Learning Accurate Low-bit Deep Neural Networks with Stochastic Quantization

Yinpeng Dong[1], Renkun Ni[2], Jianguo Li[3], Yurong Chen[3], Jun Zhu[1], and Hang Su[1]

[1] Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2] University of Virginia
[3] Intel Labs China, Beijing, China

Introduction

BinaryConnect: Quantize 32-bits weights $\mathbf{W}_i$ to binary values $\mathbf{B}_i$.

$$\mathbf{B}_i^j = \begin{cases} +1 & \text{with probability } p = \sigma(\mathbf{W}_i^j), \\ -1 & \text{with probability } 1-p. \end{cases} \quad (1)$$

BWN: Introduce a scaling factor $\alpha \in \mathbb{R}^+$ along with $\mathbf{B}_i$ to approximate $\mathbf{W}_i$.

$$\mathbf{B}_i = \mathrm{sign}(\mathbf{W}_i) \quad \text{and} \quad \alpha = \frac{1}{d}\sum\nolimits_{j=1}^{d} |\mathbf{W}_i^j|. \quad (2)$$

TWN: Approximates $\mathbf{W}_i$ with a ternary value vector $\mathbf{T}_i \in \{1, 0, -1\}^d$ along with a scaling factor α.

$$\mathbf{T}_i^j = \begin{cases} +1 & \text{if } \mathbf{W}_i^j > \Delta \\ 0 & \text{if } |\mathbf{W}_i^j| \le \Delta \\ -1 & \text{if } \mathbf{W}_i^j < -\Delta \end{cases} \quad \text{and} \quad \alpha = \frac{1}{|\mathbf{I}_\Delta|}\sum_{i \in \mathbf{I}_\Delta} |\mathbf{W}_i^j|, \quad (3)$$

where Δ is a positive threshold with following values

$$\Delta = \frac{0.7}{d}\sum\nolimits_{j=1}^{d} |\mathbf{W}_i^j|, \quad (4)$$

$\mathbf{I}_\Delta = \{j \mid |\mathbf{W}_i^j| > \Delta\}$ and $|\mathbf{I}_\Delta|$ denotes the cardinality of set $\mathbf{I}_\Delta$.

Motivations: Previous methods quantize the weights **to low-bits all together**. The quantization error **is not consistently small** for all elements/filters. The large quantization error for some elements/filters lead to **inappropriate gradient direction** during training, thus makes the model converge to **worse local minimum**.

Stochastic Quantization

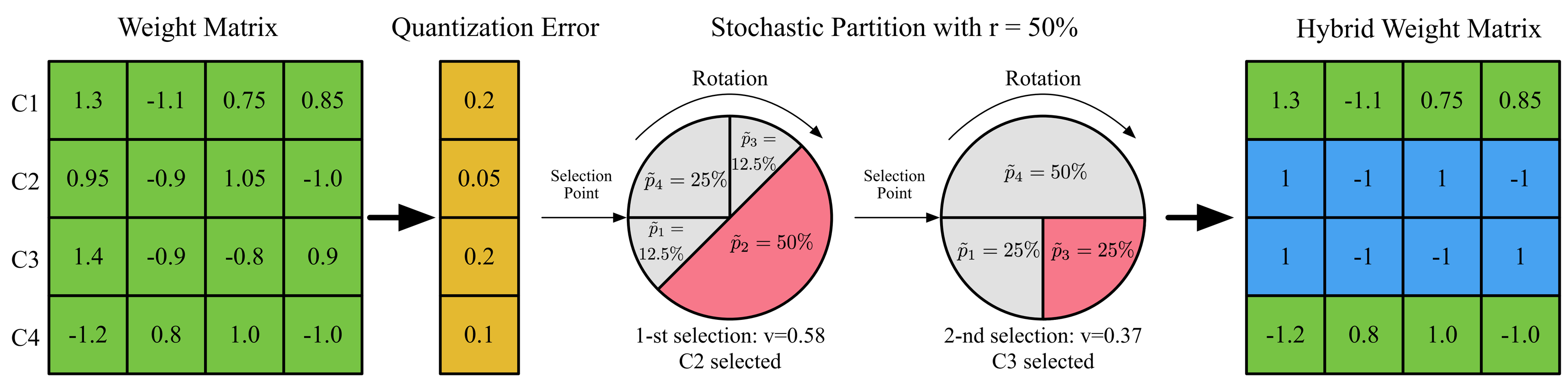

Figure 1: Illustration of the stochasitc quantization procedure.

Quantization Error: The normalized L_1 distance between $\mathbf{W}_i$ and $\mathbf{Q}_i$ (*e.g.*, $\mathbf{B}_i$, $\mathbf{T}_i$).

$$e_i = \frac{\|\mathbf{W}_i - \mathbf{Q}_i\|_1}{\|\mathbf{W}_i\|_1}. \quad (5)$$

Quantization Probability p_i: Inversely proportional to e_i (*e.g.*, linear, sigmoid functions of $1/e_i$).

Quantization Ratio r: A portion of weights to quantize. r gradually increases to 100% at the end of training.

Stochastic Partition: Partition the rows of $\mathcal{W}$ into two disjoint groups $G_q = \{\mathbf{W}_{q_1}, \cdots, \mathbf{W}_{q_{N_q}}\}$ and $G_r = \{\mathbf{W}_{r_1}, \cdots, \mathbf{W}_{r_{N_r}}\}$ ($N_q = r \times m$), which should satisfy

$$G_q \cup G_r = \mathcal{W} \quad \text{and} \quad G_q \cap G_r = \varnothing, \quad (6)$$

Training: Form the hybrid weight matrix $\tilde{\mathcal{Q}}^t$, where each row $\tilde{\mathbf{Q}}_i = \mathbf{W}_i$ *if* $\mathbf{W}_i \in G_r$; *else* $\tilde{\mathbf{Q}}_i = \mathbf{Q}_i$. Update $\mathcal{W}$ with the hybrid gradients $\frac{\partial \mathcal{L}}{\partial \tilde{\mathcal{Q}}^t}$ in each iteration as

$$\mathcal{W}^{t+1} = \mathcal{W}^t - \eta^t \frac{\partial \mathcal{L}}{\partial \tilde{\mathcal{Q}}^t}, \quad (7)$$

Inference: Use the low bitwidth weights $\mathcal{Q}$ during inference.

Codes: https://github.com/dongyp13/Stochastic-Quantization.

Experiments

Ablation Study

Several factors in the algorithm will affect the overall performance like:

- **Selection granularity:** element-wise or **channel-wise**.
- **Partition algorithm: stochastic partition—roulette algorithm**; deterministic partition—sorting; fixed partition—select once.
- **Quantization probability function:** constant—$p_i = 1/m$; **linear**—$p_i = f_i / \sum_j f_j$; softmax—$p_i = \exp(f_i)/\sum_j \exp(f_j)$; sigmoid—$p_i = 1/(1+\exp(-f_i))$, where $f_i = 1/(e_i + \epsilon)$.
- **Scheme for updating SQ ratio r:** **exponential**—$r = 50\%, 75\%, 87.5\%$ **and** 100%; average—$r = 20\%, 40\%, 60\%, 80\%$ and 100%; fine-tune—$r = 0\%, 50\%, 75\%, 87.5\%$ and 100%.

Channel-wise vs. Element-wise

	Channel-wise	Element-wise
SQ-BWN	**7.15**	7.67
SQ-TWN	**6.20**	6.53

Stochastic vs. Deterministic vs. Fixed

	Stochastic	Deterministic	Fixed
SQ-BWN	**7.15**	8.21	*
SQ-TWN	**6.20**	6.85	6.50

Quantization Probability Function

	Linear	Constant	Softmax	Sigmoid
SQ-BWN	**7.15**	7.44	7.51	7.37
SQ-TWN	**6.20**	6.30	6.29	6.28

Update Stochastic Quantization Ratio

	Exponential	Average	Fine-Tune
SQ-BWN	**7.15**	7.35	7.18
SQ-TWN	**6.20**	6.88	6.62

Benchmark Results

CIFAR

	Bits	CIFAR-10		CIFAR-100	
		VGG-9	ResNet-56	VGG-9	ResNet-56
FWN	32	9.00	6.69	30.68	29.49
BWN	1	10.67	16.42	37.68	35.01
SQ-BWN	1	9.40	7.15	35.25	31.56
TWN	2	9.87	7.64	34.80	32.09
SQ-TWN	2	**8.37**	**6.20**	34.24	**28.90**

ImageNet

	Bits	AlexNet-BN		ResNet-18	
		top-1	top-5	top-1	top-5
FWN	32	44.18	20.83	34.80	13.60
BWN	1	51.22	27.18	45.20	21.08
SQ-BWN	1	48.78	24.86	41.64	18.35
TWN	2	47.54	23.81	39.83	17.02
SQ-TWN	2	44.70	21.40	36.18	14.26

British Machine Vision Conference, September, 2017, London