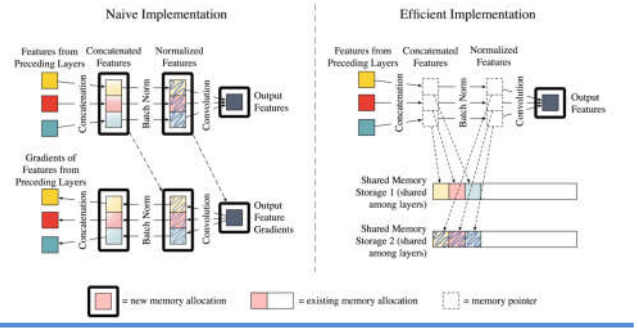
Paper

DenseNet

GitHub

- Introduction
  - What DenseNet is
  - Reading resource
- Usage/Result
  - Code
    - different version
      - original
      - BC
      - efficient



    - different framework
  - Result in different dataset
- Service
  - pre-trained models
  - suggest

wide DenseNet(Less Depth, More Growing-rate) preferred, when memory is limit.
tips: deeper means more intermediate value should be saved.

Model	Parameters	CIFAR-10+	CIFAR-100+	Time per Iteration	Memory
DenseNet-BC (L=100, k=12)	0.8M	4.51	22.27	0.156s	5452MB
Wide-DenseNet-BC (L=40, k=36)	1.5M	4.58	22.30	0.130s	4008MB
Wide-DenseNet-BC (L=40, k=48)	2.7M	3.99	20.29	0.165s	5245MB
Wide-DenseNet-BC (L=40, k=60)	4.3M	4.01	19.99	0.223s	6508MB

- Other

GitHub

Introduction

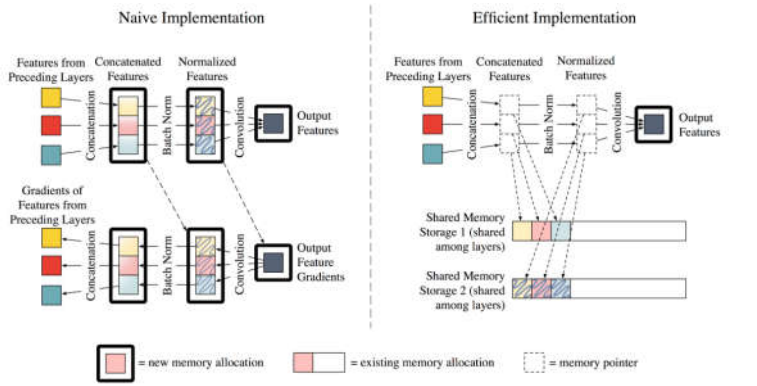
- What DenseNet is
- Reading resource

Usage/Result

Code

different version

- original
- BC
- efficient



different framework

Result in different dataset

Service

pre-trained models

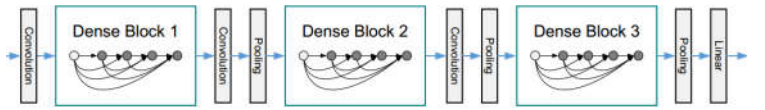
wide DenseNet(Less Depth, More Growing-rate) preferred, when memory is limit.
tips: deeper means more intermediate value should be saved.

suggest

Model	Parameters	CIFAR-10+	CIFAR-100+	Time per Iteration	Memory
DenseNet-BC (L=100, k=12)	0.8M	4.51	22.27	0.156s	5452MB
Wide-DenseNet-BC (L=40, k=36)	1.5M	4.58	22.30	0.130s	4008MB
Wide-DenseNet-BC (L=40, k=48)	2.7M	3.99	20.29	0.165s	5245MB
Wide-DenseNet-BC (L=40, k=60)	4.3M	4.01	19.99	0.223s	6508MB

Other

Paper

- layout
  

- term
  - dense layer — BN->Relu->Conv
  - dense block — only has dense layer
  - transition layer — separate dense block
  - growing rate — #channel of output of dense layer
  - global state of net — a set of state of layer
  - state of layer — output of dense layer
- consideration
  - randomly layer dropout demonstrates redundancy of ResNet
  - concatenation of features in different level can reduce redundancy
  - concatenation can make net flow more fluent than summation, and need more memory in the condition of same #param (Deep Supervision)
- advantage
  - alleviate the vanishing-gradient problem
  - encourage feature reuse
  - reduce #param and #data
  - without overfitting and optimization difficulty
- trick
  - SGD
  - weight decay
  - Nesterov momentum of 0.9 without dampening
  - weight initialization
  - Wider than Deep under limit memory
- enlightenment
  - when technology is mature, efficiency is concentration for engineer