Introduction to DNN Model Compression Techniques

Sabina Pokhrel
Xailient

2021 embedded VISION summit VIRTUAL | MAY 25-28

Xailient
See what matters

Outline of the Talk

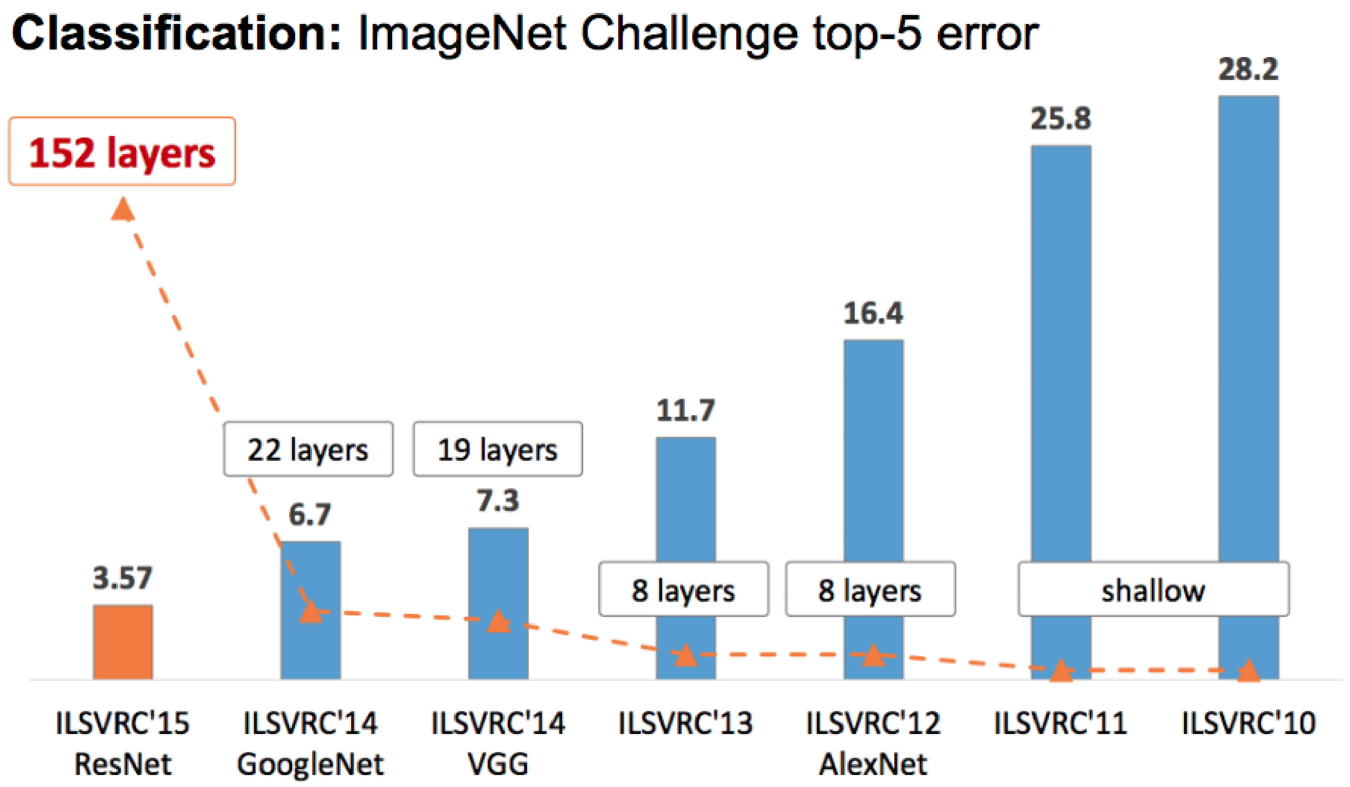
- Need for model compression
- Introduction to model compression
- Different model compression techniques:
    - Pruning
    - Quantization
    - Knowledge distillation
    - Low-rank factorization

History Of Deep Neural Networks

AlexNet achieves a very large margin of improvement with deeper network than before.

The number of layers is continuously increasing as the accuracy increases.

ResNet conquers the barrier of training network over 100 layers, much deeper than previous winners.



The ImageNet ILSVRC challenge results (top-5 error (%)) from 2010 to 2015.

Need for Model Compression

- Many real-world applications demand real-time, on-device processing capabilities.
- Deep learning models that perform well are big.
- Inference on edge device.
- Difficult to deploy on resource constrained devices.

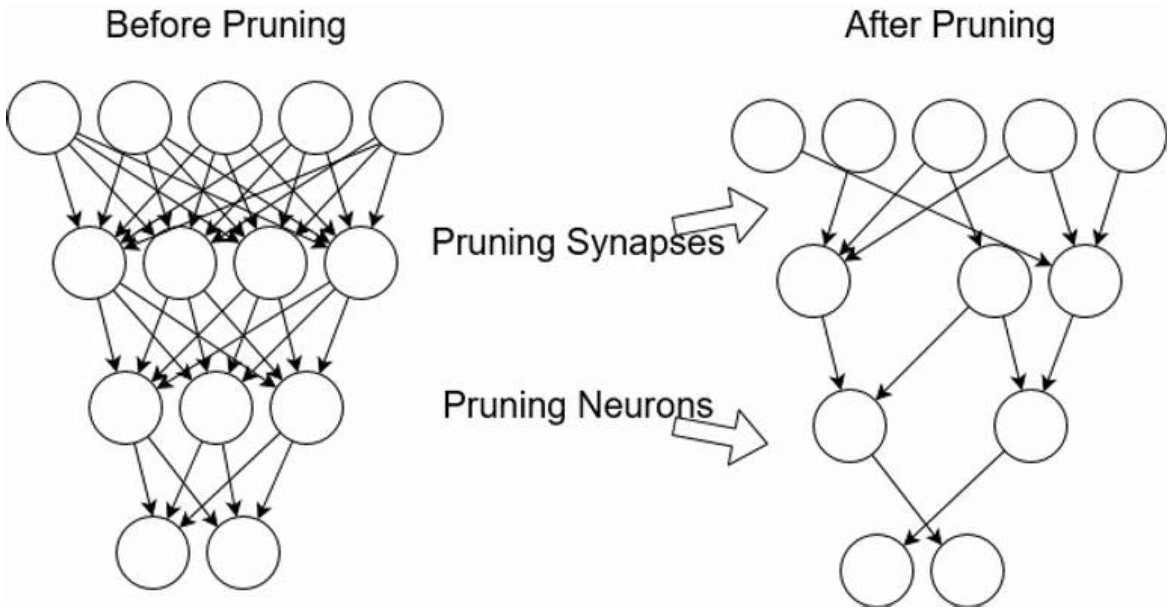
Introduction to Model Compression

- Technique to reducing size of neural network without compromising too much on accuracy.
- Resulting models are both memory and energy efficient.
- Different model compression techniques:
  - Pruning
  - Quantization
  - Knowledge distillation
  - Low-rank factorization

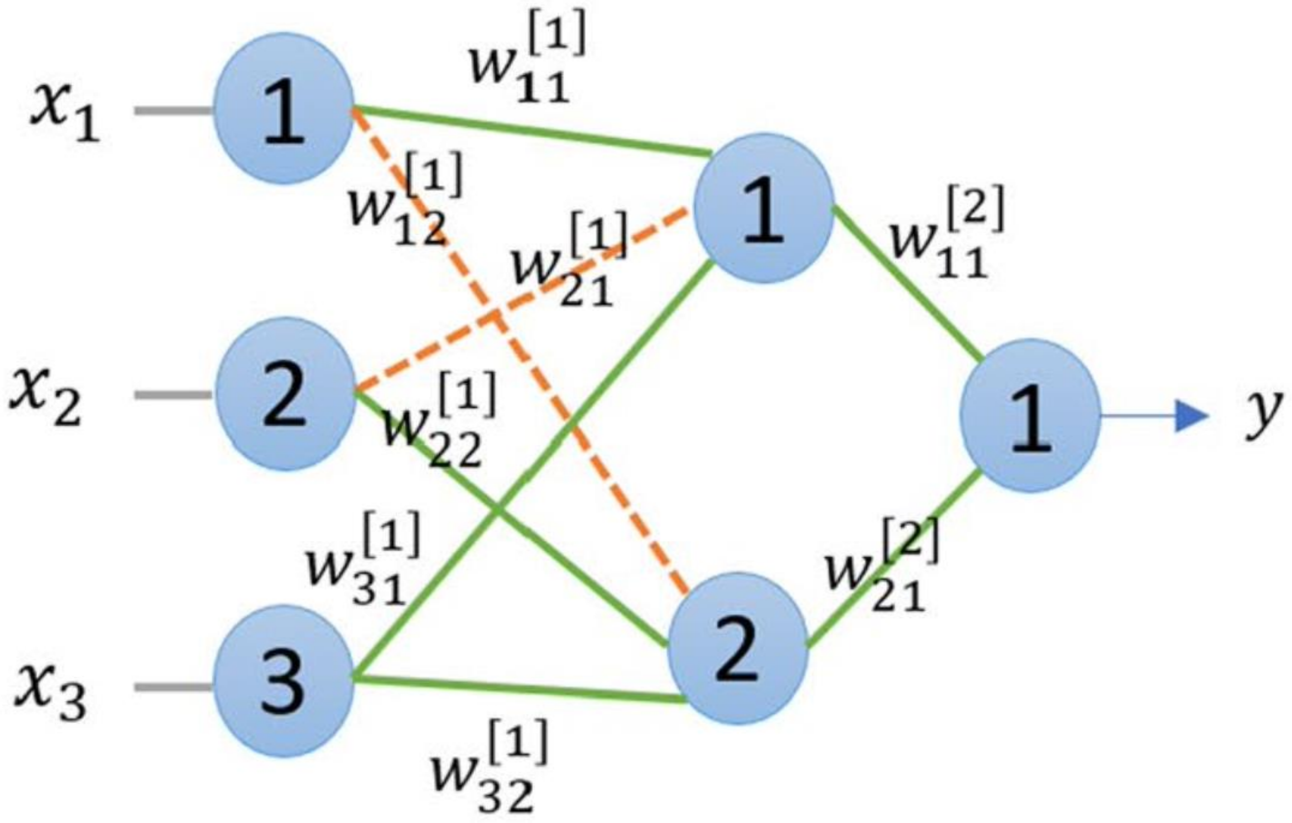
See what matters

Pruning

- Compresses the model to a smaller size with zero or marginal loss of accuracy.
- Reduces the number of parameters by removing redundant, and unimportant connections that do not affect model accuracy.
- Results in compressed neural networks that run faster.
- Reduces the computational cost involved in network training.
- Reduces the overall model size, saves computation time and energy.

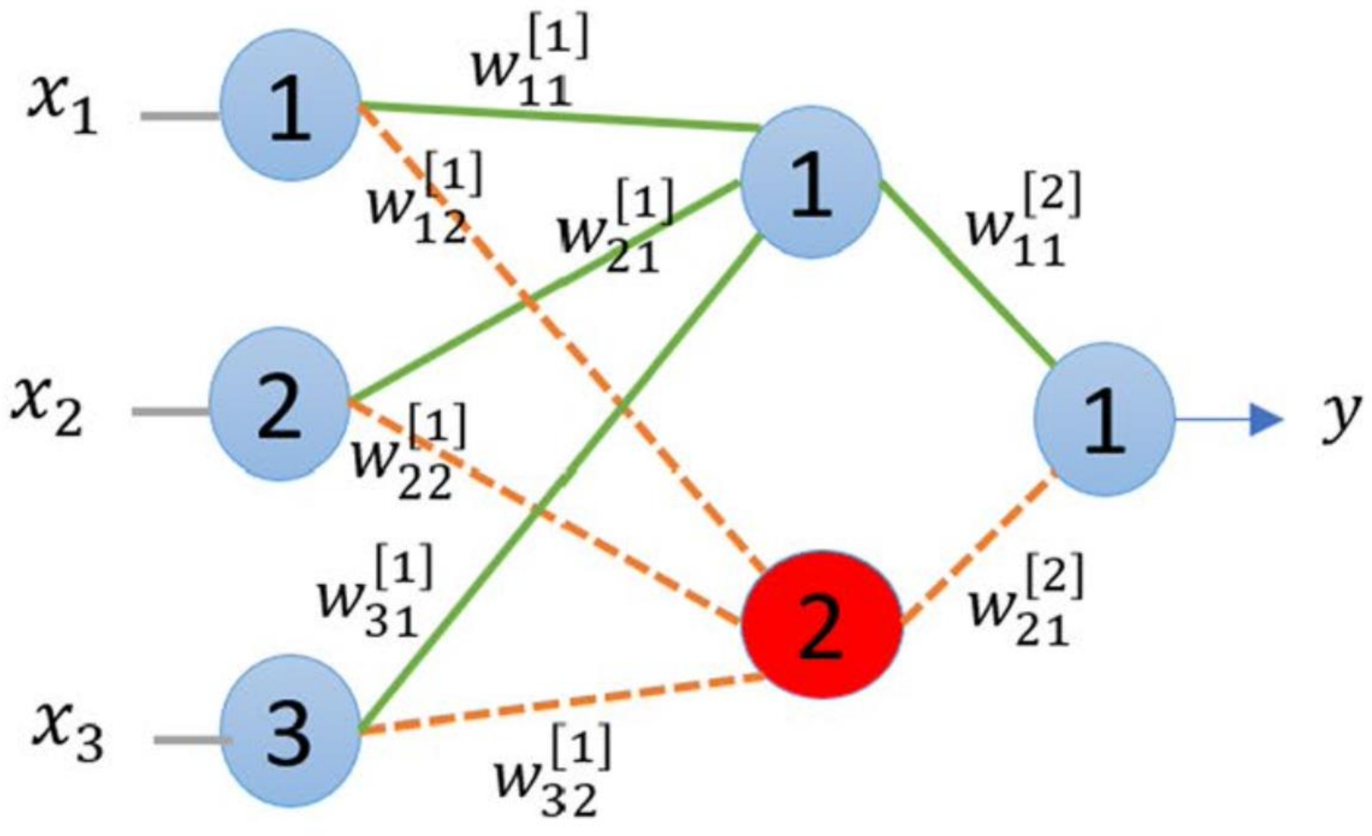


Pruning - Types



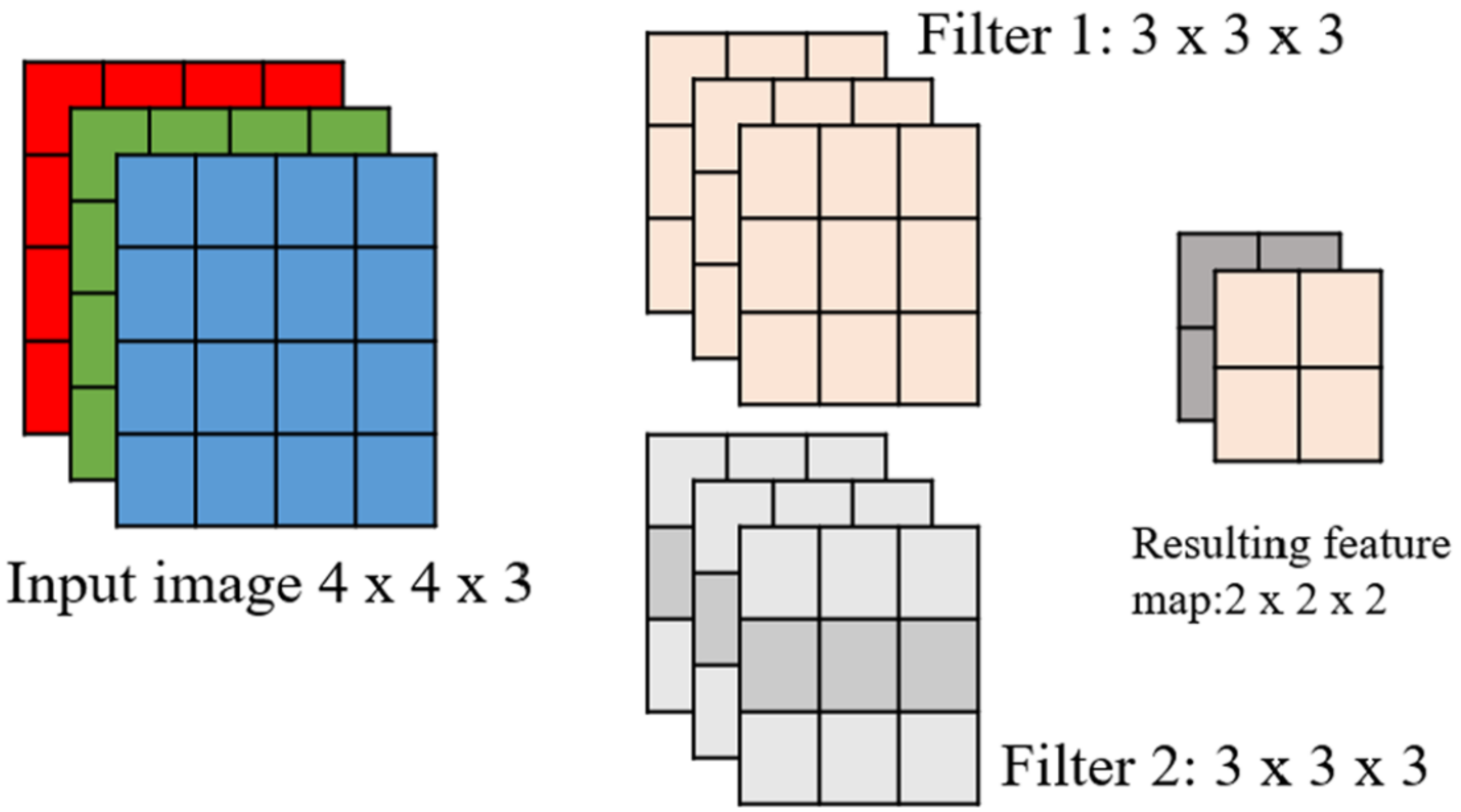
Weight Pruning

Pruning - Types



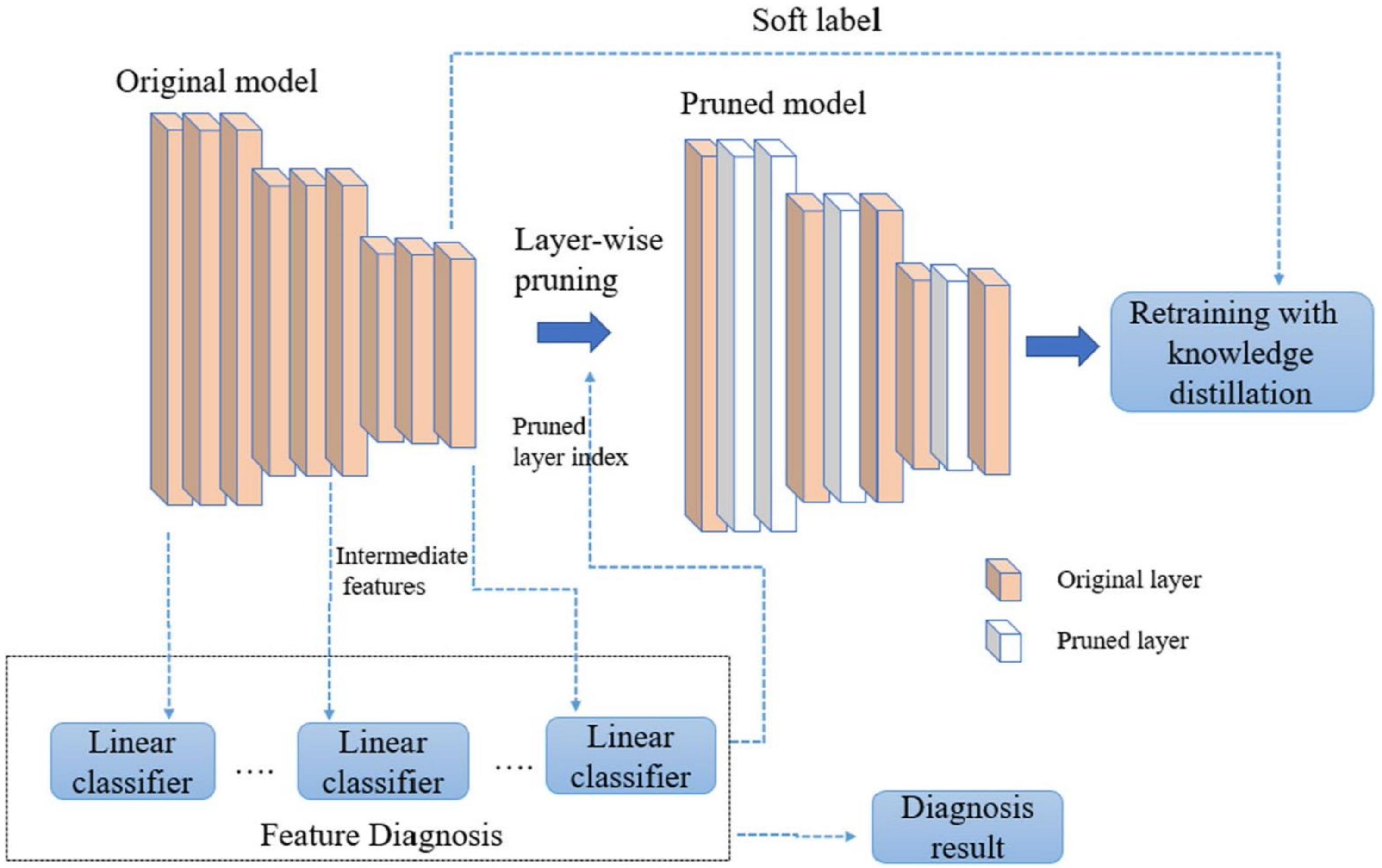
Neuron Pruning

Pruning - Types



Filter Pruning

Pruning - Types



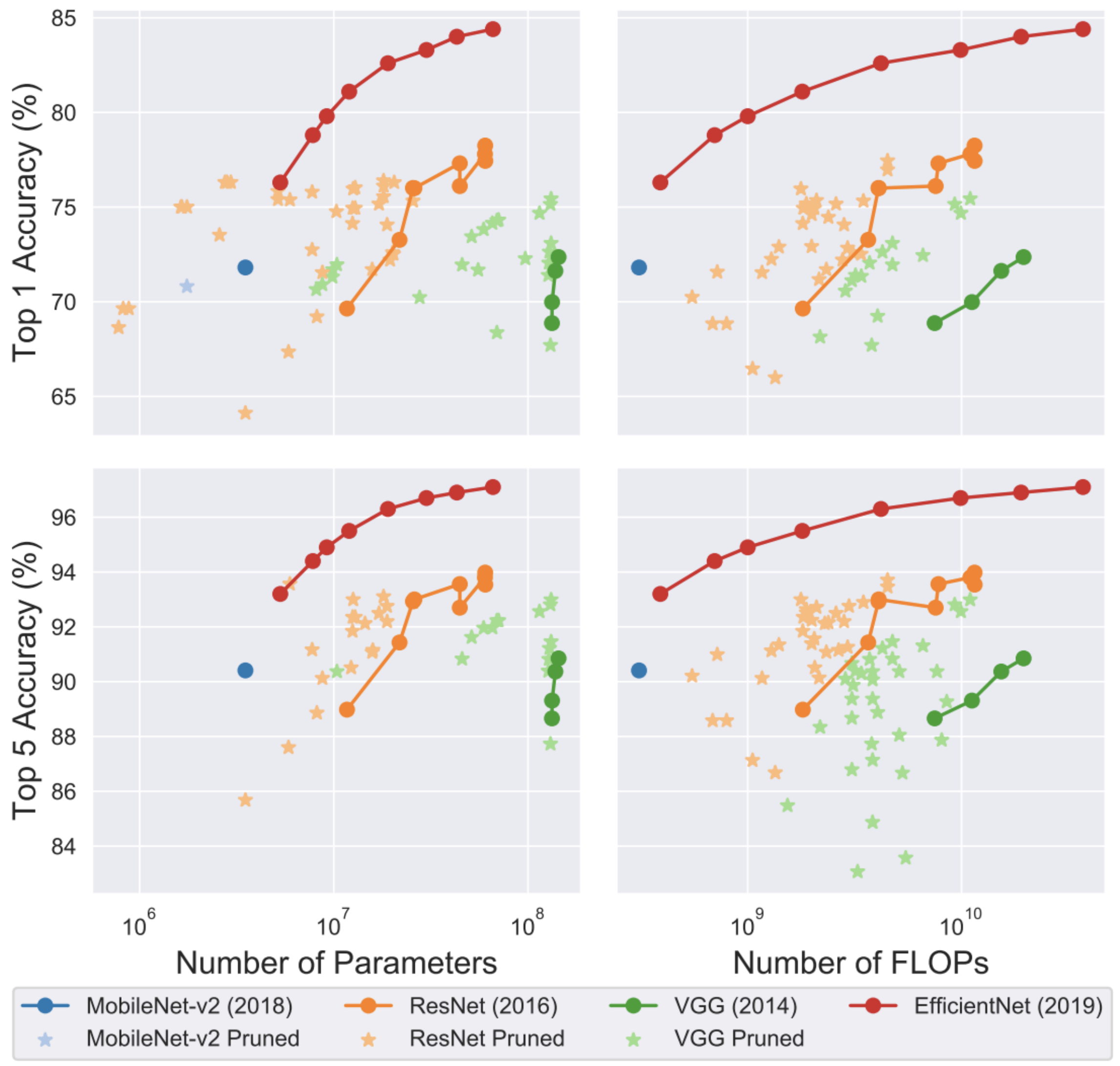
Layer Pruning

Pruning – Speed and Size Tradeoffs

Size and speed vs accuracy trade-offs for different pruning methods and families of architectures.

Pruned models sometimes outperform the original architecture, but rarely outperform a better architecture.

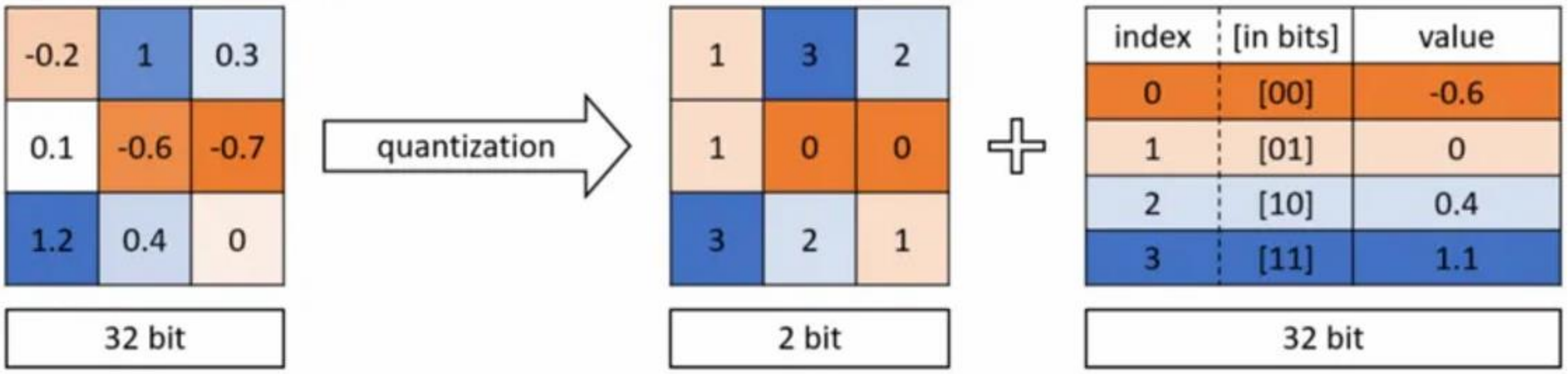
Speed and Size Tradeoffs for Original and Pruned Models



Pruning - Summary

- Can be applied during or after training.
- Can be applied to both convolutional and fully connected layers.
- Both the original and pruned model have the same architecture, with the pruned model being sparser (weights with the low magnitude being set to zeros).
- Pruned models sometimes outperform the original architecture, but rarely outperform a better architecture.

Quantization



-0.2	1	0.3
0.1	-0.6	-0.7
1.2	0.4	0

1	3	2
1	0	0
3	2	1

index	[in bits]	value
0	[00]	-0.6
1	[01]	0
2	[10]	0.4
3	[11]	1.1

Quantization

- Compresses the original network by reducing the number of bits required to represent each weight.
- The weights can be quantized to 16-bit, 8-bit, 4-bit or even with 1-bit.
- Per-channel quantization and/or per-layer quantization of weights.
- By reducing the number of bits used, the size of the deep neural network can be significantly reduced.
- Reduces inference time for hardware platforms supporting lower precision arithmetic

See what matters

Quantization - Types

- There are two forms of quantization:
  - Post-training quantization
  - Quantization aware training

Model	Non-quantized Top-1 Accuracy	8-bit Quantized Accuracy
MobilenetV1 224	71.03%	71.06%
Resnet v1 50	76.3%	76.1%
MobilenetV2 224	70.77%	70.01%

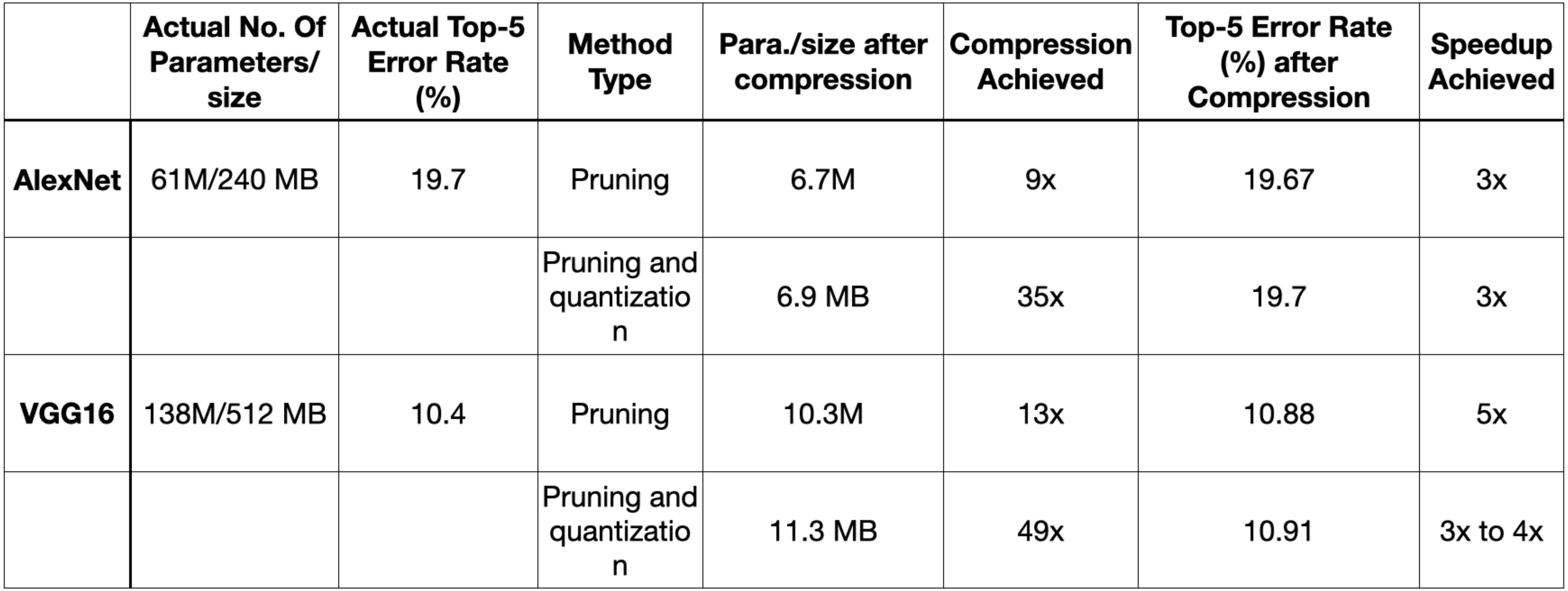
Comparison of Quantized and non-quantized models on ImageNet

See what matters

Quantization - Summary

- Quantization can be applied both during and after training.
- Can be applied to both convolutional and fully connected layers.
- Quantized weights make neural networks harder to converge. A smaller learning rate is needed to ensure the network to have good performance.
- Quantized weights make back-propagation infeasible since gradient cannot back-propagate through discrete neurons. Approximation methods are needed to estimate the gradients of the loss function with respect to the input of the discrete neurons.

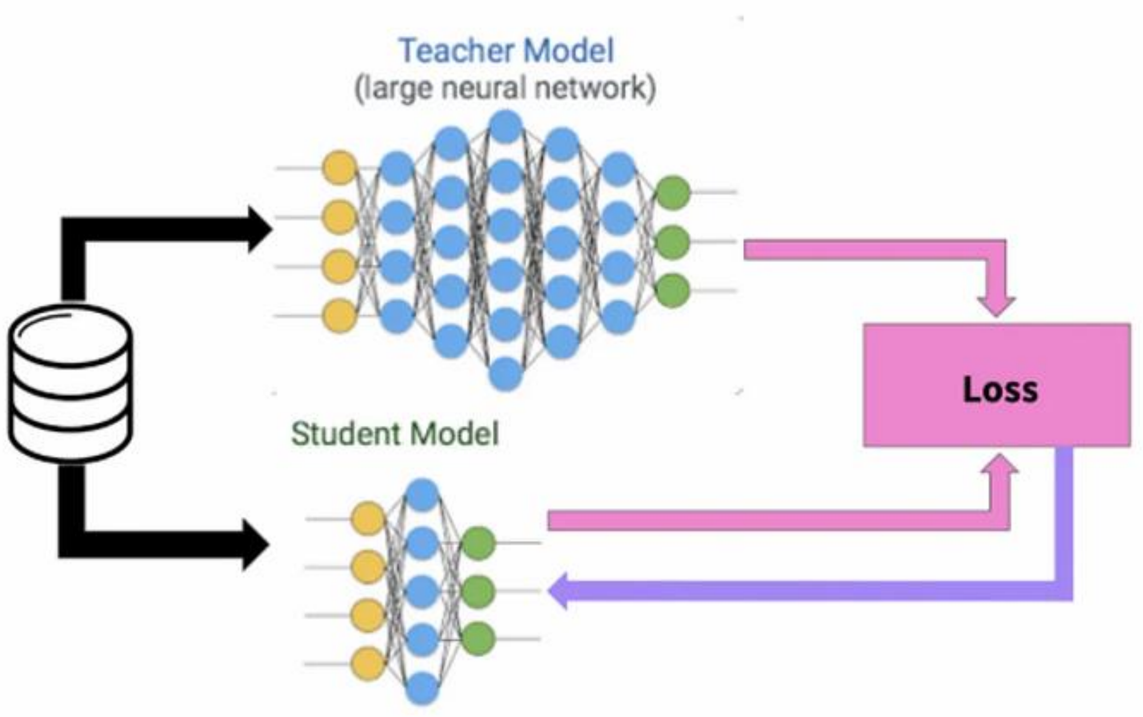
Comparison of Compression Techniques

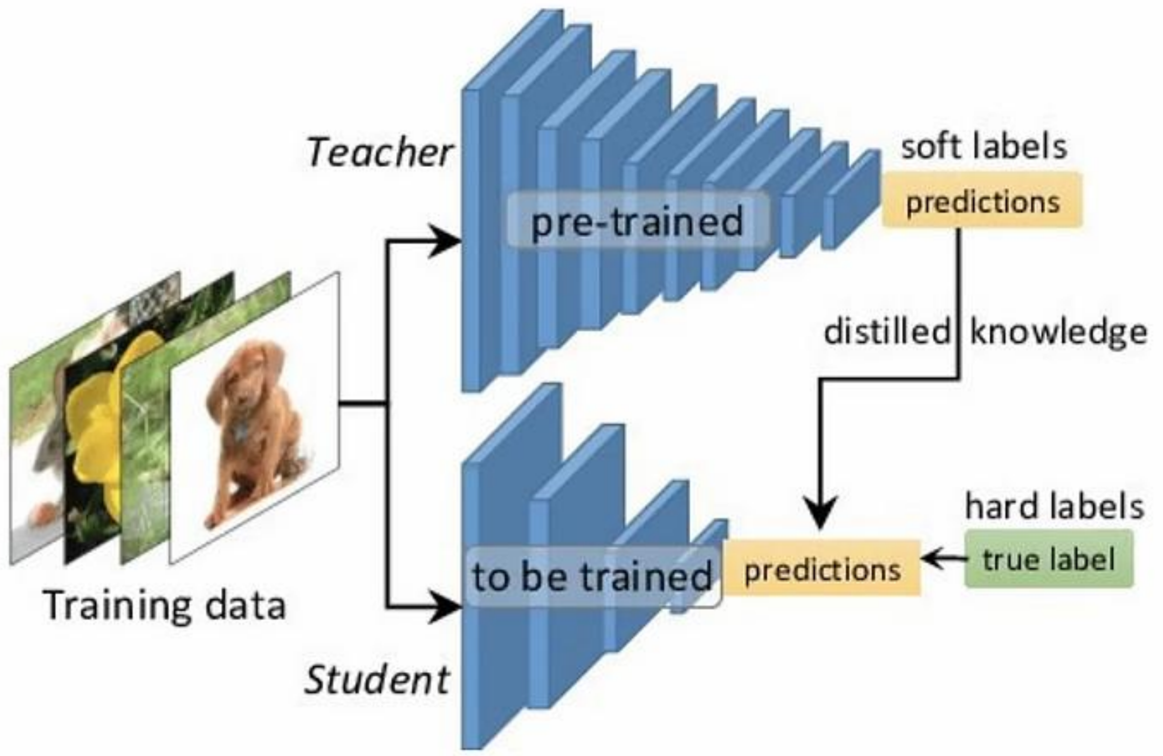


	Actual No. Of Parameters/ size	Actual Top-5 Error Rate (%)	Method Type	Para./size after compression	Compression Achieved	Top-5 Error Rate (%) after Compression	Speedup Achieved
AlexNet	61M/240 MB	19.7	Pruning	6.7M	9x	19.67	3x
			Pruning and quantizatio n	6.9 MB	35x	19.7	3x
VGG16	138M/512 MB	10.4	Pruning	10.3M	13x	10.88	5x
			Pruning and quantizatio n	11.3 MB	49x	10.91	3x to 4x

Table Comparison of Different DNN compression techniques on the ImageNet dataset and standard pre-trained models

Knowledge Distillation





Knowledge Distillation

- In knowledge distillation, a large, complex model is trained on a large dataset.
- When this large model can generalize and perform well on unseen data, the knowledge is transferred to a smaller network.
- The larger model is also known as the teacher model and the smaller network is also known as the student network.
- In knowledge distillation, knowledge types, distillation strategies and the teacher-student architectures play the crucial role in the student learning.

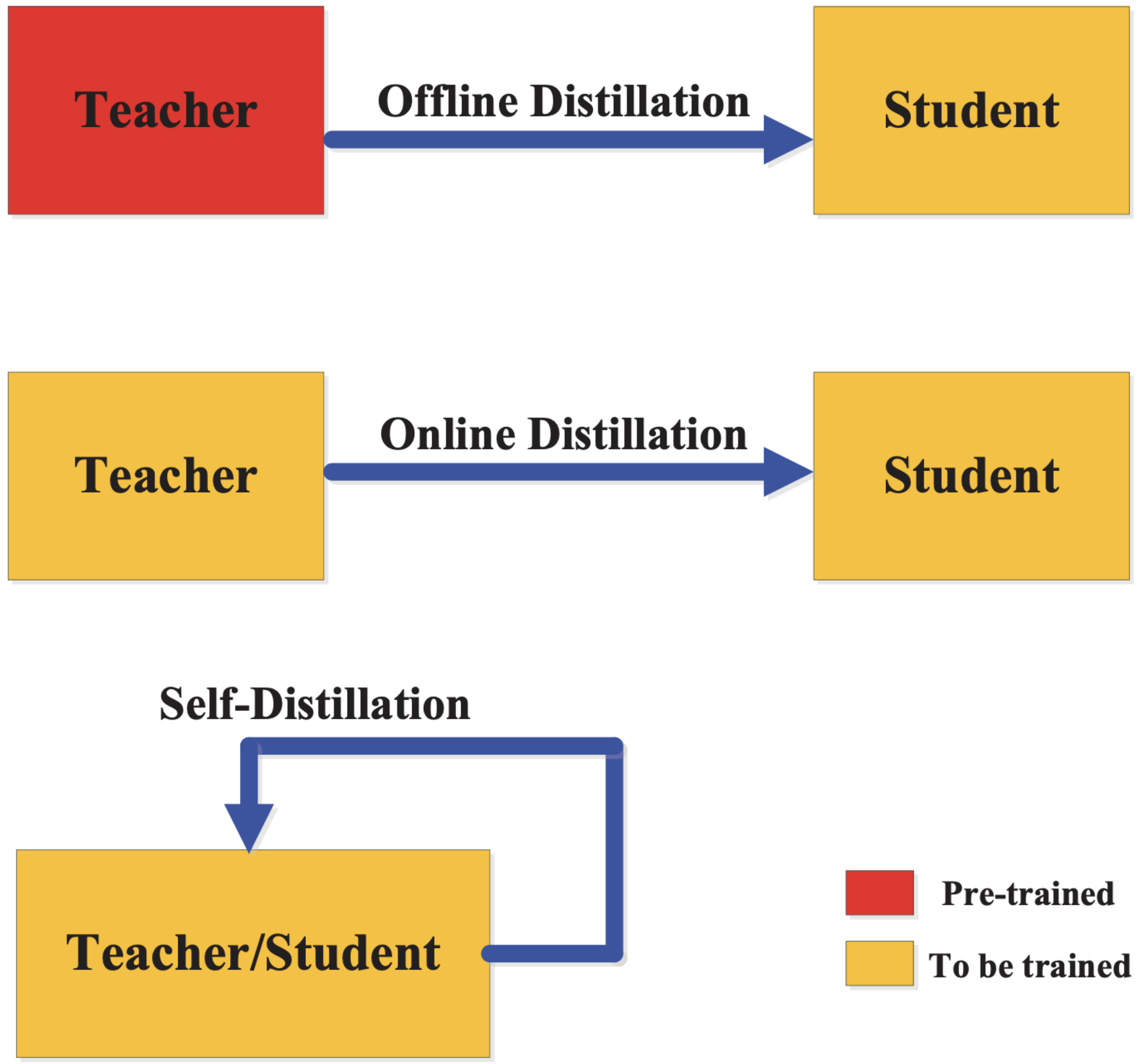
Knowledge Distillation - Knowledge

- The activations, neurons or features of intermediate layers also can be used as the knowledge to guide the learning of the student model.
- Different forms of knowledge: response-based knowledge, feature-based knowledge, and relation-based knowledge.

Knowledge Distillation - Strategies

Different knowledge distillation strategies are:

- Offline distillation
- Online distillation
- Self-distillation



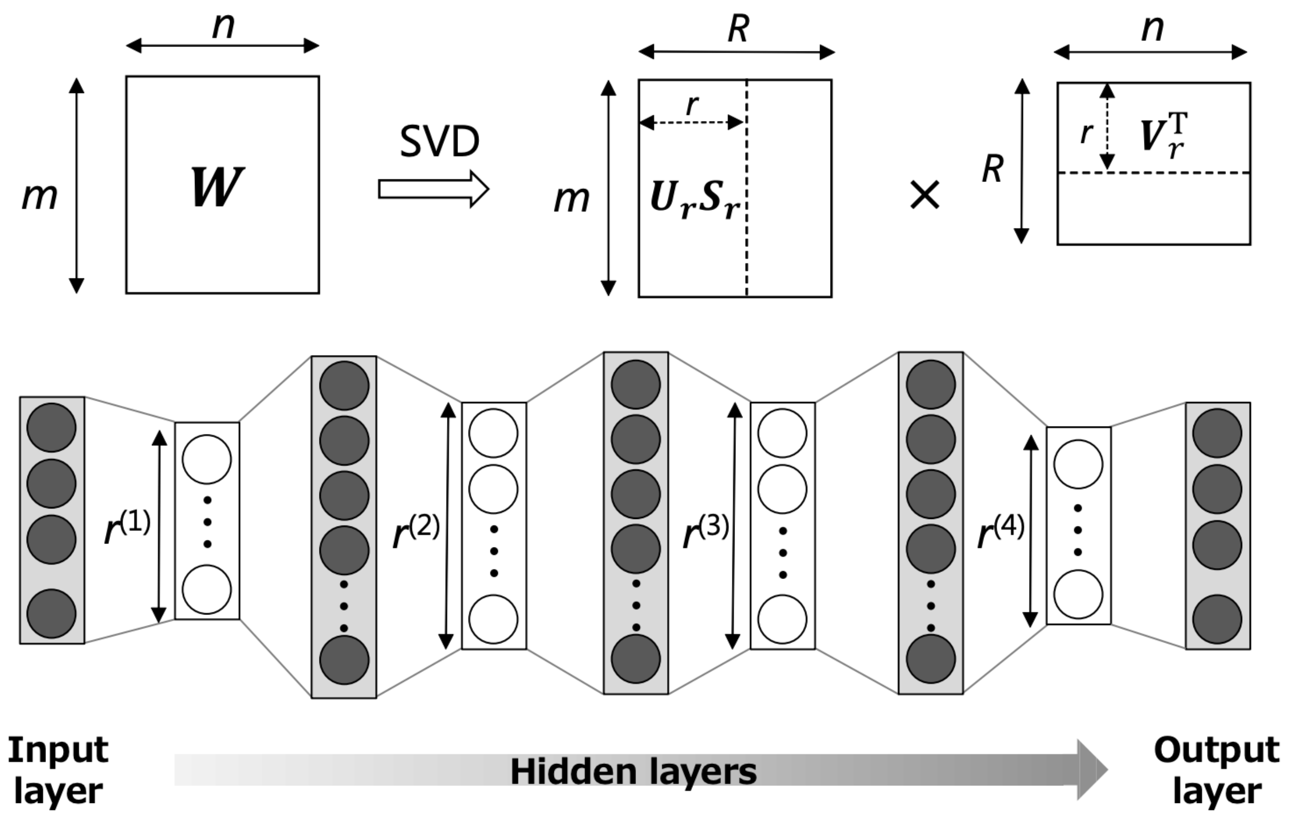
ailient
See what matters

Low-rank Factorization

Low-rank factorization identifies redundant parameters of deep neural networks by employing the matrix and tensor decomposition

A weight matrix A with m x n dimension and having a rank r is replaced by smaller dimension matrices.

This technique helps by factorizing a large matrix into smaller matrices.



Low-rank Factorization - Summary

- Can be applied during or after training.
- Can be applied to both convolutional and fully connected layers.
- When applied during training, can reduce training time.
- The factorization of the dense layer matrices reduces model size.
- Improves the speed up-to 30-50% compared to the full-rank matrix representation.

See what matters

Model Compression – Summary

- Model compression techniques are complementary to each other.
- Can be applied to pre-trained models, as a post-processing step to reduce your model size and increase inference speed.
- Can be applied during training time as well.
- Pruning and quantization have now been baked into machine learning frameworks such as Tensorflow and PyTorch.
- There are many more compression approaches beyond the four common ones covered in this talk, such as weight sharing-based model compression, structural matrix, transferred filters, and compact filters.

See what matters

Additional Compression Techniques - Resources

Transferred filters

- S. Zhai, Y. Cheng, and Z. M. Zhang, "Doubly convolutional neural networks," in Proc. Advances Neural Information Processing Systems, 2016, pp. 1082–1090.
- W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and improving convolutional neural networks via concatenated rectified linear units," arXiv Preprint, arXiv:1603.05201, 2016.

Compact filters

- S. Dieleman, J. D Fauw, and K. Kavukcuoglu, "Exploiting cyclic symmetry in convolutional neural networks," in Proc. 33rd Int. Conf. Machine Learning, 2016, vol. 48, pp. 1889–1898

See what matters

Additional Compression Techniques - Resources

Structural matrix

- Y. Cheng, F. X. Yu, R. Feris, S. Kumar, A. Choudhary, and S.-F. Chang, "An exploration of parameter redundancy in deep networks with circulant projections," in Proc. Int. Conf. Computer Vision, 2015, pp. 2857–2865.
- Z. Yang, M. Moczulski, M. Denil, N. de Freitas, A. Smola, L. Song, and Z. Wang, "Deep fried convnets," in Proc. Int. Conf. Computer Vision, 2015, pp. 1476– 1483. [32] V. Sindhwani, T. Sainath, and S. Kumar. (2015). Structured transforms for small-footprint deep learning. Advances in Neural Information Processing Systems, 28, pp. 3088–3096. [Online]. Available: http://papers.nips.cc/paper/5869-structured-transforms-for-small-footprint-deep-learning.pdf

See what matters

Additional Compression Techniques - Resources

Weight-sharing based compression

- K. Ullrich, E. Meeds, and M. Welling, “Soft weight-sharing for neural network compression,” Computing Res. Repository, vol. abs/1702.04008, 2017. [Online]. Available: https://arxiv.org/abs/1702.04008

Resource Slide - 1

- Cheng, Yu, et al. "A survey of model compression and acceleration for deep neural networks." *arXiv preprint arXiv:1710.09282* (2017).
- Choudhary, Tejalal, et al. "A comprehensive survey on model compression and acceleration." *Artificial Intelligence Review* (2020): 1-43.
- He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- Zhu, Michael, and Suyog Gupta. "To prune, or not to prune: exploring the efficacy of pruning for model compression." *arXiv preprint arXiv:1710.01878* (2017).

See what matters

Resource Slide - 2

- Blalock, D., Ortiz, J. J. G., Frankle, J., & Guttag, J. (2020). What is the state of neural network pruning?. *arXiv preprint arXiv:2003.03033*.
- Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network." *arXiv preprint arXiv:1503.02531* (2015).
- Gou, Jianping, et al. "Knowledge distillation: A survey." *International Journal of Computer Vision* (2021): 1-31.
- Yaguchi, Atsushi, et al. "Scalable deep neural networks via low-rank matrix factorization." (2019).

Resource Slide – 3

- https://towardsdatascience.com/model-compression-via-pruning-ac9b730a7c7b
- https://www.tensorflow.org/model_optimization/guide/quantization/training
- https://towardsdatascience.com/distilling-knowledge-in-neural-network-d8991faa2cdc
- https://intellabs.github.io/distiller/knowledge_distillation.html
- https://medium.com/@RaghavPrabhu/cnn-architectures-lenet-alexnet-vgg-googlenet-and-resnet-7c81c017b848

See what matters