Famous Image Classification Architectures

Presented By
Mohammad Hossein
Niki Maleki

Summer 2021

مرکز تحقیقات هوش مصنوعی پارت
هوشمندسازی فرایندهای زندگی

کالج تخصصی هوش مصنوعی پارت

Part2

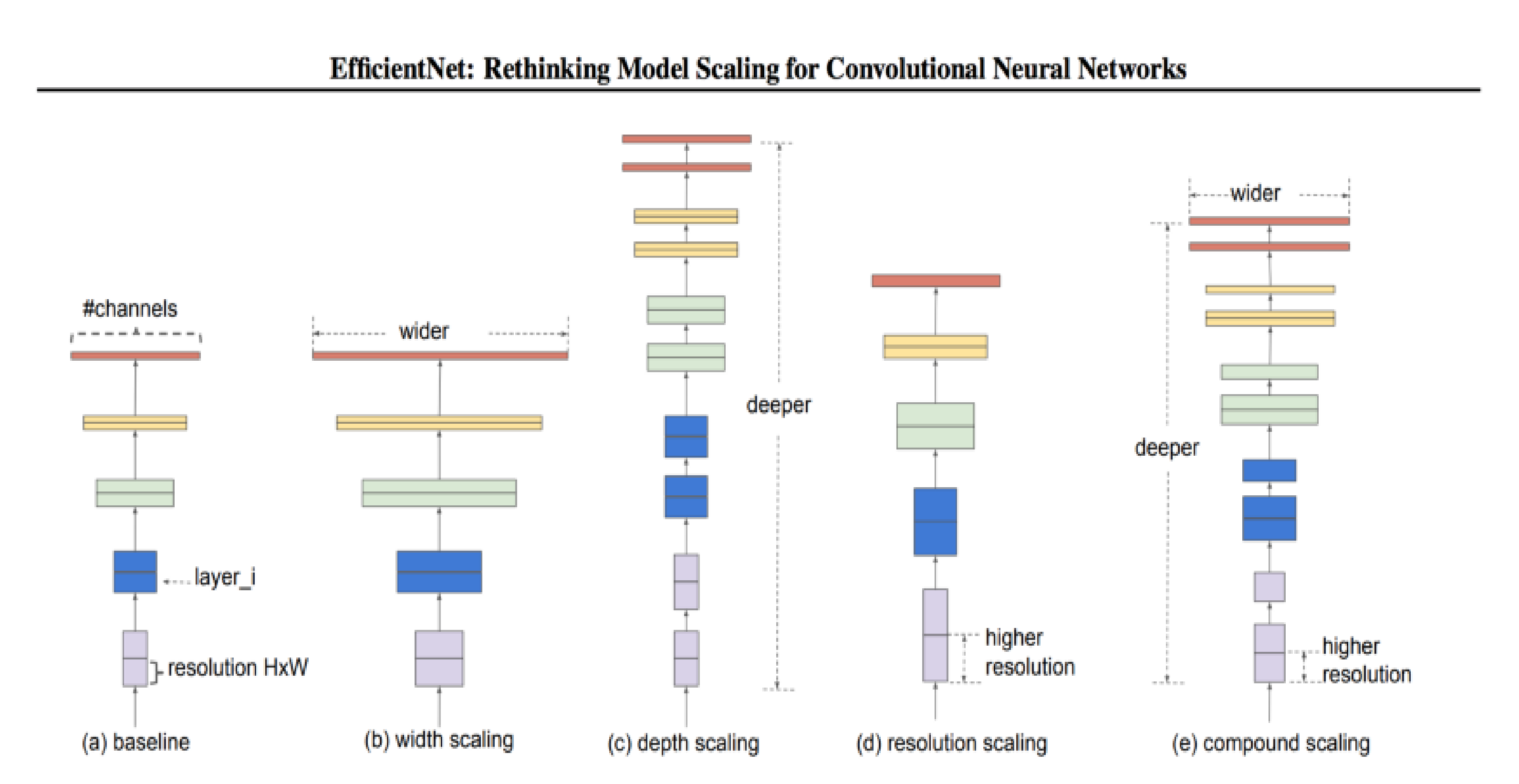
Advanced Image_Classification

21
EfficientNet

Why does scaling matter at all?
Well, scaling is generally done to improve the model's accuracy on a certain task, for example, ImageNet classification.

Scaling in the context of CNNs

- **Depth** simply means how deep the networks is which is equivalent to the number of layers in it.
- **Width** simply means how wide the network is. One measure of width, for example, is the number of channels in a Conv layer.
- **Resolution** is simply the image resolution that is being passed to a CNN.



Problems of Scaling in the context of CNNs

Depth:

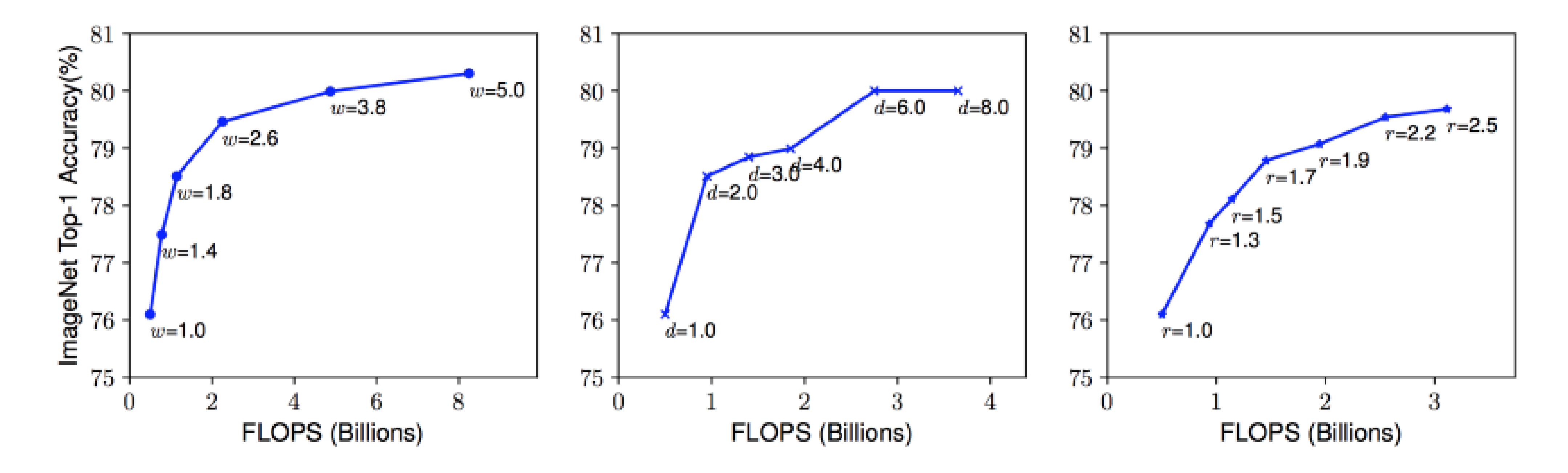
gradients to vanish; No difference between ResNet 101 to ResNet 1000

Width:

with shallow models (less deep but wider) accuracy saturates quickly with larger width

Resolution:

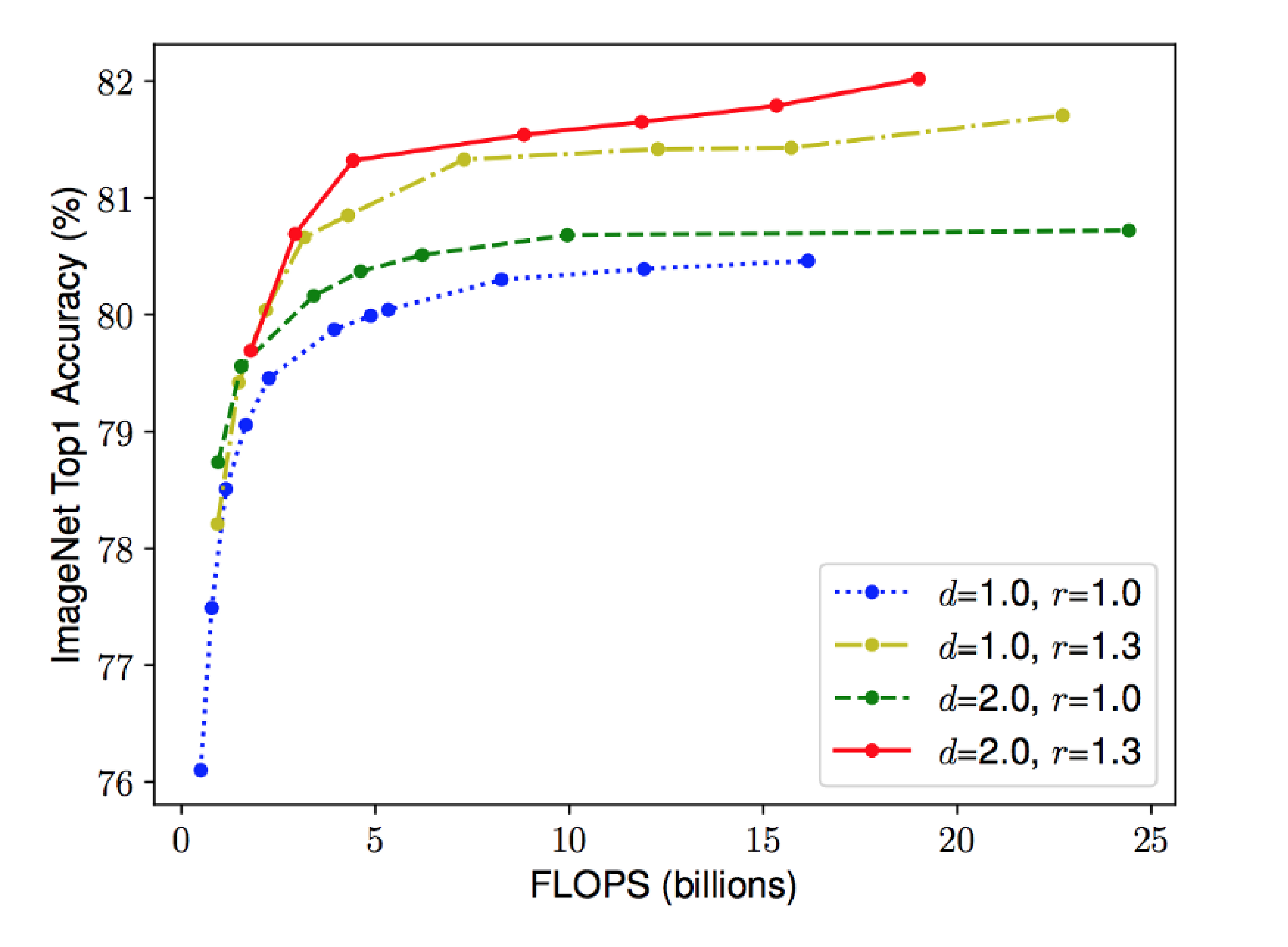
But this doesn’t scale linearly. The accuracy gain diminishes very quickly.



ImageNet Top-1 Accuracy(%)
FLOPS (Billions)
w=1.0
w=1.4
w=1.8
w=2.6
w=3.8
w=5.0
d=1.0
d=2.0
d=3.0
d=4.0
d=6.0
d=8.0
r=1.0
r=1.3
r=1.5
r=1.7
r=1.9
r=2.2
r=2.5

Combine?

It is critical to balance all dimensions of a network (width, depth, and resolution) during CNNs scaling for getting improved accuracy and efficiency and it shouldn’t be arbitrarily.



Combine ?

Model	FLOPS	Top-1 Acc.
Baseline MobileNetV1 (Howard et al., 2017)	0.6B	70.6%
Scale MobileNetV1 by width (w=2)	2.2B	74.2%
Scale MobileNetV1 by resolution (r=2)	2.2B	72.7%
compound scale (d=1.4, w=1.2, r=1.3)	**2.3B**	**75.6%**
Baseline MobileNetV2 (Sandler et al., 2018)	0.3B	72.0%
Scale MobileNetV2 by depth (d=4)	1.2B	76.8%
Scale MobileNetV2 by width (w=2)	1.1B	76.4%
Scale MobileNetV2 by resolution (r=2)	1.2B	74.8%
MobileNetV2 compound scale	**1.3B**	**77.4%**
Baseline ResNet-50 (He et al., 2016)	4.1B	76.0%
Scale ResNet-50 by depth (d=4)	16.2B	78.1%
Scale ResNet-50 by width (w=2)	14.7B	77.7%
Scale ResNet-50 by resolution (r=2)	16.4B	77.5%
ResNet-50 compound scale	**16.7B**	**78.8%**

Combine?

- In a CNN, *Conv* layers are the most compute expensive part of the network. doubling the depth will double the FLOPS while doubling width or resolution increases FLOPS almost by four times.

$$\begin{aligned} \text{depth: } & d = \alpha^\phi \\ \text{width: } & w = \beta^\phi \\ \text{resolution: } & r = \gamma^\phi \\ \text{s.t. } & \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\ & \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned}$$

Variables?

- we have a total of four parameters to search for: α, β, γ, and ϕ; in two steps:

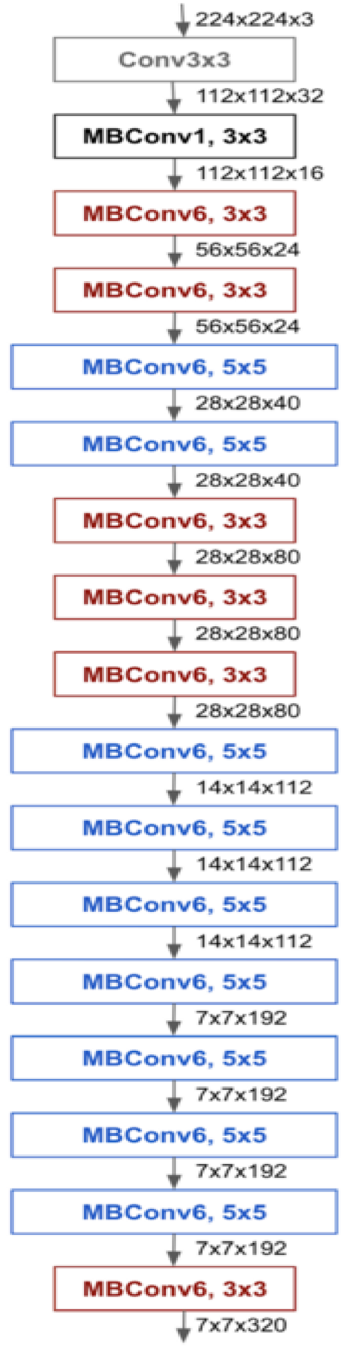
1. Fix ϕ =1, assuming that twice more resources are available, and do a small grid search for α, β, and γ. For baseline network ***B0***, it turned out the optimal values are α =1.2, β = 1.1, and γ = 1.15 such that $\alpha * \beta^2 * \gamma^2 \approx 2$

2. Now fix α, β, and γ as constants and experiment with different values of ϕ. The different values of ϕ produce EfficientNets ***B1-B7*** (Tan et al., 2019).

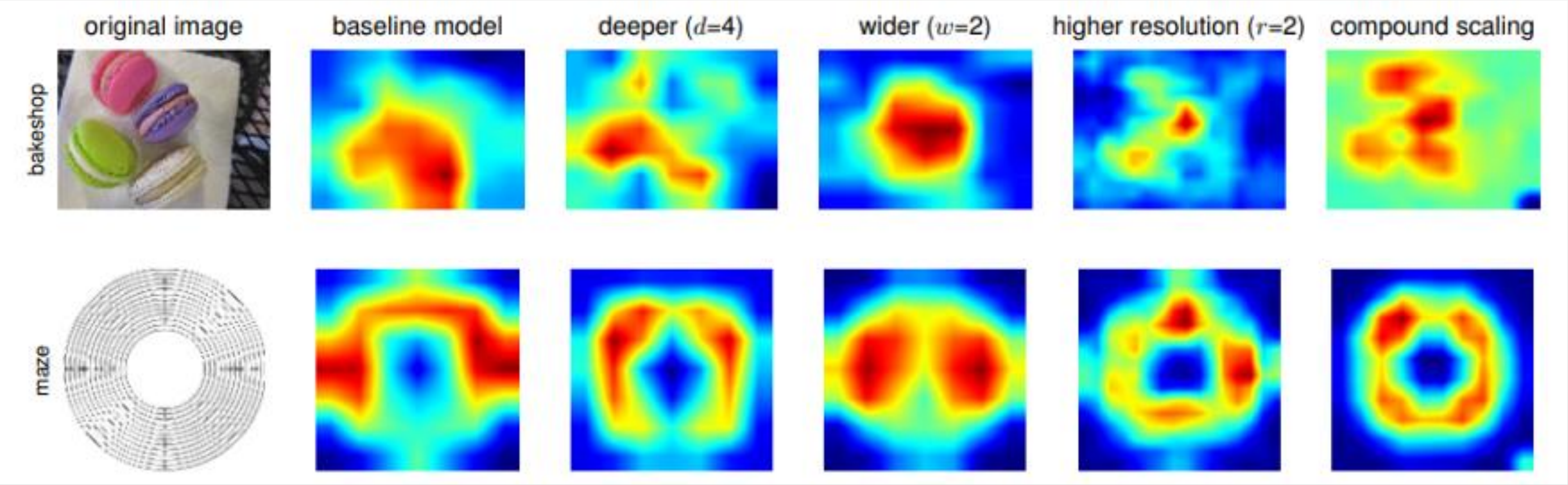
$$\begin{aligned} \text{depth: } & d = \alpha^{\phi} \\ \text{width: } & w = \beta^{\phi} \\ \text{resolution: } & r = \gamma^{\phi} \\ \text{s.t. } & \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\ & \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned}$$

EfficientNet Architecture

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels $\hat{C}_i$	#Layers $\hat{L}_i$
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	28×28	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 & Pooling & FC	7×7	1280	1

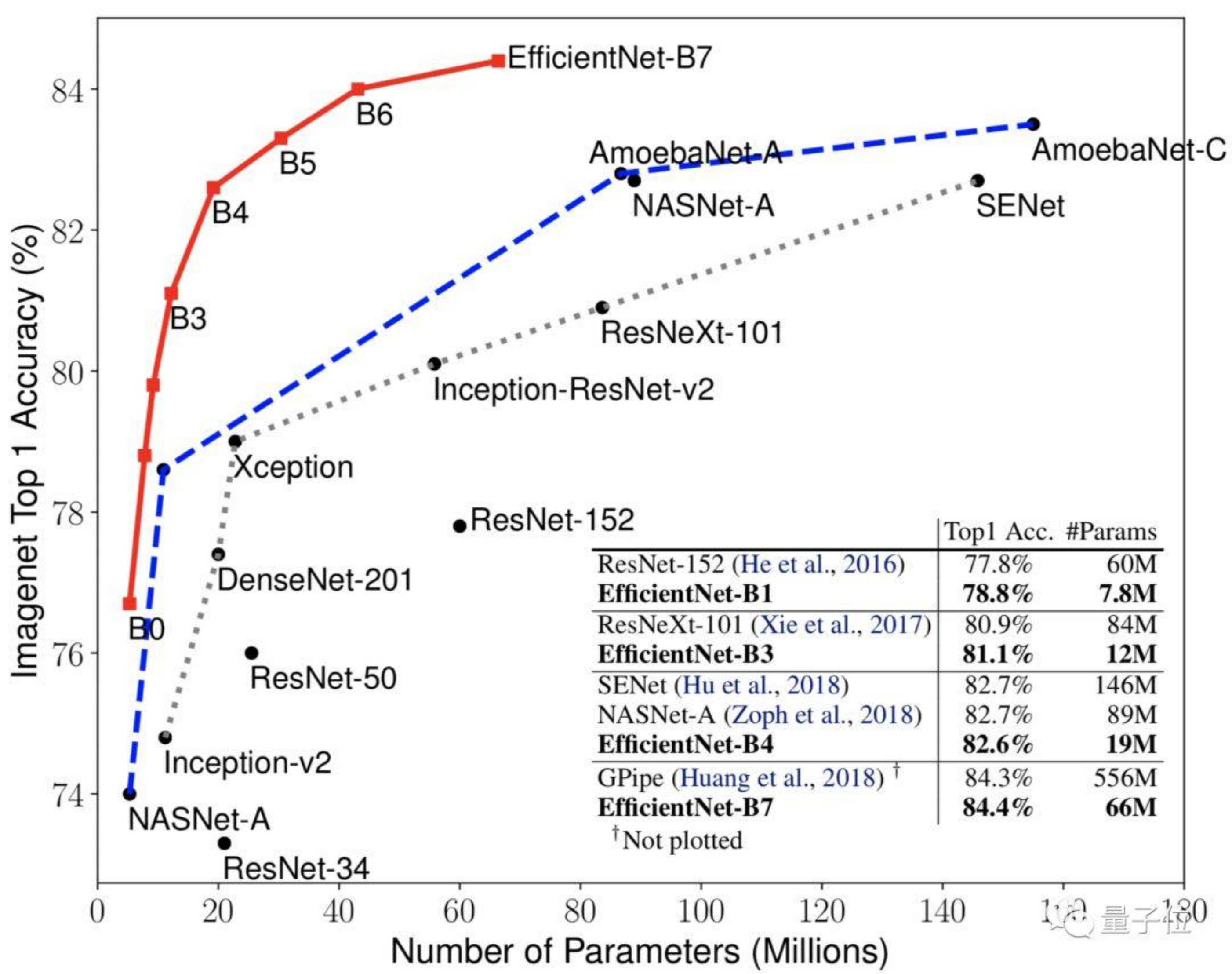


Comparison



original image
baseline model
deeper (d=4)
wider (w=2)
higher resolution (r=2)
compound scaling
bakeshop
maze

Comparison



	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**78.8%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.1%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.6%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.4%**	**66M**

†Not plotted

Part2

Advanced Image_Classification

2.2

2.2

Image Classification with Transformers

The paper on Vision Transformer (ViT) implements a pure transformer model, without the need for convolutional blocks, on image sequences to classify images.

Transformers Work Load

- Images are first tokenized and then fed into the transformers. Transformers add Attention.
- The authors of ViT solve this problem by using global attention, but not on the entire image

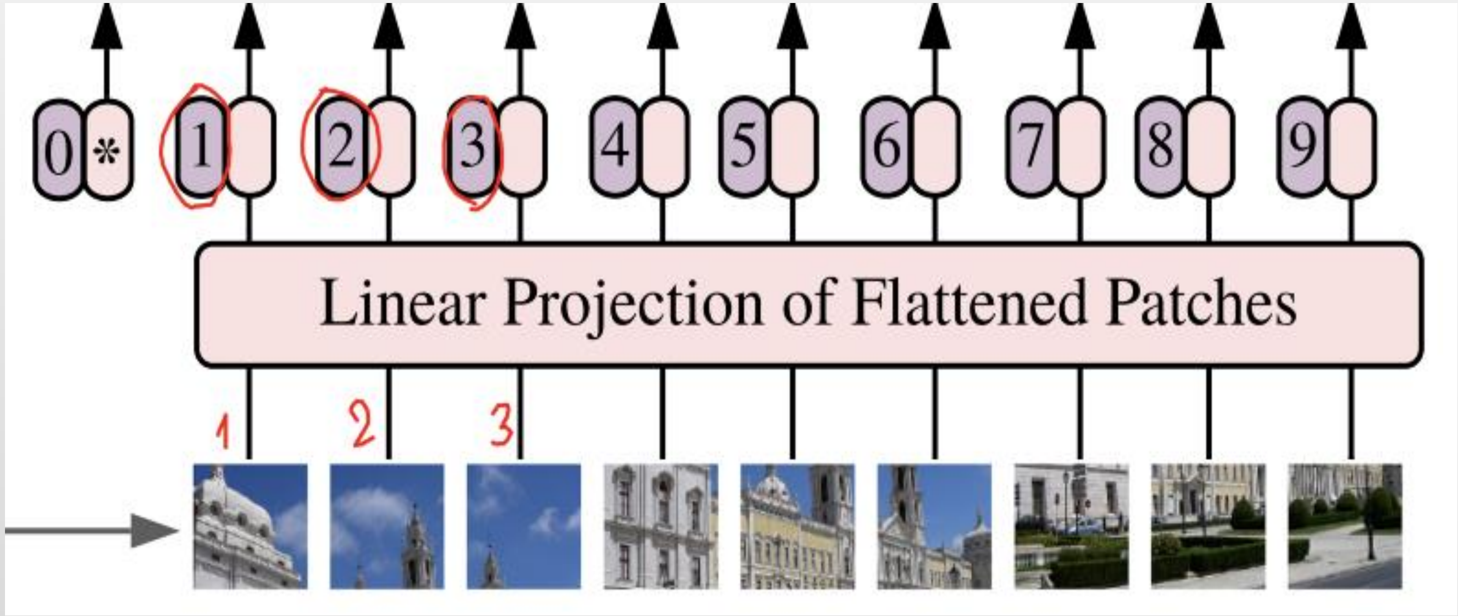
Transformers Work Load

2.2

- Then these image patches are unrolled into a sequence of images.

Transformers Work Load

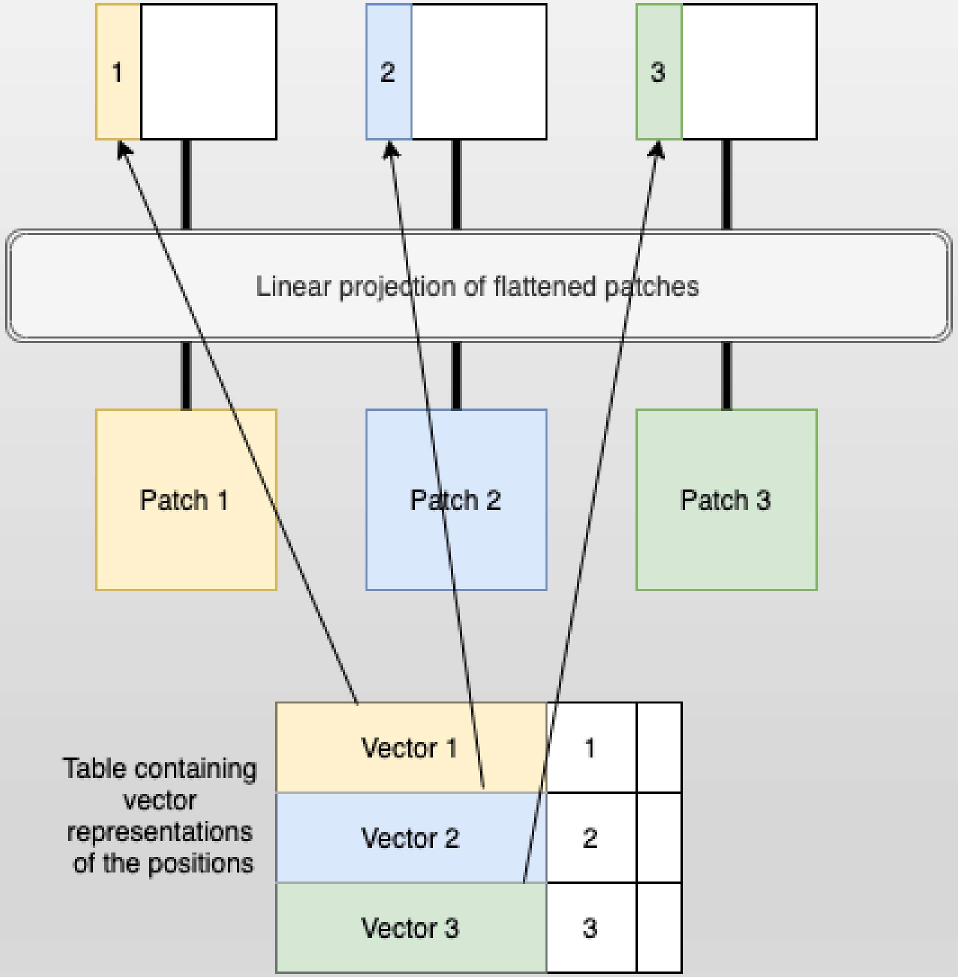
- For the first patch, the first vector from the table is grabbed and is put along with the patch into the transformer.



Transformers Work Load

2.2

- Lookup table for representation the locations.



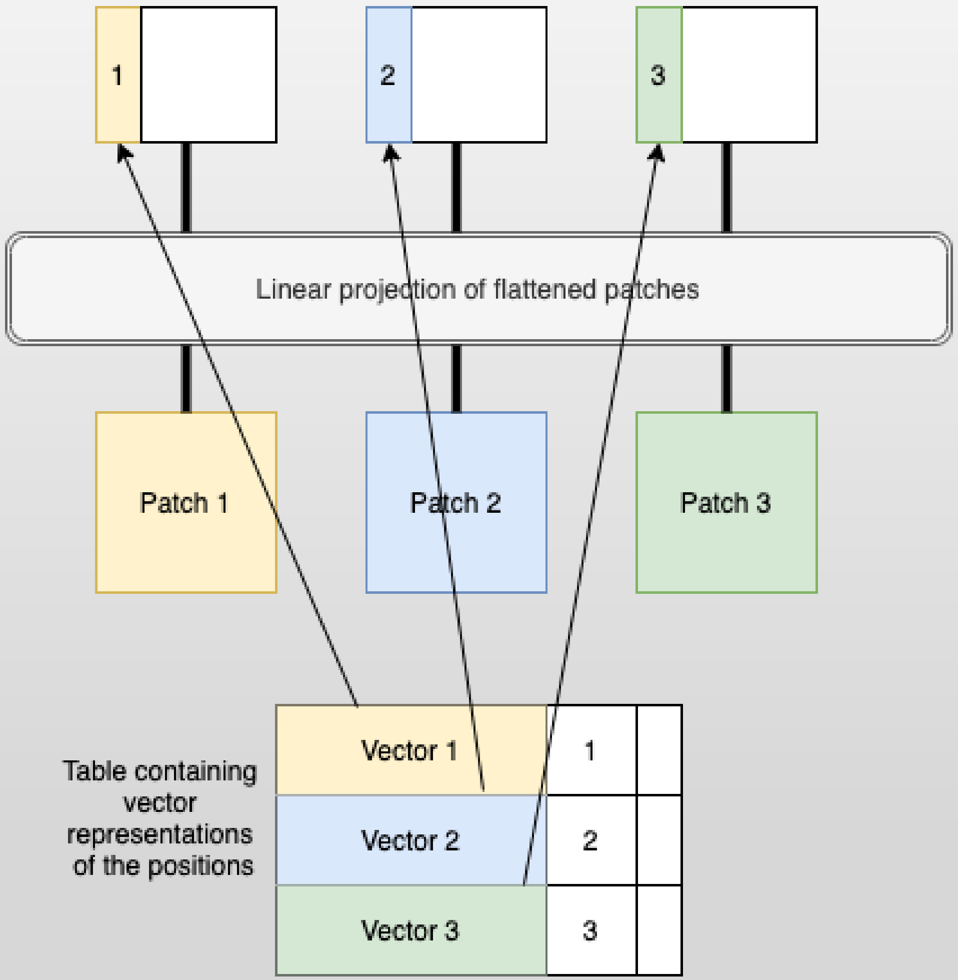
2.2

Transformers Work Load

- The image patch is a small image (16*16 pixels). This somehow needs to be fed in a way such that the transformer understands it.

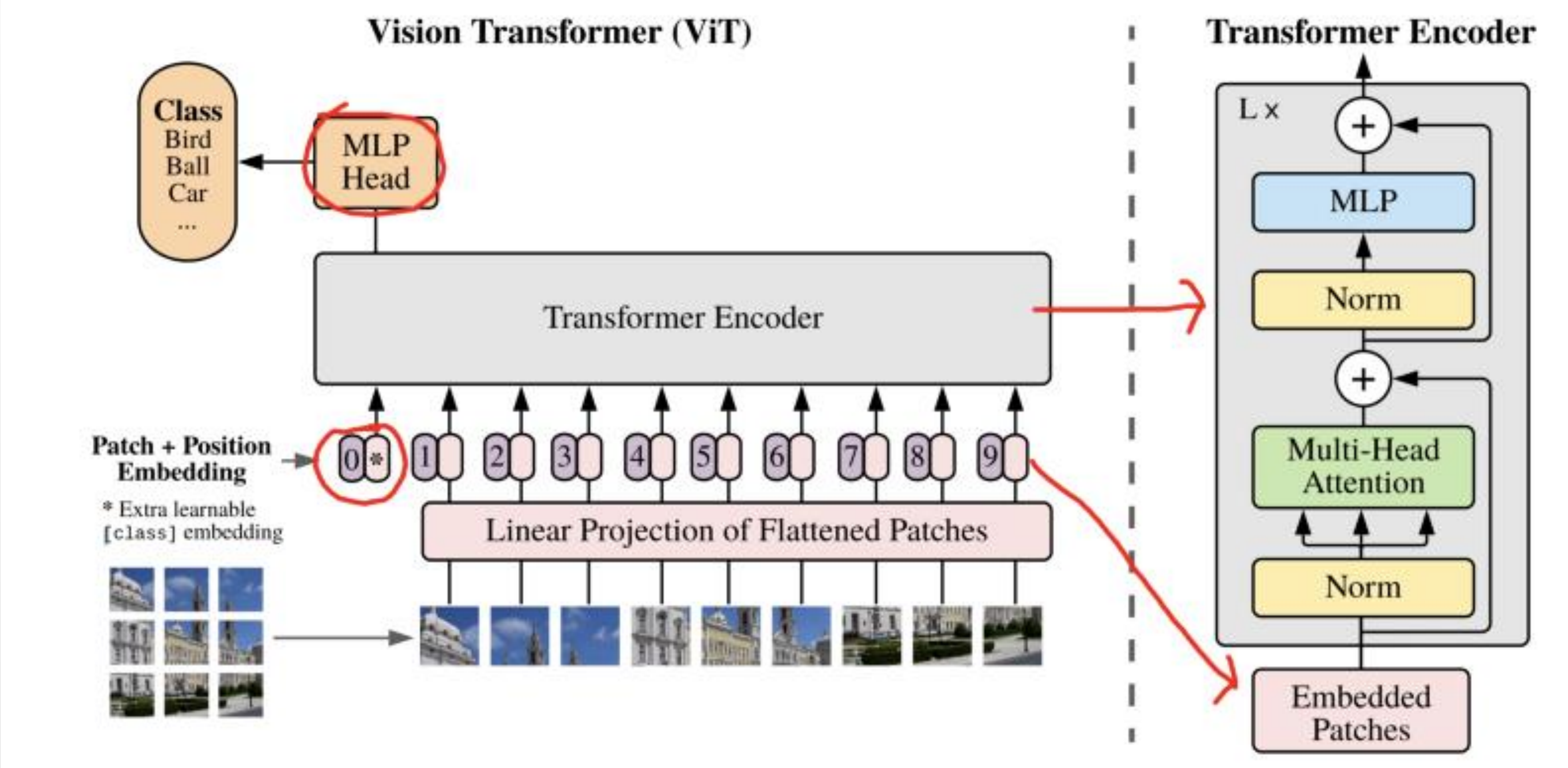
1. One way of doing so is to unroll the image into a 16*16 = 256 dimension vector.

2. Use Linear Projection Matix, a single patch is taken and first un-rolled into a linear vector. This vector is then multiplied with the embedding matrix E. The final result is then fed to the transformer, along with the positional embedding.



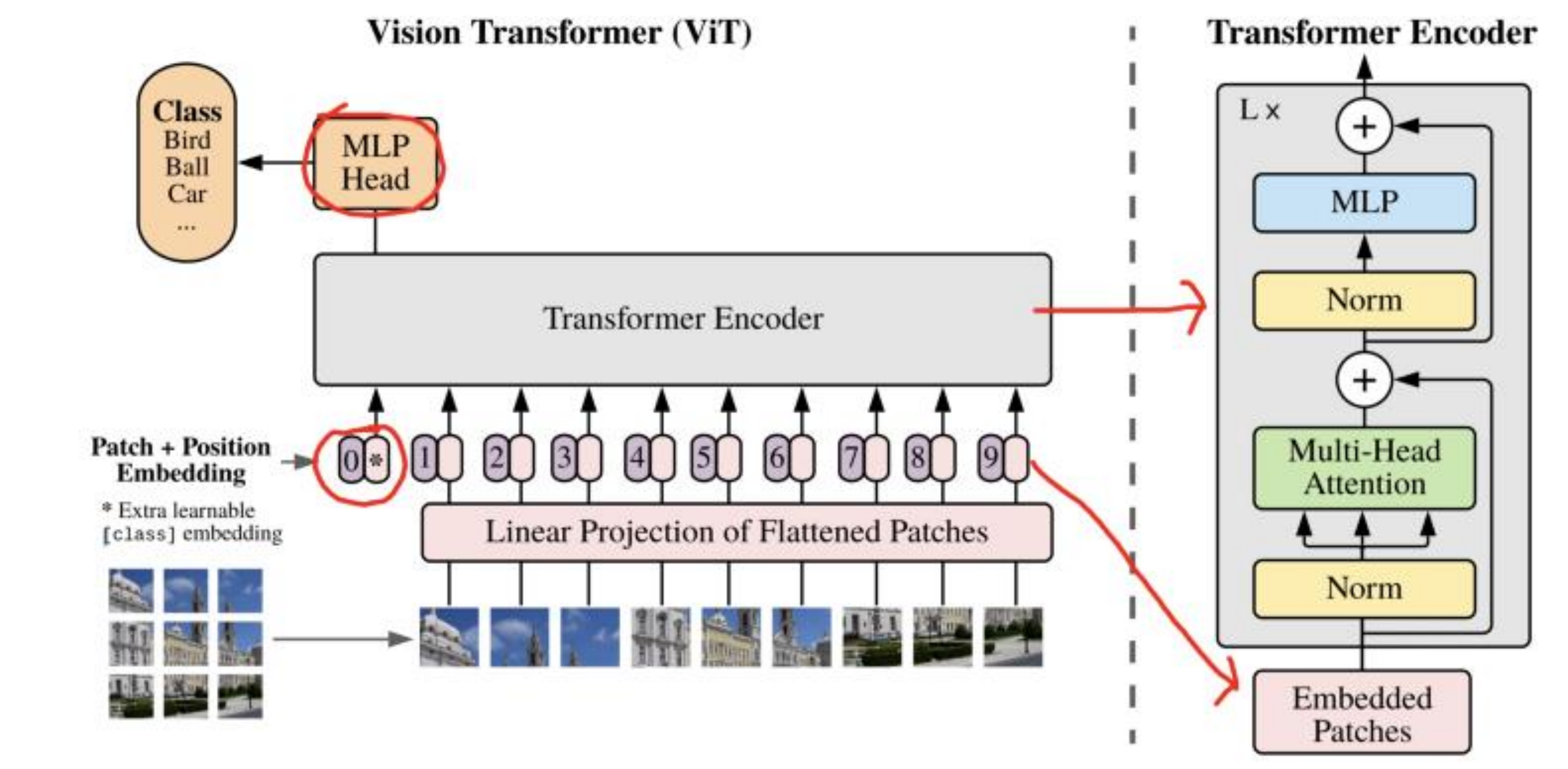
2.2

Transformers Work Load



Transformers Work Load

- The Multi-Head Self Attention layer split inputs into several heads so that each head can learn different levels of self-attention. The outputs of all the heads are then concatenated and passed through the Multi-Layer Perceptron (Dosovitskiy et al ., 2020).



Multi-Head Attention Outputs After Train

Can ViT Replace CNN?

if entire image data is fed into a model, rather than just the parts that the filters can extract (or it considers important), the chances of the model performing better are higher.

And

Transformers need efficient data to learn successfully.

the answer is, not so soon.

Just a few month back, the EfficientNet V2 model was released, which performs even better than Vision Transformers.

Benchmarking

https://paperswithcode.com/sota/image-classification-on-imagenet

Part2

Advanced Image_Classification

2.3

23

Training an Image Classification Model

We can use keras to load pretrained models and use features of pretrained models as backbone.
See:
01_training_an_image_classification.ipynb

Part2

Advanced Image_Classification

24

Transfer Learning

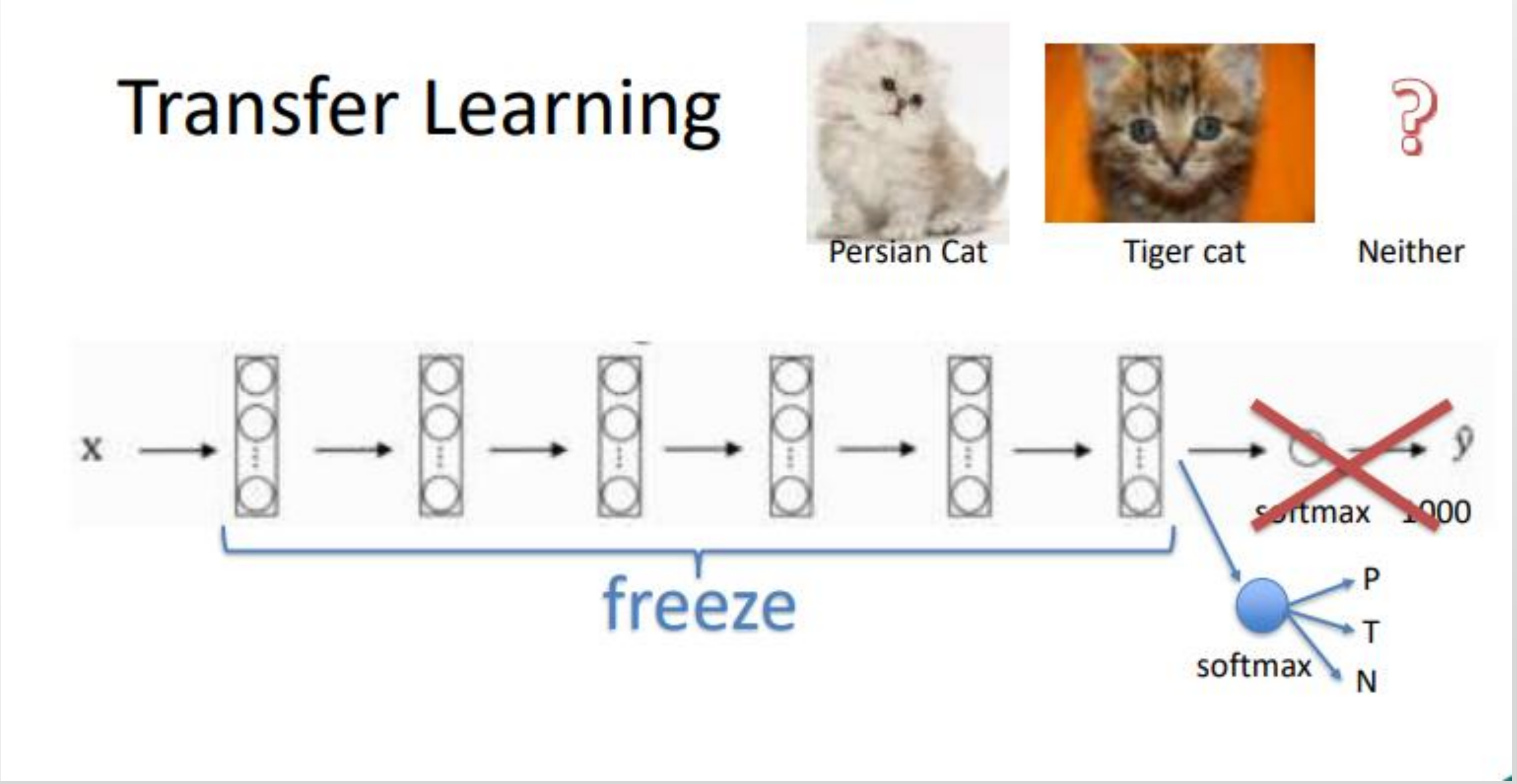
"You need a lot of a data if you want to train/use CNNs"

BUSTED

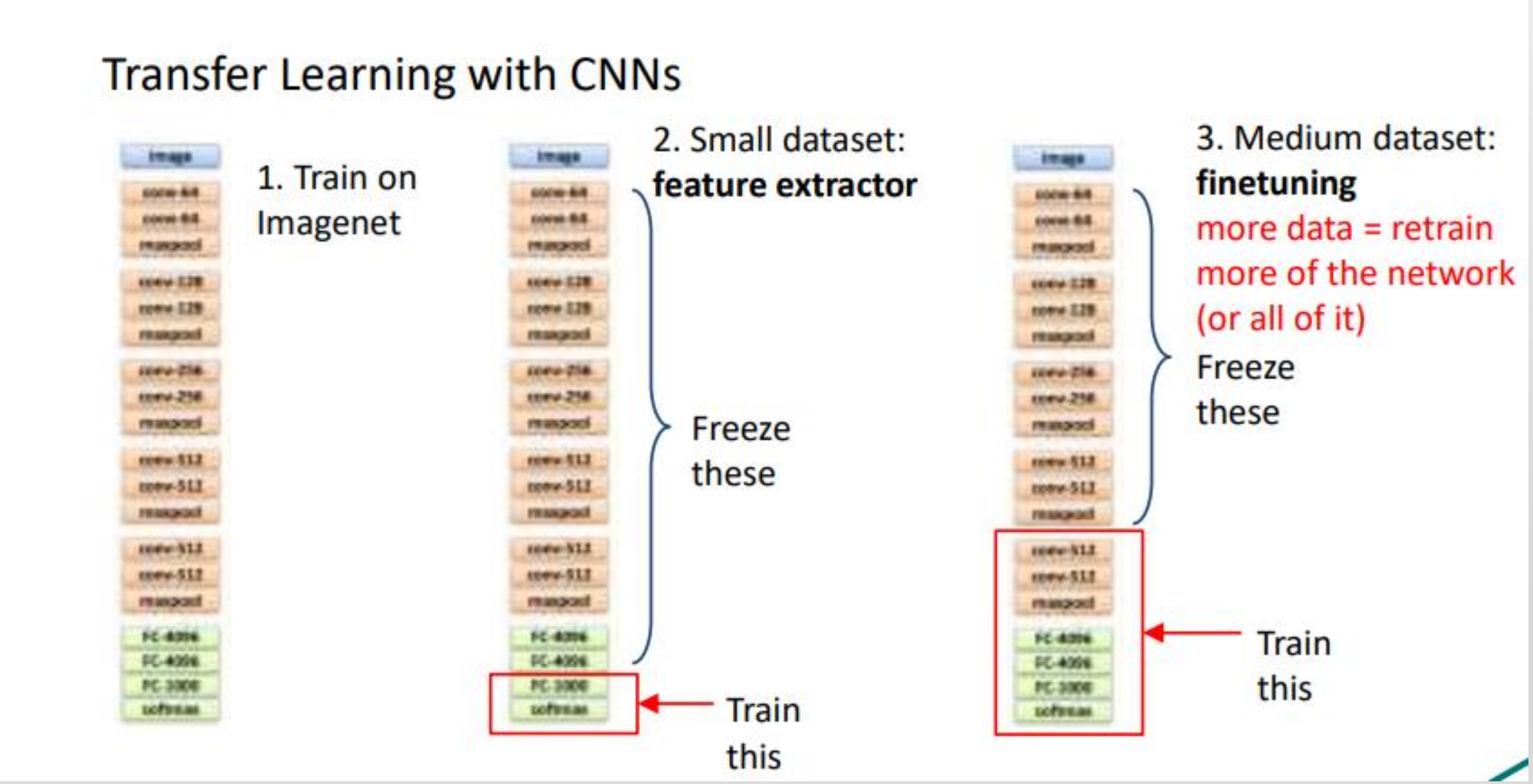
(Lisa et al., 2010)

2.4

Transfer Learning



Transfer Learning



Transfer Learning

Lets Code !

02_FineTuning.ipynb

2.4

Part2

Advanced Image_Classification

25

Improve Image Classification Accuracy

1- Add More Data: augmentation or fake even.

2- Add More Layers: if having complex and hard and strong dataset.

3- Increase/Decrease Image Size: Depend on your model, If your images are too big, you may not have enough data or your model may not be complex enough to process them.

4- More Training Time

2.5

25

Improve Image Classification Accuracy

5- different input/Color Channels: RGB, YUV, Grayscale, FFT, LBP

6- Transfer Learning

7- Change Kernel Sizes, Activation Functions

8- Progressive Resizing: both strategy

25

Improve Image Classification Accuracy

9- clean up dataset

10- Learn about dropout, L2 regularization and batch normalization

11- initialization

12- Tensorboard

2.5

Refrences

[1] Tan, Mingxing, and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks." *International Conference on Machine Learning*. PMLR, 2019.

[2] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." *arXiv preprint arXiv:2010.11929* (2020).

[3] Torrey, Lisa, and Jude Shavlik. "Transfer learning." *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI global, 2010. 242-264

R

مرکز تحقیقات
هوش مصنوعی پارت

کالج تخصصی
هوش مصنوعی پارت