Pruning neural networks

Optimizing AI - Session 3

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Course organisation

Sessions

1. Deep Learning and Transfer Learning,
2. Quantification,
3. Pruning,
4. Factorization,
5. Distillation,
6. Operators and Architectures,
7. Embedded Software and Hardware for DL.
8. Presentations for challenge.

Course organisation

Sessions

1. Deep Learning and Transfer Learning,
2. Quantification,
3. Pruning,
4. Factorization,
5. Distillation,
6. Operators and Architectures,
7. Embedded Software and Hardware for DL.
8. Presentations for challenge.

Overview of pruning

Definition

Reduce the number of parameters by eliminating neurons or connections.

Table: Comparison of obtained top-1 accuracy, number of parameters (NP) and pruning ratio (PR) on CIFAR10, CIFAR100 and ImageNet of different pruning methods applied on ResNet (RN)

Method	Network	Dataset	Baseline	Pruning	NP(M)	PR
PCAS	RN-56	C10	93.04%	93.58%	0.39	53.7%
PCAS	RN-50	C100	74.66%	73.83%	4.02	76.5%
AMC	RN-50	C10	93.53%	93.55%	NA	60.0%
ThiNet	RN-50	ImNet	72.88%	72.04%	16.94	33.7%

Pruning on pretrained networks

Basic principle (most common)

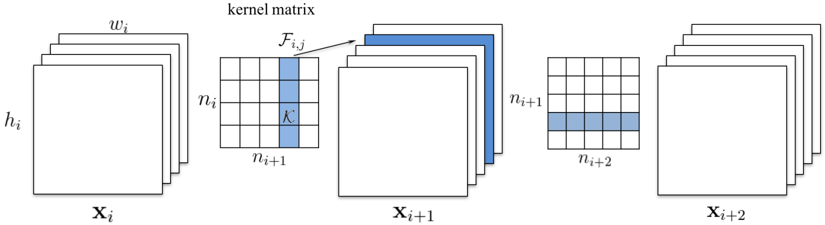
1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

Pruning on pretrained networks

Basic principle (most common)

1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

Rank filters / weights using $\sum |\mathbf{W}_{l,i,:,:,:}|$, and prune lowest filters and feature maps, then finetune. Li et al. 2016, `https://arxiv.org/abs/1608.08710`

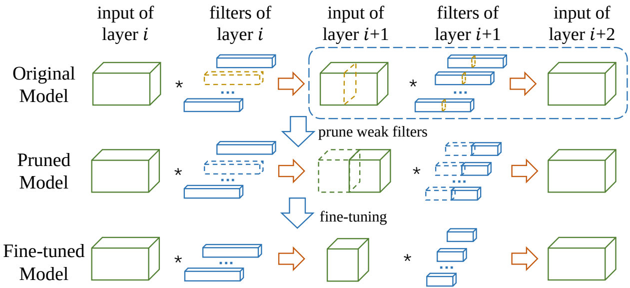


Pruning on pretrained networks

Basic principle (most common)

1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

ThiNet: rank and prune feature Maps directly.
Luo et al. 2017, `https://arxiv.org/abs/1707.06342`



Pruning on pretrained networks

Basic principle (most common)

1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

Other methods

- AutoML for Model Compression (AMC) uses reinforcement learning with a negative reward defined on the number of floating point operations
  He et al. 2018, `https://arxiv.org/abs/1802.03494`
- Pruning Channel with Attention Statistics (PCAS) uses a pretrained network, and adds an "attention" layer that learns feature map importance.
  Yamamoto and Maeno, 2018, `https://arxiv.org/abs/1806.05382`

Pruning on pretrained networks

Basic principle (most common)

1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

Other methods

- AutoML for Model Compression (AMC) uses reinforcement learning with a negative reward defined on the number of floating point operations
  He et al. 2018, `https://arxiv.org/abs/1802.03494`
- Pruning Channel with Attention Statistics (PCAS) uses a pretrained network, and adds an "attention" layer that learns feature map importance.
  Yamamoto and Maeno, 2018, `https://arxiv.org/abs/1806.05382`

Pruning on pretrained networks

Basic principle (most common)

1. Rank the importance of neurons
2. Eliminate the least important neurons
3. Fine-tune the whole network to restore accuracy

Table: Comparison of obtained top-1 accuracy, number of parameters (NP) and pruning ratio (PR) on CIFAR10, CIFAR100 and ImageNet of different pruning methods applied on ResNet (RN)

Method	Network	Dataset	Baseline	Pruning	NP(M)	PR
PCAS	RN-56	C10	93.04%	93.58%	0.39	53.7%
PCAS	RN-50	C100	74.66%	73.83%	4.02	76.5%
AMC	RN-50	C10	93.53%	93.55%	NA	60.0%
ThiNet	RN-50	ImNet	72.88%	72.04%	16.94	33.7%

Pruning while training (experimental)

(very) Recent papers have tried to prune networks while training, instead of using pretrained networks.

- Automatic Network Pruning by Regularizing Auxiliary Parameters, Xiao et al. NIPS 2019.
- Soft Threshold Weight Reparameterization for Learnable Sparsity, preprint february 2020 `https://arxiv.org/pdf/2002.03231.pdf`
- BitPruning: Learning Bitlengths for Aggressive and Accurate Quantization `https://arxiv.org/abs/2002.03090`

Lab Session and Project

Lab Session

- Implement one of the pruning methods from this course
- Apply it on MiniCIFAR

Presentation at next session

Present your current explorations on MiniCIFAR, CIFAR10 and / or CIFAR100 using the methods seen so far!