

[ホーム](#) > [ディープラーニング](#) > Residual Network(ResNet)の理解とチューニングのベストプラクティス

2016-11-30

Residual Network(ResNet)の理解とチューニングのベストプラクティス

ツイート

いいね！

シェア



[Residual Network \(ResNet\) とは](#)

- [ResNetのアイデア](#)
- [Shortcut Connectionの導入](#)

人工知能の今と一歩先を発信するメディア

- [Wide Residual Network](#)

[まとめ](#)

[参考](#)

2015年のImageNetコンペティションとCOCOセグメンテーションの最良モデルとしてDeep Residual NetworksがMicrosoft Researchから提案され、最大1000層以上の深いニューラルネットワークを構築することが可能となった。

本記事では、

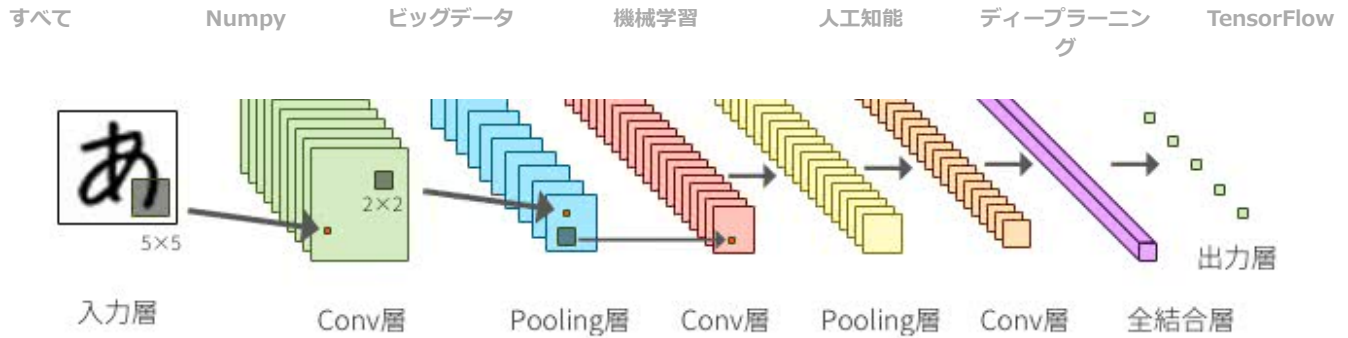
- Residual Networkとは何か
- Residual Networkのチューニング方法
- Residual Networkの派生モデル

をまとめた。

Residual Network (ResNet) とは

ResNetは、Microsoft Research(現Facebook AI Research)のKaiming He氏が2015年に考案したニューラルネットワークのモデルである。

[CNN](#)において層を深くすることは重要な役割を果たす。層を重ねるごとに、より高度で複雑な特徴を抽出していると考えられているからだ。



Convolution層はフィルタを持ち、Pooling層と組み合わせて何らかの特徴を検出する役割を持っている。低・中・高レベルの特徴を多層形式で自然に統合し、認識レベルを強化することができる。

2014年の画像認識の分野でトップを争うImageNetコンペティションにおいて、1位と2位を飾ったモデルはVGGNetと呼ばれている。

“[Very Deep Convolutional Networks for Large-Scale Image Recognition](#)”

こちらがその論文である。層の数を16層や19層まで深くしたのだ。それでVery Deepと名付けられていた。

検出部門で勝利を飾ったGoogLeNetでも22層だった。



人工知能の今と一歩先を発信するメディア

Deepの16、19層やGoogLeNetの22層がものすごく浅く感じてしまう。

ResNetのアイデア

ResNet以前も層を更に深くする試みはあったものの、思い通りに学習が進まなかった。

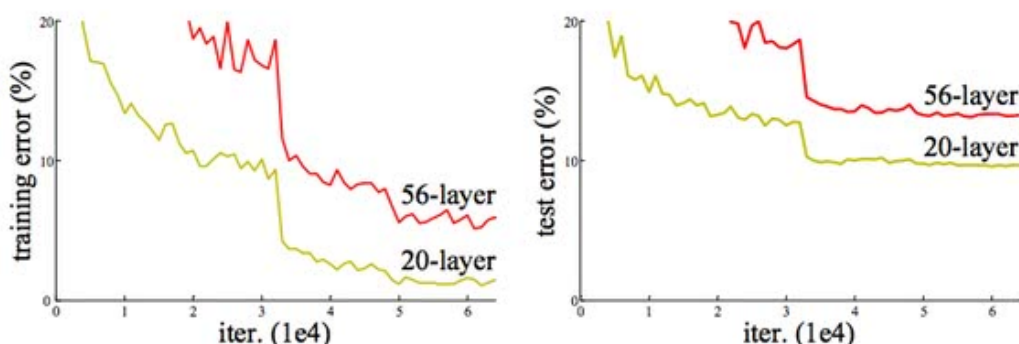


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer “plain” networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

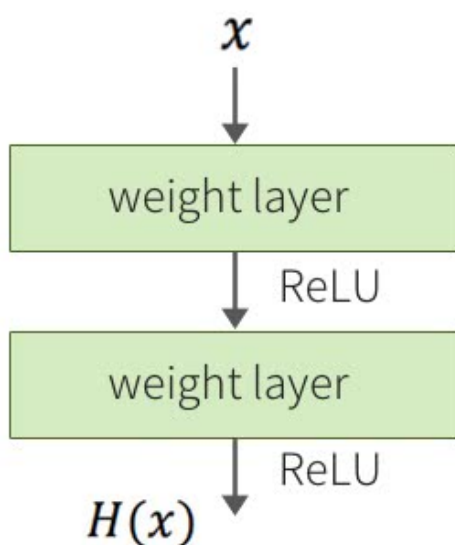
単純に層を深くすると、性能が悪化することが報告されていた。

では、ResNetはどのようにしてニューラルネットワークのモデルを深くすることを可能にしたのだろうか？

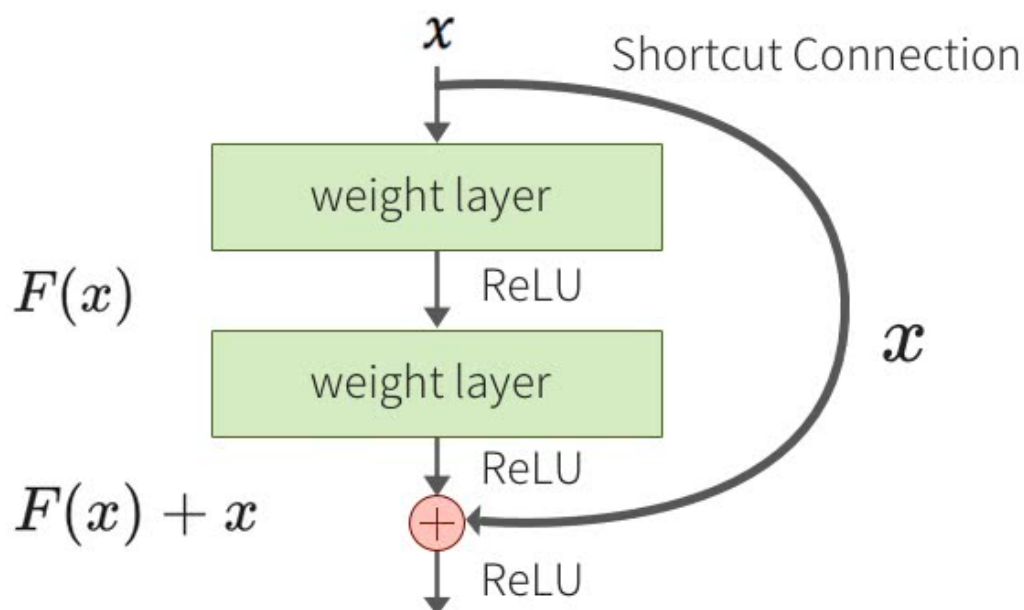
ResNetのアイデアはシンプルで、「ある層で求める最適な出力を学習するのではなく、層の入力を参照した残差関数を学習する」ことで最適化しやすくしている。

つまり、 $H(x)$ が学習して欲しい関数だとすると、入力との差分は $F(x) := H(x) - x$ となり、 $H(x) := F(x) + x$ を学習するように再定義する。

Shortcut Connectionの導入



残差ブロックはこうなる。



残差ブロックでは、畳込み層とSkip Connectionの組み合わせになっている。2つの枝から構成されていて、それぞれの要素を足し合わせる。残差ブロックの一つはConvolution層の組み合わせで、もう一つはIdentity関数となる。

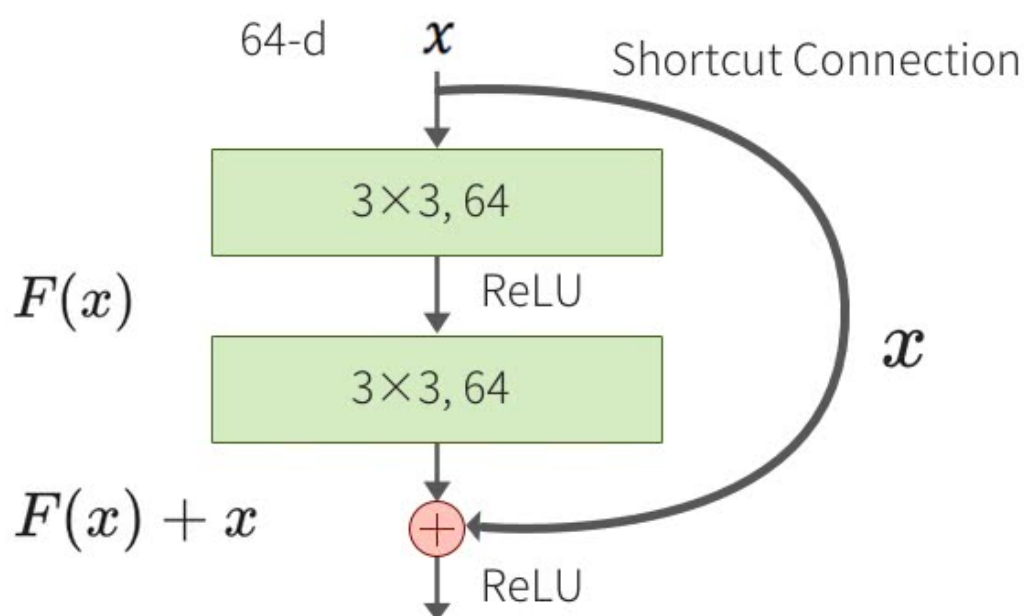
こうすれば、仮に追加の層で変換が不要でもweightを0にすれば良い。

Bottleneckアーキテクチャ

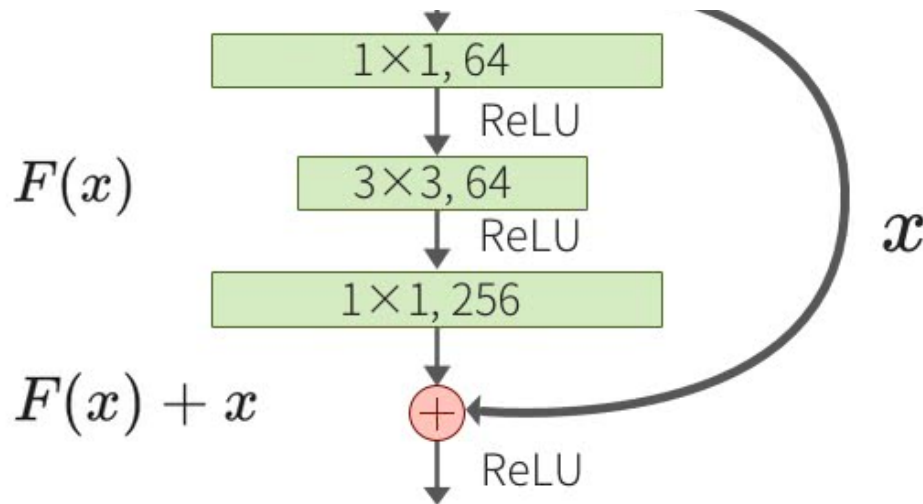
Residual Blockには2つのアーキテクチャがある。

- Plainアーキテクチャ
- Bottleneckアーキテクチャ

で、こちらがPlainアーキテクチャとなっている。



3×3 のConvolution層が2つある。そして、こちらがBottleneckアーキテクチャで残差ブロックの中が少しだけ変わっている。



この2つは同等の計算コストとなるが、Bottleneckアーキテクチャの方はPlainよりも1層多くなる。1×1と3×3のConvolution層で出力のDepthの次元を小さくしてから最後の1×1のConvolution層でDepthの次元を復元することからBottleneckという名前がついている。

ResNetの最適化ベストプラクティス

それでは、実際の実装におけるパラメータのチューニングポイントを紹介する。FacebookのTorchのブログにある[Training and investigating Residual Nets](#)と、ResNetの派生モデルの提案

- [Identity Mappings in Deep Residual Networks](#)
- [Wide Residual Networks](#)

を参考になっている。

Optimizerの選定

原論文には、SGD+Momentumを使うと書かれていたがTorchのブログでもこれが最良

- Adagrad
- SGD+Momentum

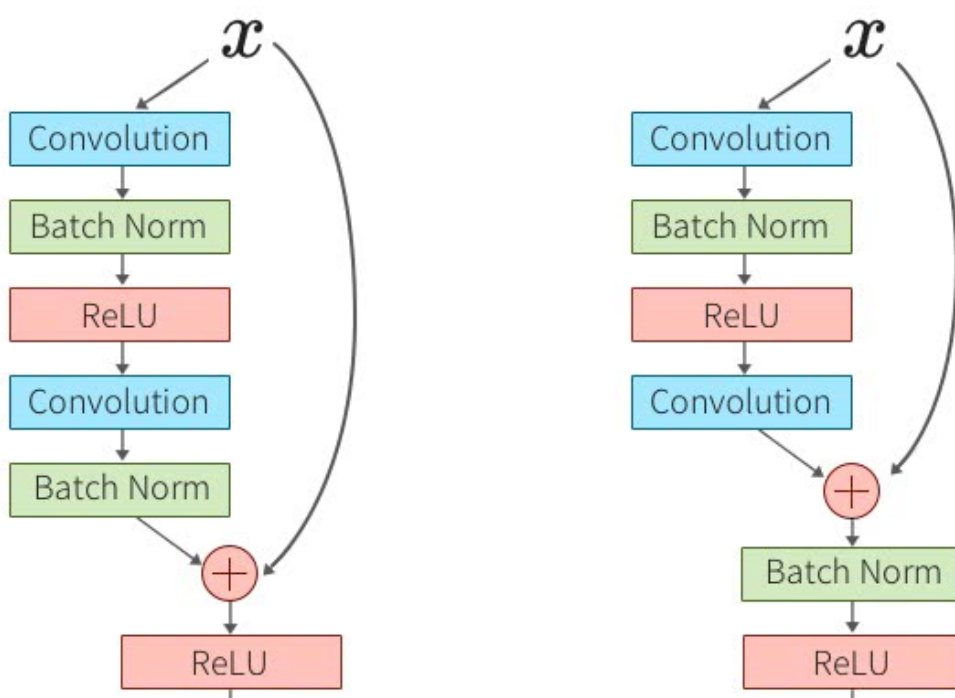
で調査したところ、唯一SGD+Momentumだけが0.7%を下回るテストエラーとなったそう。

Batch Normalizationの位置

Batch Normalizationを足し合わせる前に入れるか、足し合わせた後に入れるかをTorchのブログで検証している。

Batch Normalizationについてはこちら：

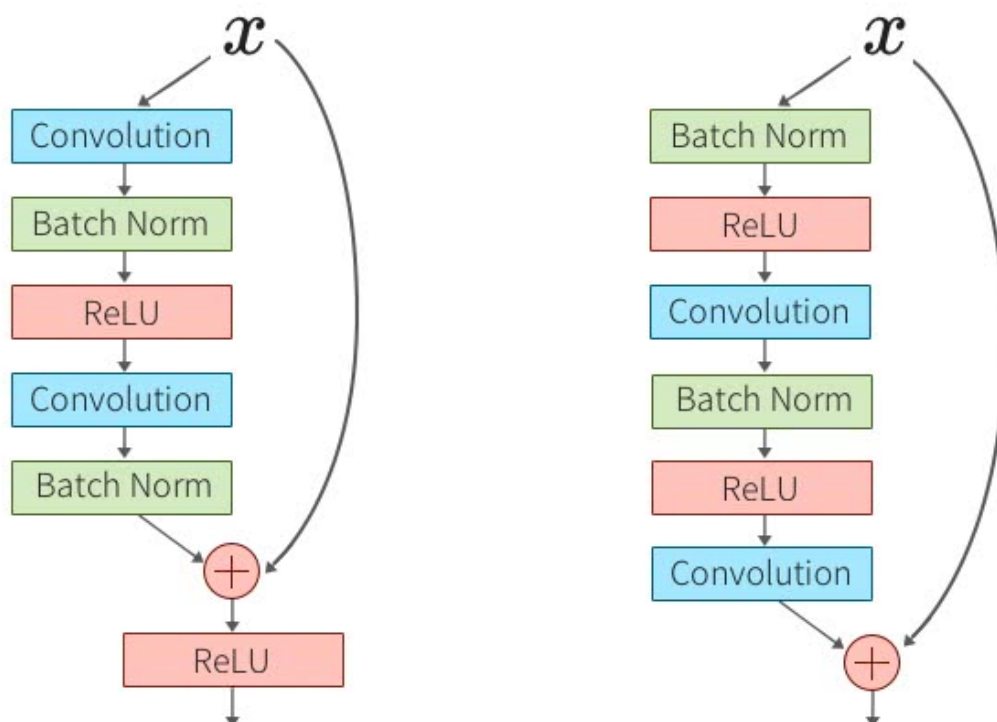
“[Batch Normalization：ニューラルネットワークの学習を加速させる汎用的で強力な手法](#)”



この検証結果は、Batch Normalizationを後に入れた場合、著しくテストデータにおける性能が落ちた。理由は、最後にBatch Normalizationをすれば残差ブロック全てが正規化されて良いように見えるが、実際にはSkip Connectionの情報をBatch Normalizationが大きく変更して情報の伝達を妨げてしまうからだそうだ。

Post Activation vs Pre Activation

次は、[Identity Mappings in Deep Residual Networks](#)における検証で下図のように元の構成が左のようになっていた場合、Activation関数とBatch Normalizationの位置を前方に持ってくるとどうなるだろうか？



この実験では、層の数が増えれば増えるほど顕著にActivation関数とBatch Normalizationを前に持ってきた後者のほうが良い結果となったそうだ。

画像認識の分類タスクImageNetにおいて、200層のResNetを使った実験ではエラーレート1.1%の改善をしている。

Wide Residual Network

最後はWide Residual Networkで、こちらはBatch NormalizationやActivation関数の順序ではなく、フィルタの数を増やしてネットワークを”広く”したらどうなるのか？

ということを検証したものだ。以下の表を見て欲しい。

group name	output size	block type = $B(3, 3)$
conv1	32×32	$[3 \times 3, 16]$
conv2	32×32	$\begin{bmatrix} 3 \times 3, 16 \times k \\ 3 \times 3, 16 \times k \end{bmatrix} \times N$
conv3	16×16	$\begin{bmatrix} 3 \times 3, 32 \times k \\ 3 \times 3, 32 \times k \end{bmatrix} \times N$
conv4	8×8	$\begin{bmatrix} 3 \times 3, 64 \times k \\ 3 \times 3, 64 \times k \end{bmatrix} \times N$
avg-pool	1×1	$[8 \times 8]$

N

が残差ブロックの数となっていて、 k が”広さ”係数だ。 \square の中のパラメータは、フィルタサイズ,フィルタの数で広いネットワークでは、フィルタの数が大幅に増えることになる。

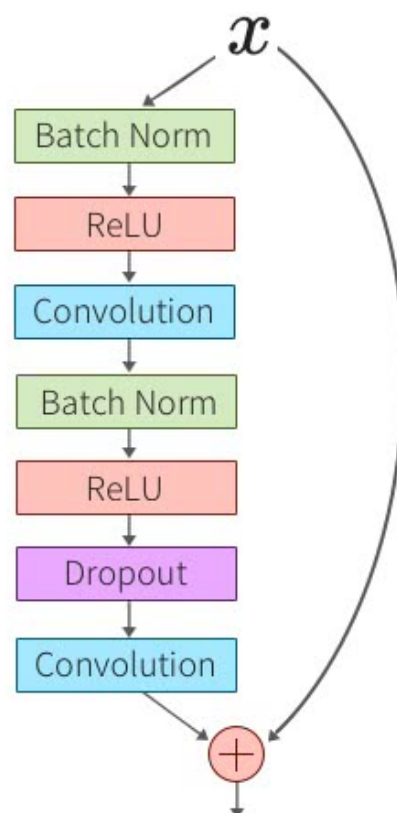
こうすることで、広さの係数を10倍にすると1000層の”薄い”ResNetと同等のパラメータ数を持つこととなり、さらにGPUによる処理をフルに活かすことができるようになる。

そして実験結果では、”薄い”ResNetよりも高い精度を50分の1の層数で、半分の時間の訓練時間になったそうだ。

Dropoutについてはこちら：

“Dropout：ディープラーニングの火付け役、単純な方法で過学習を防ぐ”

以下の図のように、最後のConvolution層の手前にDropoutを入れる。



広さの係数10で28層のWide ResNetにDropout率30~40%程度適用すると高い精度となったそうだ。

まとめ

ディープラーニングの今と一歩先を発信するメディア

人工知能の今と一歩先を発信するメディア

ResNetの実装や方針の参考にして欲しい。

参考

- [Deep Residual Learning for Image Recognition](#)
- [Training and investigating Residual Nets](#)
- [Identity Mappings in Deep Residual Networks](#)
- [Wide Residual Networks](#)

ツイート

いいね！

シェア

DeepAgeではAIに関する厳選した内容を記事にしてお届けします。
気に入って頂けたら応援お願いします！

フォローする

いいね！

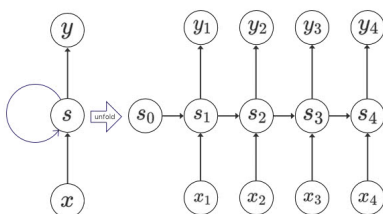


人工知能(AI)技術の事業活用に
興味はありますか？

人工知能の今と一歩先を発信するメディア

[事業概要・実績はこちら](#)[お問い合わせはこちら](#)

おすすめの投稿



RNN : 時系列データを扱う Recurrent Neural Networksとは

リカレントニューラルネットワークは、時系列データを扱うことのできるニューラルネットワークの

2017-May-23



Batch Normalization : ニューラルネットワークの学習を加速させる汎用的で強力な手法

ネットワーク内部の共変量シフトを抑えて、ニューラルネットワークの学習を加速させるBatch

2016-Oct-26



レコメンドに浸透していく Deep Learning : 大手サービスの実用例から最新アルゴリズムを概観する

各所で利用されるようになったディープラーニング。レコメンドでの応用例はどうなっているのだ

2016-Sep-26

記事一覧

[すべて](#)
[ビッグデータ](#)
[機械学習](#)

DeepAgeについて

[トップページ](#)
[採用情報](#)
[運営会社](#)

[すべて](#)[Numpy](#)[ビッグデータ](#)[機械学習](#)[人工知能](#)[ディープラーニング](#)[TensorFlow](#)

DeepAge 人工知能・機械学習・データビジネスに役立つ記事を紹介

約2年前

ブログのNumPyコラムが本になりました

NumPyの基本からニューラルネットワークの構築まで実践的な書籍として翔泳社様から出版されました。

社員研修や勉強のため、業務のために手に置いておきたい方へぜひお買い

© [Spot Inc.](#) 2016