FOCUS CALIBRATION FOR ULTRA-LOW-BIT ALL-REDUCE IN BANDWIDTH-CONSTRAINED COLLABORATIVE LLM INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

In bandwidth-constrained environments, collaborative inference of large language models (LLMs) is increasingly bottlenecked by activation communication under tensor parallelism. Although low-bit activation compression can reduce communication overhead in distributed LLM inference, further bit-width reduction is approaching practical limits, as aggressive compression often incurs substantial accuracy degradation or requires costly retraining. To address this challenge, we propose Focus Calibration, a lightweight calibration strategy for retraining-free ultra-low-bit vector quantisation, which enables an index-based all-reduce mechanism that exchanges compact centroid indices instead of full-precision activation tensors. Experimental results show that our method substantially reduces activation communication overhead, achieving up to 38.8% and 12.3% reductions in end-to-end prefill and decoding latency, respectively, while maintaining accuracy comparable to existing quantisation-based approaches at ultra-low bit-widths of 2-2.5 bits. These results establish index-based all-reduce with Focus Calibration as a practical solution for communication-efficient LLM inference in bandwidth-constrained collaborative systems.

1 INTRODUCTION

With the continued evolution and growing demands of AI applications (e.g. Large Language Model (LLM)), hardware platforms either fail to handle large-scale workload or face the risk of obsolescence due to insufficient computational resources and degrading performance. To address this challenge, researchers are now exploring solutions that can bring these legacy devices together as a collaborative cluster to support high-performance AI applications (Husom et al., 2025).

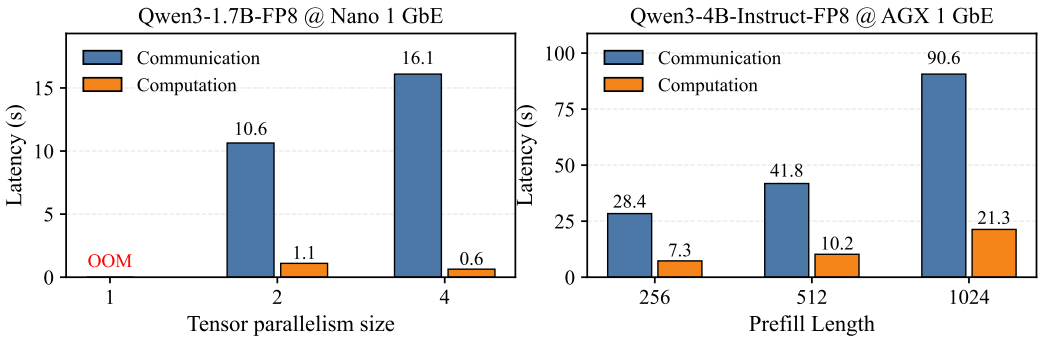

Figure 1: LLM prefill phase time cost profile in bandwidth-limited collaborative devices. **Left**: Jetson-Orin-Nano with 1 GbE bandwidth, TP=1,2,4, and prefill length=512, by using vLLM (Kwon, 2025) with max model length=8192, and max batch size=1024. The Jetson-Orin-Nano is Out-Of-Memory (OOM) in TP=1, due to the large inter-buffers and KV cache, the desired memory exceeds the available memory (8 GB Shared RAM); **Right**: Jetson-AGX-Orin 64GB with 1 GbE bandwidth, TP=2, and prefill length=256, 512, 1024, max model length=4096, and max batch size=4096.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Although collaboration alleviates the computational resource limitations of individual devices, Tensor Parallelism (TP) collaborative inference introduces significant communication overhead in bandwidth-constrained environments. Such constraints may arise from the limited support for high-speed physical interconnects on legacy devices or from interconnect contention caused by co-located background workloads. As shown in Figure 1, scaling the cluster effectively distributes the computational workload, but also increases communication cost, which grows by approximately twofold at each scaling step. As a result, communication becomes the dominant contributor to end-to-end latency, accounting for nearly 80% of the total latency for prefill lengths ranging from 256 to 1024.

Low-bit quantisation along the communication path is a promising way to reduce the communication bottleneck in TP LLM inference, but aggressive quantisation often causes substantial accuracy degradation in practical workloads. To address this challenge, we propose an index-based all-reduce mechanism that fundamentally changes how intermediate activations are communicated across TP collaborative devices. Rather than transmitting quantised activation values directly, each device maps activations to shared centroids and exchanges only their compact indices, substantially reducing communication volume at ultra-low bit-widths. To make such aggressive compression practical without model retraining, we further introduce a lightweight calibration strategy that compensates for activation distribution drift induced by quantisation. Together, the proposed index-based communication and calibration mechanisms enable bandwidth-efficient TP collaborative inference while preserving model accuracy under constrained communication budgets.

2 PRELIMINARIES

To understand the communication bottlenecks in model parallelism and the limitations of existing quantisation methods, this section introduces the key concepts of model parallelism and widely used quantisation techniques discussed throughout the paper.

2.1 MODEL PARALLELISM

Pipeline Parallelism. It exploits the fact that modern Neural Networks (NNs) are naturally amenable to layer-wise partitioning. By carefully assigning different layers to heterogeneous devices, Pipeline Parallelism (PP) enables the deployment of large-scale models across resource-constrained clusters and can improve multitasking throughput through effective inter-layer scheduling (Ma et al., 2024; Xu et al.). However, the performance of a PP collaborative devices is often constrained by bottleneck devices, which can significantly limit the latency of individual tasks within the pipeline. As a result, relying solely on inter-layer parallelism is insufficient for achieving low-latency serving.

Tensor Parallelism. It is proposed by (Shoeybi et al., 2019), is orthogonal to but compatible with PP. The core idea of TP is to partition the weights of paired linear transformations in the attention and MLP modules using column-wise and row-wise slicing. Combined with all-reduce operations, this design allows each device to perform partial local computations while synchronizing the resulting global activations across devices. As a result, TP reduces the computation and memory burden on individual devices by distributing both the model weights and the corresponding computation workload across multiple devices for collaborative execution.

We take the MLP block as an example. An MLP block consists of a pair of linear transformations—namely, the up-projection and down-projection layers-with a GeLU activation function in between. Let the corresponding weight matrices be denoted by A and B, respectively. Suppose that two devices, $Node_1$ and $Node_2$, collaboratively execute the MLP block using TP for an input activation X, as follows:

$$A = [A_1, A_2],\ B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \tag{1}$$

$$[Y_1, Y_2] = [\text{GeLU}(XA_1), \text{GeLU}(XA_2)] \tag{2}$$

$$Y_1 B_1 + Y_2 B_2 = [Y_1, Y_2] \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \tag{3}$$

The input activation X is processed sequentially by the up-projection, the GeLU activation (Equation 2), and the down-projection (Equation 3). The up-projection weight matrix is column-wise partitioned into A_1 and A_2, whereas the down-projection weight matrix is row-wise partitioned into B_1 and B_2. Accordingly, $Node_1$ and $Node_2$ compute the partial outputs Y_1B_1 and Y_2B_2, respectively. To reconstruct the final output, TP requires an all-reduce operation to aggregate these partial results across devices. However, compared with inter-layer communication in PP, the all-reduce operations incurred by TP in attention and MLP blocks introduce substantially greater communication overhead, which can lead to significant latency in bandwidth-limited collaboration environments (Ye et al., 2024). In practice, a more effective solution is to adopt Hybrid Parallelism (HP), which combines inter-layer parallelism (PP) with intra-layer parallelism (TP) to reduce communication overhead and latency (Xu et al., 2021; Zheng et al., 2022; Aminabadi et al., 2022). However, the collaboration between HP and TP incurs additional latency due to activation synchronisation. To address this issue, we propose an index based all-reduce scheme that compresses the communication payload using extremely low-bit precision, while preserving model performance.

2.2 QUANTISATION

Block-wise Quantisation. It is a data-oblivious quantisation technique that has been widely used for communication and activation compression in NN inference (Hansen-Palmus et al., 2024; Dong et al., 2024; Li et al., 2024b;a). However, it still struggles to mitigate the LLM performance collapse in extreme low-bit quantisation. As a representative example, the Microscaling (MX) format (Rouhani et al., 2023) partitions a vector into small blocks and assigns each block a shared scale, encoded as an 8-bit exponent. This scale normalises the values within the block into a suitable range for low-bit expression, such as FP4 (E2M1, consisting of 1 sign bit, 2 exponent bits, and 1 mantissa bit). For a value x in a block with shared scale s, the conversion between FP16 and MX format over the range $[-r, r]$ is defined by quantisation and dequantisation functions $Q(\cdot)$ and $Q^{-1}(\cdot)$, where $clamp(x, y, z)$ is used to confine x within the $[y, z]$:

$$q = Q(x) = clamp(x/s,\ -r,\ r),\ \ \hat{x} = Q^{-1}(q) = q \times 2^{s-127} \tag{4}$$

$$s = clamp(\log_2 [\, round[\max_{x \in block}(|x|)/r] \,] + 127,\ 0,\ 255) \tag{5}$$

Vector Quantisation. Compared to block-wise quantisation, Vector Quantisation (VQ) is a data-dependent quantisation method that requires pre-computing centroids (or prototypes) through data clustering in offline (Blalock & Guttag, 2021). In other words, VQ captures patterns instead of scaling the data to expect range, which makes it robust to channel-wise outlier (Wang et al., 2025; Dettmers et al., 2022; Lin et al., 2024). However, traditional VQ technology relies heavily on expensive retraining on massive datasets or large Lookup-Tables (LUTs) to achieve the desired performance. Specifically, VQ treats $\mathcal{X} \in \mathbb{R}^D$ as $M = D/d$ sub-vectors $[\mathbf{x}_1, \cdots, \mathbf{x}_m, \cdots \mathbf{x}_M]$. For each sub-vector $\mathbf{x}_m$, VQ pre-compute its centroids and construct a codebook $\mathcal{C}_m = [\mathbf{c}_{m,1}, \cdots, \mathbf{c}_{m,j}, \cdots, \mathbf{c}_{m,2^k}]$, where $\mathbf{x}_m,\ \mathbf{c}_{m,j} \in \mathbb{R}^d$ and D is divisible by d. The M codebooks finally construct a LUT. VQ uses Nearest Neighbor Search (NNS) to find centroid index idx_m of $\mathbf{x}_m$, and approximate $\mathbf{x}_m$ with $\mathbf{c}_{m,idx_m}$ in online computing:

$$idx_m = \text{encode}(\mathbf{x}_m, \mathcal{C}_m) \triangleq \arg\min_{0 \le j < 2^k} ||\mathbf{x}_m - \mathbf{c}_{m,2^k}||_2 \tag{6}$$

$$\hat{\mathbf{x}}_m = \text{decode}(idx_m, \mathcal{C}_m) \triangleq c_{m,idx_m} \tag{7}$$

Transformation-based Quantisation. To further reduce the Mean Squared Error (MSE) caused by outliers, (Wu et al., 2025; Han et al., 2025; Zandieh et al., 2025; Gao & Long, 2024; Ashkboos et al., 2024) proposed transformation-based quantisation solutions. These solutions uses non-linear or linear transformation to enforce data to follow the expected distribution to 'remove' outlier, thereby improving numerical fidelity in quantisation. However, it significantly degrades model quality in aggressive quantisation and introduces considerable transformation cost. By transforming the activation into an expected distribution, quantisation can be performed in a data-oblivious manner. As one of the state-of-the-art transformation-based quantisation solution, TurboQuant (Zandieh et al., 2025) assembled by two separate algorithms Turboquant$_{MSE}$ (TQ-MSE) and Turboquant$_{PROD}$ (TQ-PROD), achieves up to 2.5-bits KV-cache quantisation. In this paper, we employ only TQ-MSE for communication compression, because TQ-PROD does not correct the

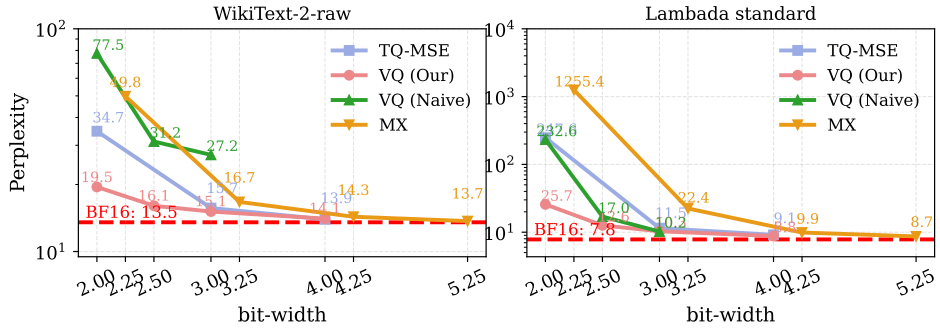

Figure 2: The Qwen3-4B-Instruct-2507 perplexity (PPL) performance across various communication path quantisation solutions at different bit-width precision. The performance is evaluated on zero-shot WikiText-2-raw test set (Merity et al., 2016) and lambada standard (Paperno et al., 2016), where the VQ (Our) utilises the proposed calibration method on 3% of WikiText-2 train set. The number of bits in VQ is discussed in Section 5.1

summation error arising from all-reduce operations. Concretely, TQ-MSE uses a random rotation matrix $\mathbf{\Pi} \in \mathbb{R}^{D\times D}$, generated via QR decomposition of a random matrix with i.i.d. Gaussian entries, to map a normalised vector $\mathcal{X} \in \mathbb{R}^D$ to $\mathcal{Y} = [y_1, \ldots, y_D]$, whose entries approximately follow a beta distribution. Each dimension is then quantised by approximating y_d with a centroid selected from $\mathbf{c} = [c_1, \ldots, c_{2^k}]$, where the centroids are defined as the mean values of 2^k disjoint intervals of the beta distribution produced by the Max–Lloyd algorithm:

$$\mathcal{Y} = \mathbf{\Pi} \cdot \mathcal{X}, \quad idx_d = encode(y_d, \mathbf{c}) \triangleq \underset{0\le j<2^k}{\arg\min} |y_d - c_j| \tag{8}$$

$$\hat{y}_d = decode(idx_d, \mathbf{c}) \triangleq c_{idx_d}, \quad \hat{\mathcal{X}} = \mathbf{\Pi}^\intercal \cdot \hat{\mathcal{Y}} \tag{9}$$

3 Challenges

Performance Collapse in Low-Bit Quantisation. Although quantisation is an effective approach for reducing communication overhead, activation quantisation at extremely low bit widths remains challenging due to distortion introduced during compression and the consequent loss in model quality. As illustrated in Fig. 2, block-wise quantisation methods, such as MX, are unable to prevent perplexity collapse when activations are quantised to 3 bits or lower. Transformation-based methods, such as TQ-MSE, achieve improved perplexity compared to MX at 3-bit precision, but still exhibit substantial degradation at more aggressive quantisation levels. Moreover, they incur additional computational overhead due to the transformation and inverse transformation steps, as shown in Equations 8 and 9. (Tang et al., 2023) indicates that the MSE introduced by VQ accumulates across NN layers, and its failure to account for training loss further exacerbates performance degradation. This imply that naively using VQ could degrade perplexity significantly.

Expensive Retraining and Limited Improvement. To maintain model performance while preserving the compression advantages of VQ, (Son et al., 2025) introduces a calibration-free VQ that leverages linear transformations and Normalise–Shift–Normalise operations to quantise KV-caches. (Liu et al., 2024b) proposes a second-order Post Training Quantisation method for VQ, achieving up to 2-bits weight quantisation. However, these methods are specifically tailored to KV-cache or weight quantisation, limiting their feasibility for activation synchronisation during all-reduce. (Schönleber et al., 2023; Van Den Oord et al., 2017; Razavi et al., 2019; Mentzer et al., 2023) employ the straight-through estimator to retrain NN that incorporates non-differentiable VQ encoding and decoding in selected computation paths. Furthermore, (Fuketa et al., 2024) enhances this approach by introducing a residual LUT to capture the residual error during backpropagation. Existing solutions typically maintain VQ performance either through full-model retraining or by increasing LUT capacity. However, retraining VQ-integrated LLMs via backpropagation is prohibitively expensive, while the performance gains from enlarging LUT are gradually diminishing (Gupta et al., 2024; Abouelhamayed et al., 2024).

The substantial communication overhead, combined with the limited bandwidth environments due to interconnect contention or low-speed physical connection, makes low-bit quantisation an attractive solution for communication compression. However, aggressive quantisation often degrades model quality, while retraining to mitigate this degradation is computationally expensive and offers limited gains. The core challenge, therefore, is to develop a TP communication compression method that reduces communication overhead through low-bit quantisation while preserving model quality and avoiding costly retraining.

4 CONTRIBUTIONS

To mitigate the performance degradation of VQ without expensive retraining, and to further reduce communication overhead in bandwidth-constrained TP collaborative devices, we propose a lightweight calibration method. Using only 3% (960 of 36,718 samples) of the WikiText train set, this method enables VQ to achieve lower perplexity while maintaining the same LUT size (i.e., 2^k in Equation 6). Compared with block-wise quantisation MX and transformation-based quantisation TQ-MSE, the calibrated VQ reduces perplexity by up to 61% at a comparable precision level on zero-shot benchmarks, such as wikitext-2-raw (Figure 2). Moreover, it achieves an effective bit width of 2.5 bits with perplexity comparable to that of 3.25-bit MX FP3. With proposed calibration strategy, VQ can be integrated into an index-based all-reduce mechanism for TP activation synchronisation, theoretically reducing communication payload by up to 87% (2 bits versus 16 bits). Compared to using original precision FP16 or BF16, this substantial compression further yields up to 69% and 44% reduction in end-to-end prefill and decoding latency in the collaborative devices with interconnect contention. The main contributions of this paper is summarised as follows:

Focus Calibration. We propose focus calibration method to alleviate the perplexity and accuracy degradation of VQ caused by data distribution drift and over-fitting, by reducing centroid mismatch and correcting outlier centroids. Compared with state-of-the-art quantisation methods, the proposed calibrated VQ enables activation quantisation down to 2 bits for collaborative communication, while achieving lower perplexity at a comparable effective bit width using only 3% of the WikiText train set. Zero-shot results across multiple benchmarks further demonstrate that the proposed calibration allows index-based all-reduce to maintain competitive accuracy in TP collaborative LLM inference.

Index-based All-Reduce. By combining VQ with the proposed focus calibration method, our index-based all-reduce reduces communication payload by up to 87% relative to the baseline precision (BF16/FP16), by transmitting packed centroid indices instead of full activation tensors. Using 2.5-bit VQ, the proposed index-based all-reduce achieves up to 36% and 11% improvements in time to first token (TTFT) and time per output token (TPOT) in bandwidth-constrained collaborative devices, compared with all-reduce using block-wise quantisation MX FP4 and transformation-based quantisation TQ-MSE at 3 bits, respectively, while preserving comparable perplexity and accuracy.

5 METHODOLOGY

5.1 INDEX-BASED ALL-REDUCE

The intense communication overhead in TP collaborative computing hinders the bandwidth-limited collaborative devices to build low-latency cooperative cluster. As shown in Figure 3, the activation needs synchronization across devices every time after the row-wise parallelism linear (i.e., the output projection in attention and the down projection in MLP as shown in Equation 3) to aggregate partial results for following computations by using all-reduce, which becomes the main bottleneck of processing the transformer block in TP collaboration. To reduce the communication overhead, we proposed the index-based all-reduce, which uses k-bits indices to represent activation to achieve equivalent low-bit precision compression by using VQ.

The index-based all-reduce builds upon VQ assisted all-gather and local reduction. Specifically, the activation is firstly sliced into M sub-vectors, each of them is encoded (Equation 6) by using 2^k centroids to obtain M k-bits indices to represent M d-dimensional sub-vectors. These indices are then gathered by all nodes and decoded (Equation 7) via node-specific LUTs (see Appendix

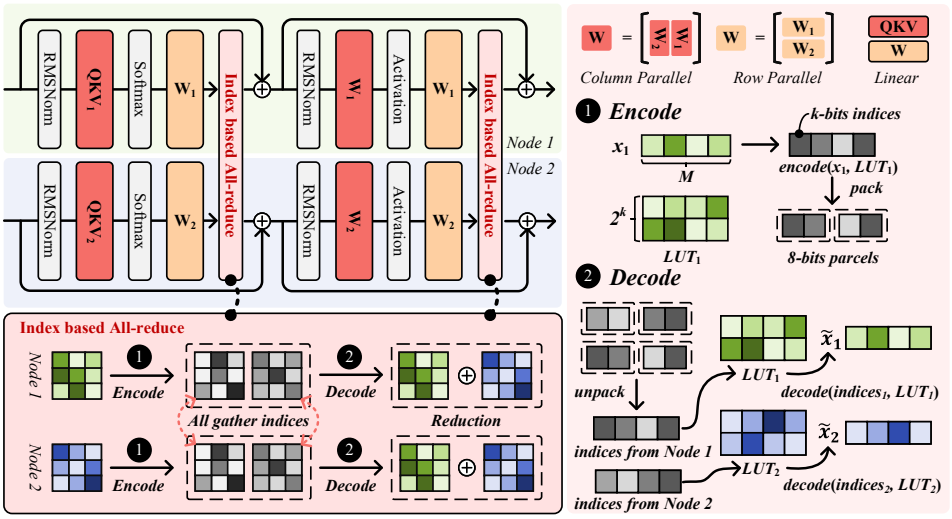

Figure 3: Compress all-reduce by transmitting the indices rather than the activation itself. The activation is firstly encoded to indices in local, and these indices are packed and exchange via all-gather across all nodes in TP collaborative cluster. Once a node has collected all the indexes, it unpacks and decodes indices to activation for local reduction.

A.1) to approximate activation for following local reduction. Although the encode, decode and local reduction introduces additional computation cost (see Appendix A.4), reducing communication overhead in bandwidth-limited collabrative devices can bring significant improvement in end-to-end latency. Suppose the activation is n-dimensional, the index-based all-reduce transmits activation by using n/d k-bits indices, which achieves k/d-bits equivalent quantisation in communication. In this paper, we use $d = 2$ and $2^k = 16, 32, 64$ centroids to implement index-based all-reduce, which equals to 2-bits, 2.5-bits and 3-bits effective bits by representing every 2 activation elements in activations as 4-bits, 5-bits and 6-bits indices, respectively. However, naively using 8-bits or 32-bits format to store these indices consuming additional bandwidth. Therefore, we pack these indices as a parcel to align with the 8-bits standard format, for instance 8 5-bits indices can be packed into 5 8-bits parcels, thereby maximising valid bandwidth utilisation.

5.2 Focus Calibration

The VQ generates centroids through offline sampling and k-means to approximate the values under the original data distribution as closely as possible. Ideally, VQ can capture the data distribution and describe real data with sufficient centroids with one-shot calibration as illustrate in Figure 4. However, the activation distribution is drifted due to the accumulated approximation error, which results in centroids mismatch and exacerbating the distribution drifts, as illustrated in Figure 5. Therefore, the encoded activation gradually gathered in fewer centroids as layer goes deep, degrading expression capability of LUT generated by using real data distribution in following layers (see Appendix A.2). To mitigate data distribution drift, we conduct the layer-wise calibration which layer-wisely constructs LUT sequentially, where each later layer is calibrated using activations produced after applying VQ to earlier index-based all-reduce. This allows the VQ in following layer can be aware the data distribution drift introduced by front layers, thereby reducing centroids mismatches.

However, because centroid generation is based on limited and noisy calibration data, layer-wise calibration remains prone to over-fitting in the VQ LUTs and therefore exhibits limited generalisation, resulting in suboptimal perplexity and accuracy. To mitigate this issue, we further introduce a subsampling strategy for layer-wise calibration. Specifically, the proposed calibration method collects pre-synchronisation activations (i.e., prior to all-reduce) for each layer and uniformly extracts $P\%$ of activations for the LUT construction. The subsamples retain the original activation distribution, while they effectively mitigate noise at the distribution boundaries. This enables the centroids to focus on capturing the primary structure of the original distribution (see Appendix A.3). Experimental results demonstrate that this focus calibration strategy improves generalisation and reduces centroid mismatch, without incurring expensive retraining or requiring extensive calibration data.

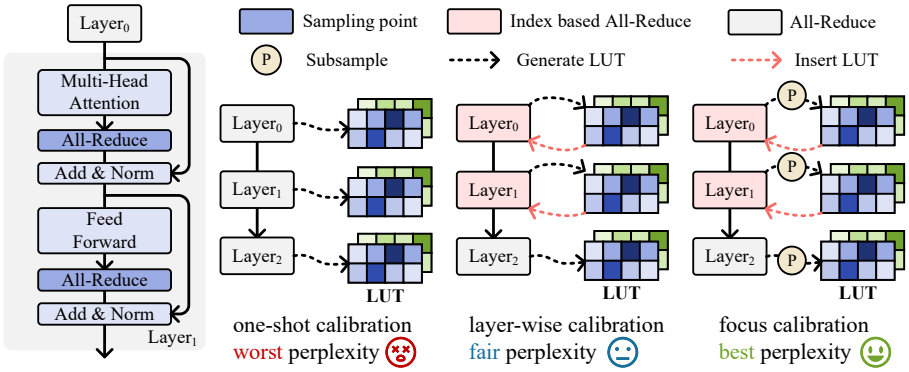

Figure 4: Three calibration strategies for VQ LUT generation. One-shot calibration: sampling internal activations before all-reduce to generate all LUTs in once; Layer-wise calibration: constructing LUTs sequentially layer, where each subsequent layer is calibrated using activations produced by applying VQ to previous index-based all-reduce; Focus calibration: Layer-wise calibration, but it uniformly extracts $P\%$ of activations to generate the LUT.

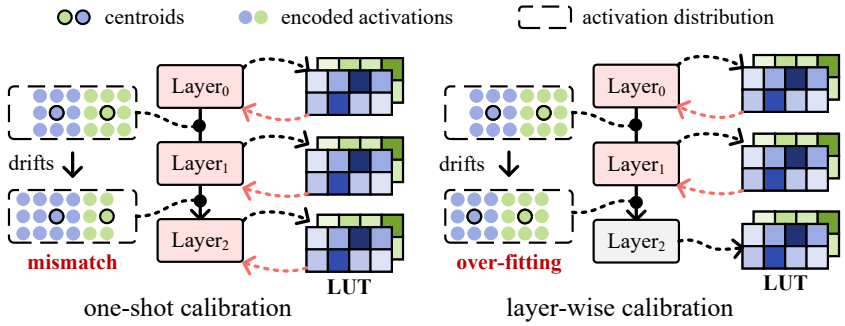

Figure 5: Activation distribution drift caused by approximation errors in the Layer_1 leads to centroids mismatches in the Layer_2 when using the one-shot calibration; While the layer-wise calibration reduce the mismatch error, but it falls into over-fitting due to the noise in limited samples.

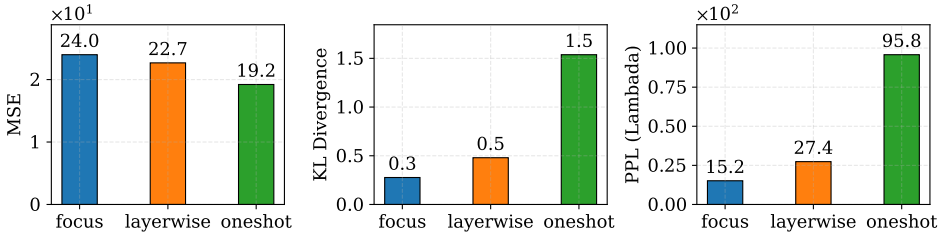

Figure 6: Qwen3-4B-Instruct-2507 logits MSE & probabilities (i.e., $softmax(logits)$) symmetric KL divergence of VQ 2-bits index-based all-reduce using focus ($P = 75\%$), layer-wise and one-shot calibration compared with BF16 naive all-reduce on lambada-openai test set.

Although focus calibration does not guarantee numerical fidelity, it significantly reduces data distribution drift compared to directly using original-precision activations, thereby improving perplexity in the index-based all-reduce. To study this effect, we calibrate the index-based all-reduce using $P = 75\%$ and 960 of 36,718 samples (according to ablation studies in Appendix A.5) from the WikiText-2 train set under three different strategies, and compare the logits of Qwen3-4B-Instruct-2507 against those produced by naive BF16 all-reduce on the Lambada-OpenAI test set. As illustrated in Figure 6, one-shot calibration achieves the lowest MSE but exhibits the highest symmetric KL divergence, indicating substantial drift in the output probability distribution (Dutta et al., 2024). In contrast, focus calibration mitigates centroid mismatch in one-shot calibration while alleviating the centroid over-fitting of layer-wise calibration under the same dataset and LUT configuration. Consequently, focus calibration reduce the centroid mismatch error introduced by distribution drift, while improving the generalisation ability across different data distributions.

6 EVALUATION

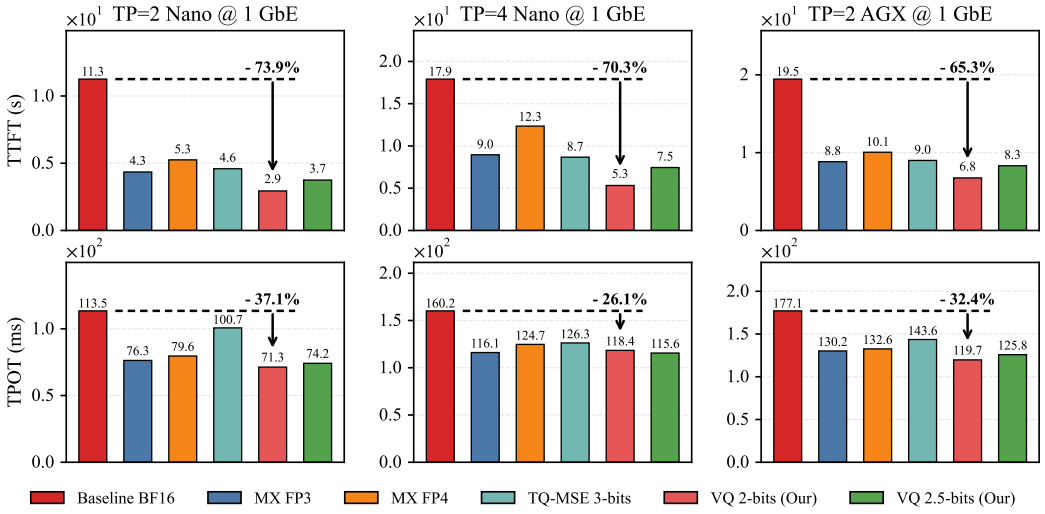

Figure 7: Time-to-First-Token (TTFT) and Time-per-Output-Token (TPOT) under 1 GbE interconnect. Qwen3-1.7B-FP8 is evaluated on TP=2 and TP=4 Jetson Orin Nano clusters with a max model length = 1024 and max batched token = 1024, while Qwen3-4B-Instruct-2507-FP8 is evaluated on a TP=2 Jetson AGX Orin 64GB cluster with max model length and max batched token set to 2048. Each experiment process 16 prompts with 512 input or output tokens per prompt (16×512 tokens).

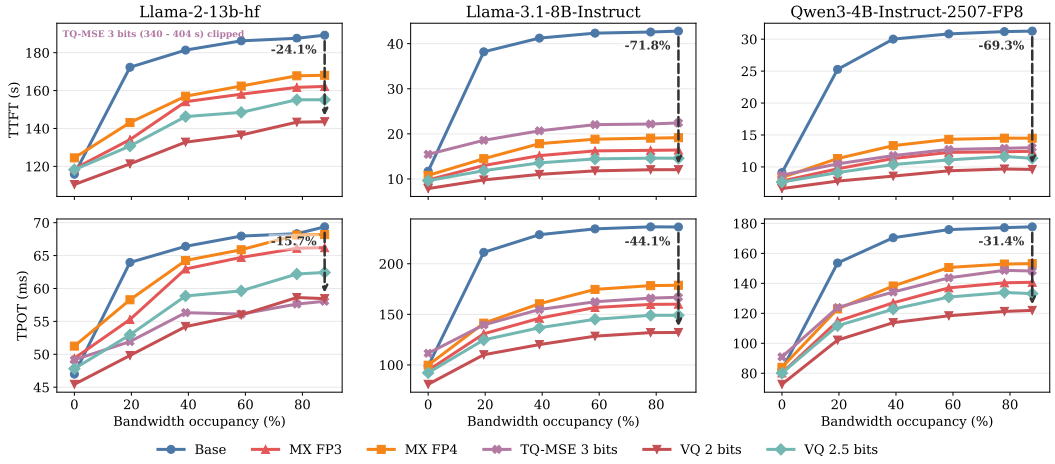

Figure 8: TTFT and TPOT of Llama-2-13B-hf, Llama-3.1-8B-Instruct, and Qwen3-4B-Instruct-2507-FP8 on a TP=2 NVIDIA A30 cluster under varying levels of background PCIe bandwidth occupancy. The system uses PCIe 3.0 $\times$ 16 with an unloaded bandwidth of 10.5 GBps, a max model length = 2048, and a max batched token = 2048. Each experiment processes 256 prompts with 512 input or output tokens per prompt (256×512 tokens).

Latency Improvement. We evaluate the end-to-end performance of TP collaborative inference on Jetson Orin Nano and Jetson AGX Orin clusters interconnected via 1 GbE. As shown in Figure 7, our index-based all-reduce consistently reduces both TTFT and TPOT compared with the BF16 baseline across all evaluated configurations. Specifically, index-based all-reduce with VQ 2-bit reduces TTFT by 65.3% - 73.9% and TPOT by 26.1% - 37.1%, while the VQ 2.5-bit variant achieves reductions of 57.3% - 66.7% and 27.9% - 34.6%, respectively. The significant TTFT improvement indicates that the proposed index-based all-reduce effectively reduces the communication payload associated with large activation transfers among TP collaborative devices during the communication-intensive prefill phase. In contrast, the decoding phase transfers a smaller amount of activation

data per token, increasing the relative contribution of synchronization and quantisation overheads and therefore limiting the improvement in TPOT. These results demonstrate that index-based all-reduce effectively alleviates communication overhead and improves end-to-end inference latency for bandwidth-constrained TP collaborative inference.

We further evaluate communication-path quantisation on a TP=2 A30 cluster under varying background bandwidth occupancy to emulate interconnect contention from co-located workloads. As shown in Figure 8, when the interconnect is unloaded (0% bandwidth occupancy), the reduction in communication volume is insufficient to offset the additional quantisation overhead, resulting in limited or no end-to-end latency improvement over BF16. However, the latency improvement of using communication path quantisation becomes significant as bandwidth occupancy increases, where communication becomes an increasingly dominant bottleneck. Specifically, the proposed index-based all-reduce with VQ 2-bit reduces TTFT by 24.1% - 71.8% and TPOT by 15.7% - 44.1%, across the evaluated models at the highest evaluated occupancy (90% bandwidth occupancy). Likewise, the VQ 2.5-bit variant achieves reductions of 18.0%-65.9% and 10.0%-36.9%, respectively. These results show that index-based all-reduce can effectively improve end-to-end latency when TP collaborative inference operates under interconnect contention.

Table 1: Performance Summary

Method	Bits	Orin Nano @ 1GbE [1]		AGX Orin @ 1GbE		A30 @ PCIe 3.0 × 16 [2]		Accuracy [3]
		TTFT (s)	TPOT (ms)	TTFT (s)	TPOT (ms)	TTFT (s)	TPOT (ms)	
Baseline	16	14.6	136.8	19.5	177.1	263.1	116.6	71.5
TQ-MSE	3	6.6	113.5	9.0	143.6	531.7	88.8	66.8
MX FP4	4.25	8.8	102.2	10.1	132.6	226.3	95.8	70.4
MX FP3	3.25	6.7	96.2	8.8	130.2	211.3	88.7	67.8
VQ (Our)	2.5	5.6	94.9	8.3	125.9	204.9	84.7	69.1
VQ (Our)	2	4.1	94.8	6.8	119.7	192.3	77.8	65.9

[1] Average TTFT and TPOT of TP=2 and TP=4 in Figure 7. [2] Average TTFT and TPOT of all bandwidth occupancy in Figure 8.
[3] Average accuracy of all models in Appendix A.6, Table 2.

Performance Trade-off. To evaluate the trade-off between accuracy and end-to-end latency, we compare the average accuracy, TTFT and TPOT of each method across all models, TP collaboration configurations and bandwidth occupancy listed in Table 2 (see Appendix A.6), Figure 7 and Figure 8. The proposed focus calibration enables the index-based all-reduce to achieve lower latency than both block-wise and transformation-based quantisation while maintaining comparable accuracy. Compared with MX FP4, VQ 2.5-bit reduces TTFT by 9.5% - 36.4% and TPOT by 5.1% - 11.6% with only a negligible accuracy degradation (69.1% vs. 70.4%). Compared with MX FP3, the proposed index-based all-reduce with VQ 2.5-bit achieves 1.3 percentage points higher accuracy (69.1% vs. 67.8%) while further reducing TTFT by 3.0% - 16.4% and TPOT by 1.4% - 4.5%. With more aggressive communication path quantisation, VQ 2-bit incurs an additional 1.9 percentage-point accuracy drop relative to MX FP3, but further reduces TTFT by 9.0%-38.8% and TPOT by 1.4%-12.3%. Meanwhile, it achieves 2.5-34.4 percentage points higher accuracy than MX FP2 and TQ-MSE 2-bit at comparable bit-widths (see Appendix A.6). These results demonstrate that the proposed index-based all-reduce together with focus calibration significantly reduces communication overhead in TP collaborative inference while mitigating the accuracy loss compared to existing communication-path quantisation approaches at lower bit-widths.

7 Conclusion

This work introduces an index-based all-reduce mechanism to address the dominant communication bottleneck in TP collaborative LLM inference under bandwidth-constrained environments, which may arise from the limited support for high-speed physical interconnects on legacy devices or from interconnect contention caused by co-located background workloads. By leveraging a lightweight focus calibration strategy, the proposed approach restricts low-bit representations to the communication path, incurring only modest additional computational. As a result, it significantly reduces activation communication overhead and prefill latency, while preserving accuracy and perplexity at extreme low-bit precision without costly retraining. These results demonstrate that the proposed method is a practical solution for collaborative LLM inference on bandwidth-limited collaborative systems, thereby enabling more efficient TP collaboration between legacy and lightweight devices.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, code implementation and clarifying mathematical notation. They were also used for literature exploration, language refinement, and improving the organization and readability of the paper. All methodological decisions, experimental settings, implementations, analyses, and conclusions were reviewed and verified by the authors. AI-generated suggestions were not accepted without manual inspection, and reported numerical results were obtained from the authors' experimental implementations rather than generated by AI. The authors take full responsibility for the final content of this work, including all text, claims, experiments, and conclusions.

ETHICS STATEMENT

This work does not involve human subjects, personal data, or privacy-sensitive information. All calibration or validation data used in the experiments were obtained from publicly available sources.

REPRODUCIBILITY STATEMENT

Implementation details are provided in methodology section 5, while the experimental settings are described in evaluation section 6. The code and detailed document are partially available in supplementary files during the review stage and will be fully open-sourced upon acceptance, further supporting reproducibility.

REFERENCES

Ahmed Abouelhamayed et al. Pqa: Exploring the potential of product quantization in dnn hardware acceleration. *ACM Transactions on Reconfigurable Technology and Systems*, 18(1):1–29, 2024.

Reza Yazdani Aminabadi et al. Deepspeed-inference: enabling efficient inference of transformer models at unprecedented scale. In *SC22: International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–15. IEEE, 2022.

Saleh Ashkboos et al. Quarot: Outlier-free 4-bit inference in rotated llms. *Advances in Neural Information Processing Systems*, 37:100213–100240, 2024.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020.

Davis Blalock and John Guttag. Multiplying matrices without multiplying. In *International Conference on Machine Learning*, pp. 992–1004. PMLR, 2021.

Christopher Clark et al. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)*, pp. 2924–2936, 2019.

Peter Clark et al. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in neural information processing systems*, 35: 30318–30332, 2022.

Harry Dong, Tyler Johnson, Minsik Cho, and Emad Soroush. Towards low-bit communication for tensor parallel llm inference. *arXiv preprint arXiv:2411.07942*, 2024.

Abhinav Dutta, Sanjeev Krishnan, Nipun Kwatra, and Ramachandran Ramjee. Accuracy is not all you need. *Advances in Neural Information Processing Systems*, 37:124347–124390, 2024.

Hiroshi Fuketa et al. Multiplication-free lookup-based cnn accelerator using residual vector quantization and its fpga implementation. *IEEE Access*, 2024.

Jianyang Gao and Cheng Long. Rabitq: Quantizing high-dimensional vectors with a theoretical error bound for approximate nearest neighbor search. *Proceedings of the ACM on Management of Data*, 2(3):1–27, 2024.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Leo Gao et al. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

Neelesh Gupta et al. Tabconv: Low-computation cnn inference via table lookups. In *Proceedings of the 21st ACM International Conference on Computing Frontiers*, pp. 180–188, 2024.

Insu Han et al. Polarquant: Quantizing kv caches with polar transformation. *arXiv preprint arXiv:2502.02617*, 2025.

Jan Hansen-Palmus et al. Communication compression for tensor parallel llm inference. *arXiv preprint arXiv:2411.09510*, 2024.

Erik Johannes Husom et al. Sustainable llm inference for edge ai: Evaluating quantized llms for energy efficiency, output accuracy, and inference latency. *ACM Transactions on Internet of Things*, 6(4):1–35, 2025.

Woosuk Kwon. *vLLM: An Efficient Inference Engine for Large Language Models*. PhD thesis, UC Berkeley, 2025.

Haoyang Li et al. Quickreduce: Up to 3x faster all-reduce for vllm and sglang, 2024a. URL https://rocm.blogs.amd.com/artificial-intelligence/quick-reduce/README.html.

Qingyuan Li et al. Flash communication: Reducing tensor parallelization bottleneck for fast large language model inference. *arXiv preprint arXiv:2412.04964*, 2024b.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6:87–100, 2024.

James Liu, Pragaash Ponnusamy, Tianle Cai, Han Guo, Yoon Kim, and Ben Athiwaratkun. Training-free activation sparsity in large language models. *arXiv preprint arXiv:2408.14690*, 2024a.

Yifei Liu et al. Vptq: Extreme low-bit vector post-training quantization for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8181–8196, 2024b.

Ruilong Ma et al. Hpipe: Large language model pipeline parallelism for long context on heterogeneous cost-effective devices. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pp. 1–9, 2024.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. *arXiv preprint arXiv:2309.15505*, 2023.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Denis Paperno et al. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1525–1534, 2016.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Bita Darvish Rouhani et al. Microscaling data formats for deep learning. *arXiv preprint arXiv:2310.10537*, 2023.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Jannis Schönleber et al. Stella nera: Achieving 161 top/s/w with multiplier-free dnn acceleration based on approximate matrix multiplication. *arXiv preprint arXiv:2311.10207*, 2023.

Or Shafran, Shaked Ronen, Omri Fahn, Shauli Ravfogel, Atticus Geiger, and Mor Geva. From directions to regions: Decomposing activations in language models via local geometry. *arXiv preprint arXiv:2602.02464*, 2026.

Ziqiang Shi, Rujie Liu, Shanshan Yu, Satoshi Munakata, and Koichi Shirahata. Scalpel: Fine-grained alignment of attention activation manifolds via mixture gaussian bridges to mitigate multimodal hallucination. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2964–2973, 2026.

Mohammad Shoeybi et al. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Donghyun Son, Euntae Choi, and Sungjoo Yoo. Nsnquant: A double normalization approach for calibration-free low-bit vector quantization of kv cache. *arXiv preprint arXiv:2505.18231*, 2025.

Xiaohu Tang et al. Lut-nn: Empower efficient neural network inference with centroid learning and table lookup. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pp. 1–15, 2023.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Tycho FA van der Ouderaa, Maximilian L Croci, Agrin Hilmkil, and James Hensman. Pyramid vector quantization for llms. *arXiv preprint arXiv:2410.16926*, 2024.

Zongwu Wang et al. Million: Mastering long-context llm inference via outlier-immunized kv product quantization. In *2025 62nd ACM/IEEE Design Automation Conference (DAC)*, pp. 1–7. IEEE, 2025.

Songhao Wu et al. Polarquant: Leveraging polar transformation for efficient key cache quantization and decoding acceleration. *arXiv preprint arXiv:2502.00527*, 2025.

Hongxin Xu, Tianyu Guo, and Xianwei Zhang. Dynapipe: Dynamic layer redistribution for efficient serving of llms with pipeline parallelism. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Yuanzhong Xu et al. Gspmd: general and scalable parallelization for ml computation graphs. *arXiv preprint arXiv:2105.04663*, 2021.

Xianglong Yan, ChengZhu Bao, Zhiteng Li, Tianao Zhang, Shaoqiu Zhang, Ruobing Xie, Samm Sun, and Yulun Zhang. d^2 quant: Accurate low-bit post-training weight quantization for llms. *arXiv preprint arXiv:2602.02546*, 2026.

Shengyuan Ye et al. Galaxy: A resource-efficient collaborative edge ai system for in-situ transformer inference. In *IEEE INFOCOM 2024-IEEE Conference on Computer Communications*, pp. 1001–1010. IEEE, 2024.

Amir Zandieh et al. Turboquant: Online vector quantization with near-optimal distortion rate. *arXiv preprint arXiv:2504.19874*, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Lianmin Zheng et al. Alpa: Automating inter-and {Intra-Operator} parallelism for distributed deep learning. In *16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22)*, pp. 559–578, 2022.

A Appendix

A.1 Using node-specific LUTs to maintain performance

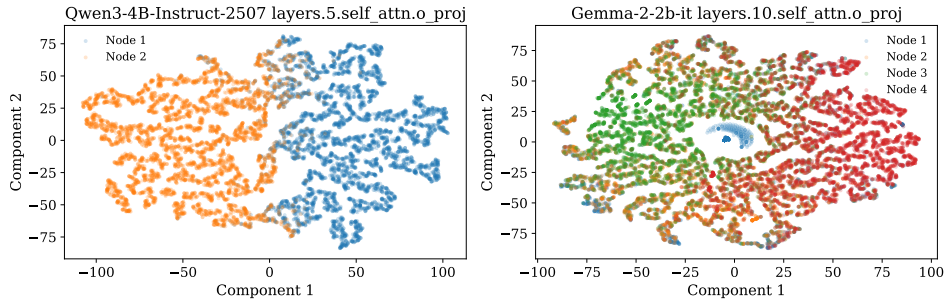

Figure 9: The t-SNE distribution map of sub-vector in activation across nodes in TP=2,4 cluster.

As illustrated in Figure 9, although the sub-vector distributions from different nodes partially overlap, it still has noticeable distribution differences across nodes. Consequently, some activation sub-vectors exhibit node-specific distributions, and using a shared LUT can incur 1% to 50% accuracy loss compared with using node-specific LUTs. Therefore, the proposed index-based all-reduce utilises node-specific LUT across the TP collaborative devices to preserve perplexity and accuracy.

A.2 Distribution drift results in centroid mismatches in one-shot calibration

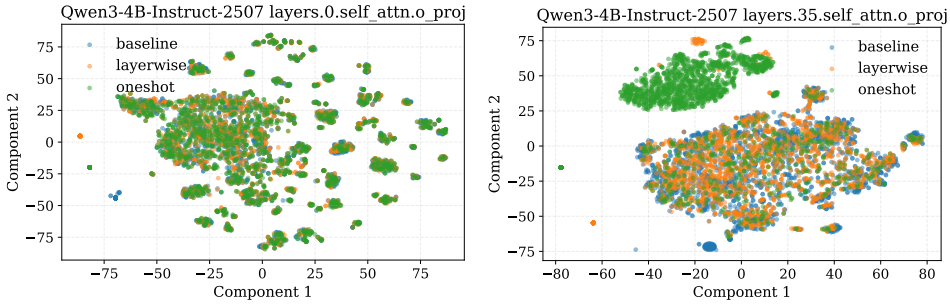

Figure 10: The first layer (left) and final layer (right) t-SNE distribution map of Qwen3-4B-Instruct-2507 internal activation on WikiText-2 train set, using original precision BF16 (baseline) and 2-bit VQ with one-shot (oneshot) and layer-wise (layerwise) calibration in all-reduce.

In the previous layer, VQ with both one-shot calibration and layer-wise calibration produces activations that follow a distribution similar to the baseline. However, in subsequent layers, VQ with one-shot calibration suffers from centroid mismatch, causing the activation distribution to drift and collapse into a narrow region. In contrast, layer-wise calibration preserves the expected distribution much more closely, as shown in Figure 9. This suggests that layer-wise calibration reduces centroid mismatch and prevents the accumulation of distribution drift across layers, thereby mitigating the performance degradation introduced by the index-based all-reduce.

A.3 Focus calibration can reduce over-fitting

Although layer-wise calibration can mitigate centroid mismatches, it still suffers from over-fitting due to outlier noise within the data. In contrast to systematic channel-wise outliers, which are intrinsic to LLM representations and critical to model performance (Dettmers et al., 2022; Lin et al., 2024), the sudden outlier noise within activation sub-vectors primarily stems from quantisation approximation errors (van der Ouderaa et al., 2024; Yan et al., 2026). Prior studies suggest that LLM activations can exhibit Gaussian-like or mixture-like distributional patterns (Shi et al., 2026; Shafran et al., 2026; Liu et al., 2024a). Motivated by these observations, we employ GMM to qualitatively

analyse the effect of uniform subsampling used in focus calibration and simulate the focus calibration process.

Expected Error Analysis of Focus Calibration. To qualitatively analyse the effect on empirical centroids with or without uniform subsampling, consider a simple scenario where K-means attempts to find a empirical centroid for an activation dataset S on a d-dimensional space. As an illustrative example, this space is modeled by a two-component GMM, which represents the clean sample distribution and the outlier distribution, respectively. Here, the dataset S comprises N total samples and is partitioned into a set of clean data points S_c and a set of outliers S_o. Specifically, S_c contains $N - m$ clean samples, where $s \sim \mathcal{N}(\mu, \Sigma_c)$ for each $s \in S_c$, with μ denoting the true target centroid. Concurrently, S_o contains m outliers concentrated in a region far from the true centroid; these outliers are modeled as $s \sim \mathcal{N}(\mu_o, \Sigma_o)$ for each $s \in S_o$.

We define $\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2]$ to assess the approximation error of using empirical centroid $\hat{\mu}_{S'}$ generated by k-means on given subset S' to approximate the true centroid μ. Denote displacement vector between the outlier center and the true centroid as $\Delta = \mu_o - \mu$. To reflect extreme anomaly scenarios, we assume $\|\Delta\|^2 \gg \text{Tr}(\Sigma_c)$.

Let S' be a subset of size $n = pN$ selected uniformly at random without replacement from S, where $p \in (0, 1)$ represents the sampling ratio. Let the random variable K denote the number of outliers contained in the subset S'. Under uniform sampling without replacement, K follows a hypergeometric distribution (i.e., $K \sim \text{Hypergeometric}(N, m, n)$) , Consequently, the probability of obtaining a completely clean subset ($K = 0$) is given by:

$$P_{\text{clean}} = P(K = 0) = \frac{\binom{N-m}{n}}{\binom{N}{n}} \approx \left(1 - \frac{m}{N}\right)^{pN} \approx e^{-p \cdot m}, \tag{10}$$

where $P_{\text{clean}} \approx e^{-p \cdot m}$ when $m \ll N$ and $N \to \infty$. Equation (10) implies that the probability of selecting clean distribution data exponentially decreases in proportion to $p \cdot m$. When m is fixed, a smaller p can increase the probability of obtaining a clean subset.

Conditioned on the event that the subset S' contains exactly k outliers (and thus $n-k$ clean samples), the empirical centroid $\hat{\mu}_{S'}$ can be expressed as a weighted sum of the empirical clean mean $\bar{X}_c$ and the empirical outlier mean $\bar{X}_o$:

$$\hat{\mu}_{S'} = \frac{n-k}{n}\bar{X}_c + \frac{k}{n}\bar{X}_o \tag{11}$$

Our objective is to derive the Mean Squared Error (MSE), or the expected squared Euclidean distance, between the empirical centroid and the true centroid μ when S' contains $K = k$ outliers:

$$\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2 \mid K = k] \tag{12}$$

We begin by decomposing the centroid error vector:

$$\hat{\mu}_{S'} - \mu = \frac{n-k}{n}(\bar{X}_c - \mu) + \frac{k}{n}(\bar{X}_o - \mu) \tag{13}$$

Since the expectation of the outlier mean is $\mathbb{E}[\bar{X}_o] = \mu_o = \mu + \Delta$, we substitute μ with $\mu_o - \Delta$ to explicitly incorporate the bias vector Δ:

$$\hat{\mu}_{S'} - \mu = \frac{n-k}{n}(\bar{X}_c - \mu) + \frac{k}{n}(\bar{X}_o - \mu_o) + \frac{k}{n}\Delta \tag{14}$$

Applying the trace operator, the expected squared norm expands to:

$$\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2 \mid K = k] = \left(\frac{n-k}{n}\right)^2 \mathbb{E}[\|\bar{X}_c - \mu\|^2] + \left(\frac{k}{n}\right)^2 \mathbb{E}[\|\bar{X}_o - \mu_o\|^2] + \left\|\frac{k}{n}\Delta\right\|^2 \tag{15}$$

In Equation (15), assuming the sampling of clean points and outliers is conditionally independent given $K = k$, the cross-terms vanish when taking the expectation. Substituting the finite-sample variances $\mathbb{E}[\|\bar{X}_c - \mu\|^2] = \frac{1}{n-k}\text{Tr}(\Sigma_c)$ and $\mathbb{E}[\|\bar{X}_o - \mu_o\|^2] = \frac{1}{k}\text{Tr}(\Sigma_o)$ into the expression yields the final closed-form conditional expectation:

$$\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2 \mid K = k] = \frac{n-k}{n^2}\text{Tr}(\Sigma_c) + \frac{k}{n^2}\text{Tr}(\Sigma_o) + \frac{k^2}{n^2}\|\Delta\|^2 \tag{16}$$

Equation (16) represents the expectation value of the squares of the deviations between empirical centroids to true centroid when the value k is exactly known. If K is random, the unconditional expected error should consider all possible Ks ($K = 0$ and $K \neq 0$.). Therefore, the approximation error of the empirical centroid $\hat{\mu}_{S'}$ and the true centroid μ, can be described by expectation:

$$\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2] = P_{\text{clean}} \cdot \underbrace{\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2 \mid K = 0]}_{\text{clean}} + (1 - P_{\text{clean}}) \cdot \underbrace{\mathbb{E}[\|\hat{\mu}_{S'} - \mu\|^2 \mid K \neq 0]}_{\text{noisy}}. \quad (17)$$

In Equation (17), the first term is $P_{\text{clean}} \cdot (\text{Tr}(\Sigma_c)/n)$ and the second term is varied according to K. In the extreme anomaly scenarios where $\|\Delta\|^2 \gg \text{Tr}(\Sigma_c)$, the noisy term $\frac{k^2}{n^2}\|\Delta\|^2$, which is a component of the second term in Equation (17), is dominant in the expected error. However, in Equation (17), enlarging P_{clean} with sub-sampling in Equation (10) would increase the first term error but decreases the second term error. The possible largest first term error is bounded to: $\text{Tr}(\Sigma_c)/n$ when $P_{\text{clean}} = 1$, which is negligible, compared to the second term error under the extreme anomaly scenarios. This suggests that, when the variance among clean samples is significantly smaller than the deviation between the true centroid and the outlier centroid in a sufficiently large dataset where outliers constitute only a small fraction of the samples (i.e, $m \ll N$), uniform subsampling can mitigate over-fitting by reducing the outliers (i.e., the noisy term) contribution.

Focus Calibration Simulation. To demonstrate how focus calibration mitigates over-fitting, we employ a GMM with 3 components (clusters) to approximate the activation distribution patterns and simulate the LUT generation process under both layer-wise ($P = 1$) and focus calibration ($P = 0.75$). As illustrated in Figure 11, suppose that some activations are clustered across a 2-dimensional space, the layer-wise calibration using full data ($P = 1$) generates an outlier centroid due to the noisy samples around the boundary between green cluster and blue cluster, while the focus calibration generates centroids fairly around the centres of the underlying data clusters by uniformly subsampling 75% of the full data. This suggests that if noise pattern generates outlier centroids, then some centroids that could represent meaningful activation values may be lost, which may degrade perplexity and accuracy because the LUT allocates part of its limited capacity to noise rather than to the meaningful activation patterns.

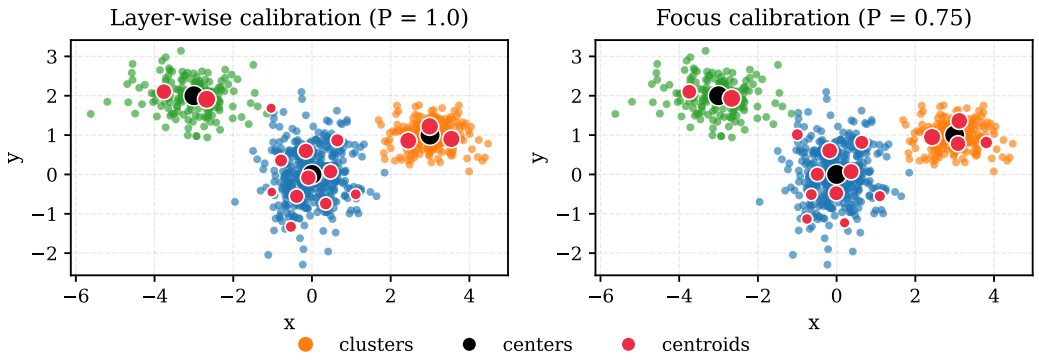

Figure 11: A simulation example for demonstrating over-fitting of VQ LUTs. Layer-wise calibration using all samples ($P = 1$) falls into over-fitting due to outlier noise, while focus calibration uniformly extracts $P\%$ samples ($P = 0.75$) can resist the effects of outlier noise.

Layer-wise MSE and KL divergence comparison. To investigate the over-fitting and generalisation of three calibration strategies, we compare the layer-wise MSE and KL divergence of Qwen3-4B-Instruct-2507 using VQ 2-bits index based all-reduce to naive all-reduce using original precision (BF16) on few-shot WikiText-2 train set and zero-shot lambada standard test set (bottom), where VQ LUT is calibrated by using 960 out of 36718 samples (about 3%) in WikiText-2 train set. As shown in Figure 12, the one-shot calibration always owns the worst performance in MSE and KL divergence due to centroid mismatches, while layer-wise and focus calibration achieve similar MSE and symmetric KL divergence on WikiText-2 train set. However, the KL divergence of layer-wise calibration degrades significantly compared to the focus calibration. This indicates that the focus

calibration can effectively reduce centroids mismatches and improve VQ generalisation compared to the one-shot and the layer-wise calibration strategies.

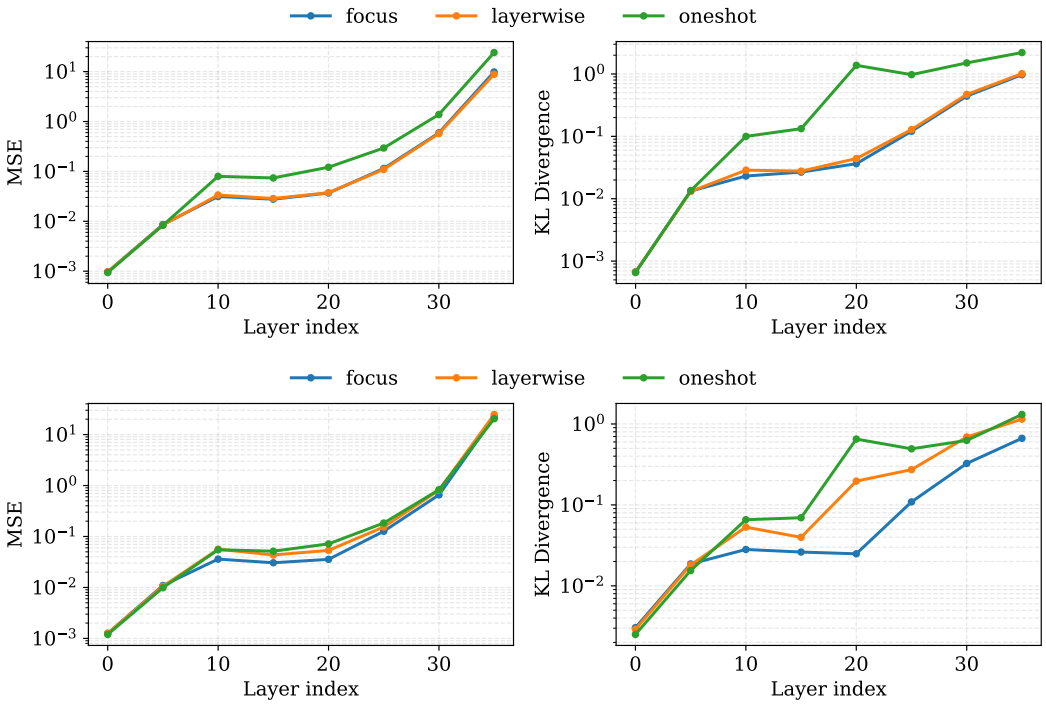

Figure 12: Qwen3-4B-Instruct-2507 layer-wise activation MSE & symmetric KL divergence of VQ 2-bits index based all-reduce using focus ($P = 75\%$), layer-wise and one-shot calibration compared with BF16 naive all-reduce on few-shot WikiText-2 train set (top) and zero-shot lambada standard test set (bottom).

A.4 Index-based All-reduce Time Cost Profile

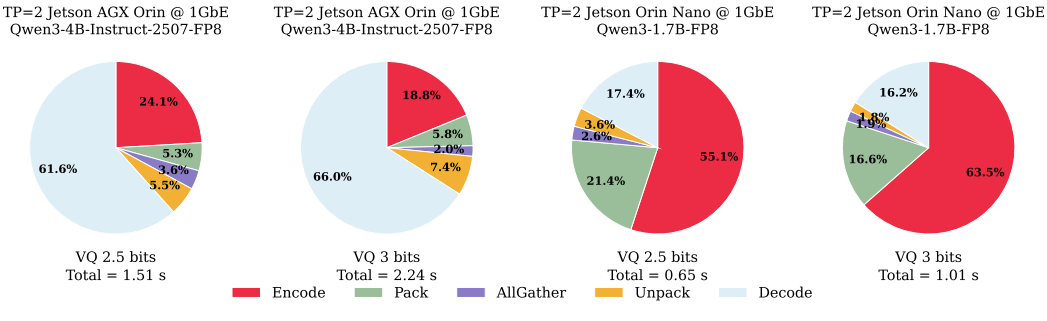

Figure 13: Average Time Cost Profile of Index-based All-reduce using VQ 2.5-bits and VQ 3-bits across different model and devices. All results are conducted in 512 input tokens, 4 round independent tests.

To assess the additional computation and memory-operation overhead introduced by index-based all-reduce, we profile the latency breakdown of its main stages, including encoding, index packing, all-gather, unpacking, and decoding. As shown in Figure 13 increasing the effective bit-width from VQ 2.5-bit to VQ 3-bit increases the overall latency on both platforms. This is expected because the larger codebook requires more prototype-distance evaluations during encoding and introduces additional LUT accesses during decoding. However, the dominant overhead differs across the two

evaluated configurations. For Qwen3-1.7B-FP8 on Jetson Orin Nano, encoding and packing account for most of the index-based all-reduce latency. In contrast, for Qwen3-4B-Instruct-2507-FP8 on Jetson AGX Orin, decoding becomes the dominant component. This difference is consistent with the characteristics of the corresponding operations: encoding and packing involve intensive distance computation and index manipulation, whereas decoding is dominated by LUT lookup, intermediate data movement, and reduction. Therefore, under the evaluated configurations, the index-based all-reduce overhead is primarily associated with compute-intensive operations on Orin Nano, while memory-intensive decoding becomes the major bottleneck on AGX Orin. These results also show that the dominant overhead of index-based all-reduce depends on the interaction between model configuration and hardware characteristics rather than being fixed across platforms.

A.5 ABLATION STUDY ON HYPER-PARAMETER OF FOCUS CALIBRATION

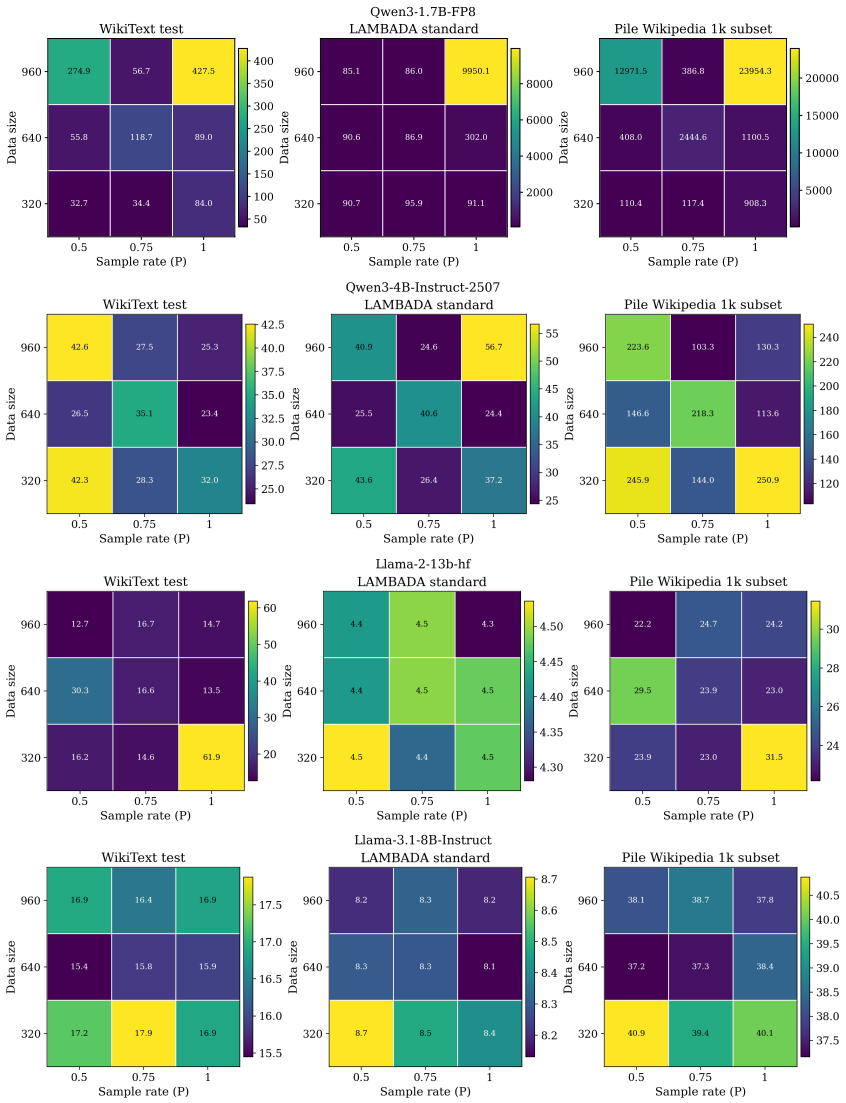

Figure 14: Ablation of Focus Calibration with different calibration data sizes (320, 640, and 960) and sampling rates ($P = 0.5, 0.75, 1$) on the WikiText-2 train set. Perplexity of index-based all-reduce with 2-bit VQ is evaluated on WikiText-2-raw, LAMBADA OpenAI, and a 1K subset of Pile-Wikipedia (Gao et al., 2020), averaged over three independent tests.

We study the sensitivity of Focus Calibration to the sampling rate P and calibration data size across different models and evaluation datasets. As shown in Figure 14, using all available calibration

samples ($P = 1$) does not necessarily yield the best perplexity at a fixed data size. In several cases, layer-wise calibration with random subsampling achieves better results than full-sample calibration. For example, reducing P from 1 to 0.75 substantially improves perplexity across multiple datasets for Qwen3-1.7B-FP8 with 960 samples, Qwen3-4B-Instruct-2507 with 960 samples, and Llama-2-13B-HF with 320 samples. From the other perspective, although the $P = 0.75$ with 960 samples is not universally optimal, this setting consistently provides competitive perplexity across WikiText, LAMBADA, and Pile Wikipedia. These results suggest that Focus Calibration with $P = 0.75$ and 960 samples provides a robust default configuration across different models and data distributions.

A.6 Zero-Shot accuracy comparison

Table 2: Zero-Shot accuracy comparison across multiple tasks

Model	Method	Bits	ARC-c[3]	ARC-e[3]	BoolQ[4]	H.S.[3]	PIQA[3]	W.G.[4]	AVG
Qwen3-1.7B-FP8[2]	Baseline	16	43.1	69.6	77.2	60.4	71.9	60.5	63.8
	TQ-MSE[5]	3	35.4	56.6	74.4	52.8	67.7	56.7	57.3
	TQ-MSE[5]	2	36.2	52.1	58.7	51.8	64.1	53.7	52.8
	MX FP4[6]	4.25	42.2	65.3	74.2	59.4	70.2	58.6	61.7
	MX FP3[6]	3.25	37.3	58.5	70.8	56.0	67.1	57.2	57.8
	MX FP2[6]	2.25	33.3	46.2	51.3	49.7	60.1	52.4	48.8
	VQ (Our)[7]	2.5	38.6	63.2	75.7	55.5	68.8	58.8	60.1
	VQ (Our)[7]	2	33.7	57.6	71.0	50.2	64.7	54.5	55.3
Qwen3-4B-Instruct-2507[1]	Baseline	16	58.5	81.2	84.7	69.1	75.8	67.7	72.8
	TQ-MSE[5]	3	54.4	76.6	84.6	65.7	74.7	64.6	70.1
	TQ-MSE[5]	2	22.8	32.9	38.5	32.6	54.7	52.5	39.0
	MX FP4[6]	4.25	56.7	79.4	83.1	68.4	76.4	67.0	71.8
	MX FP3[6]	3.25	53.5	75.9	80.4	67.1	73.6	64.5	69.2
	MX FP2[6]	2.25	37.9	59.2	61.3	65.9	68.8	58.9	58.7
	VQ (Our)[7]	2.5	55.5	80.3	83.6	65.4	74.4	65.7	70.8
	VQ (Our)[7]	2	48.8	77.3	81.9	60.7	71.3	64.0	67.3
Llama-3.1-8B-Instruct[1]	Baseline	16	55.6	79.9	85.5	79.6	81.3	73.6	75.9
	TQ-MSE[5]	3	48.8	75.4	79.5	75.2	76.4	70.5	71.0
	TQ-MSE[5]	2	25.0	28.1	39.6	26.9	51.0	49.7	36.7
	MX FP4[6]	4.25	54.1	78.0	84.8	78.9	80.6	72.8	74.9
	MX FP3[6]	3.25	50.6	75.5	82.9	77.2	78.6	70.4	72.5
	MX FP2[6]	2.25	38.7	64.2	64.9	66.6	72.8	58.3	60.9
	VQ (Our)[7]	2.5	53.0	78.6	84.6	77.0	79.8	72.0	74.2
	VQ (Our)[7]	2	48.9	76.8	82.5	73.1	77.1	68.3	71.1
Llama-2-13b-hf[2]	Baseline	16	48.8	76.5	82.1	79.6	80.4	72.4	73.3
	TQ-MSE[5]	3	45.1	71.8	76.5	73.6	78.1	67.1	68.7
	TQ-MSE[5]	2	24.7	29.6	58.2	31.6	54.8	49.7	41.4
	MX FP4[6]	4.25	49.1	76.3	80.9	79.4	80.8	72.1	73.1
	MX FP3[6]	3.25	49.1	75.0	80.0	78.2	79.2	69.3	71.8
	MX FP2[6]	2.25	33.3	46.2	51.3	49.7	60.1	52.4	48.8
	VQ (Our)[7]	2.5	47.1	73.9	78.4	77.2	79.7	70.6	71.2
	VQ (Our)[7]	2	44.9	73.7	77.2	75.1	77.5	70.0	69.7

[1] vLLM max model length = 8192. [2] vLLM max model length = 4096. [3] Normalised Accuracy.
[4] Accuracy. [5] TQ-MSE encode activations to 3 bits indices by using $2^3 = 8$ centroids, therefore it has 3 effective bits.
[6] MX FPx uses block size = 32 and 8-bits shared scale (E8M0), therefore it has $x + 8/32$ effective bits.
[7] VQ uses $d = 2$ dimensional sub-vector and 2^k centroids per sub-vector, therefore it has k/d effective bits.

We use (Gao et al., 2024) to evaluate the zero-shot accuracy of four popular models ranging from 1.7B to 13B parameters on ARC (Clark et al., 2018), BoolQ (Clark et al., 2019), PIQA (Bisk et al., 2020), HellaSwag (Zellers et al., 2019) (H.S.), and WinoGrande (Sakaguchi et al., 2021) (W.G.) under different quantisation schemes for compressing all-reduce communication. For VQ, the LUT is generated using the proposed Focus Calibration with $P = 75\%$ on approximately 3% of the WikiText-2 training set (960 out of 36,718 samples), following the ablation study in Appendix A.5. As shown in Table 2, index-based all-reduce with VQ 2.5-bit and Focus Calibration achieves accuracy comparable to, or 1%-3.5% higher than, MX FP4 and MX FP3 across the evaluated models. It also achieves 1%-3% higher accuracy than 3-bit TQ-MSE. Additionally, the VQ 2-bit variant achieves 2.5%-34.4% and 6.5%-20.9% higher accuracy than TQ-MSE 2-bit and MX FP2, respectively, at equivalent bit-widths. These results indicate that Focus Calibration enables index-based all-reduce to substantially reduce the effective bit-width of communicated activations while incurring only negligible to moderate accuracy degradation compared with existing communication-path quantisation methods.