Summary of ResNet and its variants (Part 1)

Charlotte Follow him

As a milestone model of convolutional neural networks, ResNet has been used in various fields, so it is necessary to learn such a model architecture. There are also many introductions about this network on the Internet, and many improved models have been derived from it (regardless of the size of the changes). Therefore, it is necessary to summarize the variants of ResNet.

The articles involved in this article include: the original version of ResNet[1], Wider ResNet[3], ResNeXt[4], and DarkNet53[5]. These are the articles in which I have seen relatively obvious changes. Some of the articles have relatively minor changes, such as Identity mapping[2], Bag of Trick[6], Dilated ResNet[7], and a blog under the torch framework[8]. These articles also mention changes to ResNet, but they do not change the entire block, but only a small part.

Let's start by talking about the changes to these articles.

The original ResNet[1] pointed out that it is generally believed that deeper networks usually bring better performance, but in fact, the performance of ordinary deeper networks not only does not improve, but actually decreases. So the article puts forward a hypothesis that deeper networks should bring at least the same performance as shallow networks, so how to ensure this effect? The article proposes skip connection, which is a simple addition of shallow features to deep features. This ensures that even if the features obtained by the intermediate operation have no effect, the deep network can still guarantee the same performance as the shallow layer. However, the actual effect is unexpectedly good, because of the existence of skip connection, the convolution in the middle of each feature addition block only needs to fit the "residual", and the rest can be skipped and given to the previous one. The article proposes two blocks, one is BasicBlock and the other is BottleNeck Block, as shown below:

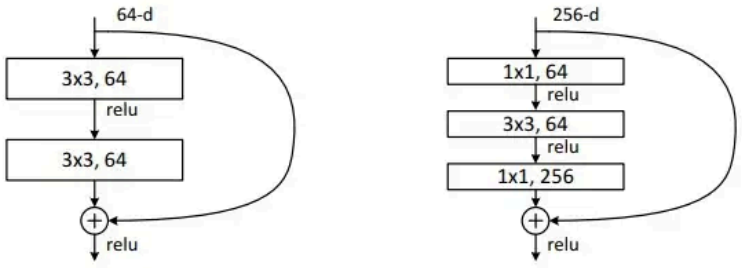


Figure 5. A deeper residual function $\mathcal{F}$ for ImageNet. Left: a building block (on 56×56 feature maps) as in Fig. 3 for ResNet-34. Right: a "bottleneck" building block for ResNet-50/101/152.

Among them, the one on the left is BasicBlock, which is used for ResNet with less than 50 layers (usually ResNet18, ResNet34); the one on the right is BottleNeckBlock, which is used for ResNet with more than or equal to 50 layers (usually ResNet50, ResNet101 and ResNet152). In

知乎

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Table 1. Architectures for ImageNet. Building blocks are shown in brackets (see also Fig. 5), with the numbers of blocks stacked. Down-sampling is performed by conv3_1, conv4_1, and conv5_1 with a stride of 2.

The two columns on the left of the network use BasicBlock, and the three columns on the right use BottleNeck.

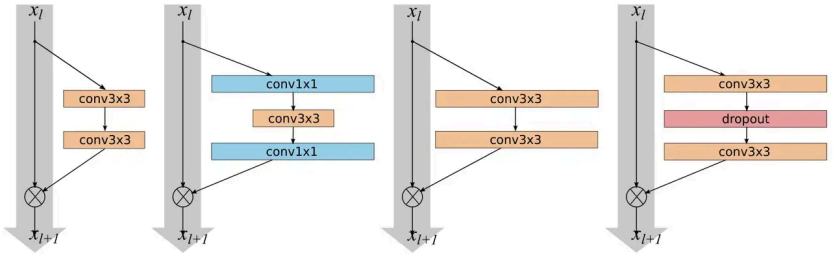
The original ResNet has made a great contribution to training convolutional neural networks, but it also has many areas for improvement. First, after the publication of [1], many subsequent studies [], [] showed that residual connections are not efficient. From the original paper in [1], we can see that from ResNet50 to ResNet152, the top5-error decreased by 1% (top1-error was not even 1%), but the FLOPs increased by about 3 times.

(Note: In the subsequent article [6] and the blog [8] in the torch framework, the top1-error of ResNet50 is 24.7%, and the figure below is obtained on arxiv)

method	top-1 err.	top-5 err.
VGG [41] (ILSVRC'14)	-	8.43†
GoogLeNet [44] (ILSVRC'14)	-	7.89
VGG [41] (v5)	24.4	7.1
PReLU-net [13]	21.59	5.71
BN-inception [16]	21.99	5.81
ResNet-34 B	21.84	5.71
ResNet-34 C	21.53	5.60
ResNet-50	20.74	5.25
ResNet-101	19.87	4.60
ResNet-152	**19.38**	**4.49**

Table 4. Error rates (%) of **single-model** results on the ImageNet validation set (except † reported on the test set).

Later [3] proposed that the convolution in each block should be wider, as shown in Figure (c):



(a) basic (b) bottleneck (c) basic-wide (d) wide-dropout

Figure 1: Various residual blocks used in the paper. Batch normalization and ReLU precede

知乎

Model	top-1 err, %	top-5 err, %	#params	time/batch 16
ResNet-50	24.01	7.02	25.6M	49
ResNet-101	22.44	6.21	44.5M	82
ResNet-152	22.16	6.16	60.2M	115
WRN-50-2-bottleneck	21.9	6.03	68.9M	93
pre-ResNet-200	21.66	5.79	64.7M	154

Table 8: ILSVRC-2012 validation error (single crop) of bottleneck ResNets. Faster WRN-50-2-bottleneck outperforms ResNet-152 having 3 times less layers, and stands close to pre-ResNet-200.

In general, this article only proposes a possibility and illustrates the role of the wide model.

ResNeXt[4] was also proposed by the Kaiming group. The 3x3 Conv in BottleNeck was added in the form of group convolution, and a hyperparameter "Cardinality" was proposed to represent the number of groups of group convolution. At the same time, the dimension reduction of 1x1 Conv was changed from 1/4 to 1/2, and then connected to 3x3 Group Conv. This keeps the parameters and floating point numbers approximated.

stage	output	ResNet-50	**ResNeXt-50 (32×4d)**
conv1	112×112	7×7, 64, stride 2	7×7, 64, stride 2
conv2	56×56	3×3 max pool, stride 2 $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	3×3 max pool, stride 2 $\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128, C=32 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3	28×28	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, C=32 \\ 1\times1, 512 \end{bmatrix}\times4$
conv4	14×14	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, C=32 \\ 1\times1, 1024 \end{bmatrix}\times6$
conv5	7×7	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, C=32 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params.		**25.5**×10^6	**25.0**×10^6
FLOPs		**4.1**×10^9	**4.2**×10^9

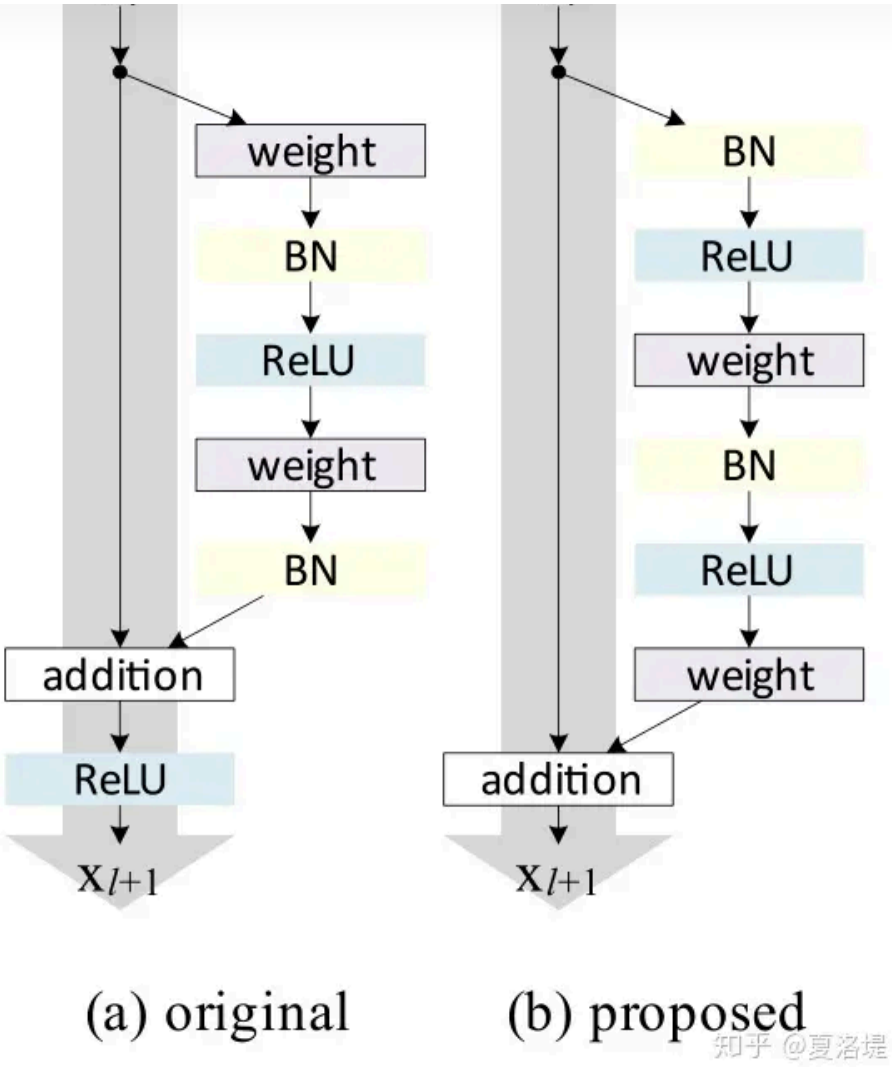
Table 1. (**Left**) ResNet-50. (**Right**) ResNeXt-50 with a 32×4d template (using the reformulation in Fig. 3(c)). Inside the brackets are the shape of a residual block, and outside the brackets is the number of stacked blocks on a stage. "C=32" suggests grouped convolutions [24] with 32 groups. *The numbers of parameters and FLOPs are similar between these two models.*

知乎

ResNet-50	1 × 64d	23.9
ResNeXt-50	2 × 40d	23.0
ResNeXt-50	4 × 24d	22.6
ResNeXt-50	8 × 14d	22.3
ResNeXt-50	32 × 4d	**22.2**
ResNet-101	1 × 64d	22.0
ResNeXt-101	2 × 40d	21.7
ResNeXt-101	4 × 24d	21.4
ResNeXt-101	8 × 14d	21.3
ResNeXt-101	32 × 4d	**21.2**

Table 3. Ablation experiments on ImageNet-1K. (**Top**): ResNet-50 with preserved complexity (∼4.1 billion FLOPs); (**Bottom**): ResNet-101 with preserved complexity (∼7.8 billion FLOPs). The error rate is evaluated on the single crop of 224×224 pixels.

After the publication of [1], Kaiming's group published the Identity mapping [2] article to explore the optimization problem of deeper residual networks. They proposed the "pre-activation" order, that is, when entering each block, first perform BatchNorm+ReLU and then Conv. In this way, each addition is directly added without BN+ReLU. Experiments have shown that this makes deeper networks, such as ResNet-1001, easier to optimize and less prone to overfitting.



The final results show that the performance of ResNet-200 with pre-activation reaches sota, which is better than InceptionV3.

Table 5. Comparisons of single-crop error on the ILSVRC 2012 validation set. All ResNets are trained using the same hyper-parameters and implementations as [1]). Our Residual Units are the full pre-activation version (Fig. 4(e)). †: code/model available at https://github.com/facebook/fb.resnet.torch/tree/master/pretrained, using scale and aspect ratio augmentation in [20].

method	augmentation	train crop	test crop	top-1	top-5
ResNet-152, original Residual Unit [1]	scale	224×224	224×224	23.0	6.7
ResNet-152, original Residual Unit [1]	scale	224×224	320×320	21.3	5.5
ResNet-152, **pre-act** Residual Unit	scale	224×224	320×320	21.1	5.5
ResNet-200, original Residual Unit [1]	scale	224×224	320×320	21.8	6.0
ResNet-200, **pre-act** Residual Unit	scale	224×224	320×320	**20.7**	**5.3**
ResNet-200, **pre-act** Residual Unit	scale+asp ratio	224×224	320×320	**20.1**†	**4.8**†
Inception v3 [19]	scale+asp ratio	299×299	299×299	21.2	5.6

However, this phenomenon is only effective in very deep networks. Even for ResNet-164, the performance of pre-activation and ordinary structures is not much different.

This is the end of the first part. The second half will describe DarkNet53 and some other

知乎

2. Identity Mappings in Deep Residual Networks
3. Wide Residual Networks
4. Aggregated Residual Transformations for Deep Neural Networks
5. YOLOv3: An Incremental Improvement
6. Bag of Tricks for Image Classification with Convolutional Neural Networks
7. Dilated Residual Networks
8. Torch | Training and investigating Residual Nets

Posted on 2019-02-04 10:54

Deep Learning

Speak rationally and interact in a friendly manner

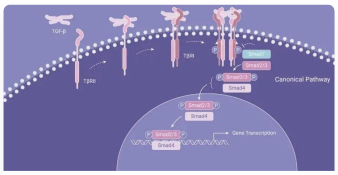
1 Comment default up to date

harry

Hello, boss, I'm catching a bug: First of all, after the publication of [1], many subsequent studies [], [] indicated that residual connections are not efficient. From the original text of [1], we can also see that from ResNet50 to ResNet152, the top5-error decreased by 1% (top1-error was not even 1%), but FLOPs increased by about 3 times.
【】【】Not fully filled in
05-19 reply like

Recommended Reading

Molecular Symmetry I: Symmetry Operations and...
Sholokhka Published in 『C.Av...



Targeting the TGF-β signaling pathway | MedChemExpress
MedChemExpress

【Textbook Intensive Reading】Chromosome...
Quanxingquanyixue Published in High School Biology...

Miscellaneous Discussions on Molecular Docking
Herba...Published in Molecular Docking Technique...