Impact of LoRA Injection in Attention versus Feed-Forward Networks on Latency-Throughput Trade-offs in Llama-3.2-3B RAG Inference

Assignee Research

June 12, 2026

Abstract

We study quality-latency-resource trade-offs in a documentation-grounded retrieval-augmented generation (RAG) system that uses Low-Rank Adaptation (LoRA) of the generator. We build a manually verified benchmark of 5,144 question-answer pairs over the official Kubernetes documentation and combine it with a fixed hybrid-retrieval pipeline (BGE-M3 dense, BGE-M3 native sparse, Reciprocal Rank Fusion, cross-encoder reranking). Over this benchmark we ablate 20 LoRA configurations on Llama-3.2-3B-Instruct and Llama-3.1-8B-Instruct across rank and target-module choices, and evaluate each on token-leve

1 Introduction

This paper examines: Analyzing Quality-Latency-Resource Trade-offs in a Technical Documentation RAG Assistant Using LoRA Adaptation. Research question: What is the impact of injecting LoRA adapters exclusively into attention mechanisms versus feed-forward networks in Llama-3.2-3B on the latency-throughput trade-off during hybrid-retrieval RAG inference?.

2 Methodology

Systematic literature search across multiple databases yielded 13 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 7.5/10.

3 Results

13 papers retrieved. 10 claims extracted; 8 independently verified. Quality review score: 7.5/10.

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
The primary quality metric is token-level F1 between the generated and the gold answer.	✓	0.22
All headline results and per-regime tables and plots in 7 and Appendix I are computed on the held-out test split (n = 7	✓	0.27
The eval split (n = 745) is used for configuration selection during the experimental loop.	✓	0.20
For paired comparisons of ΔF1 between configurations, a paired bootstrap on the same test split is used.	✓	0.23
Token F1 is the most directly interpretable similarity metric for documentation-grounded question answering with a high	✓	0.27
Two judge-based quality axes are computed by an external LLM judge, gpt-5.4-mini: correctness and groundedness.	✓	0.30
The judge assigns two independent ratings: correctness and groundedness.	✓	0.19
Correctness_pass@4 and groundedness_pass@4 are the fraction of answers that the judge rates $\geq$4 on the corresponding scal	×	0.03
Four cost quantities are measured: mean latency Linf, peak inference VRAM Minf, total training time Ttrain, and peak tra	✓	0.16
A configuration x is Pareto-optimal if there is no other configuration x' that is at least as good on quality and at lea	×	0.04

References

- http://arxiv.org/abs/2606.01947v1

- http://arxiv.org/abs/2605.28222v1
- http://arxiv.org/abs/2304.15010v1