Quantization-Aware Training and Post-Training Quantization Effects on LLM Mathematical Reasoning

Assignee Research

May 31, 2026

Abstract

This report synthesises findings from 4 peer-reviewed papers addressing the following research question: How does quantization-aware training (QAT) impact the reasoning capabilities of large language models (LLMs) on mathematical benchmarks compared to post-training quantization (PTQ) when evaluated on. Post-training quantization (PTQ) of large language models (LLMs) to extremely low bit-widths remains challenging due to the fundamental trade-off between computational efficiency and representational capacity. While existing ultra-low-bit methods rely on binary approximations or. 16 claims were extracted from source literature; 1 was independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 4.5/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: PTQTP: Post-Training Quantization to Trit-Planes for Large Language Models. Research question: How does quantization-aware training (QAT) impact the reasoning capabilities of large language models (LLMs) on mathematical benchmarks compared to post-training quantization (PTQ) when evaluated on MATH or GSM8K?.

2 Methodology

Systematic literature search across multiple databases yielded 4 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 4.5/10.

3 Results

4 papers retrieved. 16 claims extracted; 1 independently verified. Quality review score: 4.5/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
PTQTP was implemented on PyTorch platform using models from Huggingface.	×	0.03
All the weights in linear projection were quantized with tolerance ϵ = 10-4 and maximum iterations Tmax = 50.	×	0.02
Dynamic regularization with λ $\in$[10-8, 1] was employed for numerical stability.	×	0.01
No task-specific calibration, tuning, or fine-tuning was applied in any experiment.	×	0.10
All evaluations were conducted on a single NVIDIA A100 80GB GPU.	×	0.02
PTQTP was evaluated across multiple mainstream LLM families including Qwen3, LLaMA3.x, and LLaMA series.	×	0.03
Instruction-tuned variants of Qwen3, LLaMA3.x, and LLaMA series were tested to assess cross-domain generalization capabi	×	0.01
PTQTP was compared with three categories of methods: extremely low-bit PTQ methods, popular PTQ methods, and 1.58-bit QA	✓	0.18
Common 1B-3B LLMs such as SmolLM2 and MiniCPM were included for fair comparison with smaller models.	×	0.03
Perplexity measurements on WikiText-2 and C4 were used to assess language modeling capability.	×	0.02
Reasoning abilities were evaluated using ARC-Challenge, ARC-Easy, BoolQ, HellaSwag, PIQA, Winogrande, and MMLU.	×	0.02
Coding and mathematical reasoning performance were evaluated on LLaMA3.x and Qwen3 models.	×	0.10
All evaluations were conducted using the standard lm-eval benchmarking tool.	×	0.01
PTQTP consistently outperforms existing extremely low-bit (1-3 bit) quantization schemes and approaches or exceeds 4-bit	×	0.12
PTQTP delivers 4.63$\times$ end-to-end decode speed on NVIDIA RTX 3090 GPU.	×	0.08
PTQTP achieves 73.4$\times$ speedup over AQLM and 1.57$\times$ over AWQ on LLaMA-7B.	×	0.03

References

- http://arxiv.org/abs/2511.15694v1
- http://arxiv.org/abs/2509.16989v3
- http://arxiv.org/abs/2411.06084v1