ReXNet:
Diminishing Representational Bootleneck on Convolutional Neural Network

Dongyoon Han, Sangdoo Yun, Byeoungho Heo, YoungJoon Yoo

Clova AI Research, NAVER Corp.

Gio Paik
giopaik@naver.com

What is Representational Bottleneck?

- When we reduce the dimension of data, we lose some important representations.
- This leads our model to bad performance.
- Softmax layer also make representational bottleneck.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Model Structure of VGG from Karen Simonyan et al. Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR 2015

Rep. Bottleneck can occur in every layer

- The layers with limited encoding capability of generating discriminative features can be considered as the representational bottleneck.

Contributions

- Investigation of representational bottleneck problem in a DNN.
- New design principles with improved network architectures.
- SOTA result on ImageNet dataset
- Prominent transfer learning result on COCO detection and other classifications.

Preliminary: Feature Encoding

- Given an L−depth network with N features are encoded from d_0−dimensional input $X_0 \in \mathbb{R}^{d_0 \times N}$ features are represented as $X_L = \sigma(W_L(\ldots f_1(W_1 X_0)))$ with the weight matrix $W_i \in \mathbb{R}^{d_i \times d_{i-1}}$.
- We call the layer with $d_i > d_{i-1}$ an expend layer with $d_i < d_{i-1}$ an condense layer.
- Each of $f_i(\cdot)$ denotes i−th point−wise nonlinearity, such as a ReLU with a BN layer.
- $\sigma(\cdot)$ denotes Softmax Function.

Preliminary: Feature Encoding

- Let $W_i \hat{X}_{i-1}$ is convolution operation with weight $W_i \in \mathbb{R}^{d_i \times k_i^2 d_{i-1}}$ and $\hat{X}_{i-1} \in \mathbb{R}^{k_i^2 d_{i-1} \times whN}$.
- Each $i-$th layer's output X_i can be written as:

$$X_i = \begin{cases} f_i\left(W_i \hat{X}_{i-1}\right) & 1 \leq i < L, \\ \sigma\left(W_L \hat{X}_{L-1}\right) & i = L. \end{cases}$$

Softmax Bottleneck

- Output of cross-entropy loss is $\log \sigma(W_L X_{L-1})$, whose rank is bounded by the rank of $W_L X_{L-1}$, which is $\min(d_L, d_{L-1})$.
- As the input dimension d_{L-1} is smaller than the output dimension d_L, the encoded features can't fully represent the whole category due to rank deficiency.
- Then, what if we increase d_{L-1} closer to d_L?

Layer-wise rank expansion

- We conjecture the layers that expand the channel size (i.e., expand layers) such as downsampling blocks would have a rank deficiency and may have the representational bottleneck.
- To mitigate the problem, we expand the rank of weight matrix W_i.
- Given the i−th feature generated by a layer, $X_i = f_i(W_i X_{i-1}) \in \mathbb{R}^{d_i \times whN}$, rank of X_i is bounded to $\min(d_i, d_{i-1})$.

Layer-wise rank expansion

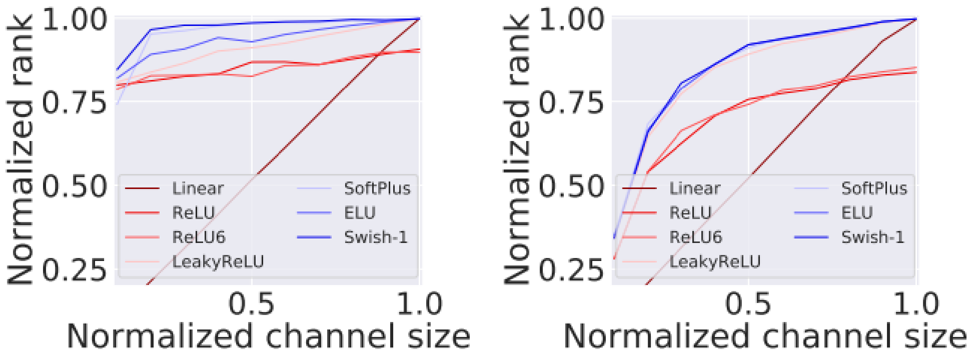
- We represent $f(X) = X \circ g(X)$, where $\circ$ denotes the pointwise multiplication with another pointwise function g.
- Following the inequality $\mathrm{rank}\big(f(X)\big) \le \mathrm{rank}(X) \cdot \mathrm{rank}\big(g(X)\big)$, the rank of X_i is bounded as,

$$\mathrm{rank}(X_i) \le \mathrm{rank}(W_i X_{i-1}) \cdot \mathrm{rank}(g_i(W_i X_{i-1})).$$

- For nonlinear function g with larger rank, we can use Swish-1 or ELU.
- When d_i is fixed, if we adjust the d_{i-1} close to d_i, we can get possibility of the unbounded rank up to the feature dimension.

Layer-wise rank expansion

- From empirical studies, we observe properly selected nonlinear functions can largely expand the rank.
- And the normalized input channel size (d_{in}/d_{out}) is closely related to the rank of the feature.



(a) A single layer (b) A bottleneck

Figure from paper

New Principles to design good model

1. Enlarge the input channel size (dimension) of a layer.
2. Equip with a proper nonlinearity.
3. Design a network with many expand layers.

Improve Network Architecture

- Representational bottleneck occur in expand layers like :
- Downsampling blocks or layers,
- First layer in a bottleneck module, inverted bottleneck blocks
- And Penultimate layers.

Improve Network Architecture

- We can improve our model by:
- Expanding the input channel size of the conv layer.
- Replacing nonlinearity like ReLU, ReLU6.

ReXNets

- Introduce new CNN model **R**ank **eX**pansion **Net**works (ReXNets)

Table 8: **Specification of ReXNet-1.0x**. Bottleneck1 and bottleneck6 denote the 3×3 inverted bottleneck with the expansion ratio of 1 and 6, respectively. In each block, SE denotes whether Squeeze Excitation Module (SE-module) [14] is used. SW denotes Swish-1 [36] is used after the convolution, and SW/RE6 denotes Swish and ReLU6 is used after the first 1×1 convolution and the 3×3 depthwise convolution [13], respectively.

Input	Operator	# of channels	SE	Nonlinearity	Stride
$224^2 \times 3$	conv 3×3	32	-	SW	2
$112^2 \times 32$	bottleneck1	16	-	SW/RE6	1
$112^2 \times 16$	bottleneck6	27	-	SW/RE6	2
$56^2 \times 27$	bottleneck6	38	-	SW/RE6	1
$56^2 \times 38$	bottleneck6	50	✔	SW/RE6	2
$28^2 \times 50$	bottleneck6	61	✔	SW/RE6	1
$28^2 \times 61$	bottleneck6	72	✔	SW/RE6	2
$14^2 \times 72$	bottleneck6	84	✔	SW/RE6	1
$14^2 \times 84$	bottleneck6	95	✔	SW/RE6	1
$14^2 \times 95$	bottleneck6	106	✔	SW/RE6	1
$14^2 \times 106$	bottleneck6	117	✔	SW/RE6	1
$14^2 \times 117$	bottleneck6	128	✔	SW/RE6	1
$14^2 \times 128$	bottleneck6	140	✔	SW/RE6	2
$7^2 \times 140$	bottleneck6	151	✔	SW/RE6	1
$7^2 \times 151$	bottleneck6	162	✔	SW/RE6	1
$7^2 \times 162$	bottleneck6	174	✔	SW/RE6	1
$7^2 \times 174$	bottleneck6	185	✔	SW/RE6	1
$7^2 \times 185$	conv 1×1, pool 7×7	1280	-	SW	1
$1^2 \times 1280$	fc	1000	-	-	1

Conclusion

- Representational bottleneck is big problem in CNNs.
- Matrix Rank is closely related to the bottleneck problem.
- Expand layers are likely to suffer from the preblem.
- So we propose a set of design principles to handle this.

ReXNet:
Diminishing Representational Bootleneck on Convolutional Neural Network

Paper: https://arxiv.org/abs/2007.00992

Official PyTorch Implementation:https://github.com/clovaai/rexnet

Thank you for watching!

Gio Paik
giopaik@naver.com