PTMQ: Post-training Multi-Bit Quantization of Neural Networks

Ke Xu^{1,2}, Zhongcheng Li^{2}, Shanshan Wang^{1*}, Xingyi Zhang^{1,3*}

[1]Information Materials and Intelligent Sensing Laboratory of Anhui Province, Anhui University
[2]School of Artificial Intelligence, Anhui University, Hefei, China
[3]School of Computer Science and Technology, Anhui University, Hefei, China
{xuke,wang.shanshan}@ahu.edu.cn lizhongcheng@stu.ahu.edu.cn xyzhanghust@gmail.com

Abstract

The ability of model quantization with arbitrary bit-width to dynamically meet diverse bit-width requirements during runtime has attracted significant attention. Recent research has focused on optimizing large-scale training methods to achieve robust bit-width adaptation, which is a time-consuming process requiring hundreds of GPU hours. Furthermore, converting bit-widths requires recalculating statistical parameters of the norm layers, thereby impeding real-time switching of the bit-width. To overcome these challenges, we propose an efficient Post-Training Multi-bit Quantization (PTMQ) scheme that requires only a small amount of calibration data to perform block-wise reconstruction of multi-bit quantization errors. It eliminates the influence of statistical parameters by fusing norm layers, and supports real-time switching bit-widths in uniform quantization and mixed-precision quantization. To improve quantization accuracy and robustness, we propose a Multi-bit Feature Mixer technique (MFM) for fusing features of different bit-widths to enhance robustness across varying bit-widths. Moreover, we introduced the Group-wise Distillation Loss (GD-Loss) to enhance the correlation between different bit-width groups and further improve the overall performance of PTMQ. Extensive experiments demonstrate that PTMQ achieves comparable performance to existing state-of-the-art post-training quantization methods, while optimizing it speeds up by 100× compared to recent multi-bit quantization works. Code can be available at https://github.com/xuke225/PTMQ.

Introduction

Model quantization reduces computation and storage by converting weights and activation values into lower precision fixed-point values, thereby enabling the deployment of deep neural networks on resource-constrained hardware platforms. However, most quantization methods (Zhang et al. 2018; Choi et al. 2018; Esser et al. 2020; Liu et al. 2020; Li et al. 2021b; Nagel et al. 2022; Liu et al. 2023) are only able to achieve a predetermined bit-width for quantization, and modifying the bit-width necessitates re-optimization. Multi-bit quantization (Jin, Yang, and Liao 2020; Shkolnik et al. 2020; Yu et al. 2021; Xu et al. 2022) provide a significant opportunity for more powerful model compression and acceleration for these very different scenarios with different platforms. It can achieve different resource budgets by adjusting the bit-width of the quantized model during inference without further training. Multi-bit quantization improves the scalability of models and their adaptability to different computational resources.

Recent works (Shkolnik et al. 2020; Yu et al. 2021; Xu et al. 2022) for multi-bit quantization rely on Quantization-Aware Training (QAT) methods (Zhou et al. 2016; Esser et al. 2020; Bhalgat et al. 2020) to achieve robust adaptive bit-width optimization, and the optimization process is time-consuming. For instance, the optimization of ResNet50 in MultiQuant (Xu et al. 2022) requires 1296 GPU hours using NVIDIA 2080Ti. Furthermore, in order to achieve stable quantized training, norm layers are typically not folded with the previous layer. When switching the bit-width, statistical parameters of the norm layers need to be recalculated, which affects the real-time response of multi-bit quantization.

Post-Training Quantization (PTQ) only requires a small number of unlabeled calibration samples to quantize the pre-trained models without retraining, which is suitable for rapid deployment. Currently, there are two mainstream approaches: searching for quantization scale factors (Migacz 2017; Banner, Nahshan, and Soudry 2019; Nahshan et al. 2019) and optimizing rounding values (Nagel et al. 2020; Hubara et al. 2021; Wei et al. 2022; Liu et al. 2023). These metrics aim to determine the ideal range by minimizing the discrepancy between FP32 and quantized feature maps such as Mean Squared Error (MSE) distance (Choukroun, Kravchik, and Kisilev 2019) and cosine distance (Wu et al. 2020). Searching only scaling factors is insufficient for achieving robust multi-bit quantization, and full-parameters fine-tuning can lead to overfitting with limited data. Fortunately, rounding value optimization methods offer possibilities for post-training multi-bit quantization. Additionally, folding norm layers with previous layers helps mitigate the influence of statistical parameters (Li et al. 2021b). However, how to optimize rounding values and scaling factors using calibration samples to achieve high precision in multi-bit quantization has become a new technical challenge.

To address the above-mentioned issues, we design a novel framework for post-training multi-bit quantization, called PTMQ. To the best of our knowledge, it is the first to collaborate the multi-bit-width quantization into the PTQ frame-

[*]Corresponding Authors.

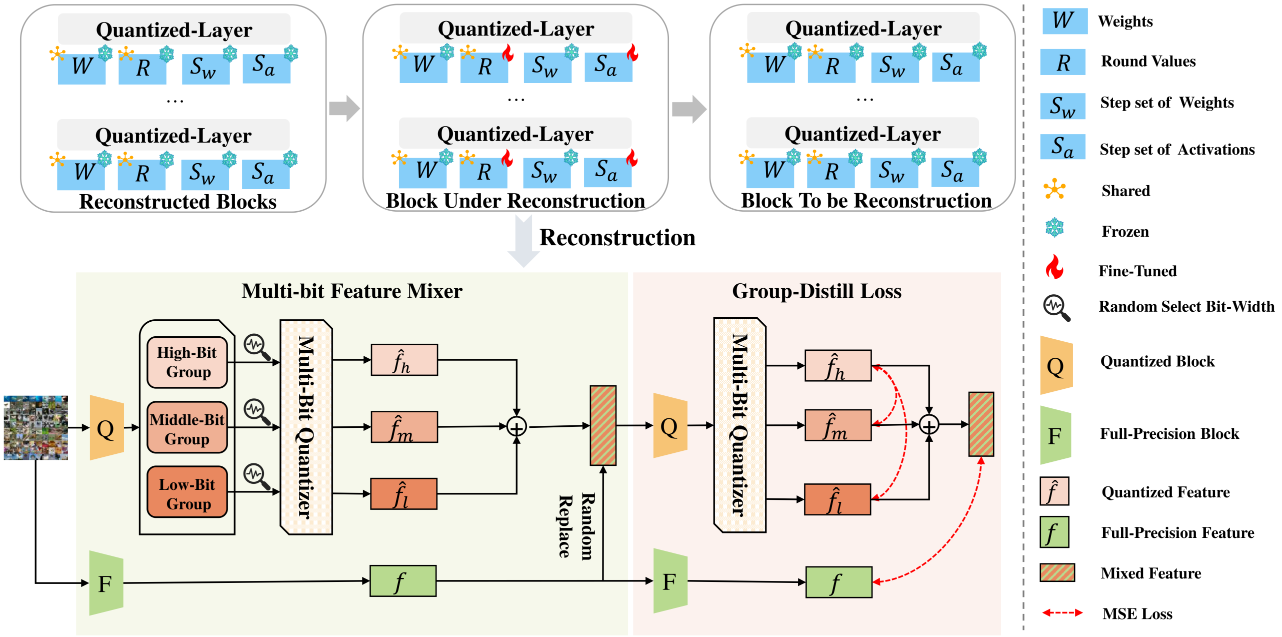


Figure 1: Overview of our proposed Post-Training Multi-Bit Quantization (PTMQ). PTMQ calibrates the step-size set S_a of activations and weight rounding values R in block-wise reconstruction, weights W and the step-size set S_w of weights are frozen. The quantitative reconstruction process consists of two stages: (1) merging multiple bit-width features from different bit groups by Multi-bit Feature Mixer; (2) optimizing rounding values using Group-Distill Loss.

work. The pipeline of PTMQ is illustrated in Figure 1. The PTMQ scheme enables real-time conversion of different bit-widths and mixed-precision quantization. To achieve robust bit-width adaptation, PTMQ utilizes shared weights and rounding values. Through block-wise PTQ reconstruction, the robust rounding values across varying bit-widths are learned. In order to enhance the robustness across various bit-widths, we propose the Multi-bit Feature Mixer (MFM), which can perform block-wise reconstruction of multi-bit quantization errors by fusing features of different bit-widths. In the reconstruction process, the mixed features serve as the input for each block. We then introduce Group-wise Distillation Loss (GD-Loss) to enhance the correlation between different bit-width groups by capturing and transferring the high-bit-width representations to the reconstruction of low-bit-width representations and further improve the overall performance of PTMQ. Overall, the PTMQ framework offers to calibrate the multi-bit quantized model only once with post-training quantization methods and supports uniform and mixed-precision quantization by adjusting the bit-width of models without additional operations. Our contributions are summarized as follows:

- We propose PTMQ, an efficient multi-bit quantization framework based on PTQ. The multi-bit quantized model with PTMQ supports uniform and mixed-precision quantization, and can perform real-time bit-width conversion.
- We propose a Multi-bit Feature Mixer (MFM), which enhances the robustness of rounding values at various bit-widths by fusing features of different bit-widths.
- We introduce the Group-wise Distillation Loss (GD-Loss), to enhance the correlation between different bit-width groups, thereby improving the overall quantization performance of PTMQ.
- Extensive experiments conducted on CNN and ViT backbones verify that PTMQ performs comparably to current PTQ methods, while achieving a 100× speed-up compared to recent multi-bit quantization approaches.

Related Works

Rounding-based PTQ. The rounding-based PTQ approach focuses on optimizing the rounding value and scaling factors with some unlabeled calibration data. AdaRound (Nagel et al. 2020) was the first to propose a method for learning the rounding mechanism by analyzing the second-order error term and suggesting a layer-by-layer reconstruction of the output. BRECQ (Li et al. 2021a) successfully extends AdaRound to a block-wise reconstruction. QDrop (Wei et al. 2022) found that higher accuracy could be achieved by randomly dropping quantized activation values and incorporating activation quantization into the weight tuning. FlexRound (Lee et al. 2023) provides weights with the opportunity to be mapped to a wider range of quantized values, rather than being limited to only the nearby 0 or 1 values during the quantization process. However, they can only quantize for a specific bit-width at a time, and it is not directly accessible to perform multi-bit quantization with them. Our work incorporates a rounding-based PTQ scheme to solve the multi-bit optimization problem.

Multi-Bit Quantization. Recently, several research works on multi-bit quantization. RobustQuant (Shkolnik et al. 2020) proves that the uniformly distributed weight tensor is more tolerant to quantization, has a higher signal-to-noise ratio, and is less sensitive to specific quantizer implementations than the typical case of normally-distributed weights, and introduces Kurtosis regularization to unify the weight distribution and improve its quantization robustness. Any-Precision (Yu et al. 2021) method trains the model using the DoReFa (Zhou et al. 2016) quantizer, but the quantized model is stored in the floating-point form. In addition, the runtime floating-point model can be flexibly set to a different bit-width directly by truncating the least significant bits. MultiQuant (Xu et al. 2022) overcomes the vicious competition between high-bit-width and low-bit-width quantization networks by using an adaptive soft-labeling strategy to enhance multi-bit supernet training. Previous studies have focused on using QAT methods to train a multi-bit quantized neural network, while the heavy training cost of QAT methods renders them less feasible for rapid deployment in the context of diverse models in different scenarios. In addition, the necessity to recalculate the statistical parameters of the norm layers in order to adapt to different bit-widths during runtime prevents real-time switching of bit-widths. Consequently, the extra computational load is introduced due to norm layers during inference. While our approach is efficient for multi-bit quantization through post-training quantization, and the impact of statistical parameters can be mitigated by fusing norm layers.

Approach

In this section, we start by modeling post-training multi-bit quantization, and then the PTMQ pipeline is described. Following that, we provide a detailed explanation of the optimization of PTMQ, including MFM and GD-Loss. Additionally, we introduce a mixed precision method for PTMQ.

Post-Training Multi-Bit Quantization Modeling

We start by modeling the multi-bit quantization problem based on the rounding-based PTQ method. Note that to simplify the description, we omit the zero-point in the quantization process. For the multi-bit quantization model, we assume that the set of bit-width candidates $\mathcal{B} = \{b|b \in \mathbb{Z}^{+}, 2 \leq b \leq 8\}$. The optimization target of multi-bit quantized models can be formulated as:

$$\min_{\mathcal{R}^{*},\boldsymbol{s_a}^{*}} \sum_{b\in\mathcal{B}} \mathbb{E}\left[\left(\mathcal{N}\left(\boldsymbol{w};\boldsymbol{x}\right) - \mathcal{N}_b\left(\hat{\boldsymbol{w}}|\left(\mathcal{R},\boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}\right);\hat{\boldsymbol{x}}|\boldsymbol{s}_{\boldsymbol{a}}^{\boldsymbol{b}}\right)\right)^{2}\right] \tag{1}$$

where $\mathcal{N}_b\left(\hat{\boldsymbol{w}}|\left(\mathcal{R},\boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}\right);\hat{\boldsymbol{x}}|\boldsymbol{s}_{\boldsymbol{a}}^{\boldsymbol{b}}\right)$ denotes the weigths and activations of quantized network under b-bit quantization, $\mathcal{R}$ represents the rounding value of weights, and $\boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}$ is the quantization step size with b-bit of weights. The quantization parameters are usually optimized as block-independent sequences in PTQ methods. For block-wise optimization, we thus end up with the following optimization problem

$$\min_{\mathcal{R}^{*},\boldsymbol{s_a}^{*}} \sum_{b\in\mathcal{B}} \mathbb{E}\left[\left(f^{\ell}\left(\boldsymbol{w};\boldsymbol{x}\right) - f_b^{\ell}\left(\hat{\boldsymbol{w}}|\left(\mathcal{R},\boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}\right);\hat{\boldsymbol{x}}|\boldsymbol{s}_{\boldsymbol{a}}^{\boldsymbol{b}}\right)\right)^{2}\right] \tag{2}$$

f_b^{ℓ} represents the ℓ-th block under b-bit quantization. Multi-bit quantization of PTQ aims to learn robust rounding values and stand-alone quantization step size set of activation with few unlabeled data under different bit-widths. For rounding values, we optimize over soft-quantized weights

$$\hat{\boldsymbol{w}} = \boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}} \cdot \mathrm{clip}\left(\left\lfloor \frac{\boldsymbol{w}}{\boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}} \right\rfloor + \mathcal{R}, -2^{b-1}, 2^{b-1}-1\right) \tag{3}$$

During reconstruction, $\mathcal{R}$ is a learnable continuous variable w.r.t. $\boldsymbol{w}$, which is regularized to converge to $\{0,1\}$. While $\mathcal{R} \in \{0,1\}$ is the rounding value for up or down in inference. The robust rounding values and quantization step size set of activations will be optimized together.

The Pipeline of PTMQ

A conceptual overview of the reconstruction process is depicted in Figure 1. ❶ For initialization, the scaling factors of the model are initialized with the given data. Block-wise optimization is adopted for reconstruction. ❷ Intuitively, the bit-width configuration $\mathcal{B}$ is divided into three groups, each requiring different scaling levels of tuning during the reconstruction process (The supplementary material D provides further details on the experiments conducted on the effects of these subgroups). ❸ During the reconstruction of a block, a random bit is selected for each bit group. The inputs of the block are obtained by integrating the output features of reconstructed blocks from different bit groups through the Multi-bit Feature Mixer. ❹ The outputs of the block undergoing reconstruction are calculated by sampling random bits from each bit group. The reconstruction loss is then aggregated over all sampled bit-widths. Then, all rounding values and step sets of activation within the block are optimized through back-propagation. PTMQ aims to enhance the robustness of the quantized block across different bit-widths by simultaneously improving the lower performance bound (low group bit-width model) and the upper performance bound (high group bit-width model).

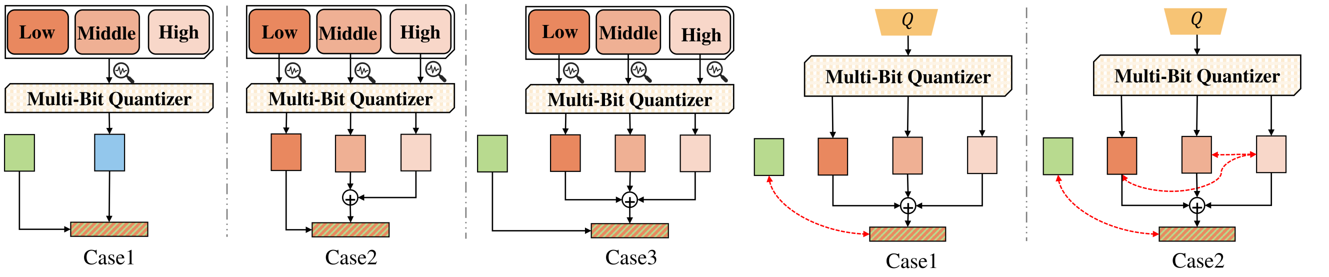
Optimization-based Multi-Bit Quantization

Absorb Multi-Bit Errors into Rounding Values. Conventionally, activation quantization is often modeled by adding noise to the full-precision counterpart, represented as $\hat{a} = a \cdot (1+u)$. The range of u is affected by the bit-width and rounding error. This transformation can absorb the noise on activation and transfer it to the weight. For instance, in a simple matrix-vector multiplication $\boldsymbol{wa}$ during the forward pass, this can be expressed as:

$$\boldsymbol{w}\left(\boldsymbol{a} \odot \left(1+\boldsymbol{u}(\boldsymbol{x})\right)\right) = \left(\boldsymbol{w} \odot \left(1+\boldsymbol{v}(\boldsymbol{x})\right)\right)\boldsymbol{a} \tag{4}$$

$1+\boldsymbol{u}(\boldsymbol{x})$ represents the activation noise, where the noise is related to a specific input data point $\boldsymbol{x}$. The perturbation on the weight is denoted as $1+\boldsymbol{v}(\boldsymbol{x})$. The symbol $\odot$ denotes element-wise multiplication for matrices or vectors.

The quantization noise $1+\tilde{\boldsymbol{u}}(\boldsymbol{x})$, which combines the activation with multi bit-widths, can be transposed into perturbation on the weight $1+\tilde{\boldsymbol{v}}(\boldsymbol{x})$, which can be denoted as $\mathcal{R}$ in the reconstruction process. Since the combination of activation quantization with multi bit-width can be seen as introducing multi-bit quantization errors into the network weights, these errors can be absorbed by rounding values.



(a) Ablation analysis of Multi-Bit Feature Mixer (b) Ablation analysis of Group-Distill Loss

Figure 2: Ablation study for the MFM and GD-Loss. In (a), different components are introduced in the mixed features. In (b), case 1 is conducted without GD-Loss. The illustration module's meaning aligns with the legend of Figure 1.

Multi-Bit Feature Mixer. As previously mentioned, mixing features with multi-bit quantization as inputs can enhance the robustness of rounding values for various bit-widths. To delve deeper into the utilization of quantization features with multi bit-width, we conducted an ablation study by employing different input feature compositions. Figure 2a demonstrates the three cases that are examined:

- Case 1: Randomly select one type of feature quantization from low, medium, and high bit-width groups, and then fuse it with full-precision features through random dropout as the input for the reconstruction block.
- Case 2: Firstly, random bit features are sampled from each bit group. Secondly, the high-bit group features are mixed with the mid-bit group features through addition operation. Finally, the fused features, obtained by randomly replacing with the low-bit group features, are used as the input for the reconstruction block.
- Case 3: Random bit features are sampled in each bit group and mixed them by addition operation, and then fused with full-precision features by randomly dropping as input to the reconstruction block.

Case 3 is verified experimentally as the most efficient Multi-bit Feature Mixer (MFM) in PTMQ. It cleverly aggregates multi-bit quantized and full-precision features, so that the input of the reconstruction block combines multiple quantization features instead of only biasing towards one quantization feature. The MFM can be formalized as follows:

$$\text{MFM}\left(\hat{f}_{\{l,m,h\}}, f\right) = \begin{cases} f & \text{with } p \\ \lambda_1 \hat{f}_l + \lambda_2 \hat{f}_m + \lambda_3 \hat{f}_h & \text{with } 1-p \end{cases} \tag{5}$$

In the MFM operation, each element is processed individually. $\hat{f}_{\{l,m,h\}}$ denotes an element in the features with $\{low, middle, high\}$-bit group selected bit of the reconstructed blocks on given mini-batch data. The full-precision feature is denoted as f. The hyperparameters $(\lambda_1, \lambda_2, \lambda_3)$ are used to control the scale of fusion for different bit groups. To enhance the flatness of the optimization landscape, we randomly drop features that are mixed with the full-precision ones. The dropping probability is set to $p =$ 0.5, which aligns with QDrop (Wei et al. 2022). Furthermore, the output features of the block are also fused with different selected bits from bit groups. This guarantees consistency with the input scheme, as the output of this block becomes the input for the subsequent block, thereby improving the adaptability of multi-bit quantization.

Group-Distill Loss. To enhance the correlation between different bit-width groups and further improve the overall performance and robustness of PTMQ, we propose a group-wise distillation strategy. The advantage of this strategy is inherent in PTMQ, as the output features of the block with different bit groups in reconstruction are always accessible. As shown in Figure 2b, in Case 1, the mixed features of each bit group are supervised by the full-precision features. In Case 2 with GD-Loss, based on Case 1, high-bit group features is introduced to supervise the features of the middle-bit and low-bit groups. The goal of supervision is to minimize the mean squared error between different features. The reconstruction loss of the block with GD-Loss is described as:

$$\begin{aligned} \mathcal{L}_{recon} &= \gamma_1 \text{MSE}(\hat{\mathcal{O}}_{mixed}, \mathcal{O}_{fp32}) \\ &+ \gamma_2 \text{MSE}(\hat{\mathcal{O}}_h, \hat{\mathcal{O}}_m) \\ &+ \gamma_3 \text{MSE}(\hat{\mathcal{O}}_h, \hat{\mathcal{O}}_l) \end{aligned} \tag{6}$$

where $\hat{\mathcal{O}}_{mixed}$ means the mixed output feature of the block with MFM, while $\mathcal{O}_{fp32}$ denotes the output of corresponding full-precision block, $\hat{\mathcal{O}}_{\{l,m,h\}}$ denotes the output feature with $\{low, middle, high\}$-bit group selected bit quantization. $(\gamma_1, \gamma_2, \gamma_3)$ are hyperparameters to control the scale of GD-Loss (Detailed sensitivity analysis of λ and γ, see Supplementary Material G).

Mixed Precision Quantization with PTMQ

To further push the performance of PTMQ, we employ mixed precision techniques with PTMQ. PTMQ enables real-time conversion for mixed-precision quantization by just switching the quantization step size of blocks without additional analysis and optimization.

Sensitivity Analysis of Blocks. After obtaining the multi-bit quantized model through PTMQ, we analyze the sensitivity of quantization for each model block while storing the

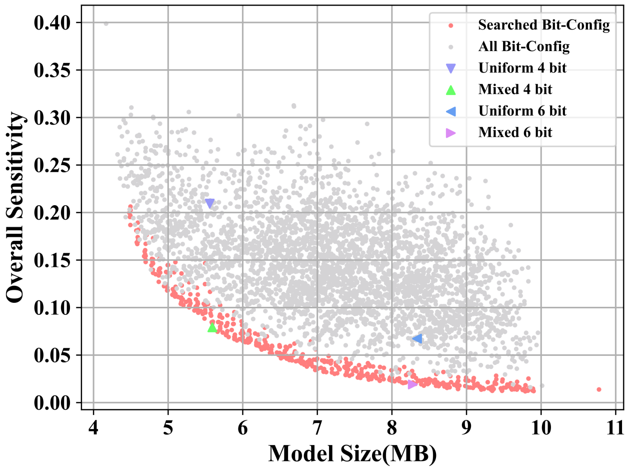


Figure 3: The Pareto frontier of ResNet-18 on ImageNet. The x-axis shows the resulting model size for each configuration, and the y-axis shows the resulting sensitivity.

measured sensitivity from the calibration dataset. The sensitivity of the quantized blocks is measured using KL divergence (Cai et al. 2020), which is defined as:

$$\Omega_i(b) = \mathrm{KL}\left(\mathcal{N}\left(\boldsymbol{w};\boldsymbol{x}\right), \mathcal{N}_b^i\left(\hat{\boldsymbol{w}}|\left(\mathcal{R}, \boldsymbol{s}_{\boldsymbol{w}}^{\boldsymbol{b}}\right); \hat{\boldsymbol{x}}|\boldsymbol{s}_{\boldsymbol{a}}^{\boldsymbol{b}}\right)\right) \quad (7)$$

Here, $\Omega_i(b)$ represents the measurement of sensitivity for the i-th block when quantized to b-bits, and $\mathcal{N}_b^i$ denotes the quantized neural network with the i-th block under b-bit precision. If $\Omega_i(b)$ is small, it indicates that the output of the quantized model does not deviate significantly from the output of the full-precision model when the i-th block is quantized to b-bits. This implies that the i-th block is relatively insensitive to b-bit quantization. Conversely, if $\Omega_i(b)$ is large, it suggests that the i-th block is more sensitive.

Pareto Frontier Search for Mixed Precision. Once we obtain the sensitivity table, it can be used to rank the blocks based on their relative sensitivity. The main idea is to allocate higher bit-width to sensitive blocks and lower bit-width to less sensitive ones. To achieve this, we employ the Pareto frontier search method to assign a specific bit width to each block. Given a target quantized model size S_t, we aim to find the bit-width configuration that minimizes the overall sensitivity while satisfying the constraint on the target model size. This optimization problem can be defined as follows:

$$\min_{\{b_i\}_{i=1}^L} \Omega_{\text{sum}} = \sum_{i=1}^{L} \Omega_i\left(b_i\right) \text{ s.t. } \sum_{i=1}^{L} P_i * b_i \leq S_t \quad (8)$$

where b_i is the quantization bit-width of the i-th block, and P_i is the size for the i-th block. We use the genetic algorithm (Guo et al. 2020) to search the optimal bit-width configuration with model size threshold.

Experiments

Performance Comparison with SOTA Method

We assess the performance of the proposed PTMQ scheme on various CNN-based architectures (ResNet (He et al. 2016), MobileNetV2 (Sandler et al. 2018), RegNet (Radosavovic et al. 2020)) and transformer-based architectures (ViT (Dosovitskiy et al. 2021), DeiT (Touvron et al. 2021)) on ImageNet (Russakovsky et al. 2014) dataset. To our knowledge, prior research has yet to be conducted on multi-bit quantization using PTQ. Therefore, we compare the performance of PTMQ with existing PTQ approaches and multi-bit quantization using QAT methods.

Results of CNN-based Architectures. We compare the accuracy results of several recently proposed quantization methods, including single-bit post-training quantization and multi-bit quantization. The summarized results are presented in Table 1. Specifically, when comparing ResNet-18 to AdaRound (Nagel et al. 2020) and BRECQ (Li et al. 2021a) using 3-bit quantization, our method achieves an accuracy improvement of 0.8% and 0.6% respectively. In the 4-bit setting, PTMQ outperforms AdaRound by 0.3% but experiences a decrease in accuracy (0.7%) compared to BRECQ. This is due to the allocation of 3- and 4-bits in the same bit group, leading to competition for different bit-widths within the group. However, with other bit-widths, such as those used in BRECQ and QDrop (Wei et al. 2022), we can achieve almost lossless accuracy with only a 0.20%/0.3% drop.

For ResNet-50 and RegNetX-600MF, the performance gap between PTMQ and AdaRound, as well as BRECQ, widens at 3-bit quantization. PTMQ achieves an accuracy improvement of 3.0% and 1.11% respectively on ResNet-50, while RegNetX-600MF demonstrates an accuracy improvement of 8.28% and 4.13%. The performance gap with QDrop narrows to 1% for ResNet-50 under 4-bit quantization, while ResNet-18 exhibits a performance gap of 2% with QDrop. This can be attributed to the fact that ResNet-50 is a heavier model than ResNet-18, where the distribution of activations across the adjacent quantization bit-widths is much more similar (Xu et al. 2022), and competition within the same group is relatively alleviated. On the compact model MobileNet-V2, our proposed scheme achieves accuracy boosts of 15.6% and 1.6% respectively compared to AdaRound at 3- and 4-bit. Due to the large differences in the distribution of output channels in MobileNet-V2 (Nagel et al. 2019), PTMQ exhibits performance degradation compared to BRECQ and QDrop, with an average drop of 0.76% and 1.31%, respectively.

Except for post-training quantization methods, we compare our approach with multi-bit methods (Shkolnik et al. 2020; Yu et al. 2021; Xu et al. 2022) based on QAT. Compared with RobustQuant (Shkolnik et al. 2020), the proposed PTMQ shows 7.6%/4.1% accuracy boosts at 3-bit for ResNet-18 and ResNet-50 models, respectively. At high bit-widths (e.g., 6 to 8 bits) the performance of PTMQ is comparable to MultiQuant (Xu et al. 2022)(within 0.5% drop on ResNet-18/50 and MobileNet-V2). Noting that PTMQ achieves 100× optimization acceleration compared to MultiQuant. Moreover, our method only requires once calibration to adapt the model for multi-bit quantization without task of repetitive optimizations.

Model	Benchmark	Criterion	Mixed	BN Folding	Bit-width				FP32	GPU Hour
					3	4	6	8		
ResNet-18	LSQ	QAT			70.60%	71%	——	71.10%	70.50%	60N
	AdaRound	PTQ		✓	64.18%	67.26%	70.17%	70.78%	71.00%	0.45N
	BRECQ			✓	64.24%	68.21%	70.44%	70.80%	71.00%	0.5N
	QDrop			✓	66.26%	69.54%	70.52%	70.83%	71.00%	0.65N
	RobustQuant	QAT-Multi-Bit			57.30%	66.90%	70%	——	70.30%	214
	AnyPrecision				——	67.96%	——	68.04%	68.16%	76
	MultiQuant		✓		67.80%	69.70%	70.50%	70.80%	69.80%	144
	PTMQ	PTQ-Multi-Bit	✓	✓	64.92%	67.57%	70.23%	70.79%	71.00%	1.4
ResNet-50	LSQ	QAT			76.90%	77.60%	——	76.80%	76.90%	240N
	AdaRound	PTQ		✓	66.66%	73.79%	76.14%	76.54%	76.80%	3.6N
	BRECQ			✓	68.56%	74.66%	76.35%	76.67%	76.80%	4.1N
	QDrop			✓	71.07%	74.98%	76.44%	76.65%	76.80%	4.7N
	RobustQuant	QAT-Multi-Bit			66.50%	74.30%	76.20%	——	76.30%	970
	AnyPrecision				——	74.75%	——	74.91%	75.00%	620
	MultiQuant		✓		75.30%	76.40%	76.8%	77%	76.10%	648
	PTMQ	PTQ-Multi-Bit	✓	✓	69.67%	73.93%	76.11%	76.52%	76.80%	7.8
RegNetX-600MF	AdaRound	PTQ		✓	51.01%	68.20%	72.40%	73.38%	73.50%	1.5N
	BRECQ			✓	55.16%	68.33%	72.88%	73.55%	73.50%	1.9N
	QDrop				64.53%	70.62%	73.26%	73.57%	73.50%	2.5N
	PTMQ	PTQ-Multi-Bit	✓	✓	59.29%	68.84%	72.85%	73.32%	73.50%	3.5
MobileNet-V2	AdaRound	PTQ		✓	——	49.28%	68.43%	71.64%	72.40%	1.15N
	BRECQ			✓	——	65.57%	71.11%	72.31%	72.40%	1.4N
	QDrop			✓	——	67.89%	71.78%	72.34%	72.40%	1.6N
	RobustQuant	QAT-Multi-Bit			——	59%	70.0%	——	71.30%	390
	MultiQuant		✓		——	69.60%	70.30%	70.50%	71.90%	334
	PTMQ	PTQ-Multi-Bit	✓	✓	——	64.94%	70.0%	72.05%	72.40%	3.1

Table 1: Comparison of SOTA quantization methods for CNNs on ImageNet. The Mixed refers to mixed-precision quantization. The time measurement is carried out with NVIDIA 3090. We use N to denote the number of up-coming deployment scenarios.

Models	#Bit-Width Weights/Activations									
	4/6		5/6		6/6		7/7		8/8	
	PTQ4ViT	PTMQ	PTQ4ViT	PTMQ	PTQ4ViT	PTMQ	PTQ4ViT	PTMQ	PTQ4ViT	PTMQ
ViT-S/224/16	71.41%	71.67%	74.94%	75.14%	75.95%	76.09%	77.69%	77.14%	78.24%	78.16%
ViT-B/224/16	75.43%	75.00%	76.17%	76.64%	77.66%	77.70%	78.85%	78.62%	78.98%	79.12%
DeiT-S/224/16	74.35%	77.20%	76.14%	78.24%	76.74%	78.74%	79.08%	79.30%	79.49%	79.53%
DeiT-B/224/16	77.95%	80.00%	79.64%	80.62%	80.07%	80.81%	81.20%	81.35%	81.50%	81.54%

Table 2: Summary of results for transformer. PTQ4ViT results reproduced through official open source code[1]. ViT-S/16/224 denotes patch size is 16 × 16, and the input resolution is 224 × 224. All results listed are the top-1 accuracy.

Results of Transformer-based Architectures. In addition, we compare our proposed method on Vision Transformers (Dosovitskiy et al. 2021; Touvron et al. 2021) with PTQ4ViT (Yuan et al. 2022), a post-training quantization framework specifically designed for quantizing transformer models. PTQ4ViT currently achieves state-of-the-art results among all transformer quantization algorithms at 6-bit. The comparative results are presented in Table 2. Interestingly, PTMQ is not limited to CNN models but can also achieve multi-bit quantization on transformer architectures. Specifically, with the bit-width set to 4 and 5, PTMQ achieves a 0.3% to 0.5% average accuracy increase over PTQ4ViT for ViT-S and ViT-B models. PTMQ demonstrates even greater improvements of 2.5% to 1.4% on DeiT-S and DeiT-B models. Remarkably, even for bit-widths ranging from 6 to 8, PTMQ maintains comparable performance to PTQ4ViT.

Results of Mixed Precision. To validate the mixed precision search algorithm, we plot the Pareto frontier with different bit-width configurations. Additionally, we compare the sensitivities of 4-bit and 6-bit quantization using uniform-precision and mixed-precision methods on ResNet-18, as shown in Figure 3. This algorithm efficiently produces the frontier for ResNet-18 within a few minutes.

In addition, We test the potential of mixed precision to fur-

[1] https://github.com/hahnyuan/PTQ4ViT

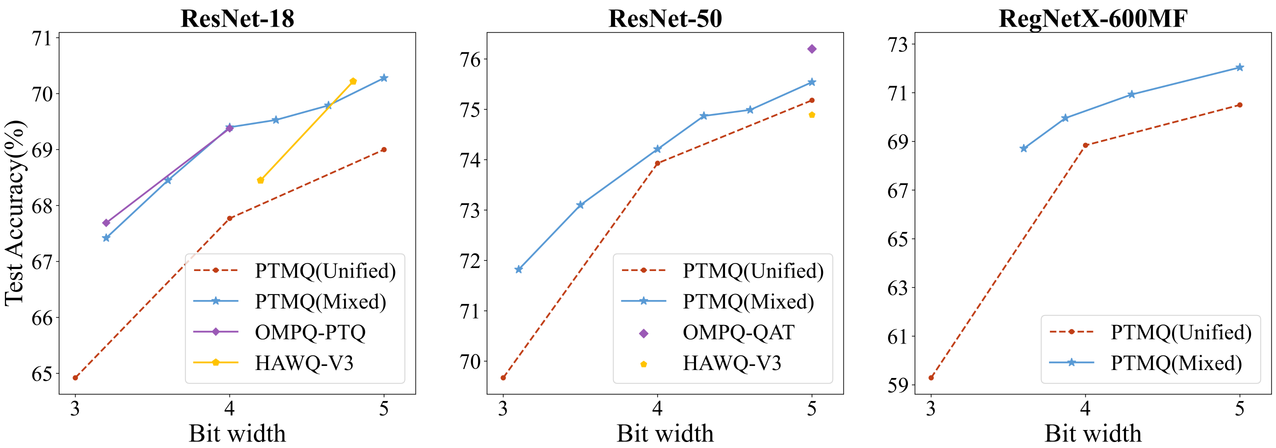


Figure 4: Comparison with HAWQ-V3, OMPQ, unified and mixed precision on PTMQ.

ther push the scalability of PTMQ, and choose ResNet-18/50 and RegNetX-600MF to validate the efficacy of our proposed approach. Experimental results show that mixed precision is superior to uniform precision, as shown in Figure 4. Specifically, 4-bit mixed quantization based on PTMQ improves the accuracy by 2% compared to uniform quantization on ResNet-18. Compared with SOTA's mixed-precision quantization method, PTMQ exceeds HAWQ-V3 (Yao et al. 2021) by 1.08% at 4.2MP and coincides with the results of OMPQ (Ma et al. 2023) on ResNet-18.

Method		W3A3	W4A4	W5A5	Avg
D-Q	Only 3-bit	66.10%	39.24%	63.06%	56.13%
	Only 4-bit	3.32%	69.54%	65.56%	46.14%
P-Q	L ⟶ H	13.71%	39.38%	57.16%	36.75%
	H ⟶ L	59.59%	62.59%	65.82%	62.67%
PTMQ		64.92%	67.57%	69.00%	67.16%

Table 3: Ablation studies for PTMQ, Direct-Quantization and Progressive-Quantization for ResNet-18 on ImageNet.

Ablation Studies

The Effectiveness of PTMQ Framework. We compare different types of PTQ-based multi-bit-width optimization methods, including Direct-Quantization (D-Q) and Progressive-Quantization (P-Q). D-Q optimizes rounding values only for one bit-width; P-Q considers two optimization strategies: gradually increasing the bit-width (from the lowest to the highest) or decreasing the bit-width (from the highest to the lowest) during the optimization process. To ensure the fairness of the experiment, we all use the QDrop (Wei et al. 2022) quantization method. Table 3 shows the results of ResNet-18 using different strategies. D-Q is not suitable for adaptive bit-width switching. P-Q shows better performance as the bit-width increases. However, there is strong competition between different bit-widths, especially from low- to high-bit optimization. In contrast, models optimized with PTMQ significantly improve the average bit-width accuracy, with an average accuracy improvement of 4.49% compared to the high- to low-bit P-D method. This validates the effectiveness of our PTMQ method.

The Effectiveness of Multi-bit Feature Mixer. To further investigate the effectiveness of the proposed Multi-bit Feature Mixer, we performed an ablation study on ResNet-18 using different input feature compositions. The goal is to comprehend the role of each component within the Multi-bit Feature Mixer, as illustrated in Figure 2a. As shown in Figure 5, clearly indicate that both Case 1 and Case 2 exhibit lower performance compared to Case 3 fusion method. Case 1 shows superior performance to Case 2 when it comes to low-bit quantization (e.g., 3- and 4-bit), suggesting that randomly dropping quantized activations can enhance performance in the presence of low-bit constraints. However, both Case 1 and Case 2 are surpassed by Case 3, highlighting the robustness and effectiveness of fusing features from different bit-widths within multi-bit groups.

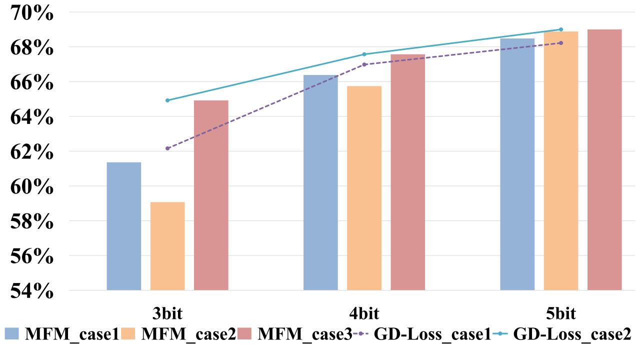


Figure 5: Ablation studies results of MFM and GD-Loss. We validate Top-1 accuracy for ResNet-18 at 3-,4- and 5-bit. The results of MFM are presented by a bar chart and the ablation results of GD-Loss are presented by the line.

The Effectiveness of GD-Loss. We investigate the effect of GD-Loss during reconstruction with PTMQ for ResNet-18. As shown in Figure 5, Case 2 (with GD-Loss) consistently outperforms Case 1 (without GD-Loss). Specifically, for 3-bit and 5-bit quantization, Case 2 achieves accuracy improvements of up to 3% and 1%, respectively. These results prove that GD-Loss effectively alleviates competition phenomena under different bit-widths, thereby enhancing the robustness of PTMQ. Especially in scenarios involving low-bit quantization, combining soft features obtained from high-bit groups can provide better regularization effects for the low-bit reconstruction.

Conclusion

This paper proposes PTMQ, which is the first attempt at efficient multi-bit quantization on the PTQ approach. This novel framework effectively resolves the time-consuming training issues in previous methods for multi-bit quantization. By combining features of different bit-widths using MFM and incorporating the GD-Loss strategy in the reconstruction process, PTMQ achieves performance comparable to state-of-the-art PTQ methods. Furthermore, 100× speed-up is attainable in terms of training time compared to recent works on multi-bit quantization.

Acknowledgments

This work was supported in part by the National Key Research and Development Project (NO. 2018AAA0100105) and the National Natural Science Foundation of China (No. 62206003, No. 62106003, No. U21A20512).

References

Banner, R.; Nahshan, Y.; and Soudry, D. 2019. Post Training 4-bit Quantization of Convolutional Networks for Rapid-deployment. In *Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 7948–7956.

Bhalgat, Y.; Lee, J.; Nagel, M.; Blankevoort, T.; and Kwak, N. 2020. LSQ+: Improving Low-bit Quantization Through Learnable Offsets and Better Initialization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2978–2985.

Cai, Y.; Yao, Z.; Dong, Z.; Gholami, A.; Mahoney, M. W.; and Keutzer, K. 2020. ZeroQ: A Novel Zero Shot Quantization Framework. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 13166–13175.

Choi, J.; Wang, Z.; Venkataramani, S.; Chuang, P. I.-J.; Srinivasan, V.; and Gopalakrishnan, K. 2018. PACT: Parameterized Clipping Activation for Quantized Neural Networks. *ArXiv*, abs/1805.06085.

Choukroun, Y.; Kravchik, E.; and Kisilev, P. 2019. Low-bit Quantization of Neural Networks for Efficient Inference. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 3009–3018.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Esser, S. K.; McKinstry, J. L.; Bablani, D.; Appuswamy, R.; and Modha, D. S. 2020. Learned Step Size Quantization. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Guo, Z.; Zhang, X.; Mu, H.; Heng, W.; Liu, Z.; Wei, Y.; and Sun, J. 2020. Single Path One-Shot Neural Architecture Search with Uniform Sampling. In *European Conference on Computer Vision (ECCV)*, 544–560.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 770–778.

Hubara, I.; Nahshan, Y.; Hanani, Y.; Banner, R.; and Soudry, D. 2021. Accurate Post Training Quantization With Small Calibration Sets. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 4466–4475.

Jin, Q.; Yang, L.; and Liao, Z. 2020. AdaBits: Neural Network Quantization With Adaptive Bit-Widths. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 2143–2153.

Lee, J. H.; Kim, J.; Kwon, S. J.; and Lee, D. 2023. FlexRound: Learnable Rounding based on Element-wise Division for Post-Training Quantization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 18913–18939.

Li, Y.; Gong, R.; Tan, X.; Yang, Y.; Hu, P.; Zhang, Q.; Yu, F.; Wang, W.; and Gu, S. 2021a. BRECQ: Pushing the Limit of Post-Training Quantization by Block Reconstruction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Li, Y.; Shen, M.; Ma, J.; Ren, Y.; Zhao, M.; Zhang, Q.; Gong, R.; Yu, F.; and Yan, J. 2021b. MQBench: Towards Reproducible and Deployable Model Quantization Benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Liu, J.; Niu, L.; Yuan, Z.; Yang, D.; Wang, X.; and Liu, W. 2023. PD-Quant: Post-Training Quantization Based on Prediction Difference Metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 24427–24437.

Liu, Z.; Shen, Z.; Savvides, M.; and Cheng, K.-T. 2020. ReActNet: Towards Precise Binary Neural Network with Generalized Activation Functions. In *European Conference on Computer Vision (ECCV)*, 143–159.

Ma, Y.; Jin, T.; Zheng, X.; Wang, Y.; Li, H.; Wu, Y.; Jiang, G.; Zhang, W.; and Ji, R. 2023. OMPQ: Orthogonal Mixed Precision Quantization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9029–9037.

Migacz, S. 2017. 8-Bit Inference with Tensorrt. In *GPU technology conference*.

Nagel, M.; Amjad, R. A.; van Baalen, M.; Louizos, C.; and Blankevoort, T. 2020. Up or Down? Adaptive Rounding for Post-Training Quantization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, 7197–7206.

Nagel, M.; Fournarakis, M.; Bondarenko, Y.; and Blankevoort, T. 2022. Overcoming Oscillations in Quantization-Aware Training. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, 16318–16330.

Nagel, M.; van Baalen, M.; Blankevoort, T.; and Welling, M. 2019. Data-Free Quantization Through Weight Equalization and Bias Correction. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 1325–1334.

Nahshan, Y.; Chmiel, B.; Baskin, C.; Zheltonozhskii, E.; Banner, R.; Bronstein, A. M.; and Mendelson, A. 2019. Loss Aware Post-Training Quantization. *Machine Learning*, 110: 3245 – 3262.

Radosavovic, I.; Kosaraju, R. P.; Girshick, R. B.; He, K.; and Dollár, P. 2020. Designing Network Design Spaces. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 10425–10433.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. S.; Berg, A. C.; and Fei-Fei, L. 2014. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115: 211 – 252.

Sandler, M.; Howard, A. G.; Zhu, M.; Zhmoginov, A.; and Chen, L. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 4510–4520.

Shkolnik, M.; Chmiel, B.; Banner, R.; Shomron, G.; Nahshan, Y.; Bronstein, A. M.; and Weiser, U. C. 2020. Robust Quantization: One Model to Rule Them All. In *Advances in Neural Information Processing Systems*, volume 33, 5308–5317.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training Data-Efficient Image Transformers & Distillation Through Attention. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 10347–10357.

Wei, X.; Gong, R.; Li, Y.; Liu, X.; and Yu, F. 2022. QDrop: Randomly Dropping Quantization for Extremely Low-bit Post-Training Quantization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

Wu, D.; Tang, Q.; Zhao, Y.; Zhang, M.; Fu, Y.; and Zhang, D. 2020. EasyQuant: Post-training Quantization via Scale Optimization. *ArXiv*, abs/2006.16669.

Xu, K.; Feng, Q.; Zhang, X.; and Wang, D. 2022. MultiQuant: Training Once for Multi-bit Quantization of Neural Networks. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 3629–3635.

Yao, Z.; Dong, Z.; Zheng, Z.; Gholami, A.; Yu, J.; Tan, E.; Wang, L.; Huang, Q.; Wang, Y.; Mahoney, M. W.; and Keutzer, K. 2021. HAWQ-V3: Dyadic Neural Network Quantization. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 11875–11886.

Yu, H.; Li, H.; Shi, H.; Huang, T. S.; and Hua, G. 2021. Any-Precision Deep Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 10763–10771.

Yuan, Z.; Xue, C.; Chen, Y.; Wu, Q.; and Sun, G. 2022. PTQ4ViT: Post-Training Quantization for Vision Transformers with Twin Uniform Quantization. In *European Conference on Computer Vision (ECCV)*, 191–207.

Zhang, D.; Yang, J.; Ye, D.; and Hua, G. 2018. LQ-Nets: Learned Quantization for Highly Accurate and Compact Deep Neural Networks. In *European Conference on Computer Vision (ECCV)*, 373–390.

Zhou, S.; Ni, Z.; Zhou, X.; Wen, H.; Wu, Y.; and Zou, Y. 2016. DoReFa-Net: Training Low Bitwidth Convolutional Neural Networks with Low Bitwidth Gradients. *ArXiv*, abs/1606.06160.