

ResNet 구현 및 실험 결과 보고



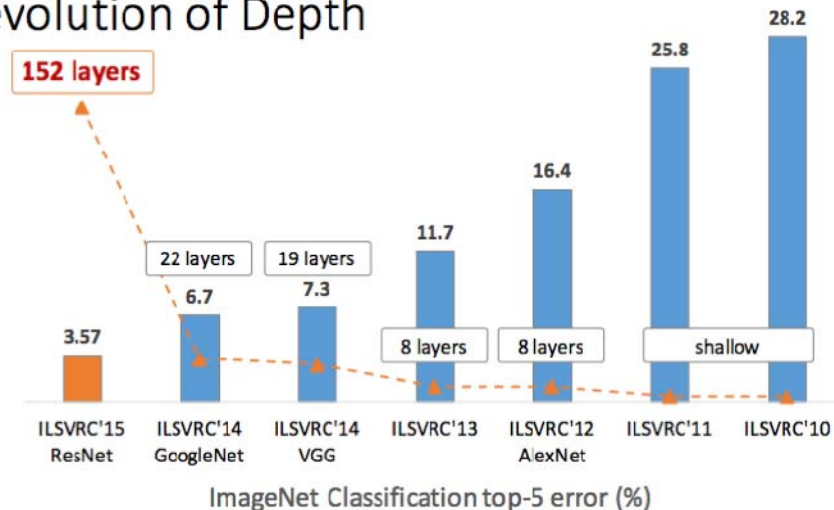
AutoML 연구실
석사과정 문아성
20.08.04

Introduction

- 2015년 ImageNet Challenge (ILSVRC) 전 분야에서 우승
- 기존 방법(VGG, GoogLeNet) 같은 경우 레이어 수가 20개 내외 였지만, ResNet은 최대 152개로 소개

=> 깊은 망의 문제점을 해결

Revolution of Depth



깊은 망의 문제점

- Vanishing / Exploding Gradient

층이 깊어질수록 너무 큰 값이나 작은 값으로 포화되어 학습 효과가 없어지거나, 학습 속도가 매우 느려지는 문제가 발생

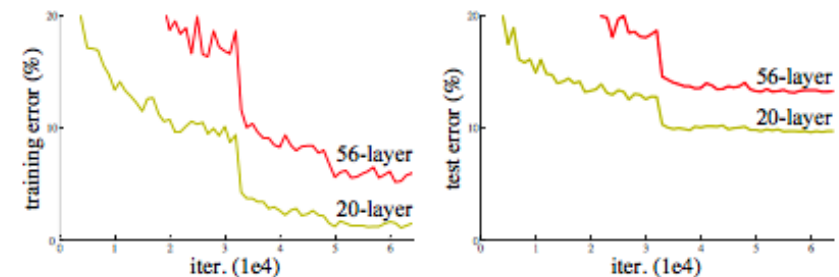


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer “plain” networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

- ➡ 기존 “plain” networks 에서 깊은 망으로 학습시켰을 때, 얇은 망에 비해 어려움이 더 높게 학습한 것을 알 수 있음

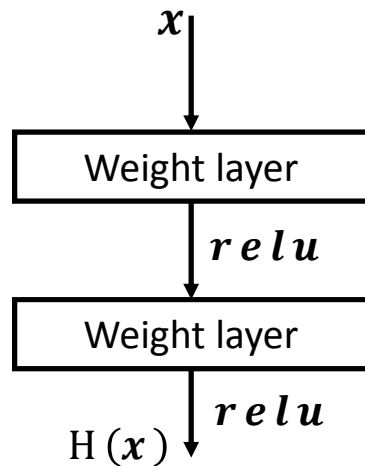
ResNet

- Residual Learning

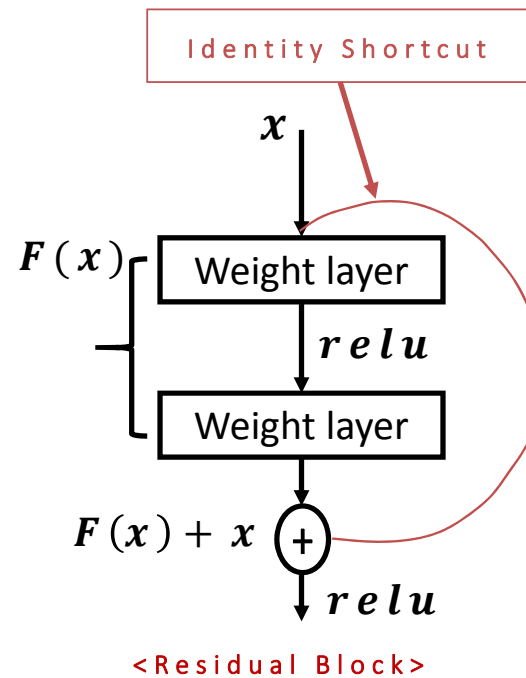
네트워크의 출력을 학습하는 것이 아닌, 출력 ($H(x)$) 과 입력(x)의 차를 학습
즉, Weighted Layer($F(x)$) 는 $H(x) - x$ 를 학습해야 하므로,

$$F(x) = H(x) - x \Rightarrow H(x) = F(x) + x$$

$\therefore$ 네트워크 출력인 $H(x)$ 는 $H(x) = F(x) + x$ 를 학습



<기존 Convolution Layers>



<Residual Block>

Identity Mapping

* Identity Mapping: 입력값을 그대로 전달한다는 의미

- Residual Unit Forward

$$y_l = h(x_l) + F(x_l, W_l), \quad x_{l+1} = f(y_l)$$

$$\therefore x_L = x_l + \sum_{i=l}^{L-1} F(x_i, W_i)$$

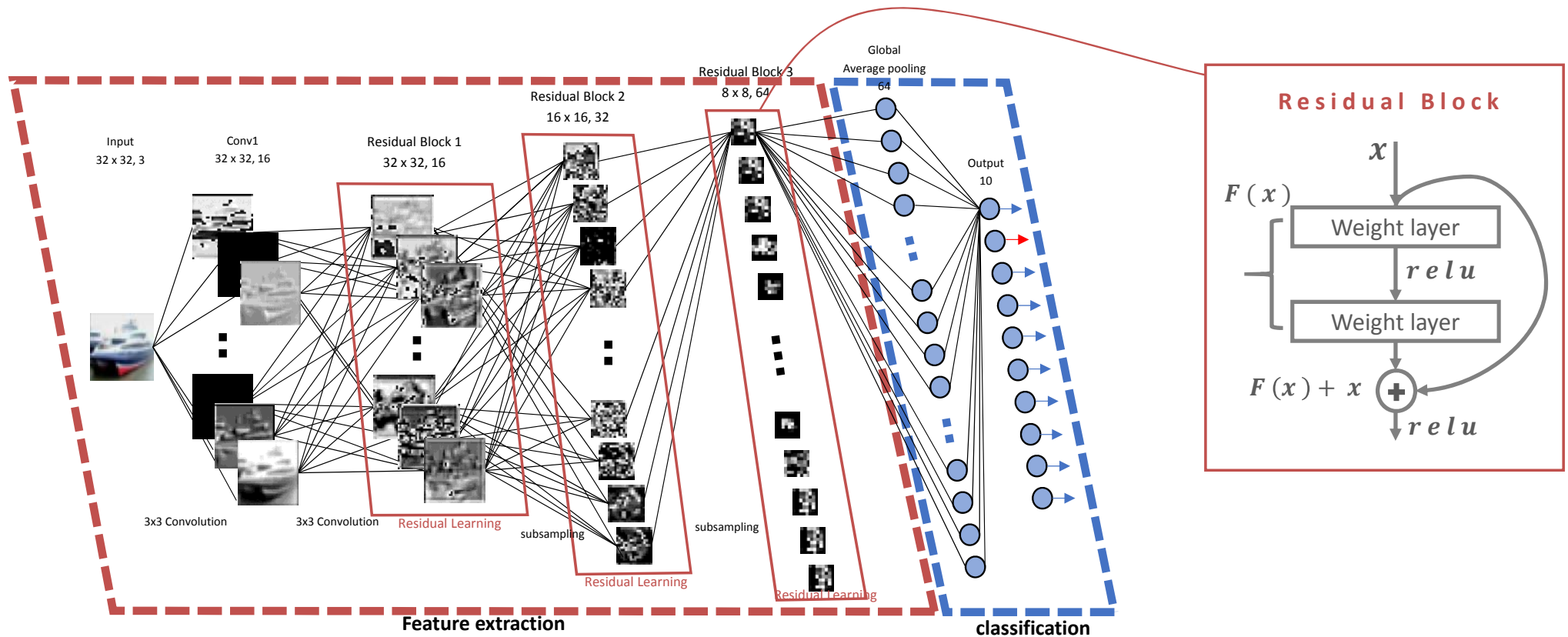
- Residual Unit Backward

$$\frac{\partial \mathcal{E}}{\partial x_l} = \frac{\partial \mathcal{E}}{\partial x_L} \frac{\partial x_L}{\partial x_l} = \frac{\partial \mathcal{E}}{\partial x_L} \left(1 + \frac{\partial}{\partial x_l} \sum_{i=l}^{L-1} F(x_i, W_i) \right)$$

→ 이 값이 0 이 되어야 Vanishing Gradient 문제가 발생하는데,

미니배치마다 $\frac{\partial}{\partial x_l} \sum_{i=l}^{L-1} F(x_i, W_i)$ 값이 -1이 되는 일은 거의 없기 때문에 vanishing이 발생할 확률은 극히 낮음

ResNet Architecture



DATASET CIFAR-10

32x32사이즈의 RGB 이미지와
10가지의 레이블로 구성 (32, 32, 3), (10)

INPUT DATA : 50000

TEST DATA : 10000

* ResNet-20는 K-fold Cross Validation 사용

Here are the classes in the dataset, as well as 10 random images from each:

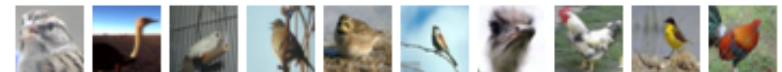
airplane



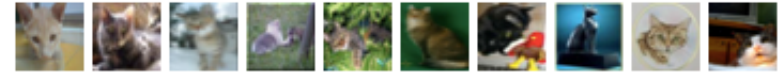
automobile



bird



cat



deer



dog



frog



horse



ship



truck



실험 결과

		실제 정답	
		True	False
분류 결과	True	True Positive	False Positive
	False	False Negative	True Negative

<Confusion Matrix>

$$\text{정밀도 (Precision)}: \frac{TP}{TP+FP}$$

$$\text{재현율 (Recall)}: \frac{TP}{TP+FN}$$

$$\text{정확도 (Accuracy)}: \frac{TP+TN}{TP+FP+FN+TN}$$

Name	# layers	# params	precision	recall	accuracy
ResNet-20	(N = 3)20	0.27M	91.69%	91.62%	91.63%
ResNet-32	(N = 5)32	0.46M	92.21%	92.18%	92.19%
ResNet-44	(N = 7)44	0.66M	91.57%	91.52%	91.53%
ResNet-56	(N = 9)56	0.85M	92.46%	92.48%	92.48%
ResNet-110	(N = 18)110	1.7M	93.19%	93.17%	93.17%

* ResNet-20는 K-fold Cross Validation 사용

Batch_1	90.27%
Batch_2	91.22%
Batch_3	90.19%
Batch_4	92.93%
Batch_5	93.01%
Batch_6	92.17%
평균	91.63%

<K-fold Cross Validation>

실험 결과

평가지표: 정확도 (Accuracy)

다중 분류 문제와

CIFAR-10 데이터셋에서 레이블당

입력데이터 개수가 같은 것을 고려하여

평가지표를 **정확도 (Accuracy)**로 선정

Name	Accuracy
ResNet-20 (N=3)	91.63%
ResNet-32 (N=5)	92.19%
ResNet-44 (N=7)	91.53%
ResNet-56 (N=9)	92.48%
ResNet-110 (N=18)	93.17%

