

NeurIPS 2025

Efficient Multi-bit Quantization Network Training via Weight Bias Correction and Bit-wise Coreset Sampling

Jinhee Kim*, Jae Jun An*, Kang Eun Jeon, and Jong Hwan Ko

**Equal Contribution*



Our Motivation & Objective

Reduce training cost of multi-bit quantization networks

Process



: Training

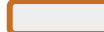


: Calibration

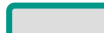


: Adaptation

Model



: Trained bits

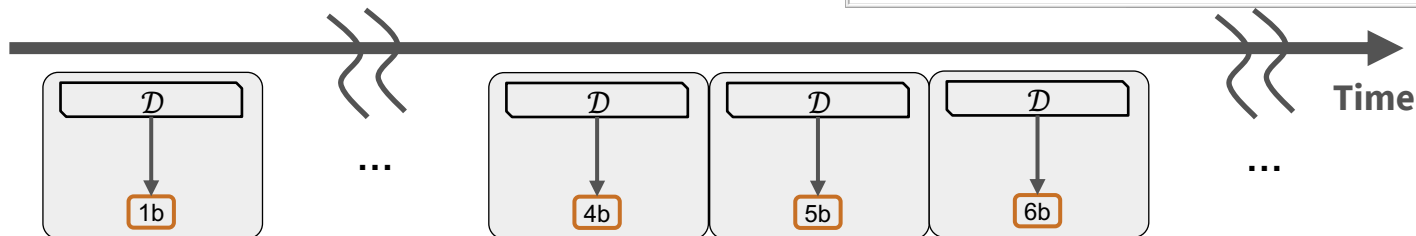


: Calibrated bits

\mathcal{D} : Full dataset

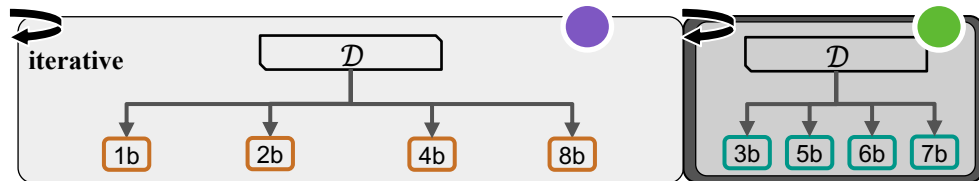
Dedicated Models

multiple INT models



AnyPrecision

*single FP parent model
quantized to INT child models*



BN Calibration Stage

Need to eliminate post-training calibration:

Requiring separate BN layers for different bit-width models force additional calibration stages.

→ Additional overhead and complicated training pipelines.




Lack of adaptive data utilization across bit-widths:

Current multi-bit networks require full-dataset updates for every supported bit-width → *Training cost that scales with the number of precisions.*

Our Motivation & Objective

Reduce training cost of mult-bit quantization networks

Process

 : Training
 : Calibration
 : Adaptation

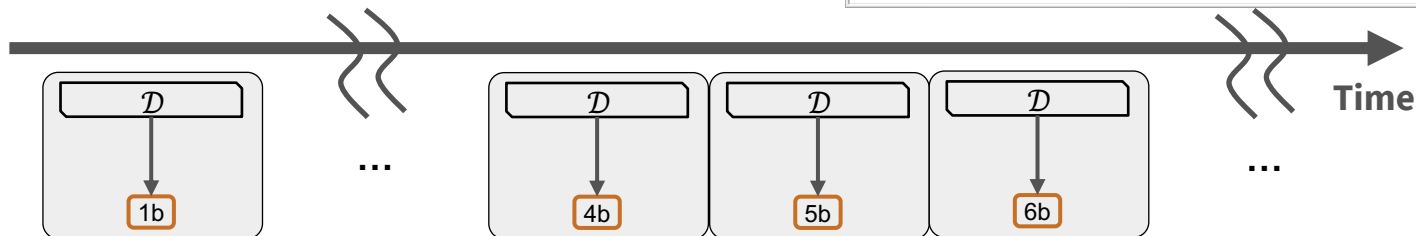
Model

 : Trained bits

 \mathcal{D} : Full dataset

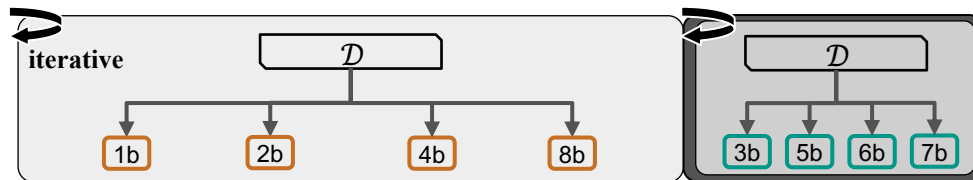
Dedicated Models

multiple INT models



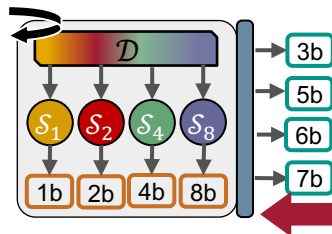
AnyPrecision

*single FP parent model
quantized to INT child models*



Ours

*With weight bias correction and
bit-wise coreset sampling,
significantly reduce training cost*



5.33×~7.88× reduction

Observation 1

Activation mismatch originates from quantization-induced weight bias

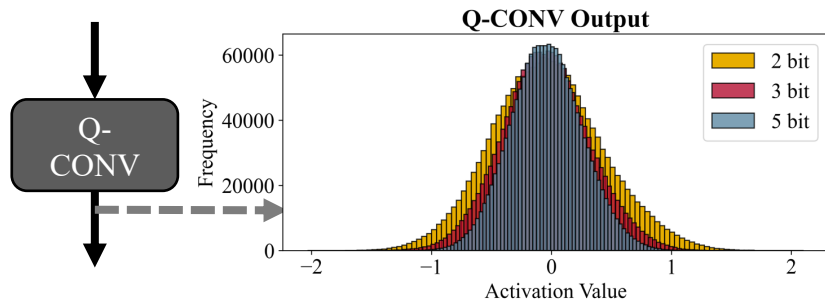


Fig. Mismatch in activation distributions between bit-widths (top), and variance ratio between quantized and original weights (bottom)

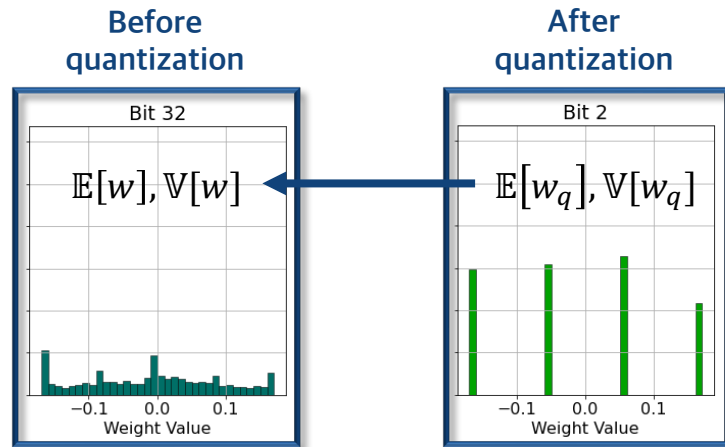
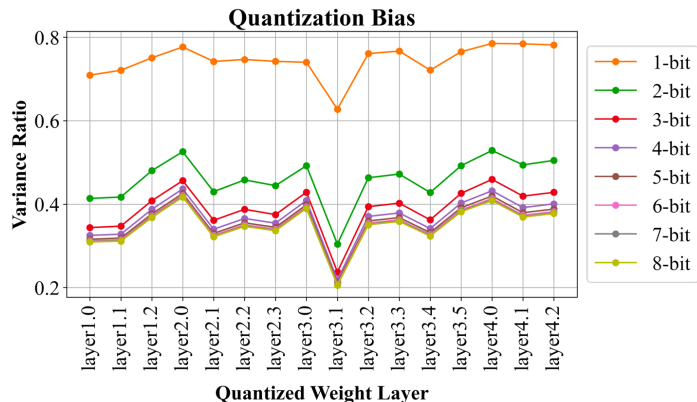


Fig. Full-precision and quantized weights

$$w'_q = \sqrt{\frac{\mathbb{V}[w]}{\mathbb{V}[w_q]}} \left(w_q + (\mathbb{E}[w] - \mathbb{E}[w_q]) \right)$$

Weight Bias Correction
→ Mean/variance correction term

**+ BN
Adaptation**

Observation 2

1) Gradient alignment across bit-widths, 2) Temporal drift in sample importance

1. Gradient alignment across bit-widths

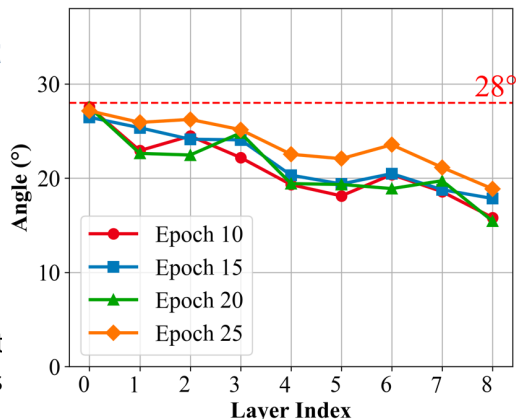


Fig. Angle between the 8-bit 2-bit gradients across layers

Bit-wise training for score extraction:

Obtain per-bit importance scores by isolating gradients, and without interference from other precisions.

2. Temporal drift in sample importance

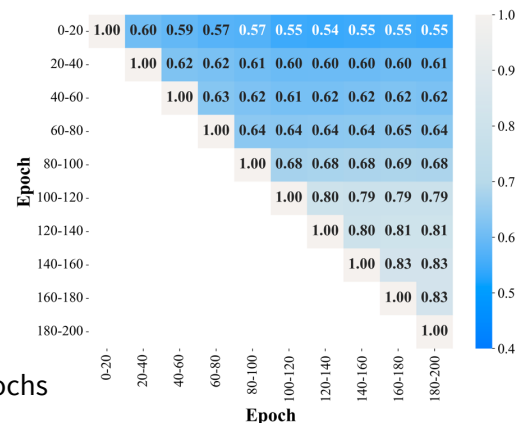
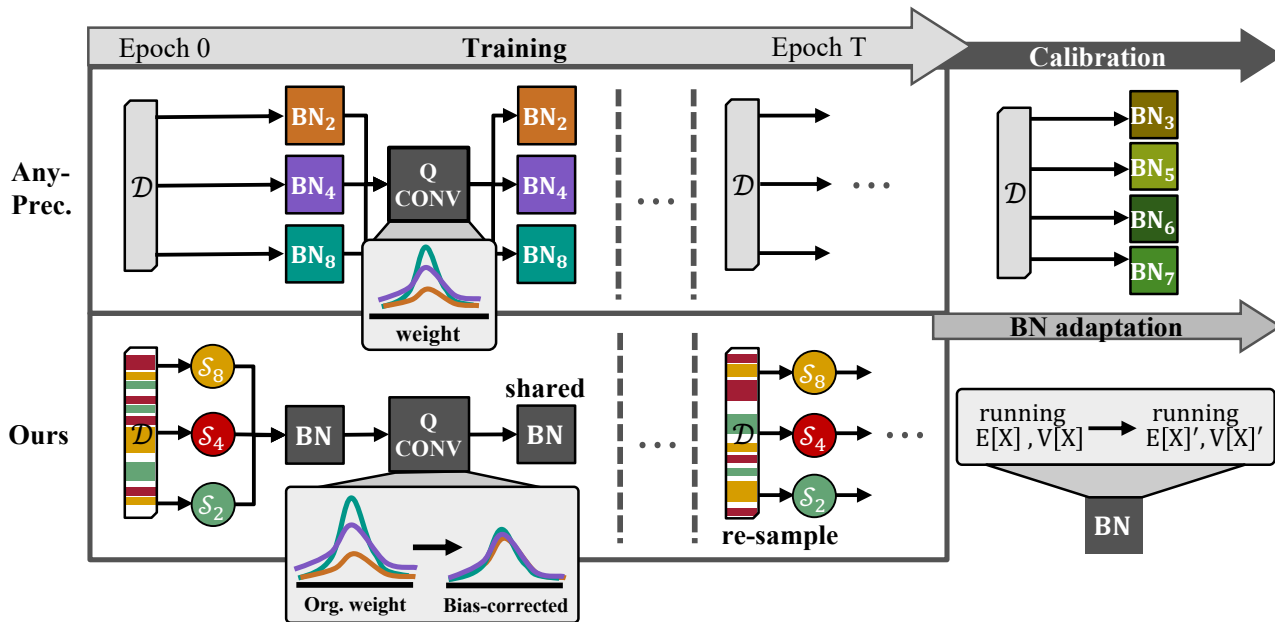


Fig. Spearman correlation between ranks at different epochs

Temperature-based sampling

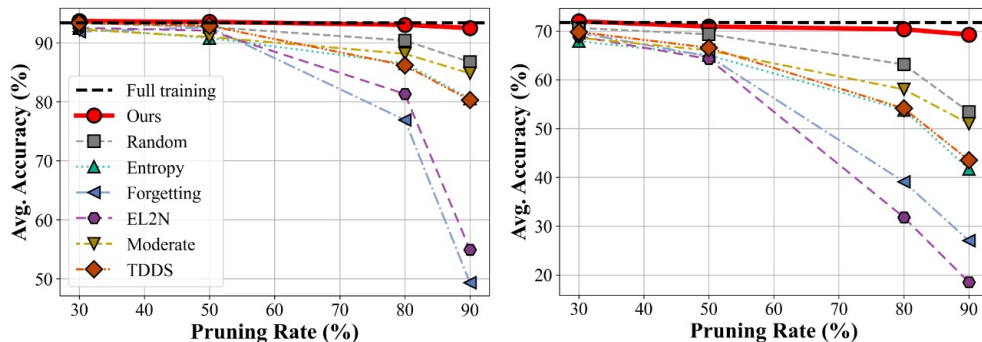
Balance focus on high-importance samples and maintaining diversity.

Overview of EMQNet



EMQNet trains a multi-bit network by applying **weight bias correction** for bit-width activation alignment and **bit-wise coreset sampling** guided by per-bit importance scores.

Accuracy Evaluations



- Our method consistently outperforms six baselines and maintains strong performance even at **90% data pruning**.

Fig. Average accuracy across all bit-widths on CIFAR-19 (left), CIFAR-100 (right)

- Our method achieves up to **~8.41× faster training** with **competitive or higher accuracy** across all bit-widths on ViTs.

Tab. ViTs on TinyImageNet for different pruning rates

Dataset	Framework	Pruning Rate	Test Accuracy					GPU hours (Speed up)
			2bit	4bit	6bit	8bit	Avg.	
CIFAR-100	Dedicated	-	87.14	87.92	87.88	88.03	87.74	41.01 (1.00×)
	Any-Prec.	-	87.52	88.30	88.20	88.21	88.08	10.47 (3.92×)
	Ours	50%	87.83	88.56	88.68	88.59	88.43	6.05 (6.78×)
		60%	87.61	88.45	88.54	88.65	88.31	5.20 (7.89×)
TinyImageNet	Dedicated	-	82.61	85.60	85.68	85.86	84.94	74.00 (1.00×)
	Any-Prec.	-	82.10	84.61	84.47	84.70	84.07	19.32 (3.83×)
	Ours	50%	82.54	84.95	85.33	85.17	84.59	10.50 (7.05×)
		60%	82.89	84.39	84.95	84.86	84.26	8.80 (8.41×)

Breakdown of GPU hours

Tab. GPU hours breakdown for CIFAR-10 and CIFAR-100

Dataset	Framework	GPU hours				Total GPU hours (Speed up)
		Training	Calibration	Adaptation	Scoring	
CIFAR-10	Dedicated	11.97	-	-	-	11.97 (1.00×)
	Any-Prec.	7.51	1.25	-	-	8.76 (1.36×)
	Bias Correction	7.51	-	0.004	-	7.52 (1.59×)
	Bias Correction + Coreset Sampling	1.52	-	0.004	0.37 (offline)	1.52 (7.88×)
CIFAR-100	Dedicated	11.19	-	-	-	11.19 (1.00×)
	Any-Prec.	7.17	1.10	-	-	8.27 (1.36×)
	Bias Correction	7.17	-	0.004	-	7.17 (1.56×)
	Bias Correction + Coreset Sampling	1.47	-	0.004	0.74 (offline)	1.47 (7.61×)

- **Coreset sampling** notably reduces training GPU hours for multi-bit quantization models.
- However, coreset sampling **cannot remove** the cost of the calibration phase by itself.
- Bias Correction and BN Adaptation are applied to **eliminate the calibration step**.

Conclusion

- **Identified the training bottleneck for multi-bit networks**
 - The extra calibration phase to align activation distributions is expensive.
 - Each supported bit-width requires full-dataset updates
- **Proposed bias correction and bit-wise coreset sampling**
 - OBS 1) Activation mismatch originates from quantization-induced weight bias
 - OBS 2) Gradients across bit-widths are highly aligned, and sample importance drifts over time
- **Accuracy and training cost reduction**
 - Consistent efficiency gains across ResNet and ViT architectures
 - Achieves up to **7.88× reduction** in training GPU hours

- END -

Questions or comments?