Cooperative Pruning in **Cross-Domain** Deep Neural Network **Compression**

Shangyu Chen, Wenya Wang, Sinno Jialin Pan

School of Computer Science and Engineering
Nanyang Techonological University, Singapore

NANYANG TECHNOLOGICAL UNIVERSITY

Outline

Background & Motivation

Cooperative Pruning

Experiment

Conclusion

NANYANG TECHNOLOGICAL UNIVERSITY

Deep Neural Network Compression

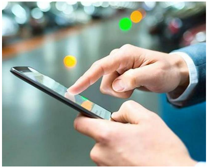


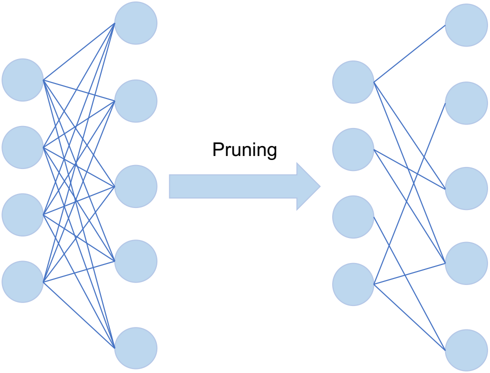
Figure 1: Smartphones

Figure 2: Cameras

- More and more deep learning applications are deployed in edge devices: cellphone, surveillance camera, etc.
- It is impossible to perform inference without optimization:
  - * Compress the model: Prune parameters from the redundant model.
  - * Accelerate computation: Skip computation with parameters as 0 (pruned).

NANYANG TECHNOLOGICAL UNIVERSITY

Pruning



$$\begin{pmatrix} 0.6 & 0.5 & -0.1 \\ 1.2 & -0.1 & -0.2 \\ 0.5 & -0.3 & -1.2 \end{pmatrix}$$

Pruning

$$\begin{pmatrix} 0.6 & 0.5 & 0 \\ 1.2 & 0 & 0 \\ 0.5 & 0 & -1.2 \end{pmatrix}$$

NANYANG TECHNOLOGICAL UNIVERSITY

Limitation of Current Pruning

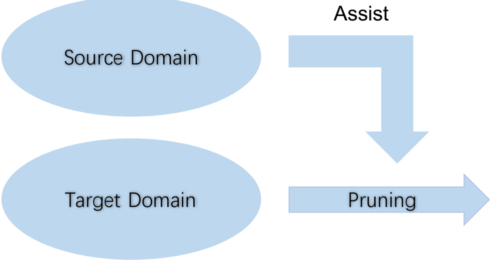
Most existing pruning methods rely access to **large amount** of training data, however:

- Training data is limited due to difficulty of collection.
- Data privacy in commercial models with high confidential requirement.

Question: How can we train a pruned neural network with limited training data?

NANYANG TECHNOLOGICAL UNIVERSITY

Using Data from Other Domain



- Can we leveage knowledge from other domains (data-affluent) to assist pruning under limited data?

Task: Transfer knowledge from other domains to improve pruning: *Cross-Domain Deep Neural Network Compression*.

NANYANG TECHNOLOGICAL UNIVERSITY

Traditional (Static) Pruning

- Given a pre-trained neural network model $\mathbf{W}$.
- Find a pruning mask $\mathbf{M}$: Whether a parameter is pruned or not (denoted by 0 or 1).
  - e.g. Prune parameters based on their **absolute magnitude**.
- The produced pruned model: $\mathbf{W}^{'} = \mathbf{W}\mathbf{M}$
  - Fine-tune the un-pruned parameters.
- This process is one-time pruning.

NANYANG TECHNOLOGICAL UNIVERSITY

Dynmaic Pruning

- Given a pre-trained neural network model $\mathbf{W}$.

- During training, pruning mask $\mathbf{M}$ is forwarded via:

$$\text{Forward}: \qquad L = \text{Loss}(\mathbf{W}^{'}) = \text{Loss}(\mathbf{WM}) \tag{1}$$

  Gradient of $\mathbf{W}$ is un-attainable due to the non-derivative of $\mathbf{M}$.

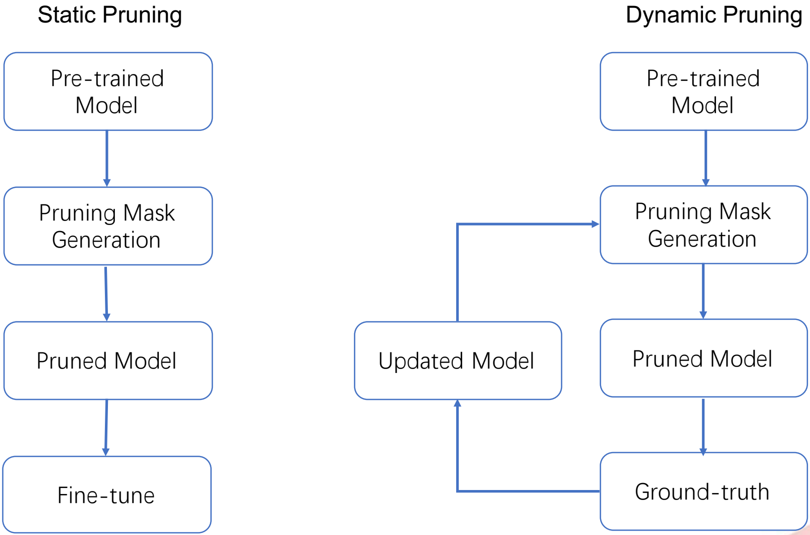
- Backward modification:

$$\text{Backward}: \qquad \frac{\partial L}{\partial \mathbf{W}} = \frac{\partial L}{\partial \mathbf{W}'} \tag{2}$$

- $\mathbf{M}$ is re-generated based on the updated value of $\mathbf{W}$.

NANYANG TECHNOLOGICAL UNIVERSITY

Pruning Algorithm



Outline

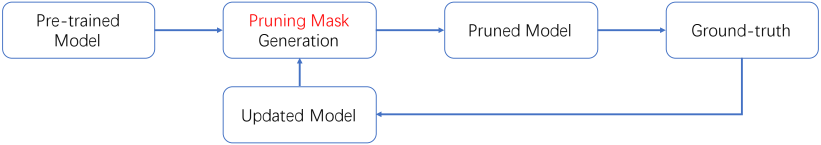
Finding a Bridge

Task: Transfer knowledge from other domains to improve pruning.

Key Questions:

- What to transfer ?
- How to transfer ?

Remember Dynamic Pruning:



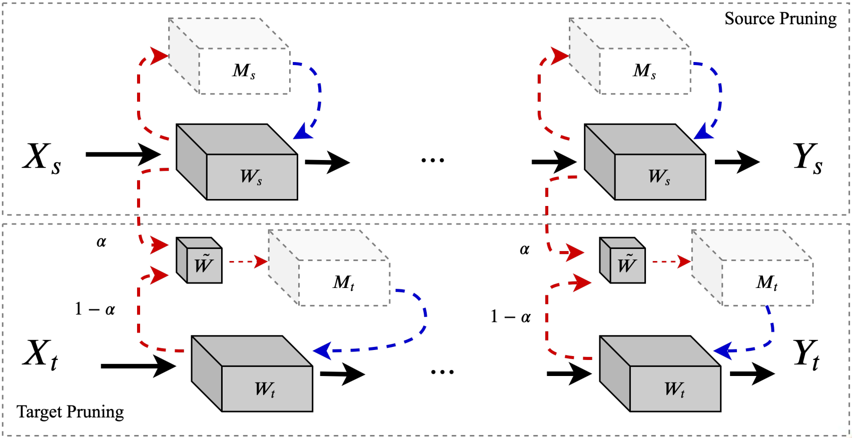
Finding a Bridge

Key Questions:

- What to transfer ?
  - Pruning mask: Knowledge is embedded in mask for transfer.
- How to transfer ?
  - Cooperative Pruning: Source / target task is **dynamically prunned** together.
  - Pruning mask is generated based on the absoluted maganitude of updated weights.

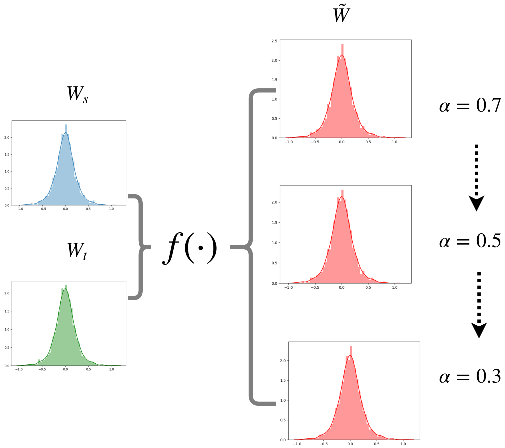
NANYANG TECHNOLOGICAL UNIVERSITY

Cooperative Pruning (Co-Prune)

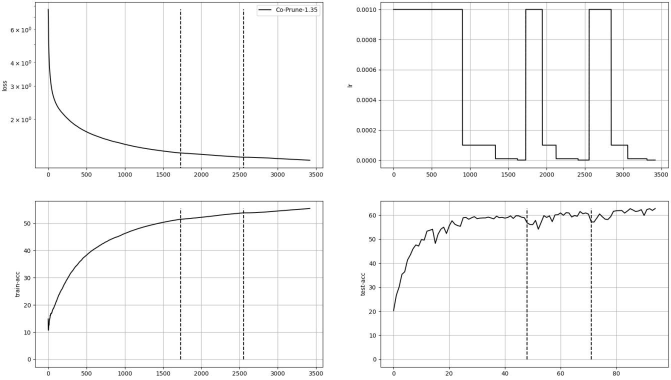


Transfer Factor α

- α determines "how much knowledge" is transfered to target.
- α decreases during the whole process.



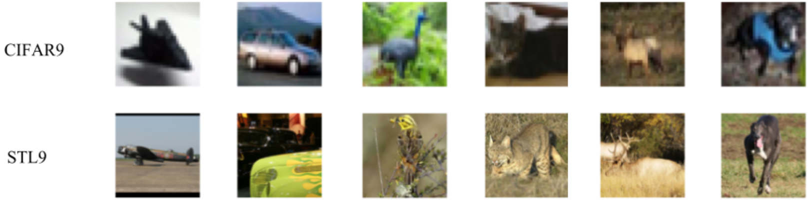
Transfer Factor α (Cont.)



Outline

Dataset: CIFAR9-STL9

- CIFAR10: 50k training data, STL10: 5k labeled training data.
- Non-overlap category is removed.
- CIFAR9: 45k training data, STL9: 4.5k labeled training data.



Experiment Results

CR (%)	Method	FP Acc (%)	Prune Acc (%)
10.4	LWC	68.03	66.26
	OBD		65.78
	DNS		66.25
	L-OBS		66.01
	DDC-DNS		66.49
	Co-Prune		**66.99**
1.3	LWC		57.47
	OBD		50.82
	DNS		58.89
	L-OBS		56.00
	DDC-DNS		56.79
	Co-Prune		**60.5**
	One-Time Co-Prune		55.36
	Distillation		53.16

Table 1: Overall results of CIFAR9-STL9 using CIFAR-Net

NANYANG TECHNOLOGICAL UNIVERSITY

Outline

Conclusion

- We proposed a framework (Co-Prune) to transfer knowledge from other domains to improve pruning in **limited-data** scenario.
- Co-Prune is conducted by **cooperatively** training different pruning models.
- Pruning mask for target domain is generated by weighted-sum of parameters from target / source neural network models.
- Codes are released at: `https://github.com/csyhhu/Co-Prune`
- Awesome project on Deep Neural Network Compression: `https://github.com/csyhhu/Awesome-Deep-Neural-Network-Compression`