Are Small Language Models Enough for Biomedical QA Tasks?

Fine-Tuning Mistral-7B Using LoRA and PubMedQA

JAVIER LAMAR LÉON, VITOR BEIRES NOGUEIRA, and PAULO QUARESMA, VISTA Lab, Algoritmi Center, University of Évora, Portugal

This paper presents a specialized fine-tuning approach for the Mistral-7B Large Language Model (LLM) tailored for biomedical applications. We employ Low-Rank Adaptation (LoRA), a parameter-efficient fine-tuning method, to adapt the model to the intricacies of biomedical language and domain-specific knowledge. By integrating LoRA, we aim to preserve the general language understanding capabilities of Mistral-7B while enhancing its performance on biomedical tasks. The fine-tuning process involves training the model on the PubMedQA dataset. Our experiments demonstrate that the fine-tuned Mistral-7B model achieves notable accuracy, 60%. This performance is particularly significant given the relatively modest size of the Mistral-7B model compared to other approaches that often require larger models to achieve comparable results. The results highlight the effectiveness of LoRA in fine-tuning large language models for domain-specific applications, particularly in the biomedical field, where precise and contextually accurate language understanding is crucial. This work contributes to the advancement of AI in healthcare by providing a robust and efficient method for adapting LLMs to biomedical applications, demonstrating that high precision can be achieved with a smaller model size.

CCS Concepts: • **Computing methodologies** → **Natural language processing**; **Machine learning**.

Additional Key Words and Phrases: LLM, Parameter-Efficient Fine-Tuning, LoRA, Biomedical, Question answering

ACM Reference Format:
Javier Lamar Léon, Vitor Beires Nogueira, and Paulo Quaresma. 2024. Are Small Language Models Enough for Biomedical QA Tasks?: Fine-Tuning Mistral-7B Using LoRA and PubMedQA. 1, 1 (December 2024), 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction and Motivation

The rapid evolution of Large Language Models (LLMs) has profoundly transformed Natural Language Processing (NLP), equipping it with unprecedented capabilities for understanding and generating human-like text. Pioneering models such as GPT, Mistral, LLaMA, and Gemini have demonstrated remarkable versatility, addressing complex tasks ranging from open-domain question answering (QA) to creative text generation. These advances have catalyzed the integration of LLMs in various sectors, fundamentally reshaping domains such as education, entertainment, and healthcare. In healthcare, LLMs offer transformative solutions to longstanding challenges. Their deep learning-based ability to process and synthesize vast, intricate datasets, including medical literature, electronic health records, and clinical research, addresses the cognitive limitations imposed by the

Authors' Contact Information: Javier Lamar Léon, jlamarleon@uevora.pt; Vitor Beires Nogueira, vbn@uevora.pt; Paulo Quaresma, pq@uevora.pt, VISTA Lab, Algoritmi Center, University of Évora, Évora, Portugal.

ACM XXXX-XXXX/2024/12-ART
https://doi.org/XXXXXXX.XXXXXXX

sheer volume of information. By mitigating information overload, LLMs empower healthcare professionals to make informed decisions, extract actionable insights, and improve diagnostic accuracy. These capabilities streamline workflows and improve patient outcomes, marking a paradigm shift in healthcare delivery [8].

One particularly impactful application of LLMs in healthcare is biomedical question-answering (QA), which enables practitioners to efficiently navigate extensive biomedical knowledge [12]. Domain-specific adaptations of foundational models, such as BioBERT, ClinicalBERT, Med-PaLM 2, and BioMistral, have demonstrated the ability to handle complexities of medical language, including intricate terminologies and context-dependent meanings. Biomedical QA systems generally fall into three categories: Extractive QA, which pinpoints precise information from structured data; Open Generative QA, which synthesizes nuanced, free-text responses; and Closed Generative QA, which relies solely on pre-trained knowledge for rapid, independent insights. Although Closed Generative QA is efficient for real-time applications, its reliance on static training data necessitates regular updates to maintain accuracy and reliability.

The integration of prompt engineering and fine-tuning techniques offers a promising strategy for optimizing LLMs in biomedical QA systems. Prompt engineering uses task-specific instructions, such as natural language prompts or learned vector representations, to guide model outputs without altering core parameters. Fine-tuning, particularly parameter-efficient fine-tuning (PEFT), complements this by adapting large models to specific tasks with minimal computational overhead. Together, these techniques enhance the contextual relevance and accuracy of LLMs in interpreting clinical queries, synthesizing biomedical knowledge, and delivering precise responses, paving the way for innovation in precision medicine and patient-centered care.

Among fine-tuning approaches, Low-Rank Adaptation (LoRA) has emerged as a pivotal method for addressing the challenges of maintaining accuracy in Closed Generative QA systems. By introducing lightweight, trainable components into pre-trained models, LoRA enables efficient adaptation to domain-specific tasks without the computational burden of retraining the entire model. In biomedical QA, LoRA facilitates the integration of evolving medical guidelines and recent clinical research, ensuring models remain contextually relevant and reliable. This approach mitigates risks associated with outdated knowledge bases while reducing the financial and computational barriers to deploying advanced LLMs in high-stakes healthcare applications.

The contrast between models with over 100 billion parameters and those with fewer than 20 billion parameters highlights the importance of efficiency in resource-constrained settings. While larger models deliver unparalleled performance, their immense hardware requirements often render them impractical for deployment outside of specialized environments. Smaller models, such as the Mistral-7B, offer a viable alternative, balancing performance and accessibility. These models, when enhanced through techniques like LoRA and prompt engineering, achieve domain-specific adaptability without overwhelming hardware limitations, ensuring broader applicability in settings like healthcare.

This paper builds on these advancements by presenting a practical adaptation of the Mistral-7B model for biomedical applications. Through the integration of LoRA and prompt engineering, the model achieves significant improvements in biomedical language processing while preserving its foundational language capabilities. Fine-tuned on the PubMedQA dataset, the Mistral-7B model attains a noteworthy 60% accuracy, showcasing the potential of smaller models to deliver domain-specific performance comparable to their larger counterparts. These findings emphasize the cost-effectiveness and computational feasibility of leveraging lightweight adaptation strategies for advancing AI in healthcare, where precision and contextual accuracy are critical.

The remainder of the paper is organized as follows: Section 2 delves into Parameter-Efficient Fine-Tuning (PEFT) methods, with a focus on Low-Rank Adaptation (LoRA). Section 3 provides an

overview of the PubMedQA dataset, detailing its structure and significance for biomedical research. Section 4 presents the experimental setup and results, showcasing the effectiveness of LoRA in fine-tuning the Mistral-7B model for biomedical applications. Finally, Section 5 concludes the paper, summarizing the key findings and discussing the broader implications of this research for advancing AI in healthcare.

2 Parameter-Efficient Fine-Tuning

Parameter-Efficient Fine-Tuning (PEFT) methods has revolutionized the process of fine-tuning by significantly reducing the computational burden, making it faster and more accessible. By leveraging PEFT, instruction fine-tuning enhances the performance of biomedical LLMs in adhering to task-specific instructions while minimizing resource consumption. This paradigm represents an efficient and cost-effective strategy to align LLMs with biomedical objectives, demonstrating strong capabilities in instruction-following and zero-shot learning with minimal computational overhead. For more details, please consult the review on PEF in [3].

Instruction Fine-Tuning (IFT) involves retraining the model using task-specific instructional data, such as question-answer pairs, to refine its performance for specialized domains. This training process optimizes model responses by minimizing response-content loss, thereby enabling LLMs to excel in understanding and executing domain-specific instructions.

2.1 Low-Rank Adaptation (LoRA)

Low-Rank Adaptation (LoRA) [11] emerges as a pivotal technique for adapting Large Language Models to domain-specific applications. LoRA not only embodies the principles of efficiency seen in PEFT, but also offers unique advantages in scalability and precision. By introducing lightweight trainable matrices to specific layers of pre-trained models, LoRA optimizes task performance while maintaining the core linguistic and contextual understanding of the base model.

Neural networks, particularly those featuring dense layers, rely heavily on matrix multiplications. The weight matrices in these layers are typically full-rank. When adapting pre-trained language models to specific tasks, [1] demonstrated that these models exhibit a low "intrinsic dimension" and can still learn efficiently despite random projections to a smaller subspace. Building on this insight, [11] hypothesized that the updates to the weights during adaptation also possess a low intrinsic rank. For a pre-trained weight matrix $W_0 \in \mathbb{R}^{d\times k}$, we constrain its update by representing it with a low-rank decompositions:

$$W_0 + \Delta W = W_0 + BA$$

where $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, and the rank $r \ll \min(d, k)$. During training, W_0 is frozen and does not receive gradient updates, while A and B contain trainable parameters. Both W_0 and $\Delta W = BA$ are multiplied with the same input, and their respective output vectors are summed coordinate-wise. For an input x, the modified forward pass yields:

$$h = W_0 x + BAx.$$

See Figure 1 for an overview of the process described above.

2.2 Integrating LoRA into Transformer Architectures

In transformers, LoRA is seamlessly incorporated into two primary components: the attention mechanism and the feed-forward network. For attention, LoRA replaces standard weight matrices in the query (q), key (k), value (v), and output projections, enabling these layers to adapt to specialized tasks without requiring extensive re-training of the entire model. Similarly, in the feed-forward network, LoRA is employed in the intermediate and output transformations.

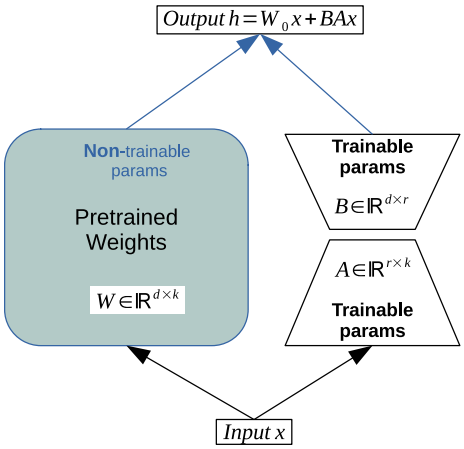

Fig. 1. Overview of the LoRA Process.

This dual integration of LoRA across both attention and feed-forward modules highlights its versatility and impact. By focusing on lightweight adaptations, LoRA ensures that transformer architectures can achieve high performance on downstream tasks while remaining scalable and resource-efficient. This makes it a crucial tool for advancing the deployment of large language models in diverse computational environments.

2.2.1 Attention mechanism. In the attention mechanism, the input tensor $X \in \mathbb{R}^{n \times d_{\text{model}}}$, where n represents the sequence length and d_{model} denotes the model's dimensionality, serves as the common source for generating the query (Q), key (K), and value (V) projected vectors. Due to the concept of self-attention, all three Q, K, and V are derived from the same input tensor, such that $X = q = k = v$. These projections are computed as:

$$Q = W_q X, \quad K = W_k X, \quad V = W_v X, \tag{1}$$

where:

- $W_q, W_k, W_v \in \mathbb{R}^{d_{\text{model}} \times (h \cdot d_{\text{head}})}$,
- h is the number of query heads,
- d_{head} is the dimension of each head.

With LoRA, each weight matrix W is decomposed into a sum of the frozen pre-trained weights ($W_{\text{pretrained}}$) and a low-rank adaptation term (ΔW):

$$W = W_{\text{pretrained}} + \Delta W, \quad \Delta W = BA \tag{2}$$

where:

- $A \in \mathbb{R}^{r \times (h \cdot d_{\text{head}})}$,
- $B \in \mathbb{R}^{d_{\text{model}} \times r}$,
- r is the rank of the low-rank decomposition, with $r \ll \min(d_{\text{model}}, (h \cdot d_{\text{head}}))$.

Thus, for LoRA-enabled projections:

$$\begin{aligned} Q &= W_{q,\text{pretrained}} X + B_q A_q X \\ K &= W_{k,\text{pretrained}} X + B_k A_k X \\ V &= W_{v,\text{pretrained}} X + B_v A_v X \end{aligned} \tag{3}$$

The queries, keys, and values are used to compute attention scores and produce the weighted sum of values. The scaled dot-product attention is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK}{\sqrt{d_{\text{head}}}} + \text{Mask}\right)V \tag{4}$$

where:

- $Q \in \mathbb{R}^{n\times(h\cdot d_{\text{head}})}$,
- $K, V \in \mathbb{R}^{n\times(h_k\cdot d_{\text{head}})}$,
- Mask ensures causal or padding constraints, if applicable.

After computing the attention output, the result is projected back to the original model dimension using the output projection weight W_o:

$$Z = W_o\,\text{Attention}(Q, K, V) \tag{5}$$

where:

- $Z \in \mathbb{R}^{n\times d_{\text{model}}}$,
- $W_o \in \mathbb{R}^{(h\cdot d_{\text{head}})\times d_{\text{model}}}$.

With LoRA applied, the weight matrix W_o is similarly decomposed as:

$$W_o = W_{o,\text{pretrained}} + B_o A_o \tag{6}$$

where:

- $A_o \in \mathbb{R}^{(h\cdot d_{\text{head}})\times r}$,
- $B_o \in \mathbb{R}^{r\times d_{\text{model}}}$.

Thus, the output projection becomes:

$$output = W_{o,\text{pretrained}}Z + B_o A_o Z \tag{7}$$

The decomposition ensures that the output adaptation is concentrated solely on the low-rank components, thereby substantially reducing the number of trainable parameters needed for fine-tuning. This methodology not only minimizes computational overhead but also optimizes hardware utilization and accelerates the training process, making it an efficient solution for fine-tuning large-scale models.

2.2.2 Feed-Forward Network. The forward pass through the layer is defined as:

$$y = W_2\big(\text{SiLU}(W_1X)\odot(W_3X)\big), \tag{8}$$

where:

- $X \in \mathbb{R}^{n\times d_{\text{model}}}$: The input matrix comes from the attention layer (Eq. 7) output after residual connections and layer normalization in the transformer architecture..
- $W_1, W_3 \in \mathbb{R}^{d_{\text{model}}\times d_{\text{hidden}}}$: Project the input X into a higher-dimensional space (d_{hidden}).
- $\text{SiLU}(z) = z\cdot\sigma(z)$: A smooth activation function (Sigmoid-Weighted Linear Unit) applied to $W_1(x)$.
- $W_2 \in \mathbb{R}^{d_{\text{hidden}}\times d_{\text{model}}}$: Projects the result back to the original dimension (d_{model}).
- $\odot$: Element-wise multiplication.

Each weight (W_1, W_2, W_3) is defined through the application of the LoRA decomposition, as outlined in 2.2.1.

3 PubMedQA: A Benchmark for Biomedical Question Answering

PubMedQA [1] is a meticulously curated biomedical question-answering dataset designed to evaluate scientific reasoning over structured abstracts in research articles. The dataset is derived from over 760,000 PubMed [2] articles with question-based titles, of which approximately 120,000 include structured abstracts with sections such as Introduction, Results, and Conclusions. The architecture of this dataset is depicted in figure 2.

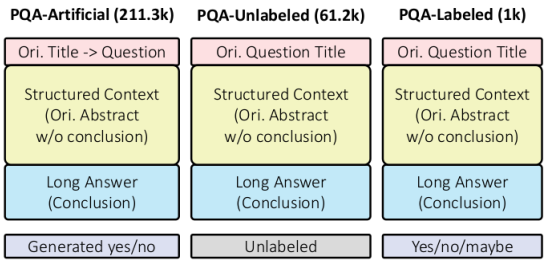

Fig. 2. Architecture of PubMedQA dataset (from [4]).

The dataset is composed of three key components. The first is the **Manually Annotated Subset**, which includes 1,000 articles manually annotated for evaluation and cross-validation purposes. The second is the **Unlabeled Subset**, designed for semi-supervised learning and serving as a valuable resource for training models without explicit labels. Lastly, the **Automatically Generated Instances** consist of 211,300 examples created by converting statement titles into questions and labeling them heuristically.

Unlike other question-answering datasets, where questions and contexts are crowdsourced or artificially paired, PubMedQA offers intrinsic consistency, as both are authored by domain experts. This ensures high accuracy and relevance, making PubMedQA a valuable benchmark for advancing NLP models in biomedical research.

Two primary approaches are used to evaluate PubMedQA. The first incorporates contextual information as model input, extensively studied in recent works (see [3] and [7]). The second excludes context, posing a more challenging task, as highlighted in [9]. These approaches highlight varying levels of difficulty in question-answering tasks with or without context.

4 Low-Rank Adaptation of Mistral with PubMedQA

In this section, we outline experiments conducted to evaluate the performance of LoRA fine-tuning applied to the Mistral 7B Instruct model using the PubMedQA dataset. The Mistral 7B Instruct model, an enhanced version of the Mistral architecture optimized for instruction-following tasks, was fine-tuned using LoRA to adapt its pre-trained parameters efficiently while minimizing computational costs. To ensure accurate alignment with the dataset's format, carefully designed prompts were employed to guide the model's responses. These experiments demonstrate the effectiveness of integrating fine-tuning techniques with prompt engineering to address the challenges of domain-specific question-answering tasks in the biomedical field.

In this approach, we evaluate the PubMedQA dataset, where answers are categorized as "Yes", "No", or "Maybe", using a more challenging approach that omits contextual information. This

[1] https://pubmedqa.github.io/
[2] https://pubmed.ncbi.nlm.nih.gov/
[3] https://pubmedqa.github.io/

evaluation strategy aligns with recent studies, such as [9], which highlight the increased difficulty of question-answering tasks when the model is not provided with contextual cues. By excluding context, the model is tasked with relying solely on the input question and its internal knowledge, making this approach significantly more demanding. This experiment aims to assess the model's ability to generate accurate answers in a more constrained setting, providing valuable insights into its performance under these challenging conditions.

The Mistral 7B Instruct [4] model was chosen for this study due to its state-of-the-art performance and suitability for instruction-following tasks, making it ideal for the PubMedQA dataset. Its transformer-based architecture, optimized for generating accurate and contextually relevant responses, aligns well with the question-answering requirements of the task. Additionally, Mistral's open access, including the pre-trained model and source code on GitHub [5], ensures transparency, reproducibility, and ease of adaptation, supporting further advancements in domain-specific applications.

4.1 Preparing the Dataset

For effective fine-tuning of the Mistral 7B model, the training data must adhere to strict formatting requirements specified by the `mistral-finetune` framework. All data must be stored in the JSONL (JSON Lines) format, with each line representing a separate data sample in valid JSON format.

In this study, we focus exclusively on the *Instruct* data format, which is specifically designed for instruction-following tasks. The data is structured under the key `"messages"`, which holds a list of dictionaries. Each dictionary contains two primary fields: `"content"` and `"role"`. The `"role"` field indicates the participant in the conversation, with possible values of `"user"`, `"assistant"`, or `"system"`. The model is trained (obtaining loss) by using only those entries where the `"role"` is `"assistant"`, as these represent the responses that the model is expected to generate.

Listing 1 presents a sample entry of the *Instruct* data format, where `"user"` entries represent the queries or inputs, while the `"assistant"` entries correspond to the desired model responses. This sequential organization ensures that the model learns to generate appropriate responses based on the preceding user input, facilitating its ability to follow instructions and engage in meaningful dialogue.

Listing 1. Example of Instruct data format

```
{
  "messages": [
    {
      "role": "user",
      "content": "User interaction n. 1 contained in document n.1"
    },
    {
      "role": "assistant",
      "content": "Bot interaction n.1 contained in document n.1"
    }
  ]
}
```

[4] https://models.mistralcdn.com/mistral-7b-v0-3/mistral-7B-Instruct-v0.3.tar
[5] https://github.com/mistralai/mistral-finetune

4.2 Prompt Training

In this section, we introduce the prompt training approach used to fine-tune the Mistral 7B model, following the specific structure designed for instruction-following tasks. The training data is organized using the *Instruct* format (see listing 1). The core objective of this approach is to guide the model in generating contextually appropriate and coherent responses based on the input provided by the user. The prompt training procedure is carefully structured to ensure the model learns how to follow instructions effectively, aligning its output with the format and nature of real-world task-specific queries. This methodology is designed to improve the model's performance on tasks such as question answering, where understanding user queries and generating accurate responses is critical.

Figure 3 illustrates the sequence of steps involved in this training process, demonstrating how the user and assistant interactions are formatted, processed, and utilized for fine-tuning the model. This example is taken from the first sample file, `ori_pqaa_.json`, of the PubMedQA dataset. In Step 1, the red square contains the "question", and the green square represents the "long_answer". In Step 2, the red and green squares show the conditioned prompt (the model's response to the question based on the context), followed by the "final_decision", which is the model's generated answer.

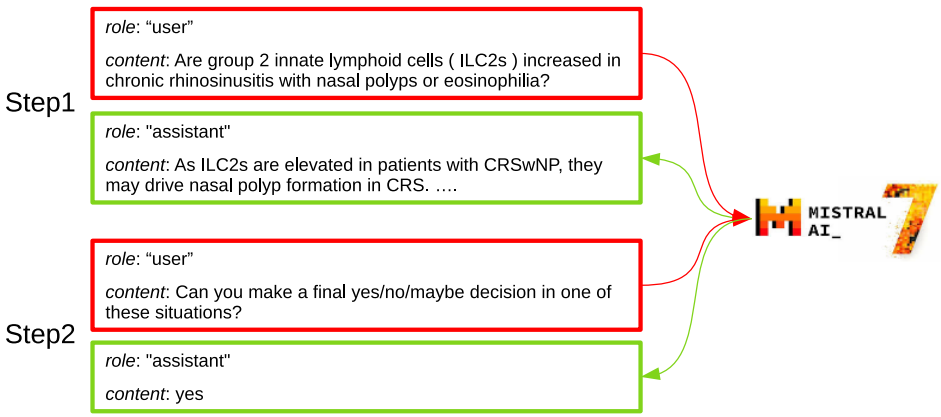

Fig. 3. Training process illustration using a sample from the *ori_pqaa_.json* of the PubMedQA.

4.3 Training Configuration

This section details the parameters and configurations used to fine-tune the Mistral 7B model on the PubMedQA dataset, ensuring efficient training and feasibility.

Training Process Parameters

- **Dataset:** `ori_pqaa.json` file.
- **LoRA Rank:** 128.
- **Sequence Length:** 28,672 tokens per batch.
- **Batch Size:** 1, the number of tokens per batch calculated as `seq_len` × `batch_size`.
- **Learning Rate:** 6×10^{-5}.
- **Optimizer:** AdamW, an adaptive moment estimation optimizer with weight decay.
- **Loss Function:** Cross entropy with masking applied.

Model Parameters Mistral 7B

- **Model Dimension (d_{model}):** 4096.
- **Number of Layers (n_{layers}):** 32.
- **Head Dimension (d_{head}):** 128.
- **Hidden Dimension (d_{hidden}):** 14,336.
- **Number of Attention Heads (n_{heads}):** 32.
- **Number of Key-Value Heads ($n_{\text{kv\_heads}}$):** 8.
- **Vocabulary Size:** 32,768.
- **Normalization Epsilon (ϵ_{norm}):** 1×10^{-5}.
- **Rope Theta:** 1,000,000.0.

Hardware Configuration

The training was performed on an NVIDIA A100 GPU with 40 GB of memory, enabling the handling of long sequences and the computational demands of fine-tuning the Mistral 7B model with LoRA.

4.4 Evaluation

The model's performance was evaluated using the dataset provided in the `ori_pqal.json` file, adhering to the evaluation protocol outlined in the source code repository at (https://github.com/pubmedqa/pubmedqa). As previously outlined, the model under evaluation receives only the question as input. To ensure that the model generates responses in the specific format of `yes`, `no`, or `maybe`, a conditional prompt is appended to the end of each question, such as:

> Can you make a final yes/no/maybe decision in one of these situations?

However, if more detailed or extended responses are desired, this prompt can be omitted, allowing the model to generate a more elaborate answer. This approach offers flexibility in the evaluation process, enabling both concise and comprehensive. Nevertheless, this method of conditioning the model's response through the use of a prompt does not always guarantee that the output will be strictly limited to `yes`, `no`, or `maybe`. However, this approach offers a high degree of reliability, ensuring 100% consistency in the desired response format. By directing the model's output with a well-defined prompt, we significantly increase the likelihood of obtaining responses in the required format, which is essential for the integrity of the evaluation process.

Table 1. Evaluation on the PubMedQA benchmark.

Method	Model Size	Fine-Tuning	PubMedQA (%)
ChatGPT [5]	175B	-	63.90
Mistral 7B LoRA (our approach)	7B	✓	60.00
LlamaCare [9]	13B	✓	57.70
Chat-Doctor [6]	7B	✓	54.30
Med-Alpaca [2]	13B	✓	53.20
LLaMA-2 [10]	13B	-	52.40

Table 1 presents the performance results of our model, highlighting its effectiveness relative to other state-of-the-art approaches. Despite being built with only 7 billion parameters, our model achieves an impressive accuracy of 60%, outperforming many contemporary models. Notably, the only model that surpasses our approach is ChatGPT, with 175 billion parameters, which demonstrates slightly superior performance.

5 Conclusions and Future Work

This study highlights the effectiveness of fine-tuning the Mistral-7B model with LoRA for biomedical applications. LoRA proved to be a cost-efficient method for improving domain-specific performance, achieving 60% accuracy on the PubMedQA dataset—showcasing the potential of smaller models to rival larger ones in precision and contextual understanding.

The results emphasize the value of parameter-efficient fine-tuning in advancing healthcare AI, enabling adaptation to specialized domains with minimal computational resources. This approach contributes to improving diagnostic accuracy, streamlining workflows, and enhancing patient outcomes.

Future efforts will optimize fine-tuning, explore additional datasets, and expand capabilities across broader biomedical applications, fostering continued innovation at the intersection of AI and healthcare.

Acknowledgments

This work is funded by Portuguese National Funds through the FCT - Fundação para a Ciência e Tecnologia, I.P., under the scope of the project 2022.03882.PTDC.

References

[1] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, Online, 7319–7328. https://doi.org/10.18653/v1/2021.acl-long.568

[2] Tianyu Han, Lisa C. Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K. Bressem. 2023. MedAlpaca – An Open-Source Collection of Medical Conversational AI Models and Training Data. arXiv:2304.08247 [cs.CL] https://arxiv.org/abs/2304.08247

[3] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. 2024. Parameter-Efficient Fine-Tuning for Large Models: A Comprehensive Survey. arXiv:2403.14608 [cs.LG] https://arxiv.org/abs/2403.14608

[4] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A Dataset for Biomedical Research Question Answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 2567–2577. https://doi.org/10.18653/v1/D19-1259

[5] Haoyang Li, Jing Zhang, Hanbing Liu, Ju Fan, Xiaokang Zhang, Jun Zhu, Renjie Wei, Hongyan Pan, Cuiping Li, and Hong Chen. 2024. Codes: Towards building open-source language models for text-to-sql. *Proceedings of the ACM on Management of Data* 2, 3 (2024), 1–28.

[6] Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. arXiv:2101.00190 [cs.CL] https://arxiv.org/abs/2101.00190

[7] Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney, Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, and Eric Horvitz. 2023. Can Generalist Foundation Models Outcompete Special-Purpose Tuning? Case Study in Medicine. arXiv:2311.16452 [cs.CL] https://arxiv.org/abs/2311.16452

[8] Jianing Qiu, Kyle Lam, Guohao Li, Amish Acharya, Tien Yin Wong, Ara Darzi, Wu Yuan, and Eric J Topol. 2024. LLM-based agentic systems in medicine and healthcare. *Nature Machine Intelligence* 6, 12 (Dec. 2024), 1418–1420.

[9] Maojun Sun. 2024. LlamaCare: A Large Medical Language Model for Enhancing Healthcare Knowledge Sharing. arXiv:2406.02350 [cs.CL] https://arxiv.org/abs/2406.02350

[10] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian,

Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288 [cs.CL] https://arxiv.org/abs/2307.09288

[11] Shaowen Wang, Linxi Yu, and Jian Li. 2024. LoRA-GA: Low-Rank Adaptation with Gradient Approximation. arXiv:2407.05000 [cs.LG] https://arxiv.org/abs/2407.05000

[12] Hua Yang, Shilong Li, and Teresa Gonçalves. 2024. Enhancing Biomedical Question Answering with Large Language Models. *Information* 15, 8 (2024). https://doi.org/10.3390/info15080494

Received 30 December 2024