Google PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Dynamic Sparsity Neural Networks for Automatic Speech Recognition

Zhaofeng Wu[1], Ding Zhao[2], Qiao Liang[2], Jiahui Yu[2], Anmol Gulati[2], Ruoming Pang[2]

[1] Paul G. Allen School of Computer Science & Engineering, University of Washington

[2]Google

Sparse Neural Networks

Sparse Neural Networks

- Goal: Models with weight matrices that have many 0 entries so that the inference-time forward pass is fast without significant quality impact
  - Sparsity of a weight matrix: (# zero entries) / (# elements)
  - May require specialized library/hardware to realize a speedup proportional to the sparsity level
- Approach: Model pruning

Model Pruning

Weight matrix W from pretrained network

-1	5	3	4
1	2	6	7
-3	2	-1	-2
7	8	3	4

Zero out entries using pruning criterion

Sparse W

0	5	3	4
0	2	6	7
-3	2	0	-2
7	8	3	4

Fine-tune remaining entries

Fine-tuned W

0	1	2	-4
0	6	9	-2
8	6	0	4
-7	-6	-2	5

Repeat; zero-out new and/or more entries

Our Model Pruning Settings

- Block Pruning: Prune 16×1 blocks instead of individual elements
- Pruning Criterion: Blocks with the smallest $||W \times \text{gradient}||_1$
- Allow pruned blocks to recover if there are blocks with smaller norms
- Sparsity Warm-Up: Cubic schedule from 0 to the target sparsity level

Issues

- In production, we need models with different sparsity levels for
    a. Different hardware types

Model	SoC	RAM	Android	Test 1, ms	Test 2, ms	Test 3, ms
Huawei P20 Pro	HiSilicon Kirin 970	6GB	8.1	144	$\boxed{130}$	2634
OnePlus 6	Snapdragon 845/DSP	8GB	9.0	$\boxed{24}$	892	$\boxed{1365}$
HTC U12+	Snapdragon 845	6GB	8.0	60	620	1433
Samsung Galaxy S9+	Exynos 9810 Octa	6GB	8.0	148	1208	1572
Samsung Galaxy S8	Exynos 8895 Octa	4GB	8.0	134	731	1512
Motorola Z2 Force	Snapdragon 835	6GB	8.0	85	823	1894
OnePlus 3T	Snapdragon 821	6GB	8.0	106	776	1937
Lenovo ZUK Z2 Pro	Snapdragon 820	6GB	8.0	115	909	2099
Google Pixel 2	Snapdragon 835	4GB	9.0	143	1264	1953
Google Pixel	Snapdragon 821	4GB	9.0	116	867	1838
Nokia 7 plus	Snapdragon 660	4GB	9.0	136	944	2132
Asus Zenfone 5	Snapdragon 636	4GB	8.0	110	1055	2405
Google Pixel C	Nvidia Tegra X1	3GB	8.0	105	1064	2585
Huawei Honor 8 Pro	HiSilicon Kirin 960	6GB	8.0	121	1720	3163
Sony XA2 Ultra	Snapdragon 630	4GB	8.0	170	1653	3424
Meizu Pro 7 Plus	Mediatek Helio X30	6GB	7.0	327	3357	4550
BlackBerry Keyone	Snapdragon 625	4GB	7.1	160	1695	3525
Sony X Compact	Snapdragon 650	3GB	8.0	111	1804	3566
Xiaomi Redmi 5	Snapdragon 450	3GB	7.1	188	1753	3707
Huawei Nexus 6P	Snapdragon 810	3GB	8.0	106	1962	4113
Meizu MX6	Mediatek Helio X20	4GB	7.1	183	2217	4981
HTC U Play	Mediatek Helio P10	3GB	6.0	239	2061	4303
Xiaomi Redmi 4X	Snapdragon 435	3GB	7.1	246	2640	5428
Samsung Galaxy J7	Exynos 7870 Octa	3GB	7.0	278	2092	4648
LG Nexus 5	Snapdragon 800	2GB	4.4	332	2182	5080
Asus Zenfone 2	Intel Atom Z3580	2GB	5.0	$\boxed{1507}$	2433	6188
Motorola Moto C	Mediatek MT6737	1GB	7.0	414	3394	7761
Samsung Galaxy S3	Exynos 4412 Quad	1GB	4.3	553	4640	10321
Fly Nimbus 15	Spreadtrum SC9832	1GB	7.0	538	5103	12618
Huawei Ascend P1	TI OMAP 4460	1GB	4.1	482	$\boxed{7613}$	$\boxed{25105}$

A. Ignatov, R. Timofte, W. Chou, K. Wang, M. Wu, T. Hartley, and L. Van Gool, “AI benchmark: Running deep neural networks on android smartphones,” in Proc. of ECCV, 2018.

Issues

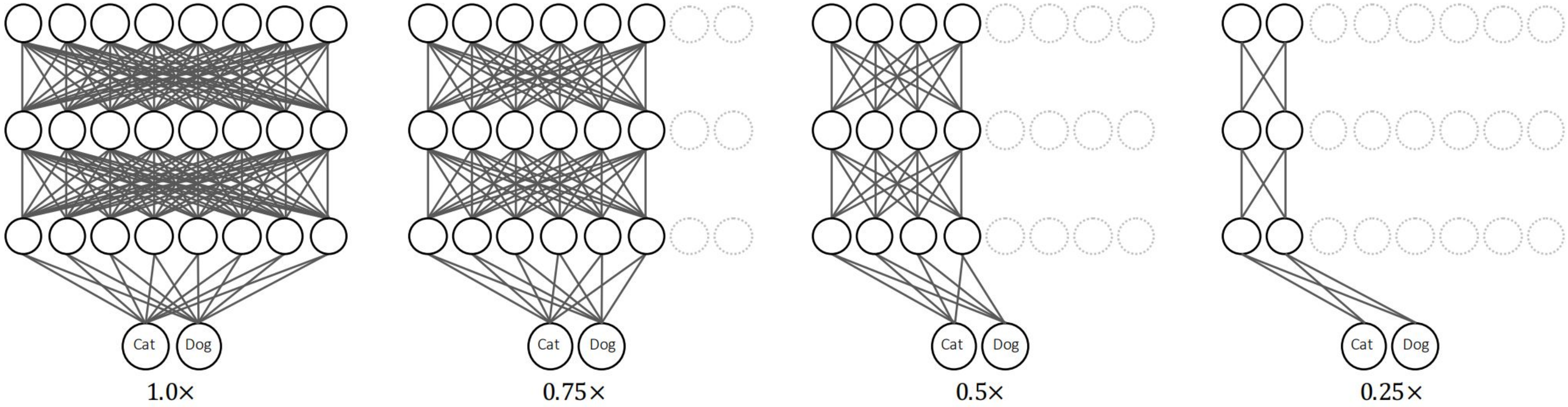
- In production, we need models with different sparsity levels for
  a. Different hardware types
     - Mobile devices, home speakers, in-car systems, etc.
  b. Different applications
     - E.g., real-time conference captioning vs. YouTube subtitle generation
  c. Different runtime resource availability

Options

- Single model with static sparsity level
  - Suboptimal resource usage
- Multiple models with static sparsity level
  - Doesn’t solve the problem entirely
  - Maintenance overhead
- Single dynamic model with adjustable sparsity levels specified at inference time?

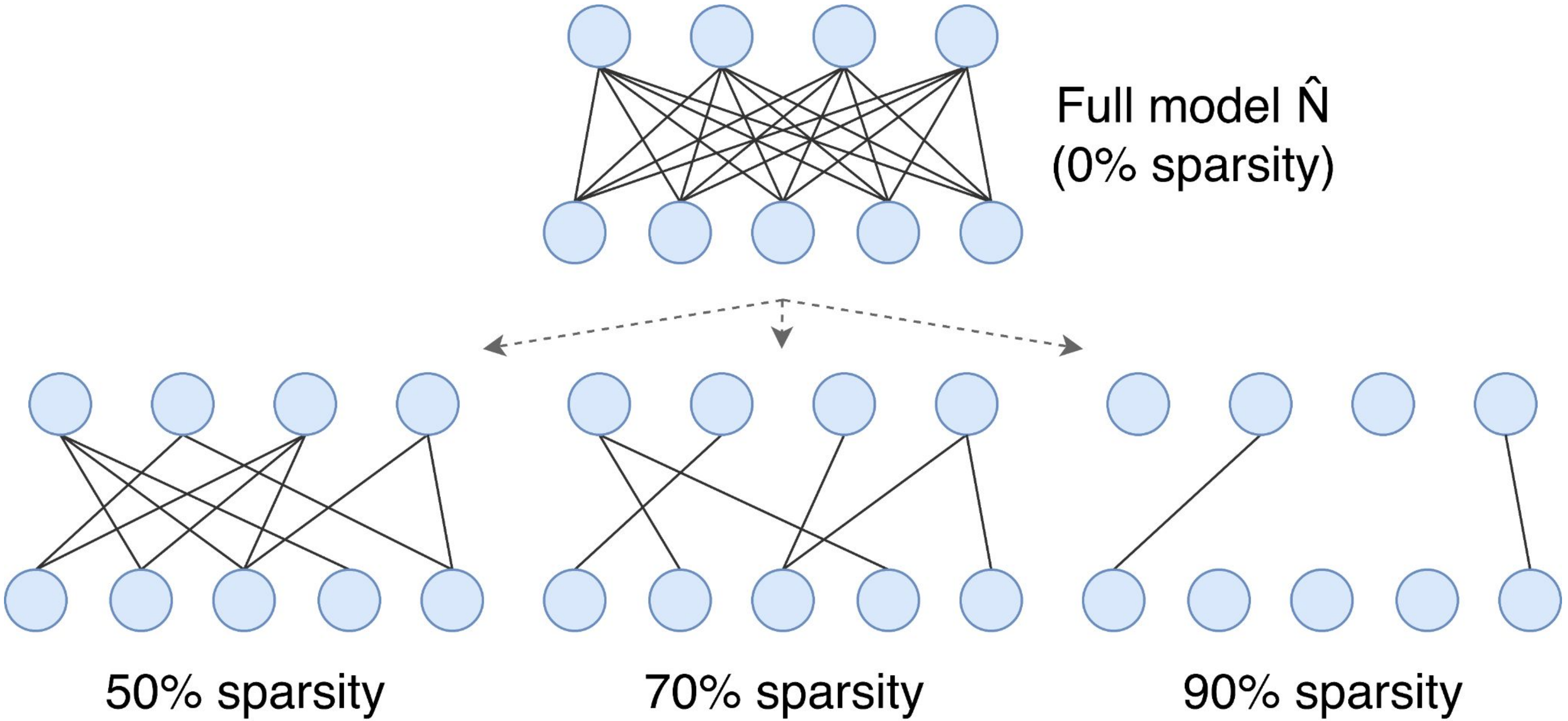
Dynamic Sparsity Neural Networks

Precursor: Slimmable Neural Networks

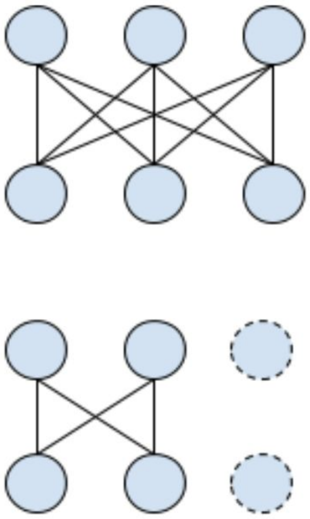


J. Yu, L. Yang, N. Xu, J. Yang, and T. Huang, "Slimmable neural networks," in Proc. of ICLR, 2019.

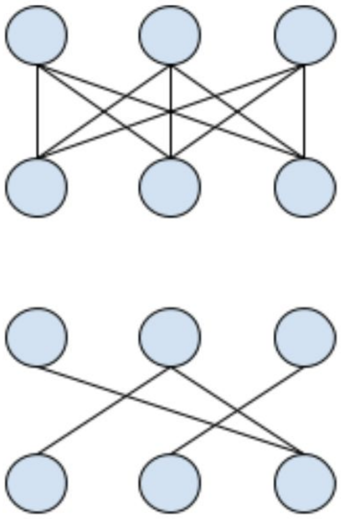
Dynamic Sparsity Neural Networks



Comparison



Slimmable Neural Networks



Dynamic Sparsity Neural Networks

Approach

- Given a predefined list of C sparsity levels
- In each epoch, perform (C+1) forward/backward passes with each sparsity level and the full model
- Lazy update
  - Gradient accumulation within each epoch
  - Lazy mask update
- Pre-training
- In-place distillation
- Progressive freezing

Experimental Setup

Task & Dataset

- Automatic speech recognition (ASR)
- In-house anonymized production dataset
  - Training: 35M English utterances/27,500 hours
  - Testing:
    - Voice Search traffic (**VS**; 15,000 utterances)
    - Noisy farfield utterances where the sound source is far from the microphone (**Farfield**; 9,000 utterances)

Model Settings

- RNN-T backbone
  - 8 LSTM encoder layers + 2 LSTM decoder layers
- Learning rate: Constant at 1e-3 after warm-up
- Prune all 2D matrices in LSTM and fully-connected layers
  - 98.7% of all model parameters
- Sparsity configurations:

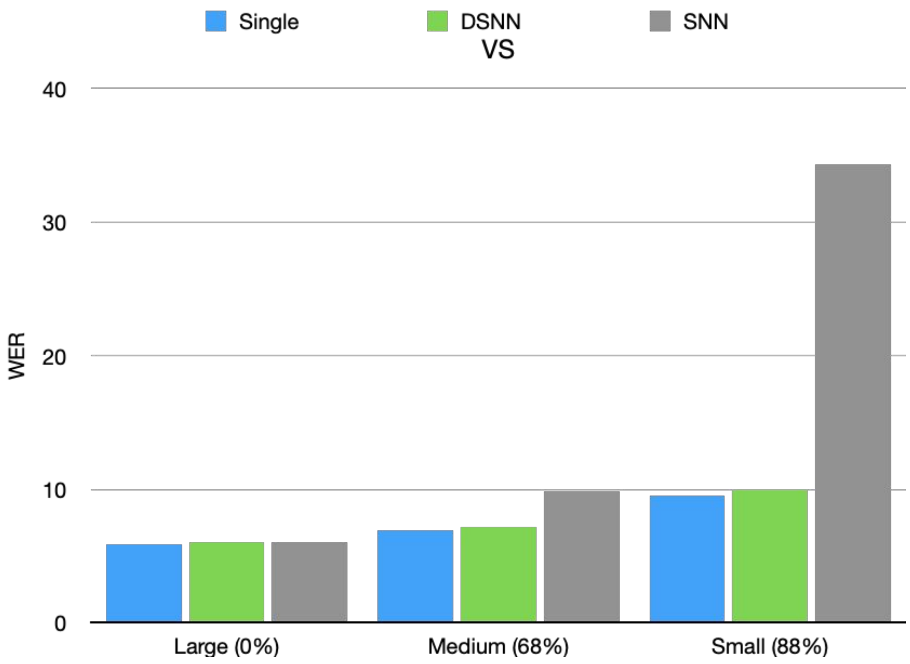
Table 1. Target sparsity configurations. The "Average" columns represents an average global sparsity level across all weights.

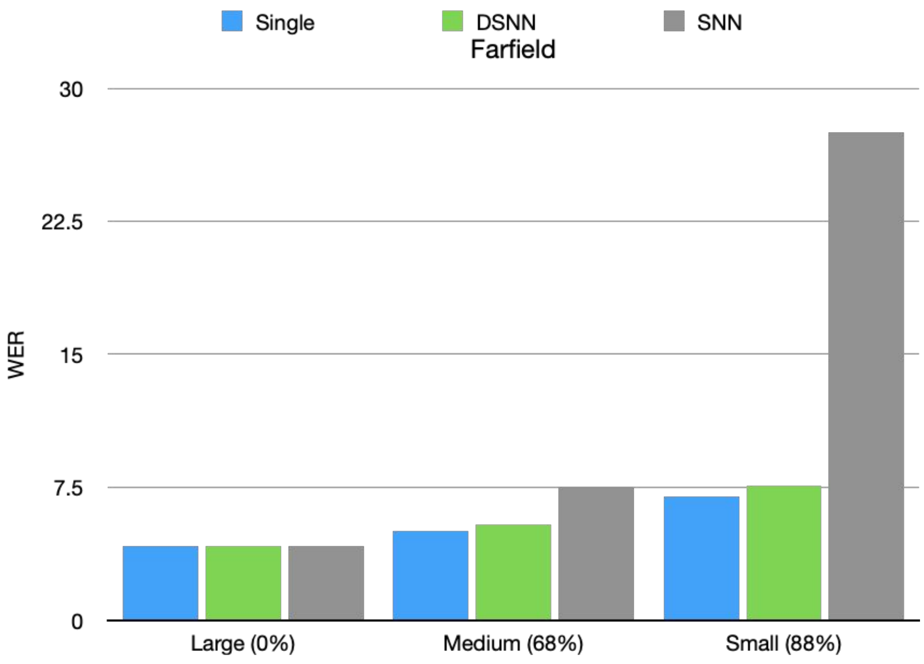
Type	Sparsity: LSTM	Sparsity: FC	Sparsity: Average	# Parameters
Large	0%	0%	0%	122.2M
Medium	70%	0%	68%	39.6M
Small	90%	50%	88%	14.6M

Google

Results

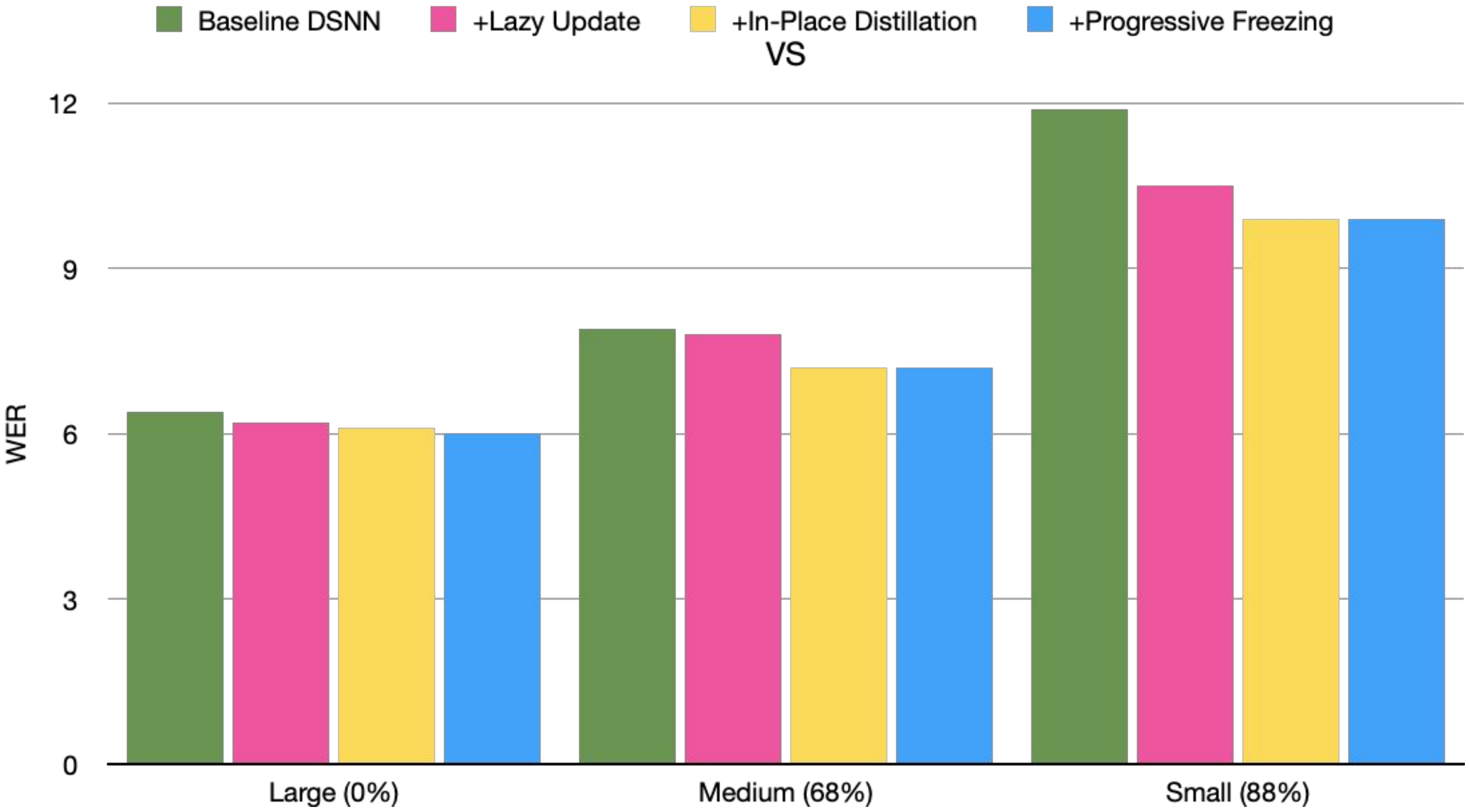
Results





Google

Ablations



Baseline DSNN
+Lazy Update
+In-Place Distillation
+Progressive Freezing
VS
12
9
6
3
0
WER
Large (0%)
Medium (68%)
Small (88%)

Summary

- Dynamic Sparsity Neural Networks (DSNN) can instantly switch to any predefined sparsity configuration at run-time
- On ASR, the performance of a DSNN model is on par with that of individually trained single sparsity network

Thank You