Affine Transformations for Outlier-Resilient Post-Training Quantization on Diffusion Models

Aarush Gupta
Massachusetts Institute of Technology
aarushg@mit.edu

Albert Tam
Massachusetts Institute of Technology
altam@mit.edu

Orion Foo
Massachusetts Institute of Technology
ofoo@mit.edu

Sumedh Shenoy
Massachusetts Institute of Technology
sshenoy@mit.edu

Abstract

Diffusion models are a powerful class of generative models that achieve state-of-the-art results in high-fidelity image generation. Quantization is crucial for enabling faster and more efficient inference for these large and complex models, but outliers in their activation and weight distributions often limit compression efficiency and degrade performance. In this study, we address outliers by applying invertible affine transformations to reshuffle and normalize weight and activation distributions, reducing the impact of extreme values prior to post-training quantization. Our approach preserves model accuracy while significantly lowering memory and compute overhead, enabling stable and effective quantization of diffusion models without extensive retraining. In particular, our experiments demonstrate an ability to quantize all the way to W6A6 with little quality loss, paving the way for more efficient deployment of these high-quality generative models. Moreover, our experiments indicate that even more aggressive quantization may be possible.

1 Introduction

Diffusion models are rapidly growing in size and complexity, which has led to significant computational overheads. As these models become more sophisticated, they will demand substantially more and more compute, which creates several bottlenecks, especially in the compute-intensive diffusion regime. Thus, there is a need for the quantization of a model to decrease both inference time and memory footprint.

An example of a naive quantization technique is round-to-nearest, where we quantize by taking

$$\bar{\mathbf{X}} = \text{round}\left(\frac{\mathbf{X}}{\Delta}\right) \cdot \Delta,$$

where $\bar{\mathbf{X}}$ is the quantized tensor, $\mathbf{X}$ is the floating-point tensor, Δ is the quantization step size, and $\text{round}(\cdot)$ represents the rounding function that is widely used often fail in diffusion models.

Especially in diffusion, naive quantization can lead to significant performance degradation, making it difficult to reduce model size and computational complexity without sacrificing output quality. A known contributor to quantization error is the occurrence of *outliers* in both the weights and activations of models [9]. Outliers make quantization difficult, as lower-magnitude information is completely erased in expanding the dynamic range to capture these outliers.

With outlier reduction being such a significant roadblock for quantization, many different approaches to mitigating outliers have been proposed. One particularly promising method from language

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

modeling literature is the use of *invertible transformations* to reduce outliers while maintaining computational invariance [12]. However, this idea has yet to be explored in diffusion models; thus, the central focus of our work is applying lightweight invertible transformations to transformer-based diffusion models to achieve quantization with minimal quality degradation.

2 Related Work

Smoothing-based quantization. The first major advancement in outlier reduction was SmoothQuant [13], which provided a simple and effective way of reducing outlier distribution in activations for language models. They proposed simply moving the quantization difficulty from the activations to the weights with a scale parameter as follows:

$$\bar{\mathbf{X}} = \left\lfloor \frac{\mathbf{X}}{\Delta} \right\rceil, \quad \Delta = \frac{\max(|\mathbf{X}|)}{2^{N-1}-1},$$

where $\mathbf{X}$ is the floating-point tensor, $\bar{\mathbf{X}}$ is the quantized counterpart, Δ is the quantization step size, $\lfloor \cdot \rceil$ is the rounding function, and N is the number of bits. While this leads to significant improvements, the difficulty of outliers is still present in this framework, motivating further work.

Rotation-based quantization. Again, primarily from language model quantization literature, there is a large body of work on applying *invertible rotations* to activations to reduce outlier channels. This idea, proposed first in Quip [2] and QuaRot [1], essentially states that when random transformations are applied to activations, the statistical phenomenon of outliers is reduced significantly since the outliers are evenly distributed across the activations. Then, since these applied rotations are invertible, they can be undone for computational invariance, allowing for identical results to be obtained while reducing the outliers in activations.

This idea was then built upon by SpinQuant [11], which proposes that these rotations can be optimized to further reduce the outliers, with calibration occurring using a small calibration set. This showed great improvement and was further developed by FlatQuant [12], which proposed to use general affine transformations, rather than just limiting to rotation matrices. Moreover, the use of affine transformations allowed for the ideas from SmoothQuant to be effectively incorporated, allowing for a scaling parameter to naturally be incorporated into the invertible transformation. This achieved extremely high performance, able to achieve W4A4 quantization with little performance cost relative to FP16 models. FlatQuant makes it possible to apply more aggressive quantization without sacrificing fidelity. However, the FlatQuant paper solely looked at the application of these affine transformations to language models, leaving room for exploring its applications to other domains.

Diffusion-focused quantization. Diffusion quantization literature, however, tackles these outliers through a different set of techniques. Q-Diffusion [10] introduced timestep-aware calibration, which samples a calibration dataset uniformly across timesteps in the diffusion process. ViDiT-Q [15] uses a mix of techniques—adding parameters to adjust for token-wise outliers, as well as adopting a SmoothQuant-based approach with time-dependent α smoothing parameters to obtain high-quality W8A8 quantization of diffusion models. SVDQuant [8], a more recent work, obtains impressive W4A4 results by moving activation outliers to the weights (via smoothing), and then handling outliers in the weights by factorizing them out with a low-rank branch via SVD. These works are extremely impressive; however, neither utilize the idea of using invertible translations from language modeling quantization.

3 Methodology

Motivated by the lack of an invertible transformation-based quantization framework for diffusion models, we develop and explore the use of these techniques in transformer-based diffusion models. In particular, we focus on adopting FlatQuant's affine transformation framework, due to its ability to be used with adaptive LayerNorms—which, critically, previous works like SpinQuant and QuaRot are unable to do, due to their structure. This is because FlatQuant applies affine transformations before and after every linear layer, as opposed to other methods, which apply them before and after whole transformer blocks. Thus, it is even feasible to use the techniques developed in FlatQuant for transformer-based diffusion models, which was not true for any rotation-based work previously (since

DiTs use adaptive layer normalization), leaving this area unexplored. Thus, our primary contribution is (a) expanding the framework of applying invertible transformations for ease-of-quantization to include cross-attention blocks, not just self-attention (which are requisite for diffusion models, but not in the decoder-only autoregressive language models) and (b) releasing code for our affine transformation-based quantization framework for diffusion models. The primary advantage of our code is that is applicable to a large class of diffusion models, since it can be easily incorporated into any diffusion model in the Huggingface library, assuming the use of regular diffusion transformer architectures.

Although our method can be easily applied to a whole host of pretrained diffusion models, for compute reasons, we focus on using the **PixArt-Σ** family of text-to-image diffusion models to run inference with [3]. However, the set of experiments we were able to run was very limited, due to strict compute constraints (and diffusion being very expensive to run). We evaluate our model on W8A8 and W6A6, and W4A8 quantization schemes.

4 Framework

We apply quantization with affine transformations to the weights and activations of a full-precision floating point 16 (FP16) version of PixArt-Σ at various levels in order to test our hypotheses. The figure below specifically illustrates the quantization mechanism in the self-attention and cross-attention blocks, where $\mathcal{P}$ indicates a trained affine transformation. A similar basic affine transformation quantization scheme is used in the network's feedforward blocks, but this is identical to FlatQuant, so we omit it.

Moreover, we experiment using different affine transformations for the latents and cross-embeddings (so $\mathcal{P}_{a,\text{latent}} \neq \mathcal{P}_{a,\text{encoding}}$) and the same affine transformation for the two ($\mathcal{P}_{a,\text{latent}} = \mathcal{P}_{a,\text{encoding}}$). As we discuss later, we found that the former works significantly better; but due to the training setup being more complex, we were only able to use this setup to train a model with W6A6 quantization. Given more time and compute, we would have attempted to train a W4A4 quantized model.

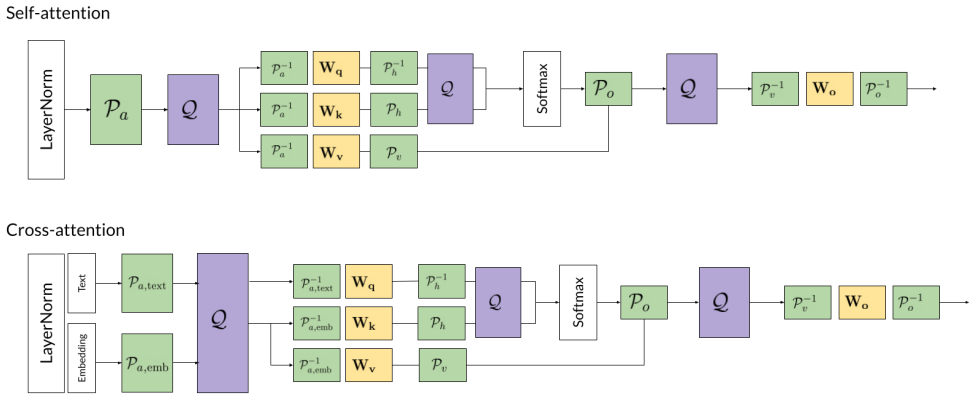

Figure 1: Figure illustration the quantization and affine transformation framework for both self-attention and cross-attention layers in the PixArt-Σ diffusion model. Affine transformations are denoted by green blocks, quantization by purple blocks, standard weight matrices by yellow blocks, and other operations by white blocks. Note that for cross-attention, separate affine transformations are applied for the text conditioning inputs and latents (embeddings) passing through the model.

Our quantized model is calibrated on **COCO-Captions** [4], a collection of human-generated captions describing various images. Training is done block-wise and is trained until convergence. In particular, we implement early-stopping and train for 15 epochs otherwise (whichever occurs first), although we generally do not end up early stopping. We run calibration on 4 prompts, sampling calibration data for each timestep at 20 timesteps.

The evaluation dataset that we use is **MJHQ-30K** [7], which is a dataset consisting of 30,000 images generated by Midjourney. This dataset consists of images falling into the following set of categories: `animals`, `art`, `fashion`, `food`, `indoor`, `landscape`, `logo`, `people`, `plants`, `vehicles`.

Our code and implementation are available in a GitHub repository.[1]

5 Results

We first display qualitative results, specifically the images generated by various quantized models for different text prompts, in Figure 2.

Prompt: *a dragon soaring through the skies, above mountainous terrain*

(a) Pixart-Σ FP16 (b) Pixart-Σ W8A8 (c) Pixart-Σ W6A6 (d) Pixart-Σ W4A8

Prompt: *endless book labyrinth illustration for The City of Dreaming Books by walter moers bookshelf dungeon glowing mushrooms scenic light under city underground rustic old place ar 169'*

(e) Pixart-Σ FP16 (f) Pixart-Σ W8A8 (g) Pixart-Σ W6A6 (h) Pixart-Σ W4A8

Prompt: *beautiful Jaguar decorated with huichol beads, in the jungle, plants everywhere, DMT colours, ultra realistic , cinematic lighting v 5*

(i) Pixart-Σ FP16 (j) Pixart-Σ W8A8 (k) Pixart-Σ W6A6 (l) Pixart-Σ W4A8

Figure 2: Comparison of generated images from the three PixArt-Σ models (FP16, W8A8, W6A6, W4A8) for three prompts. Each row corresponds to one prompt, with columns showing the output of three models. Note that the W4A8 quantized models were trained with $\mathcal{P}_{a,\text{latent}} = \mathcal{P}_{a,\text{encoding}}$, which we found to be significantly worse; however, we did not have time to retrain with $\mathcal{P}_{a,\text{latent}} \neq \mathcal{P}_{a,\text{encoding}}$.

For a more quantitative and rigorous analysis, we evaluate each quantized model using the following metrics, for each of which a high-level description is provided below.

[1] https://github.com/tam-albert/fq-diffusion

1. **Fréchet Inception Distance (FID) [6]** compares the distribution of images generated by the quantized model to the distribution of a collection of ground truth images (unquantized model) by comparing summary statistics of the images (mean, variance, etc.).
2. **CLIP Score [5]** computes the cosine similarity between image and text embeddings using the Contrastive Language-Image Pre-Training (CLIP) model.
3. **Learned Perceptual Image Patch Similarity (LPIPS) [14]** computes the similarity of embeddings of images produced by the unquantized and quantized models.
4. **Peak Signal-to-Noise Ratio (PSNR)** measures the peak ratio of signal-to-noise present between images produced by the unquantized and quantized models.

Note that only W8A8 and W6A6 schemes were trained with $\mathcal{P}_{a,\text{latent}} \neq \mathcal{P}_{a,\text{encoding}}$. We expect that W4A8 results would be better if we were able to retrain.

Model	**FID*** $\downarrow$	**CLIP Score** $\uparrow$	**LPIPS** $\downarrow$	**PSNR (dB)** $\uparrow$
Unquantized (FP16)	57.9	0.330	-	-
W8A8 Quantization	57.6	0.335	0.097	23.2
W6A6 Quantization	58.6	0.332	0.142	21.1
W4A8 Quantization*	65.3	0.318	0.562	17.2

Table 1: Comparison of models based on FID, CLIP score, LPIPS, and PSNR metrics on the MJHQ-30K dataset. Lower values of FID and LPIPS indicate better performance, while higher values of CLIP score and PSNR are preferred. Note that LPIPS and PSNR metrics are not provided for the unquantized model as this is the reference for the W8A8 metrics. Note that the FID metric is not very meaningful, due to the small sample size on which evaluations were conducted.

6 Discussion & Conclusion

In our study applying invertible affine transformations, inspired by FlatQuant, optimized for diffusion models, we found that W8A8 and W6A6 quantization performed quite well, leading to minimal degradation in image quality while theoretically saving inference time and FLOPs (we found fusing the operations and writing efficient kernels outside the scope of this project). In particular, the W8A8 quantization scheme achieved similar quality to the FP16 unquantized model, obtaining strong LPIPS and PSNR scores.

One potential reason that may make affine transformations less suitable for diffusion quantization than for language modeling is the significant timestep dependence of outlier distributions in diffusion models, which makes time-independent affine transformations less effective. Moreover, our calibration strategy may be flawed: due to GPU memory limitations, we were unable to calibrate on very large sets, which may have lead to improper representation of outlier distributions at later time steps, leading to our quantization model diverging from the FP16 version of the final output.

With our original training strategy of taking $\mathcal{P}_{a,\text{latent}} = \mathcal{P}_{a,\text{encoding}}$, we found that we obtained poor similarity to FP16 for sub-8 bit quantization. For the schemes we had time to recalibrate (W8A8, W6A6) with our new implementation that allowed us to take $\mathcal{P}_{a,\text{latent}} \neq \mathcal{P}_{a,\text{encoding}}$, we found substantial quality improvements in terms of similarity to the FP16 outputs. Our experiments were limited by computational resources and time, but future studies may look to reproduce our work and test more extreme quantization schemes (W4A4) and also obtain results for W4A8 quantization with $\mathcal{P}_{a,\text{latent}} \neq \mathcal{P}_{a,\text{encoding}}$.

7 Individual Contributions

Our group consisted of four (4) members. Every group member contributed equally to all aspects of brainstorming, paper writing, and poster-making. Additionally, Aarush wrote the code for and ran the model, Albert set up experiments and established the baselines, Sumedh worked on model quantization and training, and Orion handled the training process and set up the inference pipeline.

References

[1] Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-Free 4-Bit Inference in Rotated LLMs, October 2024. arXiv:2404.00456 [cs].

[2] Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-Bit Quantization of Large Language Models With Guarantees, January 2024. arXiv:2307.13304 [cs].

[3] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation, 2024.

[4] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server, 2015.

[5] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022.

[6] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, January 2018. arXiv:1706.08500 [cs].

[7] Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation, 2024.

[8] Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. SVDQuant: Absorbing Outliers by Low-Rank Components for 4-Bit Diffusion Models, November 2024. arXiv:2411.05007 [cs].

[9] Shiyao Li, Xuefei Ning, Luning Wang, Tengxuan Liu, Xiangsheng Shi, Shengen Yan, Guohao Dai, Huazhong Yang, and Yu Wang. Evaluating quantized large language models, 2024.

[10] Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models, 2023.

[11] Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations, October 2024. arXiv:2405.16406 [cs].

[12] Yuxuan Sun, Ruikang Liu, Haoli Bai, Han Bao, Kang Zhao, Yuening Li, Jiaxin Hu, Xianzhi Yu, Lu Hou, Chun Yuan, Xin Jiang, Wulong Liu, and Jun Yao. FlatQuant: Flatness Matters for LLM Quantization, October 2024. arXiv:2410.09426 [cs].

[13] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and Efficient Post-Training Quantization for Large Language Models, March 2024. arXiv:2211.10438 [cs].

[14] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric, April 2018. arXiv:1801.03924 [cs].

[15] Tianchen Zhao, Tongcheng Fang, Enshu Liu, Rui Wan, Widyadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. ViDiT-Q: Efficient and Accurate Quantization of Diffusion Transformers for Image and Video Generation, June 2024. arXiv:2406.02540 [cs].