UQ-ViT: Harmonizing Extreme Activations with Hardware-Friendly Uniform Quantization in Vision Transformers

Tao Jiang[1*], **Yucheng Jiang**[1*], **Xiwen Yao**[1†], **Gong Cheng**[1], **Junwei Han**[1, 2]

[1]School of Automation, Northwestern Polytechnical University, Xi'an, China
[2]Chongqing University of Posts and Telecommunications, Chongqing, China
{jtao9912, yuchengjiang}@mail.nwpu.edu.cn, {yaoxiwen, gcheng, jhan}@nwpu.edu.cn

Abstract

Post-Training Quantization enables efficient Vision Transformer (ViTs) deployment with a small calibration data, and its prevalent use of uniform quantization harnesses AI accelerator matrix cores for high-speed inference. However, the application of uniform quantization is fundamentally challenged by the extreme non-uniformity of activation distributions.Specifically, the power-law nature of post-Softmax attention scores and the significant inter-channel variance in post-GELU activations create a dilemma for conventional quantization, as it struggles to preserve critical high-magnitude values without sacrificing overall precision. To resolve this core conflict, we introduce **UQ-ViT** (Uniform Quantization for Vision Transformers), a novel uniform quantization framework designed to reconcile high precision with hardware efficiency. Central to UQ-ViT are two operators: Dynamic Elimination of Maximum (DeMax) and Normalization Quantization (NormQuant). DeMax is a quantization operator for post-Softmax attention scores that utilizes uniform quantization. It dynamically eliminates and preserves dominant values, effectively mitigating quantization loss from the extreme values in the power-law distribution. NormQuant utilizes a per-channel quantization strategy during quantization and reverts to a per-tensor format for dequantization, achieving both high accuracy and computational efficiency. Crucially, it is applicable to any linear layer, enabling effective quantization of post-GELU activations in ViTs. Through extensive experiments on various ViTs and vision tasks, including image classification, object detection, and instance segmentation, we demonstrate that our proposed approach outperforms existing methods, achieving superior accuracy while ensuring hardware friendliness.

Code — https://github.com/jiujiuwei/UQ-ViT.git

Introduction

Recent advancements in Vision Transformers (ViTs) have shown remarkable performance across various vision tasks, such as image classification (Dosovitskiy et al. 2020; Liu et al. 2021), object detection (Carion et al. 2020; Zhang et al. 2022a), and semantic segmentation (Strudel et al. 2021; Kirillov et al. 2023). Unlike convolutional neural networks (CNNs) (Krizhevsky, Sutskever, and Hinton 2012), ViTs utilize self-attention mechanisms to capture global contextual information, thereby enabling a more profound understanding of images through enhanced global context (Vaswani et al. 2017). However, their high computational and memory requirements remain a significant bottleneck for deployment on resource-constrained devices (Han, Mao, and Dally 2015), posing an urgent challenge that remains the primary focus of research.

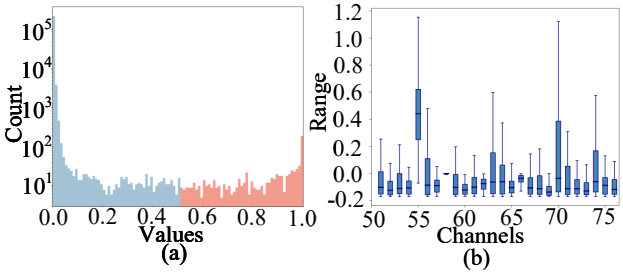

Figure 1: Two special activation distributions: (a) Post-Softmax activations with extreme values harm uniform quantization; (b) Imbalanced Post-GELU activations incur high quantization loss in pre-tensor quantization, and the lack of preceding LayerNorm prevents reparameterization.

Model quantization, a proven technique for model compression and acceleration, maps high-precision weights to low-bit representations (Jacob et al. 2018; Krishnamoorthi 2018), significantly reducing storage, bandwidth, and accelerating inference. Quantization methods are generally divided into two categories: Quantization-Aware Training (QAT) and Post-Training Quantization (PTQ). In QAT, the quantization process is simulated during training, typically yielding higher accuracy, though it incurs significant time and computational costs (Choi et al. 2018; Esser et al. 2019). In contrast, PTQ does not require extensive retraining and can perform quantization with only a small amount of calibration data, offering superior deployment efficiency (Wei et al. 2022; Liu et al. 2023a,b).

Although PTQ methods have made continuous advancements, low-bit uniform quantization of activations in ViTs remains challenging due to the extreme distribution charac-

*These authors contributed equally.
†Corresponding author.

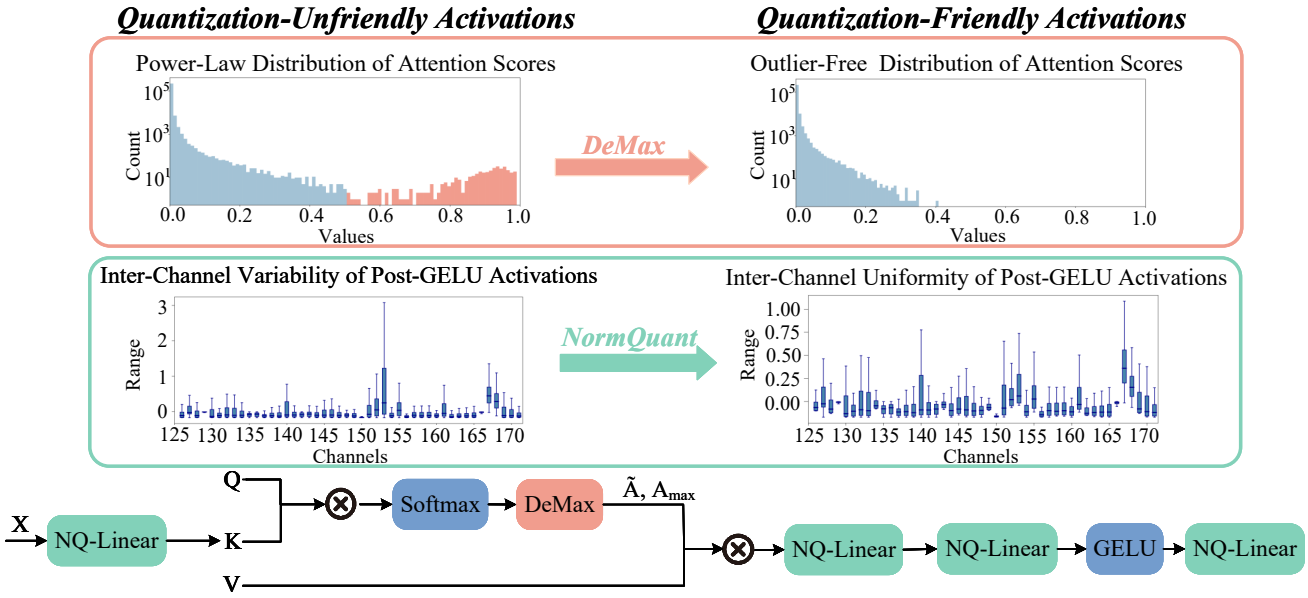

Figure 2: Illustration of UQ-ViT. LayerNorm and shortcuts are omitted for clarity. NQ denotes Normalization Quantization. DeMax processes post-Softmax activations by dynamically eliminating and preserving the dominant values. NormQuant handles the uneven activations pre-Linear by integrating channel-wise quantization with tensor-wise dequantization.

teristics of activations after specific functions, as illustrated in Figure 1. On one hand, post-Softmax attention scores follow a power-law distribution, with extreme values carrying crucial information (Ding et al. 2022; Lv et al. 2024). Clipping these dominant values can lead to severe performance degradation. While non-uniform quantization strategies can mitigate this issue to some extent, they are often incompatible with the matrix computation cores of AI accelerators, limiting their practical value on hardware. On the other hand, post-GELU activations exhibit substantial inter-channel variation. Traditional per-tensor uniform quantization relies on a unified scaling factor, failing to capture this inter-channel variation. Although reparameterization techniques have been explored, they often rely on LayerNorm, limiting their applicability to post-GELU activations.

Uniform quantization, as the most widely adopted quantization scheme, is highly compatible with matrix multiplication accelerators in AI chips, enabling significant inference speedup. However, the aforementioned extreme activation distributions arising after specific functions (e.g., Softmax and GELU) severely degrade the performance of conventional uniform quantization. To preserve the inference efficiency of uniform quantization while enhancing their accuracy, this paper proposes a uniform quantization framework, named UQ-ViT. It incorporates two novel uniform quantization operators—Dynamic Elimination of Maximum (DeMax) and Normalization Quantization (NormQuant), which jointly address the substantial quantization errors in extreme activation activations after specific functions.

Specifically, the first operator, DeMax, is designed to tackle the quantization challenges posed by extreme values in post-Softmax attention scores. Attention scores from the Softmax function typically follow a power-law distribution, where one value in each row (the dominant score) is significantly higher than others (e.g., [0.7, 0.1, 0.1, 0.05, 0.05]). DeMax addresses this by dynamically isolating and preserving these dominant values from the standard quantization range, effectively reducing quantization loss caused by extreme values. This approach ensures the preservation of critical semantic content while enhancing both accuracy and hardware friendliness. The second operator, NormQuant, addresses the issue of inter-channel distribution discrepancies in post-GELU activations. It employs a per-channel scaling strategy during quantization to better adapt to each channel's distribution. During dequantization, the representation is unified into a per-tensor format to ensure compatibility with efficient integer matrix operations. Moreover, NormQuant does not rely on LayerNorm, making it broadly applicable to any linear layer. By incorporating the DeMax and NormQuant operators, UQ-ViT addresses the issues caused by extreme activation values after specific functions in conventional uniform quantization, ensuring improved accuracy while preserving the efficiency benefits of uniform quantization.

We present the main contributions of this work as follows:

- We propose UQ-ViT, a hardware-friendly uniform quantization framework for ViTs that effectively addresses extreme activation distributions after specific functions. UQ-ViT introduces two novel operators: DeMax and NormQuant. DeMax is the first uniform quantization method specifically designed for post-Softmax attention scores, which dynamically isolates and preserves dominant values to mitigate quantization loss caused by extreme values without losing key information.
- We propose the NormQuant operator, which combines

per-channel quantization with per-tensor dequantization to effectively address post-GELU inter-channel distribution discrepancies. It can be flexibly applied to any linear layer without relying on normalization layers, significantly enhancing the method's versatility.

- Comprehensive low-bit quantization experiments on several mainstream ViT architectures validate the method's effectiveness. The proposed approach achieves state-of-the-art performance on tasks such as classification, segmentation, and detection, significantly outperforming existing baselines while maintaining hardware friendliness.

Related Work

Vision Transformers

Vision Transformer (ViT) brought the Transformer architecture (Vaswani et al. 2017) from Natural Language Processing into computer vision by representing images as patch sequences. Following this, DeiT (Touvron et al. 2021) mitigated ViT dependence on large-scale datasets by leveraging knowledge distillation. Swin Transformer further enhanced model performance with its hierarchical architecture and a sliding window self-attention mechanism. As architectural innovations continued, ViTs found widespread applications across a variety of vision tasks, including object detection, image segmentation, and video classification (Bertasius, Wang, and Torresani 2021; Liu et al. 2022).

Although ViTs excel in various vision tasks, their high computational and memory costs limit deployment on resource-constrained devices. EfficientViT (Cai et al. 2023) mitigates this by replacing softmax attention with linear attention and adding depthwise convolutions for local feature extraction. MobileViT (Mehta and Rastegari 2021) fuses CNNs and ViTs to build a lightweight, low-latency architecture for mobile scenarios. Pruning-based methods like UP-ViTs and SPViT (Yu and Wu 2023; Kong et al. 2022), as well as techniques such as knowledge distillation and parameter sharing in TinyViT and MiniViT (Wu et al. 2022; Zhang et al. 2022b), further improve ViT efficiency.

Model Quantization

Quantization is a fundamental technique for compressing models by mapping floating-point weights and activations to integers, aiming to accelerate inference with little accuracy drop. Mainstream methods fall into two categories: QAT and PTQ. QAT achieves higher accuracy but requires substantial labeled data and training costs (Choi et al. 2018; Esser et al. 2019), while PTQ uses a small unlabeled calibration set to determine quantization parameters via activation calibration and weight reconstruction, enabling low-cost deployment (Wei et al. 2022; Liu et al. 2023a,b).

Current PTQ calibration methods primarily focus on addressing the imbalance and extreme values in activation distributions, particularly at critical locations such as post-LayerNorm and post-Softmax. For instance, RepQViT (Li et al. 2023) employs a reparameterization strategy to mitigate activation distribution issues in post-LayerNorm and uses a logarithmic non-uniform quantization approach to address the power-law distribution problem in post-Softmax. AdaLog (Wu et al. 2024) introduces a non-uniform quantizer with adaptive base selection to improve the representation of imbalanced activations. IQ (Moon et al. 2024) applies a grouped quantization strategy to further enhance calibration performance. QwT (Fu et al. 2025) proposes a lightweight auxiliary structure that improves quantization efficiency and generalization. While these methods have achieved notable success in specific scenarios, most still rely on hardware-unfriendly operations, such as non-uniform quantization, which limits their adaptability to matrix computation cores commonly supported by AI accelerators.

On the other hand, PTQ reconstruction methods, mostly built upon the AdaRound framework (Nagel et al. 2020), improve accuracy by reducing quantization loss via layer-wise training. FIMA-Q (Wu et al. 2025a) derives a relationship between KL divergence and the Fisher information matrix, enhancing performance. APHQ-ViT (Wu et al. 2025b) enhances Hessian-based loss and applies MR to address GELU-induced quantization issues. Although such methods generally outperform calibration-based approaches, they rely on numerous hyperparameters and incur high computational costs, limiting deployment efficiency.

Existing PTQ methods typically employ non-uniform quantization to mitigate the imbalance and extreme values in activation distributions after specific functions. Although non-uniform quantization can improve accuracy, it is hardware-unfriendly and incompatible with the matrix computation cores of common AI accelerators, limiting its practical application.

Method

This section first outlines the background and hardware limitations of $\log\sqrt{2}$ non-uniform quantization. To address the power-law distribution of post-Softmax activations, we propose DeMax—a hardware-friendly uniform quantization operator that reduces quantization loss while preserving computational efficiency. To mitigate information loss from skewed activations, we further introduce NormQuant, which flattens the distribution of pre-linear activations to mitigate quantization loss. Compared to LayerNorm-based reparameterization, NormQuant offers greater flexibility.

Preliminaries

Uniform quantization, as a fundamental and widely adopted quantization method, can effectively align with the matrix computation cores commonly supported in AI accelerators. Its mathematical formulation is given as follows:

$$\text{Quant:} \quad \mathbf{x}^{(\mathbb{Z})} = \text{clip}\left(\left\lfloor \frac{\mathbf{x}}{s} \right\rceil) + z, 0, 2^b - 1\right) \tag{1}$$

$$\text{DeQuant:} \quad \hat{\mathbf{x}} = s\left(\mathbf{x}^{(\mathbb{Z})} - z\right) \tag{2}$$

where $\lfloor \cdot \rceil$ denotes the rounding function, $\mathbf{x}$ is the original floating-point values, $\mathbf{x}^{(\mathbb{Z})}$ is the quantized integer representation, s is the scaling factor, z is the zero-point offset, b is the bit-width of the quantized representation (e.g., 4 bits), and $\hat{\mathbf{x}}$ is the dequantized value.

To alleviate the power-law distribution issue in the post-Softmax activation values, non-uniform log2 quantization is

introduced. The $\log\sqrt{2}$ quantization adjusts the base of the logarithm, enabling more refined bit allocation. The corresponding non-uniform quantization formula is as follows:

$$\text{Quant:} \quad \mathbf{x}^{(\mathbb{Z})} = \text{clamp}\left(\left\lfloor -2\log_2 \frac{\mathbf{x}}{s} \right\rceil, 0, 2^b - 1\right) \tag{3}$$

$$\text{DeQuant:} \quad \hat{\mathbf{x}} = \tilde{\mathbf{S}} \odot 2^{\left\lfloor -\frac{\mathbf{x}^{(\mathbb{Z})}}{2} \right\rfloor} \tag{4}$$

where $\tilde{\mathbf{S}} = s\left(1[x^{(\mathbb{Z})}] \cdot (\sqrt{2}-1) + 1\right)$, $1[\cdot]$ is a parity indicator function, and $\odot$ is the element-wise multiplication operator.

Matrix multiplication is performed after dequantization, as shown in the following equation:

$$\mathbf{Y} = \tilde{\mathbf{S}} \odot 2^{\left\lfloor -\frac{\mathbf{x}^{(\mathbb{Z})}}{2} \right\rfloor} \cdot s_{\mathbf{w}} \mathbf{W}^{(\mathbb{Z})} \tag{5}$$

where $\mathbf{Y}$ is the output matrix, $\tilde{\mathbf{S}}$ is a scaling matrix applied to the input, $\mathbf{W}^{(\mathbb{Z})}$ represents the weights in integer format, and $s_{\mathbf{w}}$ is the scaling factor.

Due to the non-commutative nature of the Hadamard product and matrix multiplication, shift operations cannot be pre-executed during inference, resulting in increased implementation complexity and reduced hardware efficiency. Furthermore, the non-uniform quantization strategy adopted by AdaLog suffers from the same limitation, restricting its inference performance on AI accelerators.

Demax

In this section, we present a uniform quantization operator designed specifically for the characteristics of Softmax outputs. This operator aims to effectively mitigate quantization loss caused by extreme values in attention scores.

The Softmax function, a central normalization operation in the attention mechanism, produces outputs that exhibit a characteristic power-law distribution, as illustrated in Figure 1. Existing methods typically employ log-based non-uniform quantization to manage these values, but they overlook the intrinsic characteristics of the Softmax function. Specifically, the output of the Softmax function often contains a dominant value per row, with a magnitude much larger than the others. For example, a row of attention weights might exhibit a distribution like [0.8, 0.1, 0.05, 0.05]. This pronounced non-uniformity creates several challenges for low-bit quantization:

- **Extreme values exacerbate loss**: Large values dominate in uniform quantization, compressing smaller ones and causing severe information loss.
- **Clipping harms importance**: Clipping distorts key attention tokens, impairing semantic representation and model performance.
- **Poor hardware compatibility**: Log-based methods lack hardware support, hindering integration with AI accelerator cores and limiting the actual acceleration gains.

Therefore, there is an urgent need for a quantization strategy that preserves the critical information of the dominant attention token, mitigates the impact of extreme values, and remains compatible with efficient matrix operations. To address this, we propose the DeMax operator, inspired by the

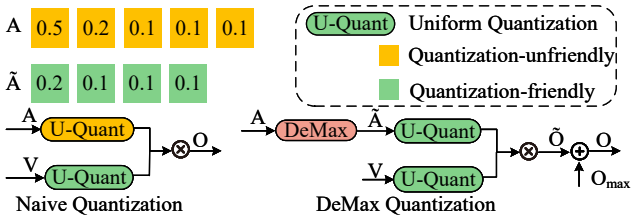

Figure 3: Flowchart of DeMax Quantization and Naive Quantization.

Softmax function's characteristic——where each row of its output typically contains a single extreme value. The core idea behind DeMax is to dynamically remove the maximum value from each row before quantization and record its positional index. During the dequantization phase, the removed values are restored using sparse matrix multiplication, where each row contains only one non-zero value. As a result, DeMax improves overall accuracy and enhances hardware friendliness.The structure of the DeMax operator is depicted in Figure 3.

The computation procedure of the DeMax operator is as follows. For input matrices $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{V} \in \mathbb{R}^{n\times p}$, the process begins by examining each row $\mathbf{A}_{i,:}$ to extract the maximum value $\mathbf{v}_i$ and its corresponding column index $\mathbf{idx}_i$. These values are then aggregated to form a value vector $\mathbf{v} \in \mathbb{R}^m$ and an index vector $\mathbf{idx} \in \mathbb{N}^m$. A sparse matrix $\mathbf{A}_{\text{max}}$ is subsequently constructed by preserving only the maximum value in each row and setting all other entries to zero. The matrix $\tilde{\mathbf{A}}$ is obtained by setting the entry at position $\mathbf{idx}_i$ in each row of $\mathbf{A}$ to zero,

$$\tilde{\mathbf{A}}_{i,\mathbf{idx}_i} = 0 \tag{6}$$

Consider the matrix multiplication $\mathbf{O} = \mathbf{AV}$, which can be decomposed into two parts: one part is $\tilde{\mathbf{O}} = \tilde{\mathbf{A}}\mathbf{V}$, and the other part is $\mathbf{O}_{\text{max}} = \mathbf{A}_{\text{max}}\mathbf{V}$. Since each row of $\mathbf{A}_{\text{max}}$ contains only one non-zero value, $\mathbf{O}_{\text{max}}$ can be simplified to the product of the maximum value index corresponding row of $\mathbf{V}$ with $\mathbf{v}_i$,

$$\mathbf{O}_{\text{max},i} = \mathbf{v}_i \cdot \mathbf{V}_{\mathbf{idx}_i,:} \tag{7}$$

and $\tilde{\mathbf{O}}$ can be computed using integer matrix multiplication accelerated by matrix computation cores,

$$\tilde{\mathbf{O}} = \tilde{\mathbf{A}}\mathbf{V} \approx s_a \mathbf{A}^{(\mathbb{Z})} \cdot s_v \mathbf{V}^{(\mathbb{Z})} \tag{8}$$

where s_a and s_v are scalar scaling factors, allowing the integer matrix multiplication to be performed first, thereby accelerating inference.

The design of DeMax considers the widely integrated matrix computation cores in AI acceleration chips. Typically, the computational complexity of an attention matrix multiplication is $O(N^2C)$, where N is the total number of tokens and C is the number of channels per token. After introducing the DeMax operator, the added complexity is $O(NC)$. In ViT, with $N = 197$, the additional computational cost of the DeMax operator is within an acceptable range compared to matrix multiplication.

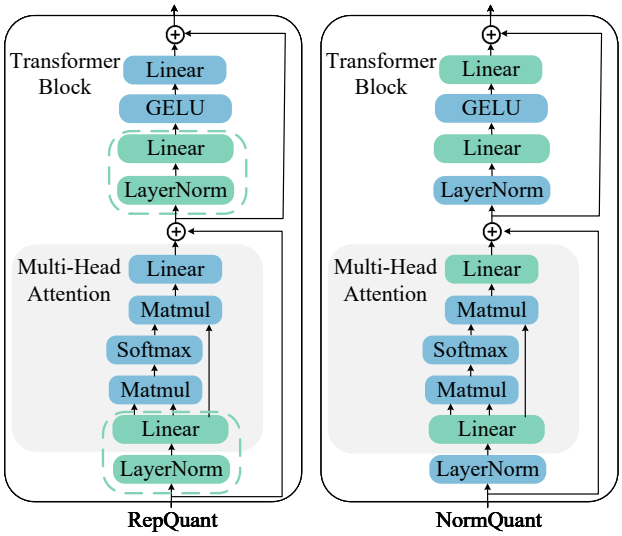

Figure 4: RepQuant is a reparameterization quantization method that relies on LayerNorm, limiting its applicability to other layers. In contrast, NormQuant can be applied to arbitrary linear layers.

Furthermore, the maximum value search and index recording operations in DeMax can reuse the traversal logic already present in the Softmax calculation process (to prevent numerical overflow by performing row-wise maximum value calculations), thus avoiding any additional statistical overhead.

NormQuant

In addition to post-Softmax activations, the activations after GELU function exhibit imbalanced activation distributions, marked by substantial variations in dynamic range across channels. Although existing reparameterization methods attempt to alleviate this issue, they often rely on LayerNorm, which limits their applicability. To address this issue and operate without normalization layers, we introduce a novel quantization operator, NormQuant.

The core idea of NormQuant is to combine the accuracy benefits of per-channel quantization with the computational efficiency of per-tensor quantization. It applies per-channel scaling during quantization to better align with the distribution characteristics of individual channels, while reverting to a per-tensor format during dequantization to preserve compatibility with efficient integer matrix operations. Moreover, NormQuant can be flexibly inserted before any linear or convolutional layer without requiring a preceding LayerNorm, thereby improving generality and flexibility.

The quantization and dequantization procedures of the NormQuant operator are as follows:

$$\text{Quant:} \quad \mathbf{x}^{(\mathbb{Z})} = \text{clip}\left(\left\lfloor \frac{\mathbf{x}}{\mathbf{s}'} + \mathbf{z}' \right\rceil, 0, 2^b - 1\right) \tag{9}$$

$$\text{DeQuant:} \quad \hat{\mathbf{x}} = s\mathbf{x}^{(\mathbb{Z})} \tag{10}$$

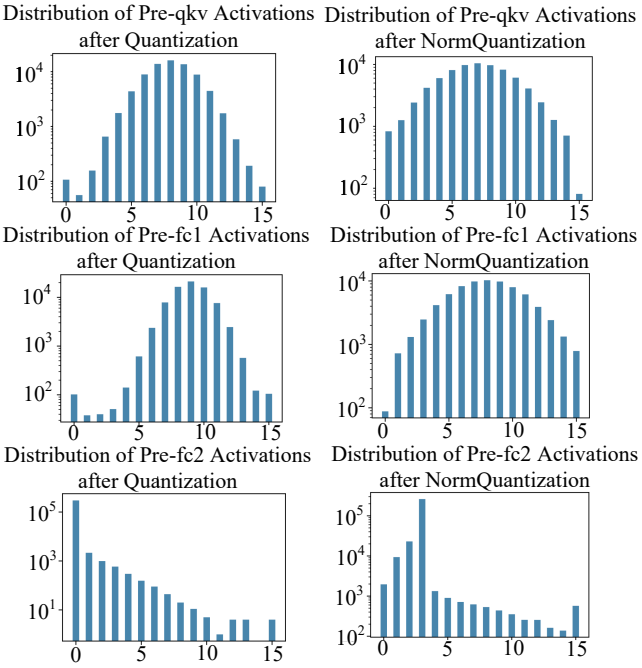

Figure 5: Illustration of the activation distributions at pre-qkv, pre-fc1, and pre-fc2 after naive quantization and NormQuant in Block 9 of DeiT-S. The horizontal axis represents the range of quantized values after 4-bit quantization, and the vertical axis denotes the count.

where $\mathbf{s}' \in \mathbb{R}^C$ is the per-channel scaling factor, $\mathbf{z}' \in \mathbb{R}^C$ is the per-channel zero-point offset, $s \in \mathbb{R}$ is the per-tensor scalar scaling factor, and C is the number of channels.

In other words, $\mathbf{s}'$ and $\mathbf{z}'$ can be regarded as the scaling and shifting coefficients of the normalization process. By applying this normalization, the originally imbalanced activation distributions become more uniform, effectively reducing quantization loss.

Following the reparameterization design, the NormQuant operator integrates the per-channel scaling factor $\mathbf{s}'$ and zero-point offset $\mathbf{z}'$ into the weights and biases, thereby incurring no additional computational overhead during inference. The updated parameters are computed as follows:

$$\mathbf{W}' = \frac{\mathbf{s}' \cdot \mathbf{W}}{s} \tag{11}$$

$$\mathbf{b}' = \mathbf{b} - \mathbf{s}'\mathbf{z}' \cdot \mathbf{W} \tag{12}$$

Here, $\mathbf{W}$ and $\mathbf{b}$ denote the original weights and biases, while $\mathbf{W}'$ and $\mathbf{b}'$ represent the reparameterized weights and biases used during inference.

After being processed by NormQuant, the activation distributions become flatter, indicating improved quantization-friendliness. As shown in Figure 5, under 4-bit quantization, the activation values are more evenly distributed across the 16 discrete quantization levels compared to the naive quantization scheme. This indicates that each quantization level retains more informative content. These results suggest

Methods	HF	W/A	ViT-S	ViT-B	DeiT-T	DeiT-S	DeiT-B	Swin-S	Swin-B
Full-Precision		32/32	81.39	84.54	72.21	79.85	81.80	83.23	85.27
PTQ4ViT (Yuan et al. 2022)	✓	4/4	42.57	30.69	36.96	34.08	64.39	76.09	74.02
APQ-ViT (Ding et al. 2022)	✓	4/4	47.95	41.41	47.94	43.55	67.48	77.15	76.48
RepQ-ViT (Li et al. 2023)		4/4	65.05	68.48	57.43	69.03	75.61	79.45	78.32
ERQ (Zhong et al. 2024)		4/4	68.91	76.63	60.29	72.56	78.23	80.74	82.44
Baseline	✓	4/4	64.62	67.61	56.80	68.98	74.41	79.65	76.35
UQ-ViT (ours)	✓	4/4	68.34	72.07	59.71	72.13	76.59	79.93	81.96
PTQ4ViT (Yuan et al. 2022)	✓	6/6	78.63	81.65	69.68	76.28	80.25	82.38	84.01
APQ-ViT (Ding et al. 2022)	✓	6/6	79.10	82.21	70.49	77.76	80.42	82.67	84.18
RepQ-ViT (Li et al. 2023)		6/6	80.43	83.62	70.76	78.90	81.27	82.79	84.57
ERQ (Zhong et al. 2024)		6/6	80.48	83.89	71.14	79.03	81.41	82.86	85.02
Baseline	✓	6/6	80.35	83.64	70.69	78.83	81.32	82.81	84.13
UQ-ViT (ours)	✓	6/6	80.71	83.81	71.17	79.07	81.45	82.82	84.99

Table 1: Results on the ImageNet dataset. Top-1 accuracy (%) is used as the evaluation metric. W/A specifies the bit-width configuration of weights and activations. HF denotes hardware-friendliness.

that NormQuant effectively reduces information loss during quantization.

Experiments

In this section, we compare UQ-ViT with existing ViT PTQ calibration methods on ImageNet (Deng et al. 2009), evaluating various quantization configurations and models. We validate its generalization on COCO object detection and instance segmentation (Lin et al. 2014), and conduct ablation studies to analyze its effectiveness.

Experimental Setup

Datasets and Models: We evaluate the proposed method on the ImageNet dataset for image classification, using representative Vision Transformer architectures including ViT, DeiT, and Swin Transformer as backbones. For object detection and instance segmentation, experiments are conducted on the COCO dataset using Mask R-CNN (He et al. 2017) and Cascade Mask R-CNN (Cai and Vasconcelos 2018), both with Swin Transformer as the backbone.

Implementation details: We adopt the RepQViT model with uniform quantization as the baseline, replacing its original $\log\sqrt{2}$ quantization scheme. For calibration, 512 images are randomly selected from the ImageNet dataset and 2 images are randomly selected from the COCO dataset. Specifically, channel-wise quantization is applied to the weights, while the pre-Linear activations are quantized using NormQuant. All post-Softmax activations are quantized using the DeMax operator. Following the SmoothQuant strategy (Xiao et al. 2023), the scaling factors for each NormQuant layer are determined via a simple search procedure. All experiments are conducted using the PyTorch framework (Paszke et al. 2019) on a single NVIDIA 2080Ti GPU.

State-of-the-Art Comparison

Quantization Results on ImageNet Dataset: We compare UQ-ViT with existing PTQ calibration methods for ViTs on the ImageNet dataset. Experimental results are reported across various Vision Transformer architectures under both 4-bit and 6-bit quantization configurations. The tables also indicate whether each method is hardware-friendly.

As shown in Table 1, under the 6-bit quantization, our method achieves performance comparable to existing non-uniform quantization approaches across multiple ViTs, while outperforming baseline methods in terms of Top-1 accuracy. Under the more challenging 4-bit quantization, our method demonstrates a significant advantage, achieving an average Top-1 accuracy improvement of 3.2 percentage points over the baseline. Notably, it even matches the performance of current state-of-the-art non-uniform PTQ calibration methods. Specifically, our approach achieves a 4.46% improvement on ViT-B and a 5.61% improvement on Swin-B compared to the baseline. Notably, UQ-ViT is entirely based on uniform quantizers, allowing efficient matrix multiplication acceleration by fully leveraging matrix computation cores commonly supported in AI accelerators.

Quantization Results on COCO Dataset: The experiments on object detection and instance segmentation are conducted on the COCO dataset, and the quantization results are summarized in Table 2. Consistent with the setup in the classification task, we also annotate whether each method is hardware-friendly. Under the 6-bit quantization, our method achieves performance comparable to that of current state-of-the-art non-uniform quantization approaches, despite using only a uniform quantization strategy, demonstrating strong generalization capability.

Under the more challenging 4-bit quantization, our method achieves a detection performance of 39.0 AP^{box} on the Mask R-CNN model with a Swin-T backbone under the configuration of 4-bit weights and 4-bit activations, significantly outperforming existing methods. This result indicates that, despite being fully based on uniform quantization, our method can still achieve performance on par with current non-uniform-based state-of-the-art approaches on complex

Methods	HF	W/A	Mask R-CNN				Cascade Mask R-CNN			
			w. Swin-T		w. Swin-S		w. Swin-T		w. Swin-S	
			AP^{box}	AP^{mask}	AP^{box}	AP^{mask}	AP^{box}	AP^{mask}	AP^{box}	AP^{mask}
Full-Precision		32/32	46.0	41.6	48.5	43.3	50.4	43.7	51.9	45.0
PTQ4ViT (Yuan et al. 2022)	✓	4/4	6.9	7.0	26.7	26.6	14.7	13.5	0.5	0.5
APQ-ViT (Ding et al. 2022)	✓	4/4	23.7	22.6	44.7	40.1	27.2	24.4	47.7	41.1
RepQ-ViT (Li et al. 2023)		4/4	36.1	36.0	42.7*	40.1*	47.0	41.4	49.3	43.1
ERQ (Zhong et al. 2024)		4/4	36.8	36.6	43.4	40.7	47.9	42.1	50.0	43.6
UQ-ViT (ours)	✓	4/4	39.0	37.3	43.3	40.3	47.3	41.6	49.6	43.4
PTQ4ViT (Yuan et al. 2022)	✓	6/6	5.8	6.8	6.5	6.6	14.7	13.6	12.5	10.8
APQ-ViT (Ding et al. 2022)	✓	6/6	45.4	41.2	47.9	42.9	48.6	42.5	50.5	43.9
RepQ-ViT (Li et al. 2023)		6/6	45.1	41.2	47.8	43.0	50.0	43.5	51.4	44.6
UQ-ViT (ours)	✓	6/6	45.6	41.4	48.0	43.0	50.1	43.7	51.5	44.7

Table 2: Results on COCO dataset. AP^{box} denotes the box average precision for object detection, and AP^{mask} denotes the mask average precision for instance segmentation. * indicates the results are re-produced by using the official code.

Model	Method	Top-1 (%)
DeiT-T	Baseline	56.80
	Baseline + DeMax	58.07
	Baseline + NormQuant	58.65
	Baseline + NormQuant + DeMax	59.71
DeiT-S	Baseline	68.98
	Baseline + DeMax	69.67
	Baseline + NormQuant	71.45
	Baseline + NormQuant + DeMax	72.13
ViT-B	Baseline	67.61
	Baseline + DeMax	68.30
	Baseline + NormQuant	70.83
	Baseline + NormQuant + DeMax	72.07

Table 3: Ablation study of DeMax and NormQuant under the W4/A4 quantization.

vision tasks, while offering better hardware friendliness.

Ablation Studies

To assess the contributions of the key components in the proposed UQ-ViT framework, we conduct ablation studies on the two core quantization operators, NormQuant and DeMax. The results, presented in Table 3, demonstrate the effectiveness of each component across multiple Vision Transformer architectures.

Introducing NormQuant alone leads to noticeable improvements, 58.65% for DeiT-T and 71.45% percent for DeiT-S, indicating that it effectively alleviates quantization loss arising from channel-wise activation imbalance. Further incorporating DeMax yields additional gains, boosting the Top-1 accuracy to 59.71% percent on DeiT-T and 72.13% percent on DeiT-S. The results indicate that NormQuant and DeMax effectively mitigate the challenges posed by quantization-unfriendly activations in ViTs, leading to improved model accuracy.

Layer Configuration	Deit-T	Deit-S	Vit-S
qkv + proj + fc1 + fc2	59.71	72.13	68.34
proj + fc1 + fc2	55.72	61.02	65.72
qkv + fc1 + fc2	58.85	72.01	68.42
qkv + proj + fc2	54.40	57.63	67.19
qkv + proj + fc1	58.50	69.92	66.01

Table 4: Ablation study of NormQuant under W4/A4 quantization across different layer configurations. Top-1 accuracy (%) is used as the evaluation metric.

To further investigate the contribution of NormQuant within individual linear layers, we conduct an ablation study by selectively disabling NormQuant in each linear layer within a transformer block. The resulting performance changes are summarized in Table 4. As shown in the table, fc1, qkv, and fc2 contribute significant gains of +6.99%, +5.61%, and +1.92% in Top-1 accuracy, respectively, while proj yields only +0.30%. This aligns with ViT's observation of imbalanced activation distributions. Since NormQuant on proj incurs no extra cost, we keep it for completeness.

Conclusion

In this paper, we propose UQ-ViT, a novel uniform quantization method for ViTs, aimed at addressing the challenges posed by the extreme distribution of attention activations. Our main contributions include: DeMax, a hardware-friendly uniform quantization operator that reduces quantization loss by dynamically preserving dominant attention tokens; and NormQuant, a flexible quantization operator that integrates channel-wise precision and tensor-wise efficiency. Experimental results show that UQ-ViT achieves accuracy comparable to existing non-uniform quantization methods, while maintaining hardware friendliness. This work provides a practical and generalizable solution for efficient low-bit quantization of ViTs.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62471398 and Grant 62136007, in part by the National Science and Technology Major Project of China under Grant 2022ZD0119005.

References

Bertasius, G.; Wang, H.; and Torresani, L. 2021. Is space-time attention all you need for video understanding? In *Icml*, volume 2, 4.

Cai, H.; Li, J.; Hu, M.; Gan, C.; and Han, S. 2023. Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, 17302–17313.

Cai, Z.; and Vasconcelos, N. 2018. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6154–6162.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

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 preprint arXiv:1805.06085*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Ding, Y.; Qin, H.; Yan, Q.; Chai, Z.; Liu, J.; Wei, X.; and Liu, X. 2022. Towards accurate post-training quantization for vision transformer. In *Proceedings of the 30th ACM international conference on multimedia*, 5380–5388.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Esser, S. K.; McKinstry, J. L.; Bablani, D.; Appuswamy, R.; and Modha, D. S. 2019. Learned step size quantization. *arXiv preprint arXiv:1902.08153*.

Fu, M.; Yu, H.; Shao, J.; Zhou, J.; Zhu, K.; and Wu, J. 2025. Quantization without tears. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 4462–4472.

Han, S.; Mao, H.; and Dally, W. J. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Jacob, B.; Kligys, S.; Chen, B.; Zhu, M.; Tang, M.; Howard, A.; Adam, H.; and Kalenichenko, D. 2018. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2704–2713.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4015–4026.

Kong, Z.; Dong, P.; Ma, X.; Meng, X.; Niu, W.; Sun, M.; Shen, X.; Yuan, G.; Ren, B.; Tang, H.; et al. 2022. Spvit: Enabling faster vision transformers via latency-aware soft token pruning. In *European conference on computer vision*, 620–640. Springer.

Krishnamoorthi, R. 2018. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Li, Z.; Xiao, J.; Yang, L.; and Gu, Q. 2023. Repq-vit: Scale reparameterization for post-training quantization of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17227–17236.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Liu, J.; Niu, L.; Yuan, Z.; Yang, D.; Wang, X.; and Liu, W. 2023a. Pd-quant: Post-training quantization based on prediction difference metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24427–24437.

Liu, Y.; Yang, H.; Dong, Z.; Keutzer, K.; Du, L.; and Zhang, S. 2023b. Noisyquant: Noisy bias-enhanced post-training activation quantization for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20321–20330.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022.

Liu, Z.; Ning, J.; Cao, Y.; Wei, Y.; Zhang, Z.; Lin, S.; and Hu, H. 2022. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3202–3211.

Lv, C.; Chen, H.; Guo, J.; Ding, Y.; and Liu, X. 2024. Ptq4sam: Post-training quantization for segment anything. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 15941–15951.

Mehta, S.; and Rastegari, M. 2021. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178*.

Moon, J.; Kim, D.; Cheon, J.; and Ham, B. 2024. Instance-aware group quantization for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16132–16141.

Nagel, M.; Amjad, R. A.; Van Baalen, M.; Louizos, C.; and Blankevoort, T. 2020. Up or down? adaptive rounding for post-training quantization. In *International conference on machine learning*, 7197–7206. PMLR.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Strudel, R.; Garcia, R.; Laptev, I.; and Schmid, C. 2021. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 7262–7272.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, 10347–10357. PMLR.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wei, X.; Gong, R.; Li, Y.; Liu, X.; and Yu, F. 2022. Qdrop: Randomly dropping quantization for extremely low-bit post-training quantization. *arXiv preprint arXiv:2203.05740*.

Wu, K.; Zhang, J.; Peng, H.; Liu, M.; Xiao, B.; Fu, J.; and Yuan, L. 2022. Tinyvit: Fast pretraining distillation for small vision transformers. In *European conference on computer vision*, 68–85. Springer.

Wu, Z.; Chen, J.; Zhong, H.; Huang, D.; and Wang, Y. 2024. Adalog: Post-training quantization for vision transformers with adaptive logarithm quantizer. In *European Conference on Computer Vision*, 411–427. Springer.

Wu, Z.; Wang, S.; Zhang, J.; Chen, J.; and Wang, Y. 2025a. FIMA-Q: Post-Training Quantization for Vision Transformers by Fisher Information Matrix Approximation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 14891–14900.

Wu, Z.; Zhang, J.; Chen, J.; Guo, J.; Huang, D.; and Wang, Y. 2025b. APHQ-ViT: Post-Training Quantization with Average Perturbation Hessian Based Reconstruction for Vision Transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 9686–9695.

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 *International conference on machine learning*, 38087–38099. PMLR.

Yu, H.; and Wu, J. 2023. A unified pruning framework for vision transformers. *Science China Information Sciences*, 66(7): 179101.

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*, 191–207. Springer.

Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L. M.; and Shum, H.-Y. 2022a. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*.

Zhang, J.; Peng, H.; Wu, K.; Liu, M.; Xiao, B.; Fu, J.; and Yuan, L. 2022b. Minivit: Compressing vision transformers with weight multiplexing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12145–12154.

Zhong, Y.; Hu, J.; Huang, Y.; Zhang, Y.; and Ji, R. 2024. Erq: Error reduction for post-training quantization of vision transformers. In *Forty-first International Conference on Machine Learning*.