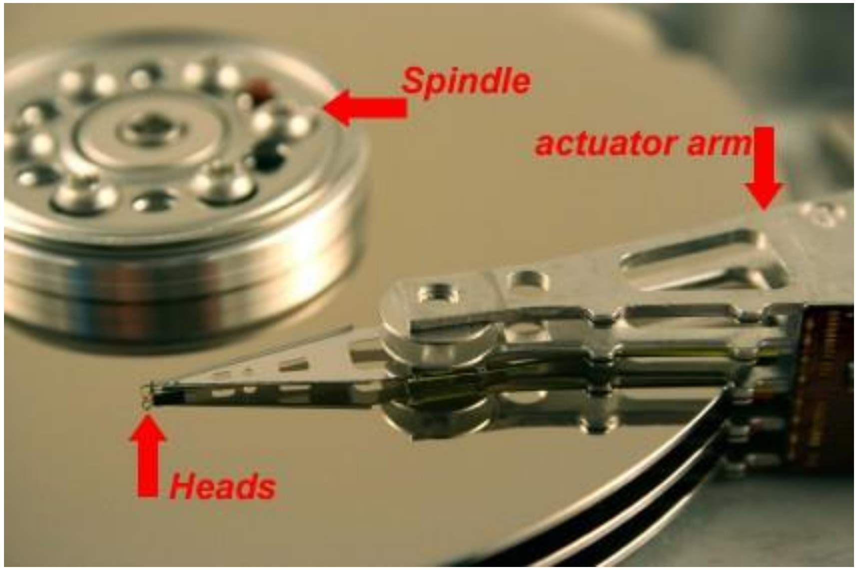
Model assumes “she=nurse”



Model assumes “he=doctor”

Multi-head self-attention (MHSA)

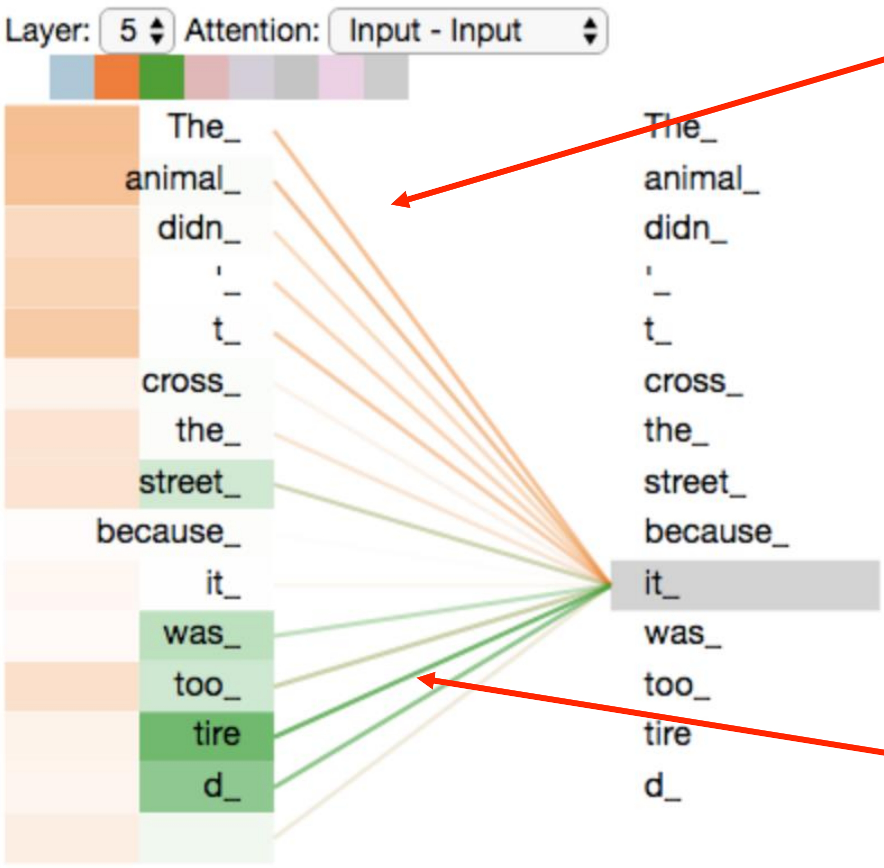
- Self-attention can be applied multiple times on the same sequence in parallel using more **heads**



Multi-head self-attention (MHSA)

`The animal didn’t cross the street because it was too tired.`

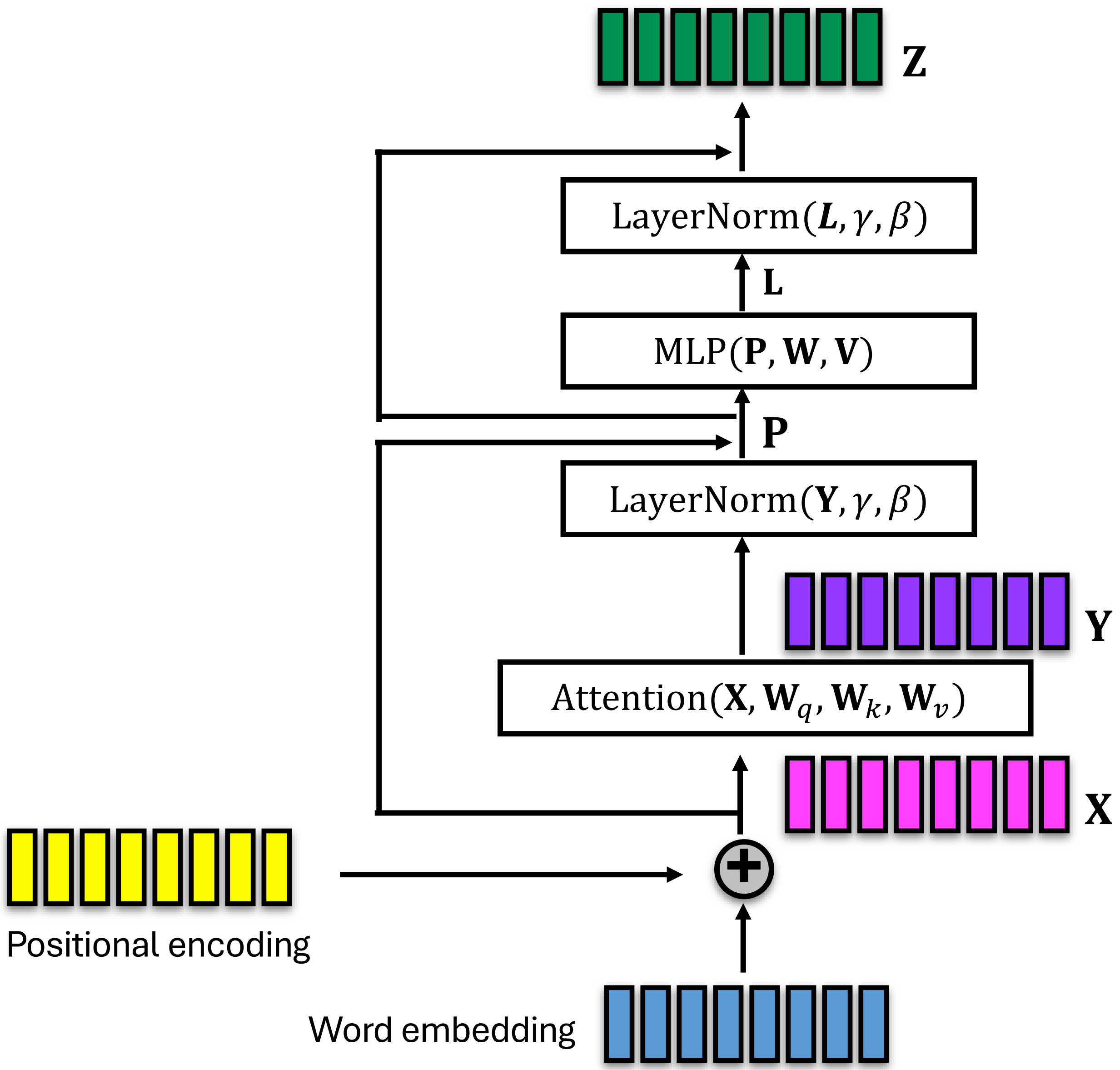
Is “it” = animal vs “it” = street ?



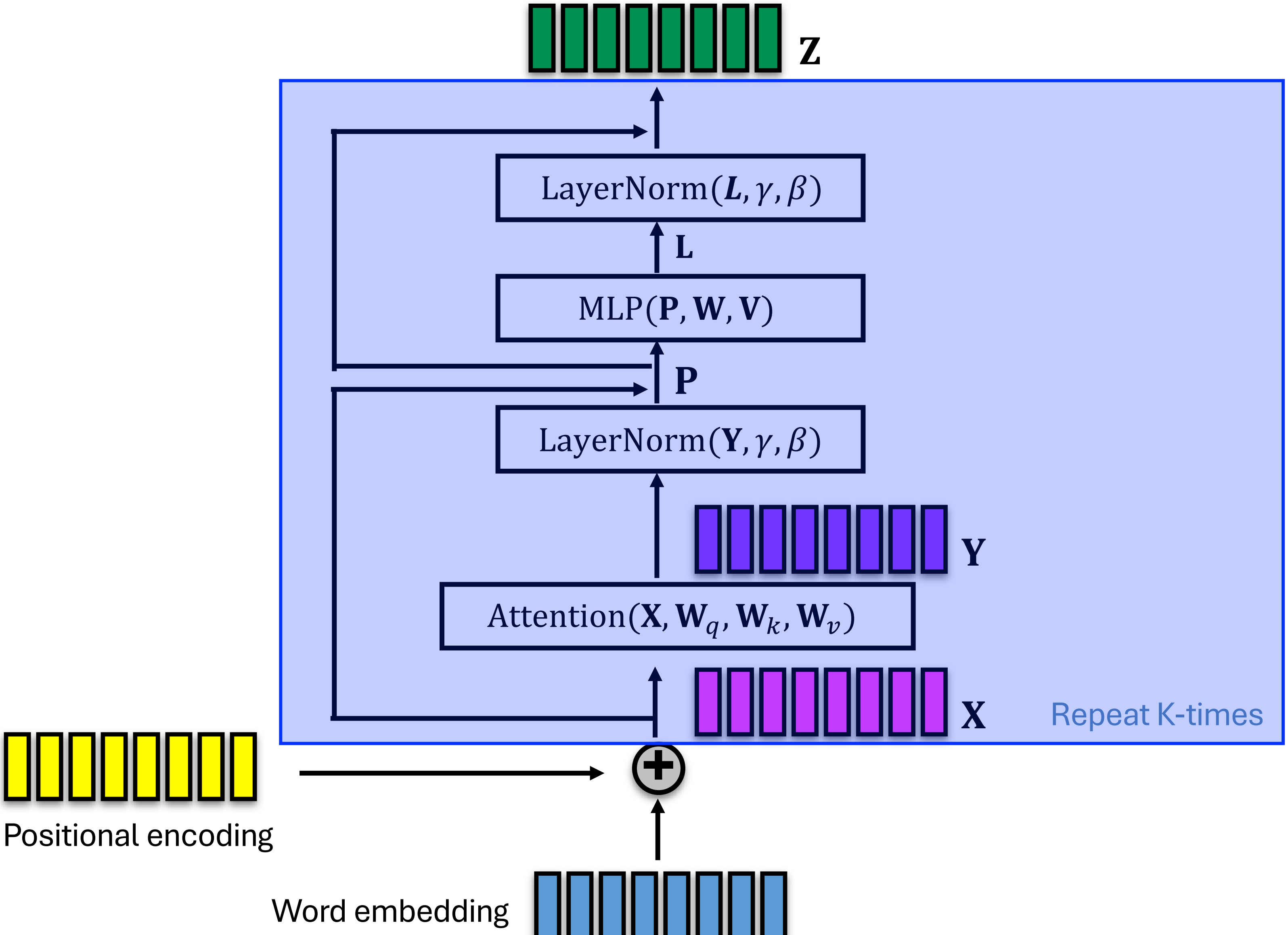
attention weights of **orange** attention head

attention weights of **green** attention head

Transformer

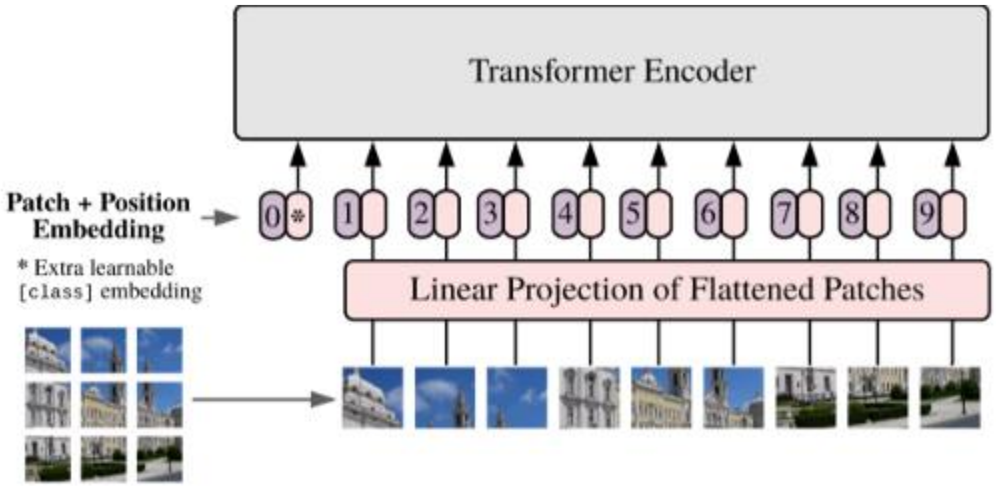


Transformer



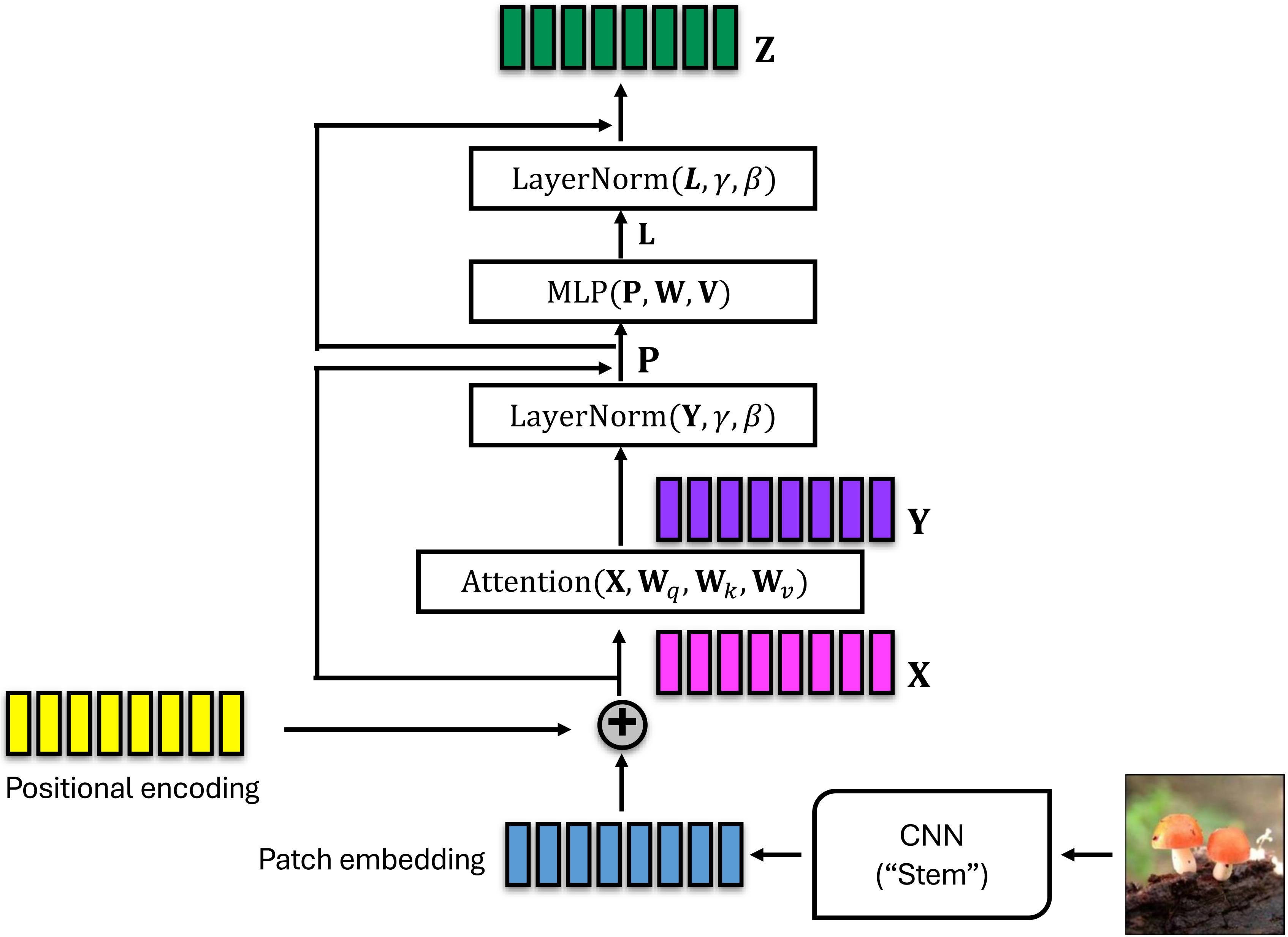
Vision Transformer (ViT)

- Split image into 16x16 patches and treat each patch as a “word”

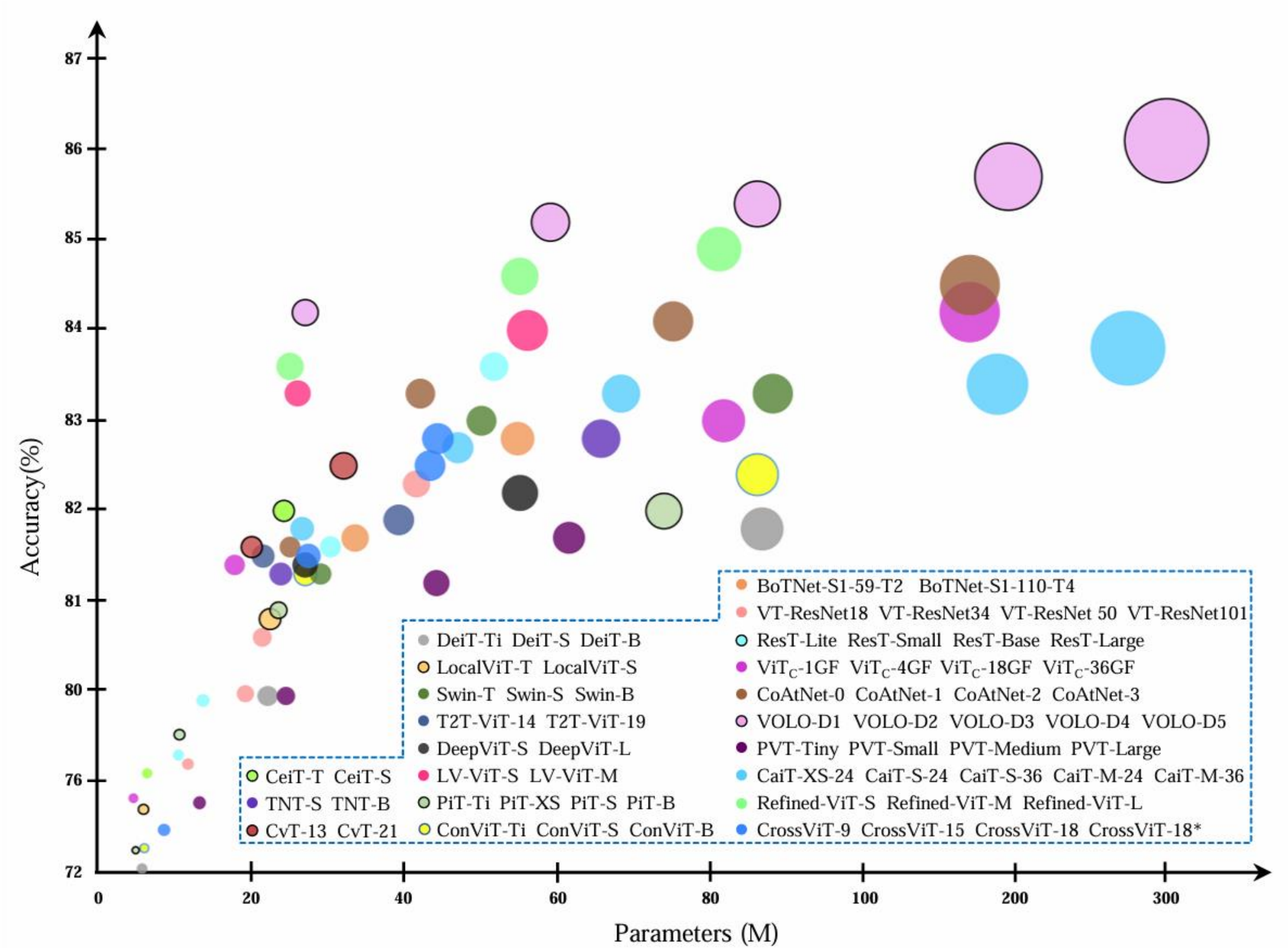


Dosovitskiy, Alexey et al. "An image is worth 16x16 words: Transformers for image recognition at scale.", arXiv 2020

Vision Transformer (ViT)



Vision Transformer (ViT)



Competencies gained for the test

- Vanishing gradient problem, ResNet
- Self-attention, Transformers
- Vision Transformer