Comparing the Effectiveness of Knowledge Distillation and Weight-Based Pruning on Neural Networks

Amber Li, Cindy Zhang, and Michelle Li

Goals

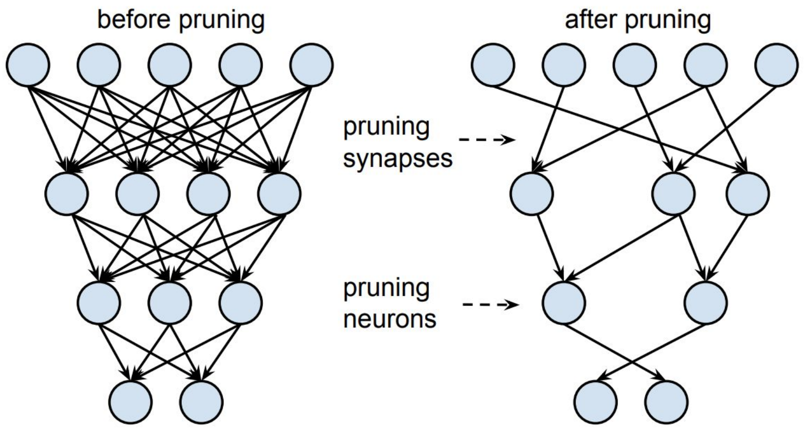
- What is the best way to compress a deep neural network?
- Popular methods:
  - Weight-based pruning
  - Knowledge distillation
- Is using a combination of these methods more effective?
  - Meaningful trend in doing so?

Knowledge Distillation

- Introduced by Hinton et al. [1] in 2015
- Train a distilled model to emulate a deep neural network
- Train on logits of larger model
- Intuition: easier for small model to generalize the same way as large model than to directly learn the true parameterization

Weight-Based Pruning

- General algorithm from Han et al. [2]:
    1. Randomly initialize the deep neural network
    2. Train to convergence
    3. Prune connections with weights below threshold
    4. Retrain the sparse network



Lottery Ticket Hypothesis

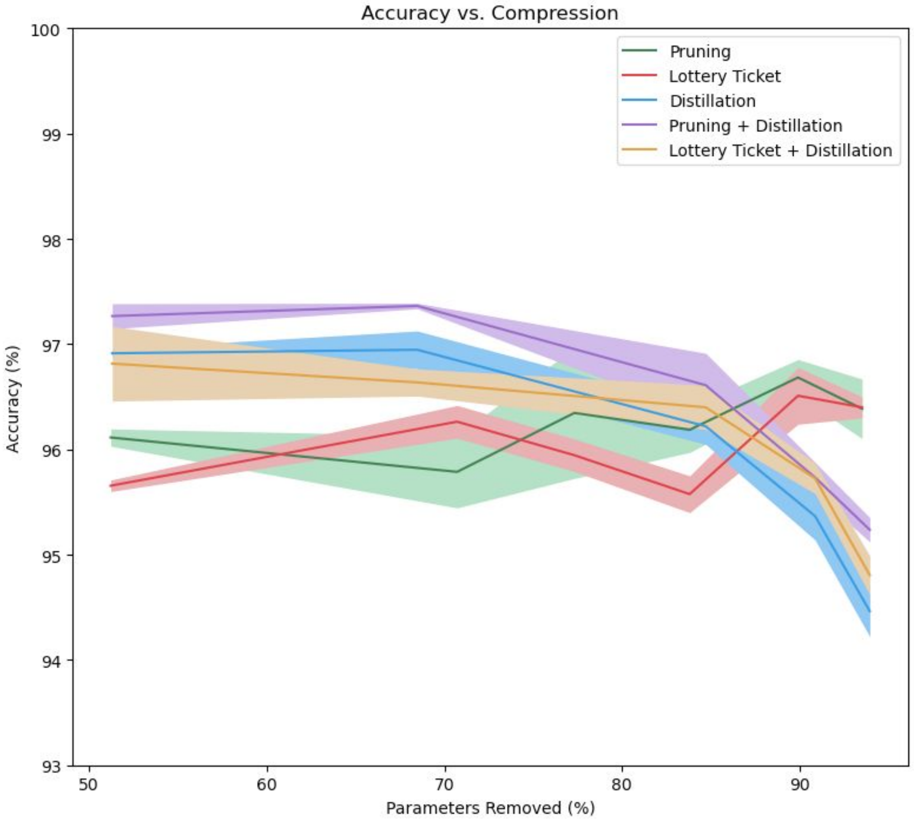
- Algorithm from Frankle and Carbin [3]:
  - Randomly initialize a deep neural network with weights W
  - Train to convergence
  - Prune connections with the lowest weights
  - Reset remaining parameters to original values in W before retraining, creating the winning ticket
- Iterative pruning rather than one-shot

Previous Work

- Oguntola et al. [4] explores effectiveness of different deep model compression methods
  - Evaluated on the VGG19 model for CIFAR-10
  - Compressed 85x and retained 96% of accuracy
  - Stacking compression methods is generally very effective

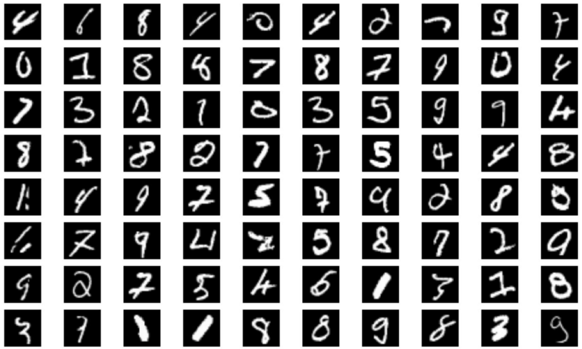
LeNet-300-100 for MNIST

- 3 fully connected layers
- Original model has 266,610 parameters, 95.84% accuracy
- Pruning + distillation works better than each method individually until ~85% compression

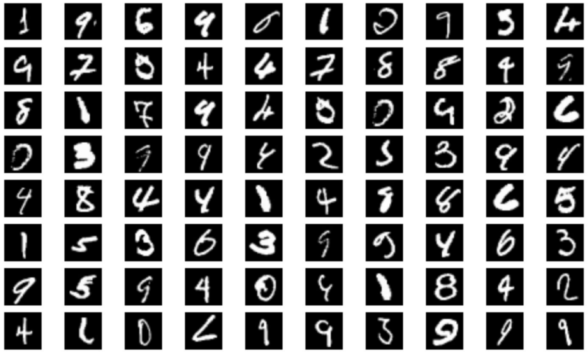


LeNet-300-100 for MNIST

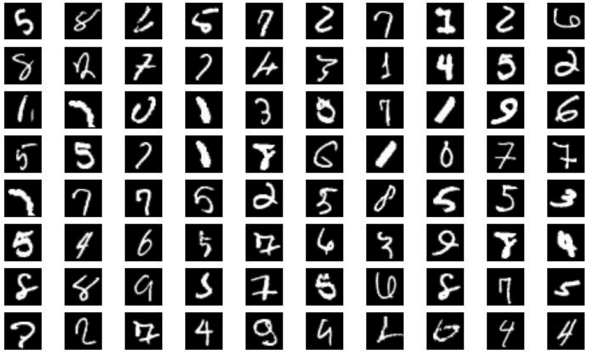
- No obvious patterns in test examples that are incorrectly classified



Original

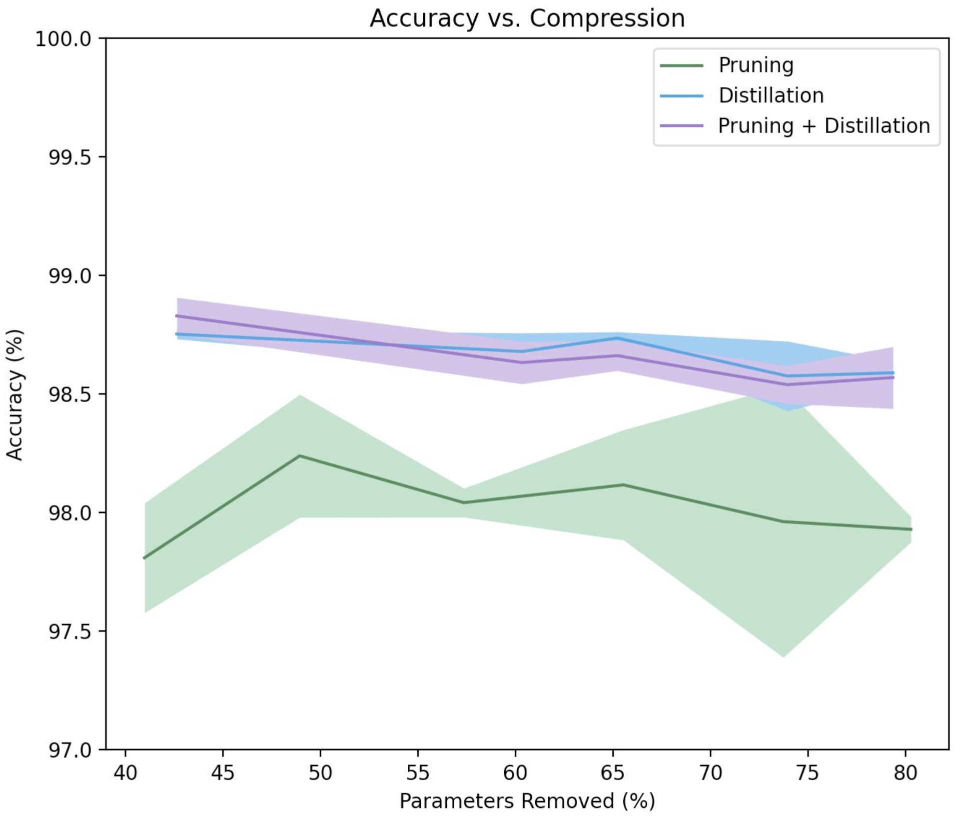


Pruned (80% params removed)



Distilled (80% params removed)

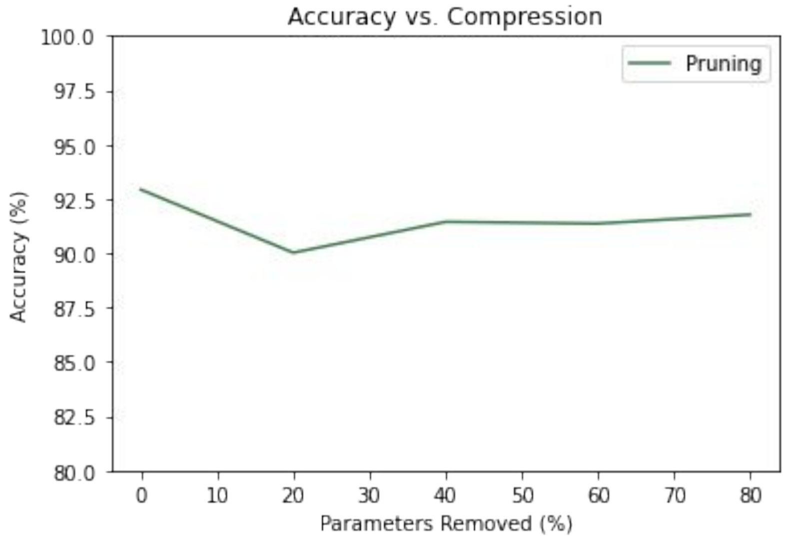
LeNet-5 for MNIST



- 3 convolutional layers followed by 2 fully connected layers
- Original model has 61,706 parameters, 98.16% accuracy
- Using only distillation produces similar results to using pruning and distillation

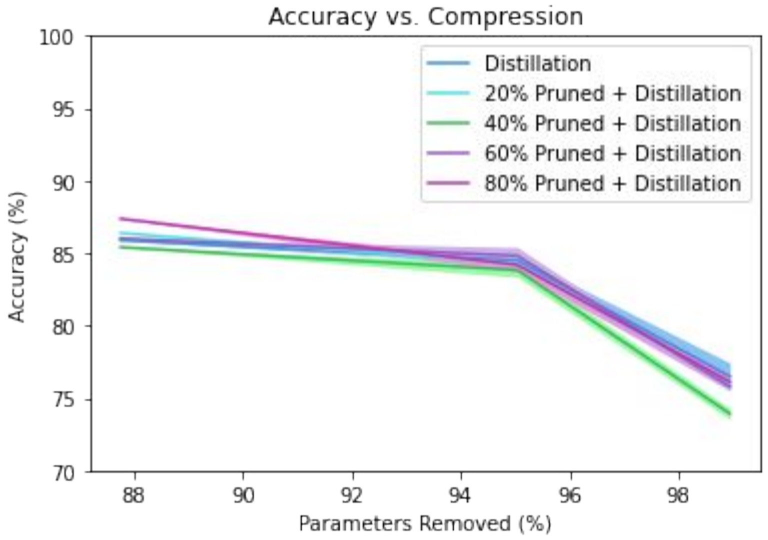
ResNet-34 for CIFAR-10

- 34 convolutional layers with residual blocks
- Original model has ~21M parameters, 92.9% accuracy (pretrained)
- Pruned 20%, 40%, 60%, 80% of parameters



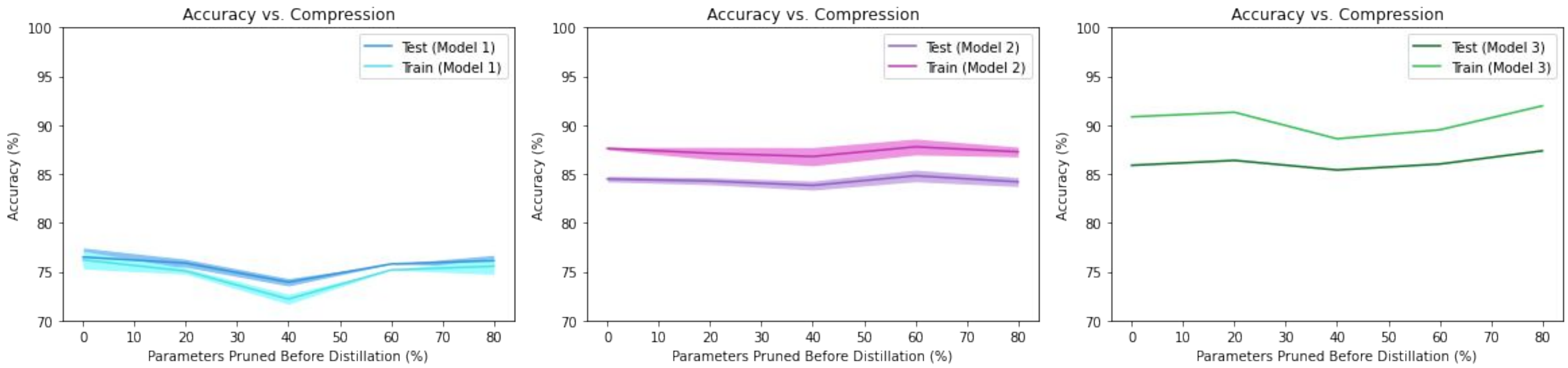
ResNet-34 for CIFAR-10

- Distillation using 3 different student models
  - 4 conv layers, 2 fc layers, increasing #s of channels
  - 1) 0.2M params (99% sparse)
  - 2) 1M params (95% sparse)
  - 3) 2.6M params (88% sparse)



ResNet-34 for CIFAR-10

- Accuracy increasing in # of parameters in student model, regardless of pruning
- Increased overfitting as # parameters increase



Comparing Neural Networks

How similar are the compressed models we produce using only distillation vs. using pruning and distillation? For LeNet-5:

Comparing Distillation and Pruning + Distillation		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
40%	127	12.46887
	147	19.80565
50%	145	19.23841
	151	19.16463
65%	155	12.17823
	158	15.24531
75%	167	11.72104
	172	17.42870

Comparing Two Distillation Models		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
40%	119	20.06279
	117	19.62773
50%	124	19.51043
	138	18.67256
65%	120	17.75762
	115	17.86453
75%	151	16.95058
	163	17.70182

Comparing Neural Networks

LeNet-300-100 for MNIST:

Comparing Distillation and Pruning + Distillation		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
50%	302	11.82822
	297	20.84023
70%	292	19.27285
	317	19.14673
90%	397	17.34818
	419	17.17063

Comparing Two Distillation Models		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
50%	201	22.32834
	190	22.56884
70%	222	20.88432
	249	20.65374
90%	360	18.54195
	372	19.82944

Comparing Neural Networks

ResNet-34 for CIFAR:

Comparing Distillation and Pruning + Distillation		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
95%	1581	42.77
	1666	44.47
	1556	45.23
	1523	43.17
99%	2226	35.75
	2299	37.29
	2348	37.94
	2300	36.03

Comparing Two Distillation Models		
Parameters Removed	# of Test Examples Classified Differently	L2 Distance
95%	1702	49.73
99%	2230	38.09

Conclusion

- Using both pruning and distillation does not perform significantly better than using only one of the methods
- Distillation vs. combination of pruning and distillation result in similar models
- Future work:
    - Experiment on other architectures/datasets
    - Try these methods on tasks beyond vision-centric classification
    - What happens when not all training data is correctly labeled?

References

[1] Hinton et al., “Distilling the Knowledge in a Neural Network.” https://arxiv.org/pdf/1503.02531.pdf.

[2] Han et al., “Learning both Weights and Connections for Efficient Neural Networks.” https://arxiv.org/abs/1506.02626.

[3] J. Frankle and M. Carbin, “The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks.” https://arxiv.org/pdf/1803.03635.pdf.

[4] Oguntola et al., “SlimNets: An Exploration of Deep Model Compression and Acceleration.” https://arxiv.org/pdf/1808.00496.pdf.