Fine-tuning a Small LLM for Medical Q&A Under Compute Constraints: TinyLlama + LoRA on MedQuAD

AMIR EXIR, University of Texas at Austin, USA

PowerPoint Slides

Code Repository

Fine-tuned Model

Data Source

Video Presentation (Google Drive)

Author's address: Amir Exir, University of Texas at Austin, Austin, TX, USA, amir.exir@utexas.edu.

Abstract. We investigate whether a small open-weight language model can be domain-adapted for medical question answering using budget hardware. Using TinyLlama-1.1B-Chat as the base model, we apply parameter-efficient fine-tuning (PEFT) with Low-Rank Adaptation (LoRA) on the MedQuAD dataset (16K Q–A pairs). We compare pretrained vs. fine-tuned outputs with ROUGE and study the effect of optimizers, learning-rate schedules, warmup, and gradient accumulation. A simple AdamW + warmup recipe with gradient accumulation yields consistent gains: ROUGE-1/2/L/Lsum improve from 0.1067/0.0320/0.0829/0.0827 to 0.1222/0.0448/0.1046/0.1060. We discuss practical lessons for low-resource fine-tuning in healthcare settings.

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of natural language processing (NLP) tasks, including text summarization, dialogue generation, and domain-specific question answering. In the healthcare domain, they offer the potential to assist with patient education, clinical decision support, and the dissemination of reliable medical knowledge. However, deploying such models in resource-constrained environments, such as on student hardware or within smaller research labs, remains challenging due to their high computational and memory requirements.

Full fine-tuning of large models, often containing billions of parameters, typically requires high-end GPUs with substantial VRAM and long training times. This presents a barrier for individuals or organizations without access to such infrastructure. As a result, there has been growing interest in *parameter-efficient* fine-tuning (PEFT) methods, which aim to achieve comparable performance while training only a small subset of model parameters. Among these, Low-Rank Adaptation (LoRA) has emerged as a particularly effective and memory-efficient technique.

In this work, we investigate whether PEFT with LoRA can be successfully applied to a relatively small open-weight model, TinyLlama-1.1B-Chat, for the task of medical question answering. We focus on the MedQuAD dataset, which contains over 16,000 curated Q&A pairs from trusted medical sources. Our goal is to determine whether such a setup—combining a compact model and LoRA—can yield measurable gains without the need for specialized hardware.

To this end, we systematically evaluate the impact of various training hyperparameters, including optimizer choice, learning rate schedule, warmup duration, number of epochs, and gradient accumulation steps. We also compare performance between the pretrained and fine-tuned models using ROUGE scores, and discuss the trade-offs between training cost and model accuracy. Finally, we reflect on practical lessons learned for low-resource fine-tuning in healthcare-related applications, providing guidance for future work in this area.

2 RELATED WORK

Parameter-Efficient Fine-Tuning (PEFT) techniques have emerged as a practical alternative to full model fine-tuning, especially in scenarios where computational resources are limited. Instead of updating all model parameters, PEFT methods train a small subset of parameters or introduce lightweight trainable modules, significantly reducing memory footprint and training time. One of the most widely adopted approaches is Low-Rank Adaptation (LoRA) [2], which injects low-rank decomposition matrices into the transformer architecture and keeps the original model weights frozen. This approach has been shown to achieve competitive performance across various NLP benchmarks while requiring only a fraction of the trainable parameters compared to full fine-tuning [5].

In the healthcare domain, the availability of open medical question answering datasets has been instrumental in advancing research without the need for protected health information (PHI). The MedQuAD dataset [1], for example, contains over 16,000 curated question–answer pairs sourced from trusted medical websites such as the National Institutes of Health (NIH) and National Library of Medicine (NLM). Its clean structure and domain-specific coverage make it a valuable resource for evaluating domain adaptation strategies in medical NLP.

Previous studies on domain adaptation for medical QA have explored a variety of fine-tuning strategies, ranging from full model updates to adapter-based methods and prompt tuning. However, few works have examined the trade-offs between training efficiency and performance when combining PEFT with compact open-weight models, such as TinyLlama, specifically for the healthcare domain. Our work addresses this gap by systematically evaluating LoRA-based fine-tuning on MedQuAD using a small model suitable for consumer-grade hardware.

Optimization strategies also play a critical role in fine-tuning performance. Adaptive optimizers such as AdamW [4] have become a standard choice in modern NLP, offering stability in training deep transformer architectures through decoupled weight decay. Additionally, evaluation metrics such as ROUGE [3] remain widely used for assessing text generation quality, particularly in question answering and summarization tasks. By leveraging these well-established tools, our work builds on proven best practices while focusing on the constraints of low-resource training environments.

3 DATA AND SETUP

Dataset. Our experiments use the MedQuAD corpus [1], a publicly available collection of medical question–answer pairs designed for domain-specific NLP tasks. After cleaning and removing duplicates or incomplete entries, the dataset contains approximately 16,404 high-quality question–answer pairs sourced from reputable medical websites, including those maintained by the National Institutes of Health (NIH) and the National Library of Medicine (NLM). The data covers a broad range of medical conditions, symptoms, treatments, and preventive care topics. We perform a stratified split of 80% for training, 10% for validation, and 10% for testing to ensure balanced coverage across categories.

Model. We adopt `TinyLlama-1.1B-Chat` as the base model due to its balance between capacity and efficiency, making it suitable for constrained VRAM environments such as consumer-grade GPUs. To further reduce memory usage, we leverage `bitsandbytes` for quantized loading in both `int8` and 4-bit precision modes, which enables fine-tuning without exceeding hardware limits.

Tokenization. All inputs are processed using a maximum sequence length of 512 tokens. Sequences shorter than this threshold are padded to the maximum length to ensure uniform batch dimensions, while longer sequences are truncated to avoid overflow during training.

PEFT Configuration. We apply Low-Rank Adaptation (LoRA) with rank $r = 8$, scaling factor $\alpha = 16$, and dropout rate $p = 0.1$. The LoRA adapters are injected into both the attention projection layers and the MLP feed-forward projections, allowing the model to adapt to the medical QA domain while keeping the majority of its parameters frozen.

Trainer. Fine-tuning is conducted using the `TRL SFTTrainer` framework with mixed precision (`fp16` and `bfloat16`), gradient checkpointing, and gradient accumulation steps to emulate larger effective batch sizes. We use the AdamW optimizer [4] with a linear learning rate scheduler and warmup steps to stabilize training.

Figure 1 presents an overview of the experimental pipeline, from dataset preparation and tokenization to LoRA-based fine-tuning and evaluation.

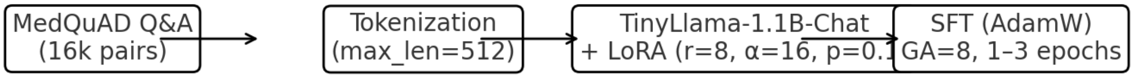

Fig. 1. Workflow: MedQuAD → tokenization → TinyLlama + LoRA → SFT with AdamW and warmup.

4 METHODS

Our approach systematically investigates the impact of key fine-tuning hyperparameters on small-scale domain adaptation using PEFT and LoRA. We conduct an ablation study across the following dimensions:

[label=()]**Optimizer:** We compare `AdamW` [4], a widely used variant of Adam with decoupled weight decay, against `Adafactor` [**?**], which offers reduced memory usage for large models. **Learning rate:** Two learning rate settings are explored: a conservative 5×10^{-5} and a more aggressive 1×10^{-3}, to capture the trade-off between stability and speed of convergence. **Warmup schedule:** We test warmup steps of $\{0, 100, 200\}$ using a linear decay scheduler. Warmup is hypothesized to prevent early-stage instability, especially when fine-tuning with low batch sizes. **Gradient accumulation:** Values of $\{2, 4, 8\}$ accumulation steps are evaluated, enabling the simulation of larger batch sizes without exceeding the VRAM limit. **Number of epochs:** We train for $\{1, 3, 4\}$ full passes over the dataset to assess how additional exposure affects model generalization and overfitting.

Unless otherwise noted, the default configuration uses a per-device batch size of 1, with gradient accumulation to meet effective batch size requirements. Mixed-precision training (`fp16` or `bfloat16`) is employed to reduce memory footprint and accelerate training.

Evaluation. We log stepwise training loss throughout fine-tuning and evaluate on the held-out test set using ROUGE metrics [3] (ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-Lsum). Decoding settings, including temperature, top-p, and maximum generation length, are kept fixed across experiments to ensure comparability.

5 RESULTS

Table 1 reports the best observed configuration versus the pretrained baseline. The strongest, stable gains came from **AdamW + warmup (100)** with **LR=1e-3**, gradient accumulation $\in [2, 8]$, and 3 epochs. This setting improved unigram/bigram overlap and sequence-level alignment while remaining compute friendly. Notably, ROUGE-2 and ROUGE-L gains suggest better phrase-level and sentence-level coherence, indicating that LoRA-based fine-tuning was effective even under strict VRAM limits.

5.1 Ablations and Observations

Optimizer. Adafactor was convenient for memory but underperformed; AdamW consistently stabilized training and improved ROUGE. **Warmup.** Nonzero warmup prevented early loss spikes; 100–200 steps worked best. **Learning rate.** Too low (5e-5) plateaued; too high (5e-3) diverged. Around 1e-3 with warmup was the sweet spot here. **Gradient**

Table 1. ROUGE on the evaluation set. Best setup: AdamW + warmup(100), LR=1e-3, GA=2–8, 3 epochs.

Metric	Pretrained	Fine-tuned (best)
ROUGE-1	0.1067	0.1222
ROUGE-2	0.0320	0.0448
ROUGE-L	0.0829	0.1046
ROUGE-Lsum	0.0827	0.1060

accumulation. Increasing GA improved stability without more VRAM. Returns diminished beyond GA=8. **Epochs.** Going from 1 to 3 helped; >4 brought little benefit and risked overfitting given the small model.

6 DISCUSSION AND LIMITATIONS

Despite measurable ROUGE gains, TinyLlama (1.1B) remains capacity-limited for tasks requiring nuanced clinical reasoning or multi-hop inference. The MedQuAD dataset covers general health Q&A, which may not capture the linguistic and structural characteristics of clinical notes, FDA/CDC bulletins, or patient–provider conversations. Moreover, our evaluation is restricted to automated ROUGE metrics; we did not conduct human expert assessment, error categorization, or bias analysis. No safety audits were performed, and generated outputs *must not* be used for clinical decision-making. Future work could incorporate domain-specific corpora, larger model backbones, and multi-metric evaluation pipelines to better assess factuality, safety, and usability. Nevertheless, our findings demonstrate that small LLMs can be effectively adapted to specialized domains with PEFT, offering a viable path for low-resource teams operating under strict compute budgets.

7 CONCLUSION AND FUTURE WORK

This work demonstrated that a compact, open-source model (TinyLlama-1.1B) can be effectively adapted to a medical Q&A domain using LoRA-based parameter-efficient fine-tuning. With an AdamW optimizer and warmup scheduling, we achieved consistent improvements in ROUGE-1/2/L/Lsum over the pretrained baseline on the MedQuAD dataset, while keeping memory and compute requirements low enough for commodity hardware.

In future work, we aim to:

(1) Incorporate instruction-style prompting and standardized formatting to improve alignment with real-world question–answer workflows.
(2) Integrate retrieval-augmented generation (RAG) over curated and trusted medical sources to enhance factual accuracy.
(3) Expand evaluation beyond ROUGE by including semantic metrics such as BERTScore and human expert ratings for factuality, clarity, and safety.
(4) Experiment with larger, yet still tractable, model backbones (e.g., 3B–8B parameters) using QLoRA to explore the trade-off between capacity and resource usage.

These directions can help bridge the gap between lightweight, resource-efficient models and the reliability standards required for real-world healthcare applications.

ACKNOWLEDGMENTS

This work was completed for AI in Healthcare (AI W395T). Thanks to the course staff and peers for feedback.

REFERENCES

[1] Asma Ben Abacha and Dina Demner-Fushman. 2019. Overview of MedQuAD: a curated multilingual medical question answering dataset. *arXiv preprint arXiv:1909.06209* (2019).

[2] Edward Hu, Yelong Shen, Phil Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*.

[3] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Workshop on Text Summarization Branches Out*.

[4] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. *ICLR* (2019).

[5] Saurabh Mangrulkar et al. 2022. PEFT: Parameter-Efficient Fine-Tuning Library. https://github.com/huggingface/peft.