Robust Binary Models by Pruning Randomly-initialized Networks

Chen Liu*, Ziqi Zhao*, Sabine Süsstrunk, Mathieu Salzmann

École Polytechnique Fédérale de Lausanne

Lausanne, Switzerland

EPFL

NeurIPS 2022

* Equal Contribution

Motivation

To improve the efficiency of robust deep neural networks.

Motivation

To improve the efficiency of robust deep neural networks.

- Pruning
- Quantization

Motivation

To improve the efficiency of robust deep neural networks.

- Pruning $\rightarrow$ Pruning without training parameters
- Quantization $\rightarrow$ Binarization

Motivation

To improve the efficiency of robust deep neural networks.

- Pruning $\rightarrow$ Pruning without training parameters
- Quantization $\rightarrow$ Binarization

Pruning as a way of training binary neural networks.

Extending *Strong Lottery Ticket Hypothesis* to the case of robust binary networks.

Methodology

Network f parameterized by $\boldsymbol{w} \in \mathbb{R}^n$. Training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.

Methodology

Network f parameterized by $\boldsymbol{w} \in \mathbb{R}^n$. Training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.

- Non-adversarial training

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(\boldsymbol{w}, \boldsymbol{x}_i), y_i)$$

Methodology

Network f parameterized by $\boldsymbol{w} \in \mathbb{R}^n$. Training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.

- Non-adversarial training

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(\boldsymbol{w}, \boldsymbol{x}_i), y_i)$$

- Adversarial Training

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \max_{\Delta_i \in \mathcal{S}_\epsilon} \mathcal{L}(f(\boldsymbol{w}, \boldsymbol{x}_i + \Delta_i), y_i)$$

Methodology

Network f parameterized by $\boldsymbol{w} \in \mathbb{R}^n$. Training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.

- Non-adversarial training

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(\boldsymbol{w}, \boldsymbol{x}_i), y_i)$$

- Adversarial Training

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \max_{\Delta_i \in \mathcal{S}_\epsilon} \mathcal{L}(f(\boldsymbol{w}, \boldsymbol{x}_i + \Delta_i), y_i)$$

- Pruning as adversarial training (sparsity ratio: r)

$$\min_{\boldsymbol{m}} \frac{1}{N} \sum_{i=1}^{N} \max_{\Delta_i \in \mathcal{S}_\epsilon} \mathcal{L}(f(\boldsymbol{w} \odot \boldsymbol{m}, \boldsymbol{x}_i + \Delta_i), y_i), \;\; s.t. \; \boldsymbol{m} \in \{0,1\}^n, sum(\boldsymbol{m}) = (1-r)n$$

Methodology

Adaptive pruning.

- Adaptively adjust the layerwise pruning rate.
- Prune fewer parameters for smaller layers.

Methodology

Adaptive pruning.

- Adaptively adjust the layerwise pruning rate.
- Prune fewer parameters for smaller layers.

Last batch normalization layer (LBN).

- Avoid gradient explosion / vanishing under binary initialization.
- Make the performance less sensitive to hyper-parameter selection.

Experimental Results

Method	Architecture	Pruning Strategy	CIFAR10		CIFAR100		ImageNet100	
			FP	Binary	FP	Binary	FP	Binary
AT	RN34	Not Pruned	43.26	40.34	36.63	26.49	53.92	34.20
AT	RN34-LBN	Not Pruned	42.39	39.58	35.15	32.98	55.14	35.36
AT	Small RN34	Not Pruned	38.81	26.03	27.68	15.85	25.40	10.44
FlyingBird	RN34	Dynamic	45.86	34.37	35.91	23.32	37.70	9.54
FlyingBird+	RN34	Dynamic	44.57	33.33	34.30	22.64	37.70	9.52
BCS	RN34	Dynamic	43.51	-	31.85	-	-	-
RST	RN34	$p = 1.0$	34.95	-	21.96	-	17.54	-
RST	RN34-LBN	$p = 1.0$	37.23	-	23.14	-	15.36	-
HYDRA	RN34	$p = 0.1$	42.73	29.28	33.00	23.60	43.18	18.22
ATMC	RN34	Global	34.14	25.62	25.10	11.09	22.18	5.78
ATMC	RN34	$p = 0.1$	34.58	24.62	25.37	11.04	23.52	4.58
Ours	RN34-LBN	$p = 0.1$	-	**45.06**	-	**34.83**	-	**33.04**
Ours(fast)	RN34-LBN	$p = 0.1$	-	40.77	-	34.45		

Table: Robust accuracy (in %) on the CIFAR10, CIFAR100 and ImageNet100 test sets for the baselines and our proposed method. "RN34-LBN" represents ResNet34 with the last batch normalization layer. "Small RN34" refers to Smaller RN34. The pruning rate is set to 0.99 except for the not-pruned methods. Among the pruned models, the best results for the full-precision (FP) models are underlined; the best results for the binary models are marked in bold. The values of ϵ for CIFAR10, CIFAR100 and ImageNet100 are 8/255, 4/255 and 2/255, respectively. "-" means not applicable or trivial performance.

Experimental Results

The pruning masks obtained by our method are *structured*.

- Many channels / kernels of the convolutional layers are totally pruned.
- Retained parameters are concentrated on a few channels / kernels.
- Pruned channels / kernels of the two consecutive layers are aligned.

Experimental Results

The pruning masks obtained by our method are *structured*.

- Many channels / kernels of the convolutional layers are totally pruned.
- Retained parameters are concentrated on a few channels / kernels.
- Pruned channels / kernels of the two consecutive layers are aligned.

Regular pruning is possible!

Thank You!

Full Paper

Code