

QuEPT: Quantized Elastic Precision Transformers with One-Shot Calibration for Multi-Bit Switching

Ke Xu^{1,2}, Yixin Wang², Zhongcheng Li⁴, Hao Cui⁴, Jinshui Hu⁴, Xingyi Zhang^{1,3}

¹State Key Laboratory of Opto-Electronic Information Acquisition and Protection Technology, Anhui University, Hefei, China

²School of Artificial Intelligence, Anhui University, Hefei, China

³School of Computer Science and Technology, Anhui University, Hefei, China

⁴iFLYTEK Research, Hefei, China

xuke@ahu.edu.cn, yxwang@stu.ahu.edu.cn, {zcli19, haocui, jshu}@iflytek.com, xyzhanghust@gmail.com

Abstract

Elastic precision quantization enables multi-bit deployment via a single optimization pass, fitting diverse quantization scenarios. Yet, the high storage and optimization costs associated with the Transformer architecture, research on elastic quantization remains limited, particularly for large language models. This paper proposes QuEPT, an efficient post-training scheme that reconstructs block-wise multi-bit errors with one-shot calibration on a small data slice. It can dynamically adapt to various predefined bit-widths by cascading different low-rank adapters, and supports real-time switching between uniform quantization and mixed precision quantization without repeated optimization. To enhance accuracy and robustness, we introduce Multi-Bit Token Merging (MB-ToMe) to dynamically fuse token features across different bit-widths, improving robustness during bit-width switching. Additionally, we propose Multi-Bit Cascaded Low-Rank adapters (MB-CLoRA) to strengthen correlations between bit-width groups, further improve the overall performance of QuEPT. Extensive experiments demonstrate that QuEPT achieves comparable or better performance to existing state-of-the-art post-training quantization methods.

Code — <https://github.com/xuke225/QuEPT>

Introduction

Model quantization reduces computational and memory requirements by representing weights and activations using low-bit fixed-point numbers, enabling deep neural networks to run efficiently on resource-constrained devices. However, most quantization methods (Nagel et al. 2020; Liu et al. 2023b; Lin et al. 2024b; Sun et al. 2025; Li et al. 2025c) are only able to achieve a single predefined bit-width for quantization, and modifying the bit-width necessitates re-optimization. Elastic precision quantization can jointly optimize multiple predefined bit widths while sharing the original weights. This approach enables a quantized model to dynamically adapt to various bit-widths through one-shot calibration, offering flexible precision-efficiency trade-offs while supporting real-time switching between uniform and mixed-precision configurations without requiring separate optimizations for each bit setting.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

While multi-bit quantization has yielded impressive gains for Convolutional Neural Networks (CNNs) (Shkolnik et al. 2020; Yu et al. 2021; Xu et al. 2023, 2024a; Kim et al. 2025), its benefits do not readily carry over to Transformers. Compared with CNNs, Transformers exhibit dense inter-token dependencies, attention-driven dynamic sparsity and expansive activation ranges, all of which magnify quantization noise and make per-layer bit-width sensitivity highly heterogeneous. Consequently, existing efforts to adapt multi-bit techniques to Transformers remain scarce, with most studies still anchored to fixed-bit-width schemes. Only very recently have a handful of works begun to explore multi-bit quantization for Transformer-based LLMs, including Any-Precision LLM (Park et al. 2024), which leverages truncated bit-widths and incremental upscaling, and MatQuant (Nair et al. 2025), which employs co-training and co-distillation regularization. However, these methods still underperform state-of-the-art single-bit-width quantization methods (Ashkboos et al. 2024; Liu et al. 2025; Sun et al. 2025) at low precision. Any-Precision LLM only implements weight quantization and does not consider activation quantization. Current approaches also fail to address how aggressive low-bit quantization negatively impacts medium and high-bit representations, resulting in overall performance being constrained by the lowest precision configuration.

To address the above challenges, we propose a novel post-training elastic quantization framework, QuEPT, as shown in Figure 1. QuEPT introduces Multi-Bit Token Merging (MB-ToMe), which addresses competitive conflicts between bit-widths by selectively preserving robust high-precision token features while fusing less stable representations across precision levels to maintain balanced performance during multi-bit optimization. Complementarily, we propose Multi-Bit Cascaded Low-Rank Adapters (MB-CLoRA), redesigning the parameter-update pathway into a cascaded low-rank flow that lets gradients harvested at high precision gently diffuse into their low-precision counterparts, turning conflict into synergy. Our method provides a unified multi-bit width quantization solution for various transformer models, including ViTs, LLMs, and Multi-modal LLM (MLLM) architectures such as LLaVA (Liu et al. 2023a; Li et al. 2025a). This approach only requires one-shot calibration and can be deployed in real time to any target bit-width by selecting the appropriate LoRA, completely eliminating re-optimization.

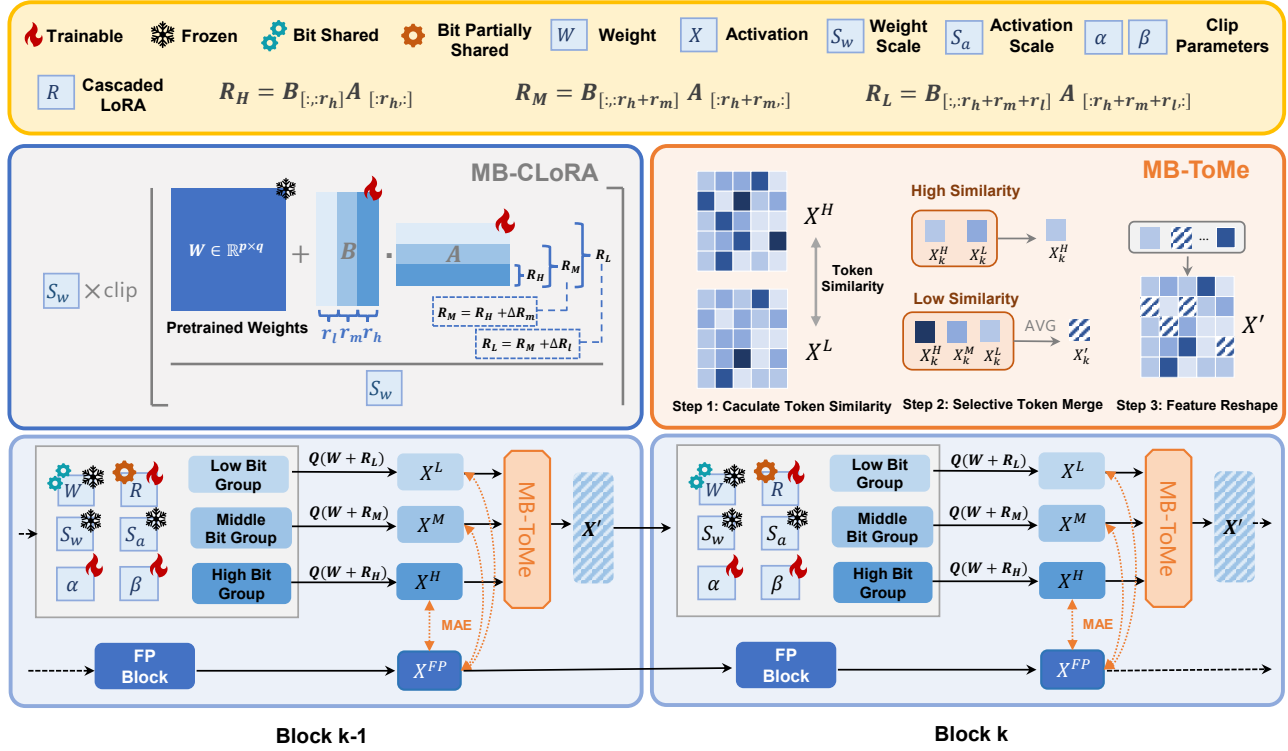


Figure 1: Overview of our proposed QuEPT. QuEPT calibrates the low-rank compensation matrix R and weight clipping parameters α and β in block-wise reconstruction, weights W and quantization scale S are frozen. The reconstruction process consists of two stages: (1) merging multiple bit-width features from different bit groups by Multi-Bit Token Merging (MB-ToMe); (2) optimizing multi-bit quantization loss using a Multi-Bit Cascaded Low-Rank Adapters (MB-CLoRA).

Moreover, it can be seamlessly converted to a training-free mixed-precision scheme by utilizing layer sensitivity data to assign different bit widths across layers.

Our contributions are summarized as follows:

- We present QuEPT, an efficient elastic quantization framework for transformers. The elastic quantized model supports real-time configuration of different bit widths without the need to repeatedly optimize parameters.
- We propose the Multi-Bit Token Merging (MB-ToMe) to reconstruct the mixed quantization error in each block by fusing tokens of different precision levels, thereby alleviating the contention conflict between different bit-widths and improving quantization robustness.
- We introduce the Multi-Bit Cascaded Low-Rank Adapters (MB-CLoRA) to enhance the correlation between different bit-width groups, thereby improving the overall quantization performance of QuEPT.
- Extensive experiments on ViT, LLM, and MLLM models verify that the performance of QuEPT is comparable to or even better than the current PTQ methods.

Related Works

Multi-Bit Quantization. Early multi-bit quantization are typically based on Quantization-Aware Training (QAT). For instance, RobustQuant (Shkolnik et al. 2020) discovered

that uniformly distributed weights are more robust to varying bit-widths and introduced kurtosis regularization to encourage a more uniform distribution during training; Any-Precision (Yu et al. 2021) flexibly converts a high-precision parent model into lower-bit-width models through truncation; EQ-Net (Xu et al. 2023) employs both skewness and kurtosis regularization to constrain the weight distribution, combining them with a group guidance distillation loss to improve robustness across different bit widths. However, the training overhead of these QAT-based methods is prohibitive, making them unsuitable for modern large-scale Transformer models. Recently, several PTQ-based multi-bit quantization methods have emerged. Among them, PTMQ (Xu et al. 2024a) absorbs multi-bit errors through rounding and validates its effectiveness on CNNs and ViTs by combining feature mixing with group-wise distillation. Despite being a PTQ method, its quantization overhead remains substantial for LLMs and ViTs. Another approach, Any-Precision LLM (Park et al. 2024), proposed a Non-uniform Quantization-based incremental upscaling to adapt the Any-Precision method for LLMs; however, its performance at low bit-widths is limited, and it fails to account for activation quantization. MatQuant (Nair et al. 2025) introduced a training approach using co-distillation regularization, but it is suboptimal as it does not consider the competitive relationship between different bit widths. Our method

improves robustness across different bit widths by integrating multi-bit token merging method and a cascaded LoRA strategy. This approach is designed to account for the importance of each bit width.

LoRA-based Quantization. The Low-Rank Adaptation (LoRA) (Hu et al. 2022) method reduces the overhead of model fine-tuning by training two low-rank matrices that are added to the frozen pre-trained weights. Recent approaches (Dettmers et al. 2023; Xu et al. 2024b; Guo et al. 2024; Li et al. 2024; Lee et al. 2024) combine LoRA with quantized models, which compensate for quantization errors and achieve the effect of fine-tuning on the downstream dataset. LR-QAT (Yelysei et al. 2024) adds LoRA to the pre-trained weights directly within the clipping operator, allowing the trained LoRA adapters to fuse back into the main weights after training. Furthermore, LoRA can also be used with PTQ. CBQ (Ding et al. 2025) replaces the weight-rounding matrix in AdaRound (Nagel et al. 2020) with LoRA, reducing the overhead of the AdaRound approach in LLM. SVDQuant (Li et al. 2025b) ingeniously combines SmoothQuant (Xiao et al. 2023) and LoRA, tasking LoRA with learning an outlier branch to make the remaining weights easier to quantize. Our method is a PTQ approach that places the LoRA module inside the clipping operator. This ensures that the LoRA branch incurs no additional overhead during inference. Furthermore, we employ a cascaded LoRA training method for LoRA that are robust across all bit widths.

Method

Post-Training Elastic Quantization Modeling

We formulate QuEPT as a block-wise optimization task for post-training quantization. Specifically, given the target set of quantization bit-widths $\mathcal{B} = \{b | b \in \mathbb{Z}^+, m \leq b \leq n\}$, $|\mathcal{B}| \geq 3$, the optimization objective of Elastic Quantization can be expressed by the following formula:

$$\min_{\mathbf{A}, \mathbf{B}, \alpha, \beta} \sum_{b \in \mathcal{B}} \mathbb{E}[(F^l(\mathbf{W}; \mathbf{X}) - F_b^l(\hat{\mathbf{W}}_b | \mathbf{A}_b, \mathbf{B}_b, \alpha_b, \beta_b; \hat{\mathbf{X}})]. \quad (1)$$

Here, \mathbf{A} and \mathbf{B} are the LoRA parameters, α and β are the weight clipping parameters, which include all target bit widths. F^l and F_b^l denote the output of the l -th block of the full-precision model and the output of the b -bit quantized model, respectively. The complete Low-Rank-based quantization includes weight and activation quantization, which can be represented as follows:

$$\hat{\mathbf{W}}_b = s_w^b \cdot (\text{clip}(\lfloor \frac{\mathbf{W} + \mathbf{B}\mathbf{A}}{s_w^b} \rfloor + z_w^b, -2^{b-1}, 2^{b-1} - 1) - z_w^b), \quad (2)$$

$$\hat{\mathbf{X}}_b = s_a^b \cdot \text{clip}(\lfloor \frac{\mathbf{X}}{s_a^b} \rfloor, -2^{b-1}, 2^{b-1} - 1), \quad (3)$$

where b represents a quantization bit width in the set \mathcal{B} , s_w , s_a represents the weight and activation quantization scales respectively, and z_w denotes the corresponding weight zero-point. $\text{clip}()$ refers to a truncated operation. The operator $\lfloor \cdot \rfloor$ rounds the continuous numbers to the nearest integers.

Unlike QLoRA, we place the low-rank parameters directly into the quantization formula. Compared to rounding-based methods, the low-rank parameters are much smaller than the full parameter size and do not need to be restricted to the 0-1 range. However, during the LoRA parameter update process, the quantized weight $\hat{\mathbf{W}}$ undergoes continuous changes. To accommodate these dynamic adjustments, we propose a method that optimizes the weight clipping threshold concurrently with the low-rank parameter. The formula of the b -bit weight quantization scale is expressed as follows:

$$s_w^b = \frac{\alpha_b \times \max(\mathbf{W} + \mathbf{B}_b \mathbf{A}_b) - \beta_b \times \min(\mathbf{W} + \mathbf{B}_b \mathbf{A}_b)}{2^{b-1}}, \quad (4)$$

$$z_w^b = -\lfloor \beta_b \times \frac{\min(\mathbf{W} + \mathbf{B}_b \mathbf{A}_b)}{s_w^b} \rfloor. \quad (5)$$

By concurrently training the clipping parameters, we enable the LoRA parameters to be optimized more effectively. The clipping mechanism handles the large-magnitude errors from weight outliers, creating a smoother and more manageable error signal for the low-rank adaptation to correct.

The Pipeline of QuEPT

As shown in Figure 1, QuEPT conducts a block-wise reconstruction that starts by initializing the weight-clipping parameters and cascaded LoRA adapters jointly with the frozen full-precision weights. Throughout the pipeline, we partition the target bit-width set \mathcal{B} into three tiers: low-bit (\mathcal{B}_L), mid-bit (\mathcal{B}_M) and high-bit (\mathcal{B}_H). At each step, one bit-width is randomly sampled from each tier, exposing the current block to a diverse combination of quantization granularities. The input to the block is produced by the preceding Multi-Bit Token Merging, which fuses the token-wise features from already reconstructed blocks while selectively anchoring the most robust high-bit tokens and softly merging the rest. The block’s forward pass is evaluated under the three sampled bit widths and the reconstruction loss, aggregated across all of them and measured with MAE, is back-propagated only into the corresponding slice of the cascaded LoRA adapters and the clipping thresholds. This simultaneous exposure to low-, mid- and high-bit quantization errors forces the adapters to co-optimize across the entire precision range, turning potential inter-bit conflicts into synergy and allowing a single calibration run to yield a model that can be deployed at any predefined bit-width simply by selecting the matching LoRA slice, without any further optimization.

Multi-Bit Token Merging (MB-ToMe)

MB-ToMe aims to design a token fusion strategy that maintains balanced performance across different bit-widths during multi-bit optimization, without favoring any specific precision level. We explored two key design dimensions: the merge operation (selection vs. fusion) and the merge policy (uniform vs. selective). Figure 2 demonstrates the three primary cases that were examined:

- **Case 1: Random Selection.** For each token, we randomly select one feature representation from the entire set of available bit-widths. This represents a pure

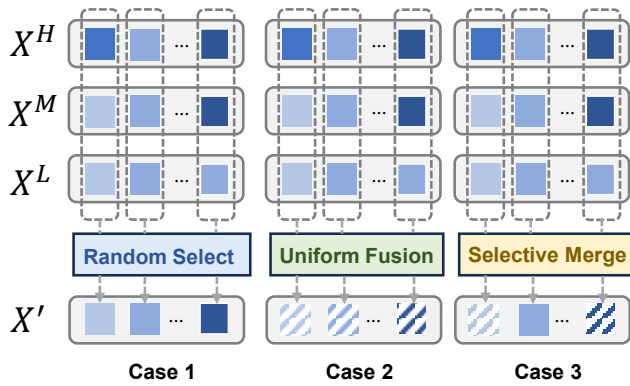


Figure 2: Three cases of multi-bit token merging strategy.

selection-based approach with a uniform policy, as every token and bit-width is treated with equal probability.

- **Case 2: Uniform Fusion.** We explicitly divide the various bit width sets into three groups: low, medium, and high. After randomly selecting one bit-width from each group, we perform a weighted fusion of the corresponding feature outputs using a 1:1:1 ratio. This strategy shifts from discrete token selection to feature-level fusion, yet maintains a uniform fusion policy across all groups.
- **Case 3: Selective Merge.** This strategy employs a selective mechanism to preserve high-precision information by identifying foundational tokens based on their quantization robustness, measured as the cosine similarity between their 8-bit and 4-bit representations. Tokens with high similarity are considered stable and are retained in their high-precision form, providing structural integrity to the feature map. This data-driven selection is more effective than random sampling. The remaining tokens are fused through a weighted average to maintain overall feature continuity. The hybrid approach thus preserves critical information via selective retention while ensuring smooth representation through fusion.

In multi-bit quantization, Case 1 with its random replacement strategy fails to produce well-integrated mixed features, leading to suboptimal representation quality. Case 2 mixes features from different bit-widths, retaining continuity but losing detailed information from high-bit-width tokens. Case 3 effectively combines both concepts, aiming to preserve the information from high-bit-width tokens as much as possible while also maintaining continuity between the different features. Table 5 confirms that Case 3 is the most effective. This method can be formalized as follows:

$$X'_k = \begin{cases} X_k^H, & \text{if } k \text{ in set } \Phi \\ \lambda_1 X_k^H + \lambda_2 X_k^M + \lambda_3 X_k^L, & \text{else} \end{cases}. \quad (6)$$

Here, Φ is a set of indices. We first pre-calculate the token-wise cosine similarity between the features of the high-bit group X_k^H , and those of the low-bit group, X_k^L . By sorting the similarity scores for all tokens, we then save the indices corresponding to the top $p\%$ of similarities as Φ .

$\{\lambda_1, \lambda_2, \lambda_3\}$ are hyperparameters that control the fusion ratio. Finally, all the X'_k are re-concatenated to form a feature map of the same size as the original one, which then serves as the quantization input for the next block. Figure 3 shows that merging dissimilar tokens leads to a more uniform numerical distribution that is more robust to varying bit-widths.

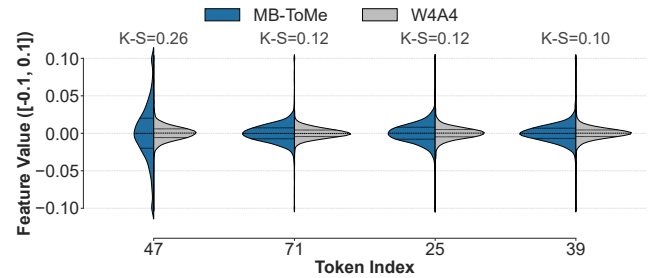


Figure 3: Feature distribution comparison for top 4 divergent tokens on the input of block 1 of LLaMA2-7b. Token sorted by Kolmogorov-Smirnov (K-S) statistic.

Multi-Bit Cascaded LoRA (MB-CLoRA)

Cascaded low-rank adaptation structure enables efficient and hierarchical parameter sharing across multiple bit-widths. Rather than maintaining separate LoRA parameters for each bit-width, we construct a unified parameter space where the LoRA matrices $A \in \mathbb{R}^{r \times q}$ and $B \in \mathbb{R}^{p \times r}$ serve all bit-widths through a hierarchical structure. The key insight is that lower bit-widths, which suffer greater quantization error, can leverage the adaptation capacity optimized for higher bit-widths while adding specialized compensation. Formally, for any given bit-width $b \in \mathcal{B}$, the corresponding low-rank compensation matrix $R^{(b)}$ is defined as:

$$R^{(b)} = B_{[:,r_b]} A_{[r_b,:]}, \quad (7)$$

where r_b represents the effective rank for bit-width b . Crucially, the rank allocation follows a cascaded pattern where r_b increases monotonically as bit-width group decreases:

$$r_b = \begin{cases} r_h & \text{if } b \in \mathcal{B}_H \\ r_h + r_m & \text{if } b \in \mathcal{B}_M \\ r_h + r_m + r_l & \text{if } b \in \mathcal{B}_L \end{cases}. \quad (8)$$

This structure establishes a natural inheritance mechanism: when processing low-bit-width $b \in \mathcal{B}_L$, the compensation matrix $R_L = B_{[:,r_h+r_m+r_l]} A_{[r_h+r_m+r_l,:]}$ will contain all parameters of medium and high bitwidth, because $R_H = B_{[:,r_h]} A_{[r_h,:]}$ and $R_M = B_{[:,r_h+r_m]} A_{[r_h+r_m,:]}$ will appear as leading submatrices.

During each training iteration, we simultaneously optimize across multiple bit-widths to ensure balanced adaptation. Specifically, we sample one bit-width from each tier: $b_H \sim \mathcal{B}_H$, $b_M \sim \mathcal{B}_M$, and $b_L \sim \mathcal{B}_L$. We then sequentially optimize the shared parameters A and B using each of these three bit-widths in turn. The unified optimization objective is to minimize the expected reconstruction error across all sampled bit-widths:

$$\min_{A, B, \alpha, \beta} \sum_{b \in \{b_L, b_M, b_H\}} \left\| \mathbf{W}X - (\widehat{\mathbf{W}} + R^{(b)})X' \right\|_1, \quad (9)$$

where $(\widehat{W + R^{(b)}})$ denotes the joint quantization of the weight and its compensated LoRA, while X' are the merged features from the previous block, obtained via MB-ToMe. It is worth noting that we use Mean Absolute Error (MAE) loss instead of MSE loss, as our experiments (Table 6) show that MAE generally yields superior results.

Experiments

Models and Evaluations. We evaluated our method on the Vision Transformer family such as ViT (Dosovitskiy et al. 2021), DeiT (Touvron et al. 2021), and Swin (Liu et al. 2021); the LLaMA (Dubey et al. 2024) family of large language models; and the LLaVA-OneVision (Li et al. 2025a) multimodal model. For the Vision Transformer models, we evaluated accuracy on ImageNet. For the LLaMA models, we assessed perplexity on the WikiText2 (Merity et al. 2017) and C4 (Raffel et al. 2020) datasets. Additionally, we evaluated accuracy on several zero-shot tasks, including the PIQA (Bisk et al. 2020), ARC-C (Clark et al. 2018), ARC-E (Clark et al. 2018), HellaSwag (Zellers et al. 2019), and WinoGrande (Sakaguchi et al. 2019) datasets. For the multi-modal model evaluation, we selected the representative LLaVA-OV model and assessed its accuracy on the MMLU (Yue et al. 2024), TextVQA (Singh et al. 2019), VizWiz (Gurari et al. 2018), OCRBench (Liu et al. 2023c), and SEED (Li et al. 2023a) datasets.

Calibration Datasets. For training the Vision Transformer models, we randomly selected 1024 unlabeled images from ImageNet as the calibration dataset; for the LLaMA family models, we used 128 samples from C4 (Raffel et al. 2020) datasets for calibration; and for the LLaVA-OV model, we followed the same setup with MBQ (Li et al. 2025c), also using 128 image-caption pairs sampled from the improved COCO Caption dataset proposed by ShareGPT4V (Chen et al. 2023) as the calibration dataset.

Baselines. We compare with PTQ methods on Vision Transformers, such as PTQ4ViT (Yuan et al. 2021), PDQuant (Liu et al. 2023b), RepQ-ViT (Li et al. 2023b), ERQ (Zhong et al. 2024), and PTMQ (Xu et al. 2024a). For LLMs, we considered the quantization methods AWQ (Lin et al. 2024b), GPTQ (Frantar et al. 2023), OmniQuant (Shao et al. 2024), AffineQuant (Ma et al. 2024), SmoothQuant (Xiao et al. 2023), QLLM (Liu et al. 2024), QuaRot (Ashkboos et al. 2024), and DuQuant (Lin et al. 2024a). For multi-modal models, our primary comparison was with the MBQ (Li et al. 2025c) method.

Main Results

Quantization of Vision Transformer Models. For weight-activation quantization on ViTs, we partition the bit-widths $\{4,5,6,7,8\}$ into a low-bit group $\{4\}$, a mid-bit group $\{5,6\}$, and a high-bit group $\{7,8\}$. This same weight-activation quantization configuration is applied to other models, including LLMs and MLLMs. We use the reparameterization technique (Li et al. 2023b) to initialize the activation quantizer. The quantization results for Vision Transformers under both single-bit and multi-bit scenarios

are compared and summarized in Table 1. For the ViT-S model, it achieves 6.2% higher accuracy than ERQ on the W4A4 setting and a 1.46% higher average accuracy across the five bit-widths. Unlike single-bit-width methods, which require five separate optimizations, our approach necessitates only one training run. As a result, the total training time is reduced to less than half of that required by ERQ. Our method is both more effective and significantly faster to train than the multi-bit quantization method PTMQ. For instance, on the ViT-B model, our method outperforms PTMQ by 6.1%, 5.6%, and 5.2% on the W6A6, W7A7, and W8A8 settings, respectively. Compared to PTMQ, our method has significantly lower training overhead because it discards the computationally expensive rounding values, while requiring only 1/26 of the time overhead on a single Nvidia RTX 3090 GPU.

Weight-Act Quantization of LLaMA Series Models.

Our method also demonstrates competitive performance in weight-activation quantization for the LLaMA family of models, as shown in Table 2. Since LLMs have outliers, we first apply a smoothing process with a Hadamard matrix (Ashkboos et al. 2024) before quantization. On LLaMA2-7B with W4A4 quantization, our approach not only yields lower PPL on WikiText2 and C4 but also outperforms QuaRot and SpinQuant by 7.07% and 0.28%, respectively, in average accuracy across five benchmarks (PIQA, ARC-E, ARC-C, HellaSwag, and WinoGrande).

Quantization of LLaVA-OneVision Model. Notably, for the weight-only quantization of LLaVA-OV and LLaMA, we explore a bit-width range of 2 to 8. The detailed settings are provided in the Appendix. In the multimodal model LLaVA-OneVision-7B, QuEPT achieves better results than the current SOTA method MBQ, as shown in Table 3. In a comparison of average accuracy across five datasets (MMMU, OCRBench, TextVQA, VizWiz, and SEED), QuEPT is 1.5% higher than MBQ under W3A16 quantization and 4.3% higher under W4A8 quantization.

Results of Mixed Precision of LLaMA2-7B Model. As our method is an elastic quantization approach, it can be seamlessly transformed into a mixed-precision quantization method by simply modifying the LoRA configuration for each layer. To achieve this, we first use KL divergence to evaluate the per-layer sensitivity of the model. We then employ a DP algorithm to determine the optimal per-layer bit-width allocation that minimizes the total sensitivity, with our bit-widths ranging from 2 to 8 bits. As shown in Figure 4, we compare the PPL of our method on the LLaMA2-7B model against several weight mixed-precision and non-uniform quantization methods, including SKIM (Bai, Liu, and Liu 2025), SqueezeLLM (Kim et al. 2024), and QuIP# (Tseng et al. 2024). The results indicate that with average bit-widths of 2.25/3.00/4.00, our method achieves a perplexity of only 8.97/5.93/5.54, respectively, on WikiText2.

Ablation Studies

The Effectiveness of Multi-Bit Cascaded LoRA. To validate the Effectiveness of MB-CLoRA, we designed two

Model	Method	Opti.	Uniform	Criterion	Bit-Width					Time (min)	FP32
					W4A4	W5A5	W6A6	W7A7	W8A8		
ViT-S	PTQ4ViT	✗	✗	Single-Bit	42.6	72.7	78.6	80.5	81.0	7×N	81.3
	PD-Quant	✓	✓		21.2	65.1	71.4	–	–	110×N	
	RepQ-ViT	✗	✗		65.1	78.4	80.4	80.8	81.2	4×N	
	ERQ	✓	✗		68.9	78.8	80.5	80.9	81.2	9×N	
	PTMQ	✓	✓	–	–	76.1	77.1	78.2	430		
	QuEPT	✓	✓	Multi-Bit	75.1	79.7	80.6	81.0	81.2	17	
ViT-B	PTQ4ViT	✗	✗	Single-Bit	30.7	72.3	81.7	83.7	84.3	13×N	84.5
	PD-Quant	✓	✓		34.9	58.4	63.1	–	–	230×N	
	RepQ-ViT	✗	✗		68.5	82.0	83.6	84.3	84.4	8×N	
	ERQ	✓	✗		76.6	82.8	83.9	84.3	84.4	17×N	
	PTMQ	✓	✓	–	–	77.7	78.6	79.1	950		
	QuEPT	✓	✓	Multi-Bit	80.7	83.3	83.8	84.2	84.3	36	
DeiT-S	PTQ4ViT	✗	✗	Single-Bit	34.1	70.3	76.3	79.0	79.5	7×N	79.8
	PD-Quant	✓	✓		64.9	74.9	77.6	–	–	110×N	
	RepQ-ViT	✗	✗		69.0	77.0	78.9	79.5	79.7	4×N	
	ERQ	✓	✗		72.6	77.6	79.0	79.5	79.6	9×N	
	PTMQ	✓	✓	–	–	78.7	79.3	79.5	430		
	QuEPT	✓	✓	Multi-Bit	75.3	78.1	79.3	79.7	79.7	17	
Swin-S	PTQ4ViT	✗	✗	Single-Bit	76.1	80.9	82.4	82.9	83.1	17×N	83.2
	PD-Quant	✓	✓		70.4	81.3	82.3	–	–	270×N	
	RepQ-ViT	✗	✗		79.5	82.1	82.8	83.0	83.0	8×N	
	ERQ	✓	✗		80.7	82.4	82.9	83.1	83.2	20×N	
	QuEPT	✓	✓	Multi-Bit	81.9	82.8	82.9	83.1	83.2	38	

Table 1: Comparison of the top-1 accuracy across various ViTs on ImageNet. “Opti.” indicates whether a method is optimization-based, and “Uniform” indicates whether it uses a uniform quantizer. The best results are highlighted in bold.

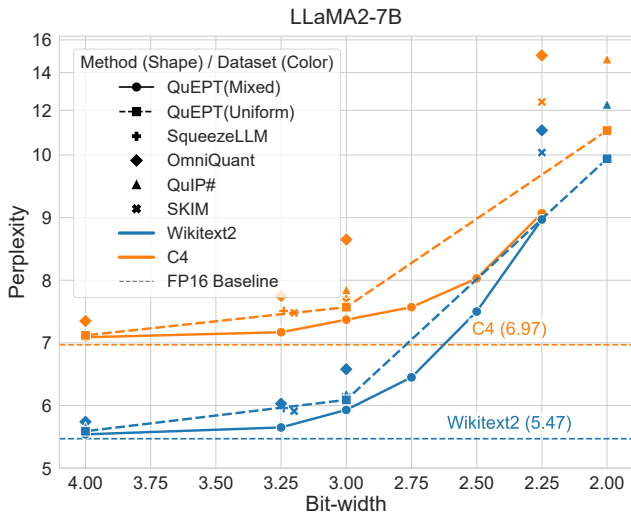


Figure 4: Results of Mixed Precision of LLaMA2-7B.

sets of experiments. We evaluated the accuracy of the ViT-S model on ImageNet and the average accuracy of the LLaMA2-7B model across five benchmarks. We compared three LoRA sharing strategies: (1) Case 1 (Fully Shared): All bit-widths share a single LoRA module. The LoRA rank was

set to 12 for ViT-S and 48 for LLaMA2-7B. (2) Case 2 (Independent): The LoRA is partitioned by bit-width into a low-bit group {4}, a mid-bit group {5, 6}, and a high-bit group {7, 8}. For the ViT-S model, the rank for each group was 4, and for the LLaMA2-7B model, it was 16. (3) Case 3 (MB-CLoRA): This approach maintains the same total LoRA rank as the other settings but employs the cascaded sharing strategy described in the methods section. The results in Table 4 demonstrate that our proposed MB-CLoRA performs better at lower bit-widths. For instance, on LLaMA2-7B with W4A4, MB-CLoRA’s accuracy is 2.4% and 0.7% higher than that of the Independent and Fully Shared strategies, respectively. At higher bit-widths, the performance of MB-CLoRA is comparable to the Independent strategy.

The Effectiveness of Multi-Bit Token Merging. We compared the performance of several multi-bit-width feature fusion strategies. The results in Table 5 confirm that Case 3 achieves the best performance across all bit-widths. Compared to Case 1, Case 3 prevents random feature selection from affecting the results. In contrast to Case 2, Case 3 preserves high-bit features as much as possible, leading to more accurate feature information. Therefore, Case 3 yields the best results. For instance, on the LLaMA2-7B model, MB-ToMe shows performance gains of 5.9%, 0.8%, and 0.2% over the Case 1 method at 4, 6, and 8 bits, respectively.

Model	Bits	Method	Wiki.(↓)	C4.(↓)	0-shot ⁵ Avg.(↑)
L2-7B	FP16	-	5.47	6.97	65.71
		OmniQ	14.26	18.02	49.13
	W4A4	QLLM	11.75	13.26	51.60
		QuaRot	9.66	11.98	54.55
		DuQuant	6.28	7.90	59.11
		QuEPT	6.33	7.86	61.62
	W5A5	QuEPT	5.66	7.17	64.42
	W6A6	SmoothQ	6.20	7.76	61.40
		OmniQ	5.87	7.48	58.56
		QuEPT	5.53	7.03	65.48
W8A8	QuEPT	5.48	6.98	66.23	
L2-13B	FP16	-	4.88	6.46	69.35
		OmniQ	12.30	14.55	53.13
	W4A4	QLLM	9.09	11.13	54.31
		DuQuant	5.42	7.05	62.84
		QuEPT	5.53	7.16	66.05
		W5A5	QuEPT	5.10	6.69
	W6A6	SmoothQ	5.18	6.76	64.30
		OmniQ	5.14	6.74	64.71
		QuEPT	5.00	6.59	68.72
	W8A8	QuEPT	4.94	6.53	69.09
L3-8B	FP16	-	6.14	8.88	72.65
		OmniQ	3.64e3	2.80e3	35.70
	W4A4	AffineQ	2.12e4	3.46e4	35.44
		QuaRot	10.41	14.33	57.16
		DuQuant	8.56	11.98	65.05
		QuEPT	8.25	11.67	67.04
	W5A5	QuEPT	6.76	9.72	70.99
	W6A6	SmoothQ	7.07	9.57	70.50
		OmniQ	7.24	9.82	69.51
		AffineQ	7.35	9.99	69.22
W8A8	QuEPT	6.34	9.12	71.56	
W8A8	QuEPT	6.20	8.96	72.87	

Table 2: Weight-Activation quantization results of LLaMA models. We report the perplexity on WikiText2 and C4 (↓ indicates lower is better), and the average accuracy (Avg.) denotes the average accuracy across 5 datasets.

Bits	Method	mmmu	ocrbench	textvqa	vizviz	seed
FP16	-	48.0	62.2	76.1	60.4	72.6
W2	QuEPT	32.2	47.8	64.0	55.2	67.0
W3	AWQ	36.6	59.3	70.0	56.4	51.5
	MBQ	42.0	61.1	73.3	60.7	66.4
	QuEPT	44.6	60.6	74.1	60.3	71.6
W4A4	QuEPT	33.9	49.1	64.4	59.8	66.5
W4A8	SmoothQ	30.9	32.0	56.9	56.7	41.6
	MBQ	42.6	52.3	68.3	58.9	64.4
	QuEPT	43.4	61.2	71.5	61.3	70.7

Table 3: Weight-only and weight-activation quantization results of LLaVA-OneVision-7B model.

The Effectiveness of Different Modules. As presented in Table 6, we evaluated combinations of different modules.

Model	Method	W4A4	W5A5	W6A6	W7A7	W8A8
ViT-S	Case 1	73.7	78.6	80.6	79.9	81.1
	Case 2	73.5	78.9	80.7	81.1	81.2
	Case 3	74.7	79.4	80.6	81.0	81.1
L2-7B	Case 1	60.9	64.2	64.9	65.2	65.5
	Case 2	59.2	64.4	65.3	65.7	65.7
	Case 3	61.6	64.5	65.5	65.6	65.6

Table 4: Ablation studies for Multi-Bit Cascaded LoRA.

Model	Method	W4A4	W5A5	W6A6	W7A7	W8A8
ViT-S	Case 1	74.5	79.4	80.3	80.5	80.9
	Case 2	74.9	79.3	80.2	80.6	80.8
	Case 3	75.1	79.7	80.6	81.0	81.2
L2-7B	Case 1	55.7	63.8	64.7	64.9	65.4
	Case 2	60.2	64.4	65.2	65.5	65.5
	Case 3	61.6	64.5	65.5	65.6	65.6

Table 5: Ablation studies for Multi-Bit Token Merging.

Our baseline uses 3 independent LoRAs ($r=4$) with MSE loss to compensate for low, medium, and high bit-width groups. With MB-CLoRA, W4A4 accuracy increased, while W6A6 and W8A8 accuracy dropped. This is because MB-CLoRA boosts the effective rank for low bit-widths via rank sharing, but this optimization harms the LoRA compensation for medium and high bit-widths. The combination of MB-CLoRA and MB-ToMe led to an improvement in low-bit-width performance.

Clip	MB-CLoRA	MB-ToMe	MAE	W4A4	W6A6	W8A8
✗	✗	✗	✗	69.2	80.1	81.1
✓	✗	✗	✗	70.1	80.2	81.1
✓	✓	✗	✗	72.2	79.2	80.7
✓	✗	✓	✗	73.0	80.1	80.9
✓	✓	✓	✗	73.6	80.1	81.0
✓	✓	✓	✓	74.7	80.7	81.1

Table 6: QuEPT modules ablation study on ViT-S model.

Conclusion

This paper presents QuEPT, the first multi-bit-width PTQ method validated on ViTs, LLMs, and MLLMs. First, we introduce MB-CLoRA, which uses a cascaded LoRA to enable efficient parameter collaboration across bit-widths. Second, we propose MB-ToMe, a selective feature retention and fusion mechanism that safeguards performance from the negative impact of low-bit quantization. Finally, we have validated our method on various models, establishing it as a state-of-the-art multi-bit-width quantization approach. Nevertheless, our approach is not without its limitations. We have not explicitly handled outliers in LLMs, meaning that integrating our implementation with outlier mitigation techniques (e.g., SpinQuant (Liu et al. 2025)) would likely yield better results. Additionally, our method’s performance remains constrained at extremely low bit-widths, which requires further exploration in the future.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62576001, No. 62206003, No. U21A20512) and the National Key Research and Development Project (NO. 2018AAA0100105).

References

- Ashkboos, S.; Mohtashami, A.; Croci, M. L.; Li, B.; Jaggi, M.; Alistarh, D.; Hoefler, T.; and Hensman, J. 2024. QuaRot: Outlier-Free 4-Bit Inference in Rotated LLMs. In *NeurIPS*.
- Bai, R.; Liu, B.; and Liu, Q. 2025. SKIM: Any-bit Quantization Pushing The Limits of Post-Training Quantization. In *ICML*.
- Bisk, Y.; Zellers, R.; Le bras, R.; Gao, J.; and Choi, Y. 2020. PIQA: Reasoning about Physical Commonsense in Natural Language. In *AAAI*.
- Chen, L.; Li, J.; wen Dong, X.; Zhang, P.; He, C.; Wang, J.; Zhao, F.; and Lin, D. 2023. ShareGPT4V: Improving Large Multi-Modal Models with Better Captions. In *ECCV*.
- Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *ArXiv*.
- Dettmers, T.; Pagnoni, A.; Holtzman, A.; and Zettlemoyer, L. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. In *NeurIPS*.
- Ding, X.; Liu, X.; Tu, Z.; Zhang, Y.; Li, W.; Hu, J.; Chen, H.; Tang, Y.; Xiong, Z.; Yin, B.; and Wang, Y. 2025. CBQ: Cross-Block Quantization for Large Language Models. In *ICLR*.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houslyby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.
- Dubey, A.; Jauhri, A.; Pandey, A.; and Abhishek Kadian, e. a. 2024. The Llama 3 Herd of Models. *ArXiv*.
- Frantar, E.; Ashkboos, S.; Hoefler, T.; and Alistarh, D. 2023. OPTQ: Accurate Quantization for Generative Pre-trained Transformers. In *ICLR*.
- Guo, H.; Greengard, P.; Xing, E.; and Kim, Y. 2024. LQ-LoRA: Low-rank plus Quantized Matrix Decomposition for Efficient Language Model Finetuning. In *ICLR*.
- Gurari, D.; Li, Q.; Stangl, A.; Guo, A.; Lin, C.; Grauman, K.; Luo, J.; and Bigham, J. P. 2018. VizWiz Grand Challenge: Answering Visual Questions from Blind People. In *CVPR*.
- Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*.
- Kim, J.; Yoon, S.; Lee, T.; Lee, J. C.; Jeon, K. E.; and Ko, J. H. 2025. TruncQuant: Truncation-Ready Quantization for DNNs with Flexible Weight Bit Precision. *ArXiv*.
- Kim, S.; Hooper, C.; Gholami, A.; Dong, Z.; Li, X.; Shen, S.; Mahoney, M. W.; and Keutzer, K. 2024. SqueezeLLM: Dense-and-Sparse Quantization. In *ICML*.
- Lee, C.; Jin, J.; Kim, T.; Kim, H.; and Park, E. 2024. OWQ: Lessons learned from activation outliers for weight quantization in large language models. In *AAAI*.
- Li, B.; Wang, R.; Wang, G.; Ge, Y.; Ge, Y.; and Shan, Y. 2023a. SEED-Bench: Benchmarking Multimodal LLMs with Generative Comprehension. *ArXiv*, abs/2307.16125.
- Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Zhang, P.; Li, Y.; Liu, Z.; and Li, C. 2025a. LLaVA-OneVision: Easy Visual Task Transfer. *Transactions on Machine Learning Research*.
- Li, M.; Lin, Y.; Zhang, Z.; Cai, T.; Li, X.; Guo, J.; Xie, E.; Meng, C.; Zhu, J.-Y.; and Han, S. 2025b. SVDQuant: Absorbing Outliers by Low-Rank Component for 4-Bit Diffusion Models. In *ICLR*.
- Li, S.; Hu, Y.; Ning, X.; Liu, X.; Hong, K.; Jia, X.; Li, X.; Yan, Y.; Ran, P.; Dai, G.; Yan, S.; Yang, H.; and Wang, Y. 2025c. MBQ: Modality-Balanced Quantization for Large Vision-Language Models. In *CVPR*.
- Li, Y.; Yu, Y.; Liang, C.; Karampatziakis, N.; He, P.; Chen, W.; and Zhao, T. 2024. LoftQ: LoRA-Fine-Tuning-aware Quantization for Large Language Models. In *ICLR*.
- Li, Z.; Xiao, J.; Yang, L.; and Gu, Q. 2023b. RepQ-ViT: Scale Reparameterization for Post-Training Quantization of Vision Transformers. In *ICCV*.
- Lin, H.; Xu, H.; Wu, Y.; Cui, J.; Zhang, Y.; Mou, L.; Song, L.; Sun, Z.; and Wei, Y. 2024a. DuQuant: Distributing Outliers via Dual Transformation Makes Stronger Quantized LLMs. In *NeurIPS*.
- Lin, J.; Tang, J.; Tang, H.; Yang, S.; Dang, X.; and Han, S. 2024b. AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration. In *MLSys*.
- Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023a. Visual Instruction Tuning. In *NeurIPS*.
- Liu, J.; Gong, R.; Wei, X.; Dong, Z.; Cai, J.; and Zhuang, B. 2024. QLLM: Accurate and Efficient Low-Bitwidth Quantization for Large Language Models. In *ICLR*.
- Liu, J.; Niu, L.; Yuan, Z.; Yang, D.; Wang, X.; and Liu, W. 2023b. PD-Quant: Post-Training Quantization Based on Prediction Difference Metric. In *CVPR*.
- Liu, Y.; Li, Z.; Huang, M.; Yang, B.; Yu, W.; Li, C.; Yin, X.; lin Liu, C.; Jin, L.; and Bai, X. 2023c. OCRBench: on the hidden mystery of OCR in large multimodal models. *Sci. China Inf. Sci.*
- Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *ICCV*.
- Liu, Z.; Zhao, C.; Fedorov, I.; Soran, B.; Choudhary, D.; Krishnamoorthi, R.; Chandra, V.; Tian, Y.; and Blankevoort, T. 2025. SpinQuant: LLM Quantization with Learned Rotations. In *ICLR*.
- Ma, Y.; Li, H.; Zheng, X.; Ling, F.; Xiao, X.; Wang, R.; Wen, S.; Chao, F.; and Ji, R. 2024. AffineQuant: Affine Transformation Quantization for Large Language Models. In *ICLR*.
- Merity, S.; Xiong, C.; Bradbury, J.; and Socher, R. 2017. Pointer Sentinel Mixture Models. In *ICLR*.

- Nagel, M.; Amjad, R. A.; Van Baalen, M.; Louizos, C.; and Blankevoort, T. 2020. Up or Down? Adaptive Rounding for Post-Training Quantization. In *ICML*.
- Nair, P. A.; Datta, P.; Dean, J.; Jain, P.; and Kusupati, A. 2025. Matryoshka Quantization. In *ICML*.
- Park, Y.; Hyun, J.; Cho, S.; Sim, B.; and Lee, J. W. 2024. Any-Precision LLM: Low-Cost Deployment of Multiple, Different-Sized LLMs. In *ICML*.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *JMLR*.
- Sakaguchi, K.; Bras, R. L.; Bhagavatula, C.; and Choi, Y. 2019. An Adversarial Winograd Schema Challenge at Scale. *Communications of the ACM*.
- Shao, W.; Chen, M.; Zhang, Z.; Xu, P.; Zhao, L.; Li, Z.; Zhang, K.; Gao, P.; Qiao, Y. J.; and Luo, P. 2024. OmniQuant: Omnidirectionally Calibrated Quantization for Large Language Models. In *ICLR*.
- 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 *NeurIPS*.
- Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards VQA Models That Can Read. In *CVPR*.
- Sun, Y.; Liu, R.; Bai, H.; Bao, H.; Zhao, K.; Li, Y.; Hu, J.; Yu, X.; Hou, L.; Yuan, C.; et al. 2025. FlatQuant: Flatness Matters for LLM Quantization. In *ICML*.
- Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jegou, H. 2021. Training data-efficient image transformers & distillation through attention. In *ICML*.
- Tseng, A.; Chee, J.; Sun, Q.; Kuleshov, V.; and Sa, C. D. 2024. QuIP $\$$: Even Better LLM Quantization with Hadamard Incoherence and Lattice Codebooks. In *ICML*.
- Xiao, G.; Lin, J.; Seznec, M.; Wu, H.; Demouth, J.; and Han, S. 2023. SmoothQuant: Accurate and Efficient Post-Training Quantization for Large Language Models. In *ICML*.
- Xu, K.; Han, L.; Tian, Y.; Yang, S.; and Zhang, X. 2023. EQ-Net: Elastic Quantization Neural Networks. In *ICCV*.
- Xu, K.; Li, Z.; Wang, S.; and Zhang, X. 2024a. PTMQ: Post-training Multi-Bit Quantization of Neural Networks. *AAAI*.
- Xu, Y.; Xie, L.; Gu, X.; Chen, X.; Chang, H.; Zhang, H.; Chen, Z.; ZHANG, X.; and Tian, Q. 2024b. QA-LoRA: Quantization-Aware Low-Rank Adaptation of Large Language Models. In *ICLR*.
- Yelysei, B.; Chiaro, D.; Riccardo; and Markus, N. 2024. Low-Rank Quantization-Aware Training for LLMs. *arXiv preprint arXiv:2406.06385*.
- Yu, H.; Li, H.; Shi, H.; Huang, T. S.; and Hua, G. 2021. Any-Precision Deep Neural Networks. In *AAAI*.
- Yuan, Z.; Xue, C.; Chen, Y.; Wu, Q.; and Sun, G. 2021. PTQ4ViT: Post-training Quantization for Vision Transformers with Twin Uniform Quantization. In *ECCV*.
- Yue, X.; Ni, Y.; Zhang, K.; and et al. 2024. MMMU: A Massive Multi-Discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *CVPR*.
- Zellers, R.; Holtzman, A.; Bisk, Y.; Farhadi, A.; and Choi, Y. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? In *ACL*.
- Zhong, Y.; Hu, J.; Huang, Y.; Zhang, Y.; and Ji, R. 2024. ERQ: Error Reduction for Post-Training Quantization of Vision Transformers. In *ICML*.