machine learning & knowledge extraction

MDPI

Article

Systematic Analysis of Retrieval-Augmented Generation-Based LLMs for Medical Chatbot Applications

Arunabh Bora and Heriberto Cuayáhuitl *

School of Engineering and Physical Sciences, University of Lincoln, Brayford Pool, Lincoln LN6 7TS, Lincolnshire, UK; 27647565@students.lincoln.ac.uk
* Correspondence: hcuayahuitl@lincoln.ac.uk

Abstract: Artificial Intelligence (AI) has the potential to revolutionise the medical and healthcare sectors. AI and related technologies could significantly address some supply-and-demand challenges in the healthcare system, such as medical AI assistants, chatbots and robots. This paper focuses on tailoring LLMs to medical data utilising a Retrieval-Augmented Generation (RAG) database to evaluate their performance in a computationally resource-constrained environment. Existing studies primarily focus on fine-tuning LLMs on medical data, but this paper combines RAG and fine-tuned models and compares them against base models using RAG or only fine-tuning. Open-source LLMs (Flan-T5-Large, LLaMA-2-7B, and Mistral-7B) are fine-tuned using the medical datasets Meadow-MedQA and MedMCQA. Experiments are reported for response generation and multiple-choice question answering. The latter uses two distinct methodologies: Type A, as standard question answering via direct choice selection; and Type B, as language generation and probability confidence score generation of choices available. Results in the medical domain revealed that Fine-tuning and RAG are crucial for improved performance, and that methodology Type A outperforms Type B.

Keywords: large language models (LLMs); medical chatbots; fine-tuning; quantization of LLMs; retrieval-augmented generation (RAG); natural language processing

Citation: Bora, A.; Cuayáhuitl, H. Systematic Analysis of Retrieval-Augmented Generation-Based LLMs for Medical Chatbot Applications. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 2355–2374. https://doi.org/10.3390/make6040116

Academic Editor: Laura Po

Received: 9 September 2024
Revised: 6 October 2024
Accepted: 15 October 2024
Published: 18 October 2024

1. Introduction

Recent advancements in artificial intelligence (AI) suggest a high potential for revolutionising the medical and healthcare sectors. The COVID-19 pandemic accelerated the adoption of AI-driven solutions, particularly in response to shortages of medical workers and the need for efficient healthcare delivery. AI technologies, such as medical AI assistants and chatbots, have emerged as promising tools to address these challenges by offering automated support and improving access to medical information. But effective ways of creating and deploying them remain to be researched and demonstrated.

While AI—especially machine learning—has made significant strides in various industries, its application in healthcare is expected to be particularly impactful. The use of AI in healthcare is not entirely new since early implementations date back to the 1980s with technologies like brain biopsy trajectory mapping [1]. AI's role in healthcare has expanded rapidly since then, with AI systems now being used in hospital logistics, pharmaceutical applications, and specific medical procedures, among others [2].

This paper explores the potential of Retrieval-Augmented Generation (RAG)-based Large Language Models (LLMs) in healthcare, with a focus on their ability to enhance the transparency and reliability of AI-driven medical systems. Unlike traditional LLMs, which generate content based solely on pre-existing training data, RAG models retrieve external data in real-time and integrate it into the content generation process [3]. This retrieval capability allows RAG-based LLMs to leverage vast amounts of medical information from structured databases, enabling more accurate and contextually relevant responses. As a result, RAG systems offer a promising solution to one of AI's greatest challenges in healthcare: the generation of factually correct, reliable, and contextually appropriate information.

Given the complexity of healthcare data and the need for precise information, the application of RAG-based LLMs represents a significant step forward in AI-assisted healthcare technologies. This paper conducts a comprehensive analysis of RAG models, including their design, implementation, and potential for integration into healthcare applications. In doing so, we aim to address a critical gap in the literature: how to effectively deploy LLMs in healthcare environments—where accuracy, trust, and transparency are paramount?

Large Language Models (LLMs) [4,5] and the creation of LLM-related technologies are considered as one of the most significant technological advancements in recent times. To create an LLM, massive amounts of text and data are required to train the model using deep learning technology. This data includes books, internet websites, articles, video transcripts, and other different language-related content. In that way, LLMs can be used to understand the language content directly in the form of chatbots and response generators, which has led to querying internet content more directly and efficiently than previously (based on document retrieval). However, researchers are continuously trying to enhance the quality of LLMs and their applications by handling diverse types of data. The primary differences between Pre-trained language models like BERT, GPT2 and current LLMs like GPT4o, LlaMA3/LlaMA3.1, Mistral, Claude [6–11] are the in-context learning (ICL), instruction following and step-by-step reasoning [2]. ICL refers to the ability to generate an output based on given instructions or demonstrations without requiring additional training or gradient updates. That means the model can learn new things by just following instructions without any additional treatment. The ability of step-by-step reasoning is considered a crucial factor of LLMs. It refers to the Chain-of-Thought (CoT) prompting [12], which helps LLMs solve complex problems. In CoT, instead of asking the question directly, users need to provide some of the pre-requisite knowledge. After that, the model breaks down the problem into smaller steps.

Our approach leverages the inherent strengths of RAG systems to mitigate hallucinations [13], a prevalent issue with LLMs. Hallucination refers to the phenomenon where LLMs generate information that may be factually incorrect or nonsensical, especially when dealing with complex, high-stakes domains like healthcare. To address this challenge, our system retrieves relevant, factual data from external sources—such as trusted medical knowledge bases like the USMLE MedQA books and Gale Encyclopedia—before generating responses. This retrieval step ensures that the generated content is grounded in reliable information, minimising the risk of hallucinations. However, while RAG helps in curbing hallucinations, they are still possible, especially in scenarios where the retrieved documents lack sufficient relevance or depth. A recent study by [14] says that hallucinations may continue to occur in the future, and rather than fully eliminating them, we may need to find ways to manage and coexist with this challenge. Retrieval-Augmented Generation or RAG is an AI technique used for enhancing the accuracy of LLMs and reliability of generated responses with facts fetched from external sources. RAG [15,16] offers several advantages such as the accuracy and relevancy of the answers. In other words, RAG effectively reduces the problem of generating factually incorrect content. Storing data in a vectorised database format is the first step of a RAG system. The collected data for RAG can be in any text format—PDF, HTML, .txt, CSV, or JSON files—which is then converted into a uniform text format through a process called indexing. To optimise the process, text data are segmented into smaller chunks. These chunks are then encoded into a vectorised format using an embedding model and stored in a vector database. When an LLM receives a query, the RAG system employs the same embedding model used in the encoding process to retrieve the context from the vector database. This process is known as retrieval, where the query prompts are transformed into vector representations and similarity scores between the query vector and the vectorised chunks are computed. The system prioritises retrieving the top K chunks that show the highest similarity scores. Using these retrieved chunks, the LLM generates an answer for the given query prompt, a process known as *generation*. Whilst the research and development of RAG is still ongoing, researchers are classifying RAG into three types illustrated in Figure 1: Naive RAG, Advanced RAG, and Modular RAG.

Whilst Naive RAG uses basic retrieval models like BM25, Advanced RAG uses embedding models such as BERT and generalisations of it. Modular RAG allows various retrieval strategies depending on the task [15]. Among all of them, the Advanced RAG type is more popular in chatbot development research [15,17]. Because in a medical chatbot setting, the context-awareness is highly crucial. Medical dialogues often require precise, real-time access to highly specific and detailed information, such as drug interactions, recent research findings, or updated clinical guidelines. The Naïve RAG type retrieves information without much consideration for contextual relevance, and Modular RAG, which decouples retrieval and generation but may lack deep integration between the two. However, the Advanced RAG type provides a more sophisticated communication between retrieval and generation. Advanced RAG is designed to handle such complexity by integrating retrieval deeply into the generation process, continuously refining the relevance of the retrieved documents based on the ongoing interaction. This helps to reduce errors and ensures that responses are up-to-date and medically sound—depending on the quality of external/retrieved data.

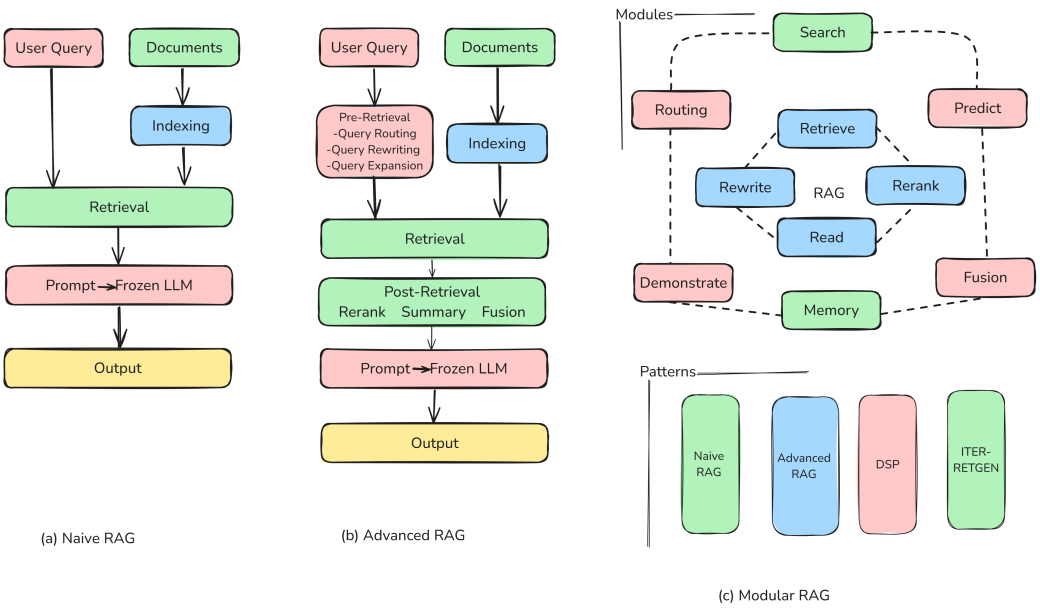


Figure 1. Illustrative comparison of the three RAG paradigms. On the left (**a**), Naive RAG comprises three primary stages: indexing, retrieval, and generation. In the middle (**b**), Advanced RAG introduces several optimisation strategies, both before and after retrieval, while maintaining a similar linear process to Naive RAG, structured in a chain-like sequence. On the right (**c**), Modular RAG builds upon the previous approaches to offer greater flexibility through the incorporation of multiple functional modules and to replace existing ones as needed. Its process is no longer constrained to sequential retrieval and generation but also includes iterative and adaptive retrieval techniques [15].

In light of the increasing importance of LLMs in the development of medical applications, this study offers a focused comparative analysis of RAG-based LLMs tailored for use in computationally resource-constrained environments (for single-GPU computers instead of large servers). Resource-constrained environments are highly relevant to the practical deployment of medical chatbots, particularly in real-world settings where access to high-end computational resources may be limited. In such scenarios, it is crucial to optimise LLMs for efficiency without compromising performance. Unlike existing research, which often concentrates on isolated aspects of LLM performance, our work provides a holistic evaluation that encompasses model selection, architecture classification, and specialised evaluation methodologies. In this study, while we do not provide a direct comparison with readily available traditional medical AI models like *GPT-4* and *Med-PaLM 2* due to their closed nature, we believe that our findings are still generalisable within the context of models available to the research community for continued development and experimenta-

tion. We provide a comprehensive assessment of the effectiveness of RAG-based LLMs in handling both subjective and objective medical queries. Those models can play important roles in medical chatbots or conversational AIs. The key contributions of this paper include:

- A comprehensive analysis of three open-source LLMs (Flan-T5-Large, LLaMA-2-7B, Mistral-7B) and four datasets (MedQA, MedMCQA, Meadow-MedQA, Comprehensive Medical Q&A) examines their performance in medical chatbot applications. Experimental results comparing base models with RAG and fine-tuned models with and without RAG reveal that RAG and Fine-tuning are key for best results.
- A systematic analysis of subjective questions and multiple-choice questions (MCQs). The MCQs are further divided into two methodologies: Type A (similar to NLP question-answering tasks) and Type B (similar to language generation tasks). While *Type A* assesses model accuracy through exact matches between predicted options and reference answers, *Type B* generates text based on the question and context retrieval, measures similarity with all candidate answers, and calculates a probability confidence score to select the best possible answer for performance evaluation. While Type A methodology has shown superior results in our experiments, there is room for further development of the Type B methodology to bridge the performance gap.
- A comparative analysis of fine-tuned LLMs for medical data in resource-constrained environments using techniques such as quantization, PEFT, and LoRA. This involves evaluating their effectiveness in improving model performance and efficiency. These techniques significantly reduce memory usage and inference time, with fine-tuned models using only 5 to 8 GB of GPU RAM compared to higher memory requirements of unoptimised models, highlighting their potential for future work.

2. Literature Review

The rapid advancement of Conversational AI, particularly Large Language Models (LLMs), is profoundly transforming various sectors, including healthcare. These AI systems are becoming increasingly adept at performing both basic and complex tasks, with projections suggesting that within the next 5 to 10 years, AI will play an even more significant role in shaping our daily lives.

In the medical field, the integration of AI is gaining momentum across a range of applications. Research is actively exploring areas such as augmented care, diagnostic imaging, diabetic retinopathy, AI-driven drug discovery, and precision therapeutics [18]. These innovations are not just theoretical; they are gradually being integrated into real-world medical practices, pushing the boundaries of what is possible with AI in healthcare. For instance, the MedMCQA project introduces a novel approach to developing medical chatbots aimed at guiding pre-medical and postgraduate examination aspirants in India [19]. This project leverages a large dataset encompassing 194K multiple-choice questions across 21 medical subjects and utilizes various pre-trained language models. Among these models, PubMedBERT, a biomedical domain-specific model, demonstrated the highest accuracy of 47% after fine-tuning and integrating a retrieval pipeline [20].

The growing interest in AI-driven medical chatbots has also led to focusing on evaluating and optimising LLMs for accuracy and relevance. A recent study [21] provides a comprehensive comparison of LLM evaluation methodologies specifically for medical chatbot applications, highlighting significant research gaps. The study identifies key limitations in current medical chatbot development, such as the need for real-time data from patient-doctor interactions, and lack of awareness regarding medical administrative tasks. Additionally, the study emphasises the need for better approaches to define and quantify bias in LLMs, particularly when applied across different clinical specialities.

In the realm of disease-specific chatbots, recent research has introduced models like "LiVersa", a liver disease-specific chatbot [22]. This RAG-based LLM model was trained on guidelines from the American Association for the Study of Liver Diseases (AASLD) and other specialised documents. The study addresses a common issue with general-purpose chatbots like ChatGPT, which tend to hallucinate or provide unclear and sometimes in-

correct medical information. "LiVersa", however, demonstrated a robust understanding of context and offered accurate and contextually appropriate responses to yes/no medical queries—and only 70% when considering detailed and justified responses. Similarly, another study [23] introduced a RAG-based multilingual chatbot designed for cataract patients. This study recognised the anxiety and uncertainty of patients' feelings regarding their medical conditions, which can negatively impact treatment outcomes. The chatbot, integrated with GPT-4 and a custom knowledge-based dataset, was tested in real-time scenarios using WhatsApp. Results were promising though: out of 343 messages sent to the chatbot, about 70% of the responses received direct approval from an expert team.

Further advancing the field, Google DeepMind introduced Med-PaLM 2, a new medical LLM built upon the PaLM 2 base model [24]. This model was fine-tuned specifically for the medical domain and introduced the concept of ensemble refinement, a novel prompt strategy that significantly enhances the model's reasoning abilities. Med-PaLM 2 achieved an impressive accuracy of 86.5% on the MedQA dataset, outperforming physician-provided answers in terms of medical reasoning. This work sets a strong foundation for future AI developments in healthcare, particularly concerning safety and ethical considerations. Despite these advancements, there remain significant challenges in the adoption of AI-based medical chatbots. As noted by [25], there are currently three primary types of chatbots being developed: therapy, prevention, and diagnosis-focused chatbots. However, the issue of trust remains a major barrier to widespread adoption. Users often hesitate to trust chatbots, partly because these systems struggle to provide clear explanations for their diagnoses and lack of transparently displaying the sources of their information. This lack of transparency can erode trust, making it difficult for users to rely on these tools for critical healthcare decisions. There are some other challenges as well, ref. [26] points out the problems of hallucination: LLMs automatically generate text that is not factually supported, which poses a big risk in making incorrect diagnoses and treatments. Some remedies are based on approaches to enhance model accuracy, reasoning strategies, and using external resources for verification. A further challenge is the lack of evaluation benchmarks tailored to medical-specific metrics like trustworthiness and explainability, emphasising the need for new evaluation standards. Limited domain data further hamper LLM performance, and while synthetic data generation is a solution, it raises concerns with respect to model retention. Lastly, ethical, regulatory, and behavioral alignment challenges require better oversight and alignment with medical professionals' practices.

3. Theoretical Background

3.1. Embedding Model

An embedding model transforms data (words, sentences, images, videos) into dense vectors, which represent semantics in the data. When similar data are provided as input, similar dense vectors are generated, therefore, helping AI systems to generalise better in the case of unseen inputs. Embedding models play a crucial role within Retrieval-Augmented Generation (RAG) systems. Numerous open-source embedding models, available on the Hugging Face platform, for example, are primarily built on the sentence-transformers library [8]. Among these, models such as 'all-mpnet-base-v2', 'all-MiniLM-L6-v2', and 'multi-qa-MiniLM-L6-cos-v1' have been extensively evaluated across various datasets. Google's GTR (Generalizable T5 Retriever) series models represent a significant advancement over traditional embedding approaches [7]. These models employ a fixed-size bottleneck embedding technique, which compresses embeddings into a uniform size, enhancing computational efficiency. GTR models tend to excel in out-of-domain generalisation. As noted in [7], the largest model in the series, GTR-XXL, demonstrated superior performance compared to other models in similar tasks.

3.2. LLMs

LLMs are fundamentally built upon neural embedding models, leveraging these underlying architectures to process and comprehend text more effectively. This paper

evaluates several open-source LLMs based on various criteria, with a particular focus on optimising a RAG-based LLM system. The selection and effective utilisation of LLMs are of importance to potentially improve their performance. The open-source models selected for the study reported in this paper include "T5" by Google, "LLaMA-2-7B" by Meta, and "Mistral-7B" by Mistral AI; see Table 1. Unlike proprietary models like *GPT-4* and *Med-PaLM 2*, the models in this study are fully open-source, increasing accessibility and enabling wider collaboration and development.

Table 1. Brief technical specifications of employed LLMs in this paper.

Feature	Flan-T5-Large	LLaMA2-7B	Mistral-7B
Architecture	Transformer (Encoder-Decoder)	Transformer (Decoder)	Transformer (Decoder)
Model Parameters	780 million	7 billion	7 billion
Training Data	Massive text corpus	Publicly Available Data	Publicly Available Data
Training Objective	Text-to-Text	Autoregressive Language Modelling	Autoregressive Language Modelling
Use Cases	Translation, Summarisation, Q&A	Text Generation, Summarisation, Code Generation	Text Generation, Summarisation
Inference Speed	Fast	Moderate to High	Moderate to High
Memory Requirements	Moderate to High	High	High
Open-Source	Yes	Yes	Yes
Quantization	Yes	Yes	Yes
Licensing	Apache 2.0	Apache 2.0	Apache 2.0

T5, developed by Google [27], is based on a Transformer architecture employing an encoder-decoder framework. It processes text using a text-to-text approach, enabling it to perform a wide array of NLP tasks such as translation, summarisation, and question-answering. While "T5" is a widely recognised language model, it demands significant computational resources. To address that, this paper utilises the "Flan-T5-Large" model [28], an enhanced and fine-tuned version of "T5" that offers improved efficiency.

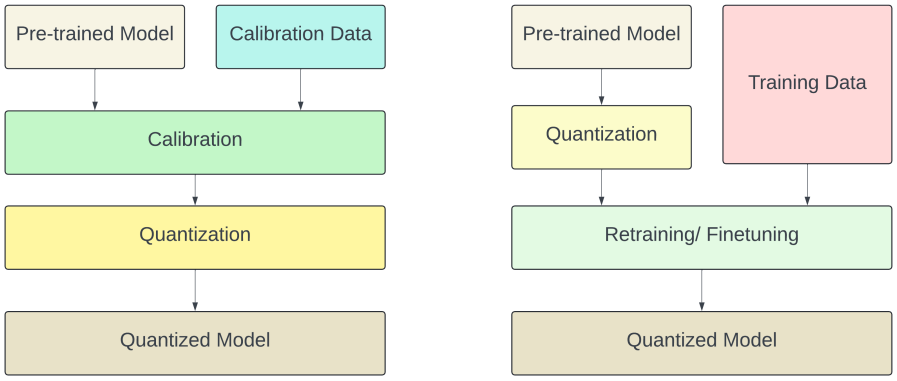
LLaMA-2-7B, developed by Meta [29], is an updated version of "LLaMA-1". The "LLaMA-2" model is available in several parameter sizes, including 7B, 13B, and 70B (B = billion parameters). This paper specifically employs the 7B variant for experimental purposes. "LLaMA-2-7B" is based on a transformer decoder architecture, with its primary training objective being autoregressive language modelling. "LLaMA" models are known for their efficiency in text generation, summarisation, and code generation. Architecturally, "LLaMA" models incorporate a pre-normalisation variant of RMSNorm, enhancing their stability and performance.

Mistral-7B, designed by the Mistral AI team [6], also utilises a Transformer-based architecture and it claims to outperform "LLaMA-2-13B". Despite its similarities with "LLaMA", "Mistral-7B" introduces several distinct architectural innovations, including Sliding Window Attention (SWA), a rolling buffer cache, and a pre-fill and chunking mechanism. SWA allows the model to attend information beyond a fixed window size by leveraging stacked Transformer layers. The rolling buffer cache reduces and fixes cache memory size, enabling the model to incorporate new data without sacrificing quality. The pre-fill and chunking method divides longer prompts into manageable chunks, allowing the model to efficiently apply attention masks over both the cache and the chunks.

3.3. Quantization of LLMs

In the field of machine learning, Quantization refers to the process of compressing deep learning models by mapping high-precision values to lower-precision equivalents, to make a model more efficient in terms of memory and computations but possibly sacrificing performance. LLMs are typically trained using full-precision (float32) or half-precision (float16) floating-point numbers. For example, a model trained with float16 precision and containing one billion parameters would require approximately two gigabytes of VRAM or GPU memory to store its parameters [30]. Quantization methods are broadly categorised into two types: Post-Training Quantization (PTQ) and Quantization-Aware Training (QAT); see Figure 2. In PTQ, Quantization is applied after the model has been trained, i.e., during

the inference phase. Conversely, QAT integrates Quantization throughout the training process. By simulating quantization effects during training, QAT enables the model to learn to handle quantized noise, often resulting in more effective performance compared to PTQ.



(a) Post-Training Quantization (PTQ) **(b) Quantization-Aware Training (QAT)**

Figure 2. Schematic diagram of PTQ and QAT quantization techniques [17].

3.4. PEFT and LoRA

There are several other techniques available for fine-tuning, particularly designed for low-cost memory utilisation. Parameter-Efficient Fine-Tuning (PEFT) techniques are used to fine-tune large pre-trained models with a relatively small number of additional parameters. Low-Rank Adaptation (LoRA) is a specific PEFT technique designed to adapt pre-trained models efficiently by inserting low-rank matrices into the model.

In LoRA [31], let the parameter matrices be $A \in \mathbb{R}^{r \times i}$ and $B \in \mathbb{R}^{o \times i}$, where i and o refer to the original input and output dimensions of the weight matrix. Here, $r \ll i, o$ is the rank of the LoRA matrices, and α is a constant that adjusts the LoRA parameters.

Assuming $W_o \in \mathbb{R}^{o \times i}$ is the original frozen weight matrix, for a given input activation $X \in \mathbb{R}^{i \times s \times b}$, where i is the input dimension, s is the sequence length, and b is the batch size, the output can be represented as follows:

$$Y = (W_o + \alpha BA)X = W_o X + \alpha BAX, \quad \text{where } Y \in \mathbb{R}^{o \times s \times b}. \tag{1}$$

The gradient loss L is then given by the following:

$$\frac{\partial L}{\partial A} = \alpha \frac{\partial L}{\partial \tilde{X}} X^T, \quad \frac{\partial L}{\partial B} = \frac{\partial L}{\partial Y} \tilde{X}^T, \tag{2}$$

where $\tilde{X} := AX$ is the intermediate input activation of B.

Although both LoRA and PEFT are known to be effective in fine-tuning LLMs, recent research has introduced quantization-aware PEFT approaches, such as QLoRA [32] and QA-LoRA [33]. This paper utilises a hybrid method known as QAT-LoRA, which integrates both Quantization-Aware Training (QAT) and LoRA PEFT approaches. In past studies, such as in paper [34], LoRA has been used to compare full-parameter fine-tuning approaches for medical LLMs like Med42, demonstrating its computational efficiency without significantly compromising performance. In our study, we similarly applied these techniques to improve the computational efficiency of the LLM while maintaining its performance in medical domain question-answering tasks.

4. Methodology

Our research methodology is divided into the following parts: medical data retrieval, LLM fine-tuning, vector database development, and a testing framework for evaluating three types of systems illustrated in Figure 3: (a) Base Model with RAG, (b) Fine-tuned

Model Without RAG, and (c) Fine-tuned Model with RAG. Be mindful that models (b) and (c) will be further divided based on the fine-tuning training dataset.

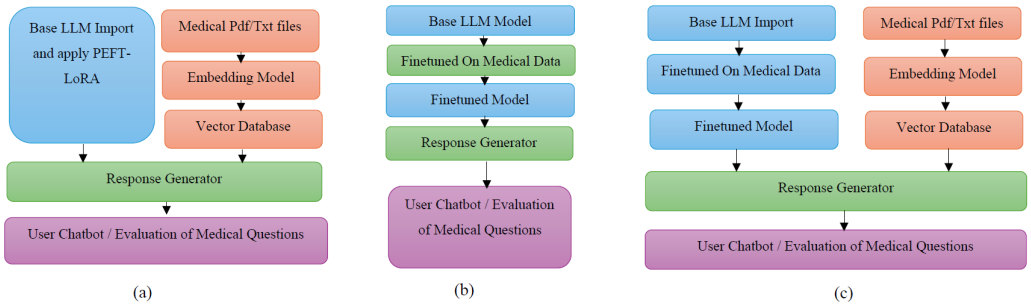


Figure 3. Schematic diagram of system types: (**a**) Base Model with RAG, (**b**) Fine-tuned Model Without RAG, and (**c**) Fine-tuned Model with RAG.

4.1. Medical Data Retrieval for Fine-Tuning, RAG Database and Evaluation

This research utilises a diverse range of medical datasets for fine-tuning and evaluation. The following datasets are employed for LLM fine-tuning: Meadow-MedQA [35] dataset from Huggingface containing 10,178 training rows and the MedMCQA [19] dataset from Kaggle with 194K multiple-choice questions, 2.4K healthcare topics, and 21 medical subjects. The justification for choosing Meadow-MedQA and MedMCQA for fine-tuning lies in their ability to provide a diverse and comprehensive testbed for evaluating medical decision-making capabilities. Unlike other medical datasets, such as PubMedQA [36], which primarily focuses on simple yes/no/maybe questions derived from PubMed abstracts and is more suitable for factual retrieval in clinical research, Meadow-MedQA and MedMCQA offer greater complexity and variety in question formats.

Our RAG system is constructed using an extensive collection of medical books and journals, including 18 English medical note collections from the USMLE MedQA [24] dataset, the open-source PDF of 'Gale Encyclopedia of Medicine' [Second Edition], the 'Current Essentials of Medicine' book [Fourth Edition], and three open-access journals.

For model evaluation, two dataset types are used: multiple-choice question (MCQ) data and medical knowledge-based subjective question data. The USMLE MedQA English test [24] dataset is used for MCQ evaluation, and the Comprehensive Medical Q&A [37] dataset is used for subjective question evaluation. The models are assessed using the first 100 MCQs and the first 20 subjective questions from these datasets—for reasons of high compute requirements. Figure 4 shows the different datasets used in the paper.

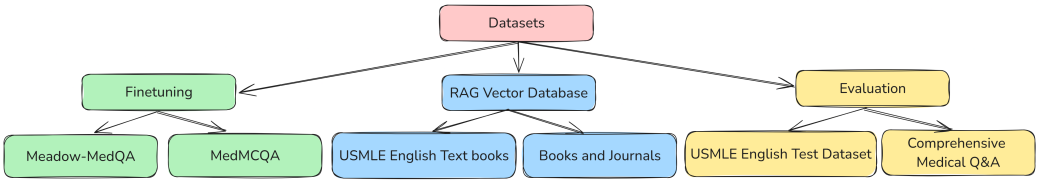


Figure 4. Various types of datasets and their uses in this paper.

4.2. Fine-Tuning LLMs

This research incorporates three LLMs for medical chatbot development and comparative analysis: Google's Flan-T5-Large, Meta's LLaMA-2-7B, and Mistral's Mistral-7B. To fine-tune these models efficiently in a computationally resource-constrained environment, deep learning techniques such as LoRA, PEFT, and quantization are employed.

The fine-tuning process follows a structured algorithmic flow as shown in Figure 5. It begins by importing the necessary modules and the LLM base model. The base model is then quantised following Post-Training quantization (PTQ) rules, preparing it for k-bit training. Next, the model is configured with LoRA. Datasets are loaded either directly from Hugging Face or from pre-downloaded sources. A pre-processing function is defined,

which is essential for correctly loading the dataset. This involves visual inspection and may include data analysis techniques. The pre-processing function consists of three main steps: (a) Input Data Extraction, (b) Formatting Inputs and Labels, and (c) tokenisation. Following tokenisation, the model is trained using a trainer configured with relevant training arguments, including all necessary hyperparameters for optimised fine-tuning. After training, the fine-tuned model and tokeniser are saved for evaluation or further use.

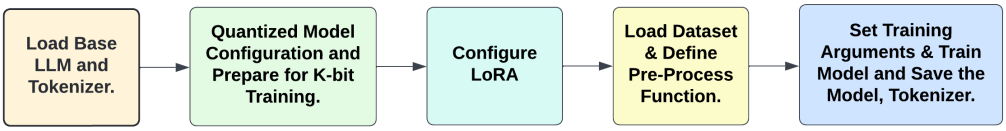


Figure 5. Block diagram illustrating the fine-tuning methodology (The colors are purely for visual reference and do not hold any analytical significance).

4.3. Vector Database for RAG

In natural language processing (NLP), encoding and decoding are essential techniques for converting text into numerical representations. Encoding transforms textual data into numerical vectors, utilising various techniques such as one-hot encoding, word embeddings, and contextual embeddings, depending on the specific application. Decoding is the reverse process, which converts numerical vectors back into human-readable text.

The creation of a vector database is an encoding process. Using an embedding model, text from PDFs and other files is converted into vectorised data. We employ the `gtr-t5-large` embedding model for encoding text data and the following pipeline:

- Document Loading: The documents are first loaded into the system.
- Text Chunking: The text is then divided into smaller, manageable chunks.
- Encoding: These chunks are passed through the embedding model, where they are transformed into vector data.
- Indexing: The generated vector data are stored as indexes in the vector database.

4.4. Testing Framework for Evaluation

4.4.1. Performance Metrics

Evaluating the effectiveness of language models in NLP is crucial, with several traditional metrics being commonly used. These metrics are generally divided into three categories: (a) Multi-Classification, (b) Token-Similarities, and (c) Question-Answering [38]. This paper employs two types of metrics: Token-Similarities and Question-Answering. The former category includes BERTScore [9], BLEU Score [39], ROUGE Score [40], METEOR Score [41], and Perplexity [38], while the later category includes Exact Match (EM), Mean Reciprocal Rank (MRR), and Lenient Accuracy (LaCC) [38].

4.4.2. Evaluation of Subjective Questions

For subjective questions, the methodology employed in this research uses auto-regressive response generation, i.e., generated word by word given previous outputs. In addition, our methodology utilises the few-shot prompting technique to enhance answer generation performance [26]. A pipeline system for handling questions and answers from a dataset is illustrated in Figure 6. After processing a predefined number of questions, responses are saved in a text file along with the ground truths and corresponding questions. Subsequently, separate code is used for performance evaluation—using Token-Similarity metrics—by comparing the generated answers to the reference texts and ground truths.

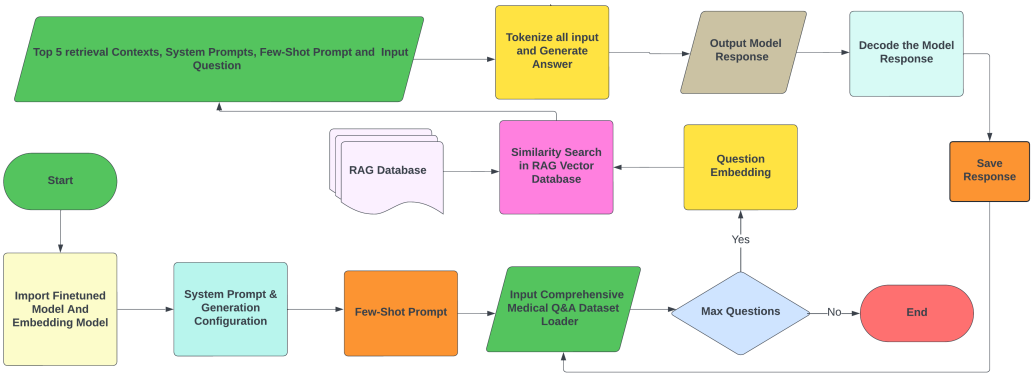


Figure 6. Subjective question evaluation flow-Chart, for fine-tuned model with RAG classification. (The colors are purely for visual reference and do not hold any analytical significance).

4.4.3. Evaluation of Multiple-Choice Question Answering

For evaluating Multiple-Choice Questions (MCQs), this research follows two methodologies: (Type A), based on a typical NLP question-answering task, and (Type B), which experiments with Large Language Model (LLM) capabilities for generating probable textual answers and calculating probability confidence scores for each option (candidate answer).

- Type A Methodology: Answers are generated according to an NLP question-answering task based on the context retrieval, system prompt, few-shot prompt, question, and reference options (candidate answers). For few-shot prompting, the model is provided with a small number of examples related to the input prompt. In this way, the Few-Shot prompt is provided to enhance the model's efficiency in answer selection and to avoid unnecessary text generation. Figure 7 illustrates this methodology. For computational reasons, only the first 100 MCQs from the USMLE Test Data are used for the evaluation. The process generates answers for these questions, saving responses, questions, ground truth, exact matches of answers, and generation times in a text file. Typically, LLMs generate text one token at a time, and for each token, they calculate a probability distribution. In our methodology, since the model takes first the input question, system prompt, and retrieved context, the probability of that sequence can be expressed as follows:

$$P(Y|X) = \prod_{t=1}^{T} P(y_t|y_{<t}, X), \tag{3}$$

  where $P(y_t|y_{<t}, X)$ is the probability of token y_t given all previous tokens $y_{<t}$, Y is the output word sequence, and X includes all inputs. When candidate answers (options) are provided, the model generates scores for each option, selecting the one with the highest score [42] according to

$$\text{Selected Option} = \arg\max_{O_i} S(O_i), \tag{4}$$

  where $S(O_i) = \sum_{t=1}^{T_i} \log P(o_{i,t}|o_{i,<t}, X)$ represents the option's log probability.
- Type B Methodology: It is designed to experiment with textual answer generation, while simultaneously calculating confidence scores for each option. This method generates text based on the input question and retrieved context without any few-shot prompts or candidate options. A function converts the generated answer and all reference options into vector representations using an embedding model, calculating the cosine similarity between each option and the generated answer as

$$\text{Similarity}(v_g, v_o) = \frac{v_g \cdot v_{o_i}}{\|v_g\| \|v_{o_i}\|}, \tag{5}$$

  where v_g is the vector representation of the generated answer, and v_{o_i} is the vector representation of option i.

Logits and probability are distinct mathematical concepts. While probabilities represent the likelihood of an event occurring, logits are raw, unnormalised scores that indicate how confident the model is in one outcome over another. In this methodology illustrated in Figure 8, for any given question, the mean logits score for every option is calculated. Then, mean logits are converted into probabilities using the 'Sigmoid' function: $P(o_i) = \sigma(l_i) = \frac{1}{1+\exp(-l_i)}$, where l_i is the mean logit score.

Last but not least, mean reciprocal rank (MRR) is calculated according to $\text{MRR} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{\text{rank}_q}$, and Lenient Accuracy with Cutoff at $k = 1$ and $k = 3$ (LaCC) according to $\text{LaCC} = \frac{\text{Number of queries where rank}_q \leq k}{Q}$, where Q is the total number of questions, and k is the cutoff rank, set to 1 and 3.

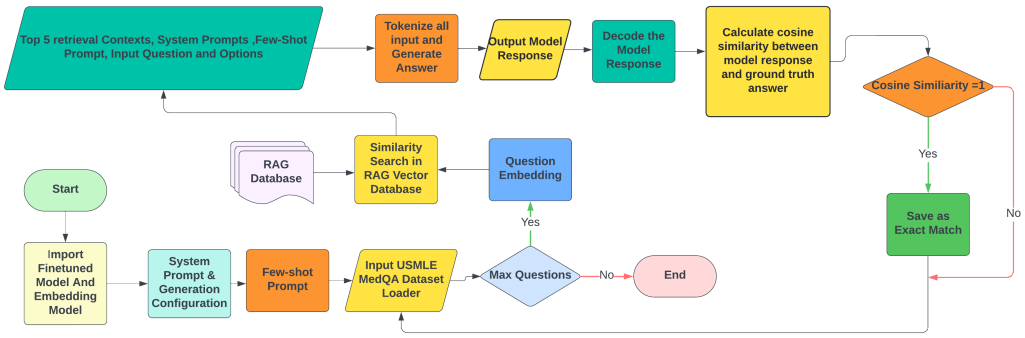


Figure 7. Type A MCQ evaluation flow-chart—for fine-tuned model with RAG. (The colors are purely for visual reference and do not hold any analytical significance).

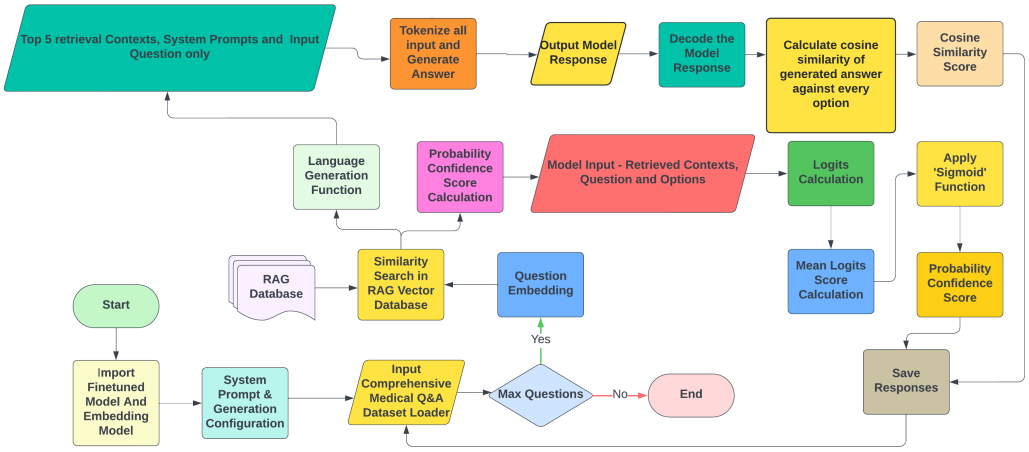


Figure 8. Type B MCQ evaluation flow-chart—for fine-tuned model with RAG. (The colors are purely for visual reference and do not hold any analytical significance).

4.5. Implementation

The procedures and software implementation for this research are publicly available in our GitHub repository. It contains the codebase and detailed documentation for the various components, including data preprocessing, fine-tuning of LLMs, the development of the RAG vector database, and the evaluation process. The code is designed to be easily reproducible, utilising widely available open-source frameworks and tools such as PyTorch, LangChain, and FAISS. The training and testing of models were conducted on a machine with the following specifications: CPU: Intel i7-6950 @ 3.00 GHz, 10 cores; RAM: 32 GB; GPU: NVIDIA TITAN X 12 GB. For further details and access to the implementation, please see Appendixes A.1, A.2, A.3, A.4 and visit the following link: https://github.com/arunabh-alt/Comparative-Analysis-of-RAG-based-LLMs-for-Medical-Chatbot.git (accessed on 5 October 2024).

5. Results and Discussion

5.1. Subjective Question Evaluation

The process involved using the 'Comprehensive Medical Q&A' dataset to generate answers for the first 20 subjective questions. These questions—unseen from training data containing complex medical terminology—were selected to evaluate the model's performance in handling challenging medical concepts. The evaluation results are summarised in Tables 2 and 3, which detail the models' performance on subjective question evaluation.

Table 2. Results of subjective questions using Meadow-MedQA data, where FT = Fine-tuned and bold fonts indicate best performing values.

Metrics	Classification	Flan-T5-Large	Llama-2-7B	Mistral-7B
BERT F1	Base with RAG	0.071	0.068	0.181
	FT without RAG	0.063	0.16	0.17
	FT with RAG	0.066	**0.233**	0.221
ROUGE-1	Base with RAG	0.162	0.232	**0.3456**
	FT without RAG	0.153	0.339	0.339
	FT with RAG	0.136	0.284	0.308
ROUGE-L	Base with RAG	0.133	0.168	0.2512
	FT without RAG	0.129	**0.254**	0.246
	FT with RAG	0.113	0.23	0.221
BLEU	Base with RAG	0.01	0.05	**0.127**
	FT without RAG	0.02	0.119	0.103
	FT with RAG	0.019	0.095	0.091
METEOR	Base with RAG	0.086	0.16	0.271
	FT without RAG	0.084	**0.259**	0.238
	FT with RAG	0.077	0.219	**0.288**
Perplexity	Base with RAG	**4.0188**	7.967	6.4691
	FT without RAG	7.822	8.393	6.9795
	FT with RAG	12.592	7.797	4.84
Avg. Time p/Q (s)	Base with RAG	17.6975	8.128	78.5243
	FT without RAG	**6.8529**	33.7098	56.7355
	FT with RAG	13.0491	116.372	150.658

Table 3. Results of subjective questions using MedMCQA data, where FT = Fine-tuned and bold fonts indicate best performing values.

Metrics	Classification	Flan-T5-Large	Llama-2-7B	Mistral-7B
BERT F1	Base with RAG	0.071	0.068	**0.181**
	FT without RAG	0.072	0.039	0.113
	FT with RAG	0.074	0.152	0.073
ROUGE-1	Base with RAG	0.162	0.232	**0.3456**
	FT without RAG	0.166	0.204	0.292
	FT with RAG	0.159	0.295	0.31
ROUGE-L	Base with RAG	0.133	0.168	**0.2512**
	FT without RAG	0.1376	0.15	0.222
	FT with RAG	0.13	0.217	0.181
BLEU	Base with RAG	0.01	0.05	**0.127**
	FT without RAG	0.011	0.049	0.091
	FT with RAG	0.01	0.085	0.05
METEOR	Base with RAG	0.086	0.16	0.271
	FT without RAG	0.09	0.146	0.21
	FT with RAG	0.09	0.208	**0.278**
Perplexity	Base with RAG	**4.0188**	7.967	6.4691
	FT without RAG	5.823	6.99	5.95
	FT with RAG	8.67	7.088	6.253
Avg. Time p/Q (s)	Base with RAG	17.6975	**8.128**	78.5243
	FT without RAG	9.4703	151.56	81.871
	FT with RAG	9.8312	123.47	54.715

The performance metrics across both tables highlight key differences. For BERT F1, Llama-2-7B leads after fine-tuning with RAG (0.233 on Meadow-MedQA, 0.152 on MedMCQA), outperforming Mistral-7B, which performs better in base settings. ROUGE-1 and ROUGE-L also favour Llama-2-7B post-fine-tuning, demonstrating strong word overlap and sentence structure retention. In BLEU, Llama-2-7B excels with fine-tuning without RAG (0.119 on Meadow-MedQA, 0.085 on MedMCQA), while METEOR favours Llama-2-7B on Meadow-MedQA but Mistral-7B on MedMCQA with RAG (0.271).

Further results reveal that Mistral-7B exhibits low perplexity in the fine-tuned setting with RAG, especially on Meadow-MedQA (4.84), indicating more fluent text generation than Llama-2-7B. In contrast, average time per question shows a trade-off between performance and speed, where Llama-2-7B's fine-tuned models take significantly longer to process each question (116.372 s on Meadow-MedQA and 123.47 s on MedMCQA), whereas Mistral-7B is faster after fine-tuning with RAG on MedMCQA (54.715 s). Thus, while Llama-2-7B demonstrates strong performance across most metrics, Mistral-7B's efficiency and perplexity in certain conditions highlight its strength in balancing accuracy with processing speed.

5.2. MCQ Evaluation

We utilised two distinct methodologies for conducting the MCQ evaluations. The first, Type A Methodology, was applied across all model classifications. The second, Type B Methodology, was an exploratory approach specifically applied to the 'Finetuned Model with RAG', for comparing the best models of Type A methodology against others. For the evaluations, the first 100 MCQs were selected from the USMLE English Test dataset.

5.2.1. Type A Methodology

This methodology evaluated model performance by calculating (1) exact match accuracy and (2) average time per question, providing a standardised comparison across all model classifications. Table 4 and Figure 9 report that fine-tuning with RAG achieves the best performance in terms of exact match followed by fine-tuning without RAG.

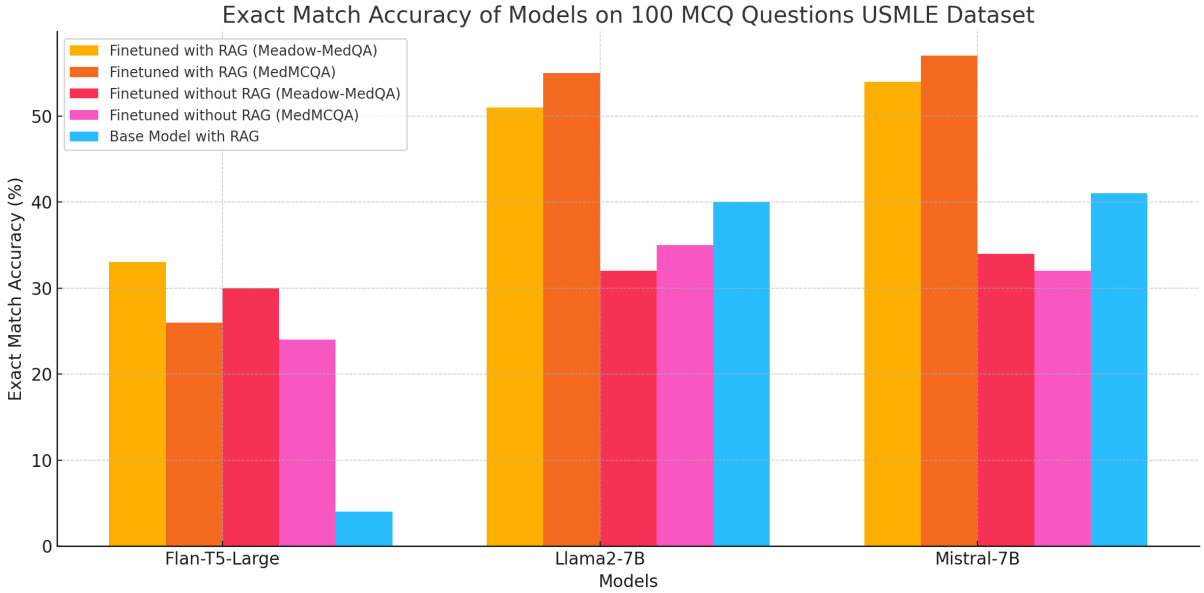


Figure 9. Exact Match accuracy of different models and their classifications.

Table 4. Model performance on Type A MCQ evaluation—bold numbers indicate best performance.

Type	Fine-Tuning Dataset	Model	Exact Match Accuracy	Avg. Time p/Question (In Seconds)
Base with RAG	-	Flan-T5-Large	4%	**3.0446**
	-	Llama-2-7B	40%	13.4805
	-	Mistral-7B	**41%**	18.5270
FT without RAG	Meadow-MedQA	Flan-T5-Large	30%	0.6309
	MedMCQA	Flan-T5-Large	24%	**0.2713**
	Meadow-MedQA	Llama-2-7B	32%	10.4805
	MedMCQA	Llama-2-7B	**35%**	10.5743
	Meadow-MedQA	Mistral-7B	34%	15.3815
	MedMCQA	Mistral-7B	32%	11.1937
FT with RAG	Meadow-MedQA	Flan-T5-Large	33%	4.4675
	MedMCQA	Flan-T5-Large	26%	**0.3956**
	Meadow-MedQA	Llama-2-7B	51%	14.1219
	MedMCQA	Llama-2-7B	55%	14.2405
	Meadow-MedQA	Mistral-7B	54%	14.1282
	MedMCQA	Mistral-7B	**57%**	12.2159

5.2.2. Type B Methodology

Type B evaluated MCQ performance as a language generation task, employing a distinct approach to answer generation and probability confidence score calculation. The evaluation process employed ranking-based metrics—Mean Reciprocal Rank (MRR) and LaCC (k = 1 and k = 3). These metrics were specifically designed to assess the model's ability to rank relevant information accurately. It should be noted that this methodology was applied only to 'Fine-tuned with RAG' models. Table 5 on the one hand reports that Mistral-7B is the best LLM across metrics, which agrees with the exact match accuracy reported by Type A methodology. On the other hand, since exact match accuracy and LaCC (k = 1) are equivalent, that reveals Type A methodology outperforms its counterpart Type B.

Table 5. Results of Type B Methodology—bold numbers indicate best performance.

Model	Fine-tuning Training Dataset	Mean Reciprocal Rank (MRR)	LaCC (K = 1)	LaCC (k = 3)
Flan-T5-Large	Meadow-MedQA	0.44	21.42%	61.05%
	MedMCQA	0.43	18.6%	56.84%
Llama-2-7B	Meadow-MedQA	0.44	22.8%	58.90%
	MedMCQA	0.47	26.2%	61.80%
Mistral-7B	Meadow-MedQA	0.51	30.7%	67.74%
	MedMCQA	**0.53**	**32.8**%	**70.83**%

5.3. Discussion

The comparative analysis of LLMs Flan-T5-Large, Llama-2-7B, and Mistral-7B across various evaluation tasks demonstrates clear distinctions in their strengths and weaknesses. Among them, Mistral-7B consistently outperformed the others, particularly in tasks requiring high accuracy including Type A and Type B MCQ evaluations. Its best performance is attributed to advanced fine-tuning with RAG, achieving the highest exact match accuracy (57%) and superior ranking metrics like MRR and LaCC. However, its strength in accuracy comes at the cost of longer response times and higher computational demands. On the other hand, Llama-2-7B showcased a balanced approach, offering competitive accuracy (55%) and good performance in subjective question generation and MCQ evaluations, while maintaining a slightly more efficient processing time than Mistral-7B. This could make it well-suited for broader applications, such as user medical chatbots. But the average time per question is something that remains to be made more efficient in future works.

In contrast, Flan-T5-Large lagged behind in all accuracy metrics, particularly in the MCQ evaluations (Type A: 33%, Type B: lowest MRR), but it demonstrated notable strengths in inference speed and computational efficiency. This suggests that while it may not be ideal for tasks requiring high accuracy, it can excel in scenarios where speed is prioritised requiring near real-time responses. Despite its lower performance, fine-tuning improved Flan-T5-Large's relevance and accuracy, even when it remained outperformed by the other models across metrics and datasets. Our results indicate that while these fine-tuned models are capable of generating accurate responses, the integration of RAG significantly improves performance, especially in scenarios requiring detailed, context-specific information. In particular, the exact match accuracy improved by 25% on the MedMCQA dataset when we compared the fine-tuned Mistral-7B model with RAG to its fine-tuned counterpart without RAG in Type A multiple-choice question (MCQ) evaluations. This suggests that while fine-tuned models can perform adequately on their own, RAG's retrieval mechanism enhances their capability to deliver more precise and contextually relevant answers.

In this study, the Type A MCQ evaluation methodology demonstrated superior performance compared to Type B. For instance, with the Mistral-7B model, Type A achieved an exact match accuracy of 57%. This straightforward approach allows the model to focus solely on selecting the correct answer from the provided options; therefore, it minimizes the complexity and potential errors in generation or scoring. In contrast, the Type B's LaCC ($k = 1$) metric can be considered the counterpart of Type A's exact match accuracy. For the same Mistral model classification, the Type B LaCC ($k = 1$) achieved only 32.8%. This significant difference arises from the methodological distinctions of Type B, where the Type B method calculates the reference answers' overall mean logit scores and then converts them to probabilities, a process not typically applied in this simple manner by traditional LLMs. In contrast, Type A applies few-shot prompting, where reference answers are provided, and the LLM automatically generates or scores the best answer based on those examples. Type A outperformed Type B by leveraging the LLM's inherent strength in context-based generation, while Type B introduced additional manual steps, which reduced both efficiency and accuracy due to potential errors in the process.

The above findings emphasise the importance of model configuration and hyperparameter optimisation for performance improvement. While Mistral-7B is well suited for tasks demanding high precision, Llama-2-7B, on the other hand, is very useful for many applications, whereas Flan-T5-Large is fast and efficient even though it lacks high preciseness for the tasks above. These findings highlight the need for continued research into fine-tuning techniques and RAG advancements to further enhance model performance. From a comparison standpoint, while the Flan-PaLM model, with its impressive 540 billion parameters, achieves an accuracy of 67.6% on the USMLE MedQA dataset [24], our much smaller Mistral-7B model with a RAG system, despite having far fewer parameters, achieves a commendable 57% accuracy on the same dataset. More refined fine-tuning techniques remain to be found to improve application-specific details of the domain requirements, together with the integration of more advanced RAG systems like Graph-RAG for effective knowledge retrieval and processing [43], model pruning, and distillation for best performance in resource-constrained scenarios. These advances, among others, are likely to bring forth more sophisticated AI systems integrated into medical chatbots and other conversational AI technologies in the very near future.

6. Summary and Conclusions

This research conducted a systematic analysis of Retrieval-Augmented Generation (RAG)-based Large Language Models (LLMs) for medical subjective and multi-choice questions, specifically focusing on their performance and suitability in computationally resource-constrained environments. The study evaluated three open-source LLMs across three different architectures (with/without Fine-tuning and RAG) and datasets.

This analysis revealed that out of three LLMs, *Mistral-7B* consistently outperformed the other models across various tasks and configurations. It demonstrated superior accuracy, relevance, and overall performance, particularly when fine-tuned with RAG. The model achieved an exact match accuracy of 57% in the MCQ evaluation (Type A) for the fine-tuned model with RAG. Different models and their classifications displayed diverse strengths and weaknesses based on their architectural designs and training. For instance, *Flan-T5-Large*, while fastest in processing, showed lower accuracy and performance across tasks, indicating it may be more suited for speed-oriented applications rather than complex content generation. *Llama-2-7B* offered a balanced performance, making it suitable for scenarios requiring a compromise between accuracy and efficiency.

In summary, with further modifications, these models can be integrated into clinical workflows to assist healthcare providers by answering medical queries both quickly and accurately [44], particularly in complex decision-making scenarios. Our novel methodology (Type B) could be applied in real-time applications by incorporating probabilistic confidence scores, allowing clinicians to assess how certain the model is in its recommendations. For patient education, medical chatbots powered by these models can provide accessible, reliable, and confidence-rated information on health conditions and treatments [45–47]. To support our argument, we calculated the Pearson correlation between model confidence and ground truth correctness. A strong correlation would demonstrate that the model's confidence is a reliable indicator of accuracy, adding value for trainees. Preliminary analysis shows a moderate to strong positive correlation of 0.618 between confidence scores and ground truth of the *Mistral-7B* finetuned model with RAG classification, indicating that the model's confidence is reasonably well-aligned with reality.

Our analysis included hybrid architectures, combining RAG with fine-tuned models and base models with RAG and fine-tuned models. Furthermore, a unique MCQ evaluation was carried out comparing MCQ performance as question answering and as language generation. Unlike previous studies, this paper distinguishes itself by fine-tuning models on diverse datasets and conducting evaluations across different datasets—providing a more comprehensive assessment. The main limitation of this study is that all experiments were conducted in a controlled computational environment, which may not fully capture the challenges and performance variations that could arise in more dynamic and real-world settings. Future research should focus on testing these models in diverse, real-world medical scenarios and exploring further optimisations for both accuracy and efficiency for near real-time response generation. Additionally, there is potential for refining Type B methodology to improve its usefulness in LLMs applied to different domains.

Author Contributions: Conceptualisation, A.B. and H.C.; methodology, A.B. and H.C.; software, A.B.; validation, A.B. and H.C.; formal analysis, A.B. and H.C.; investigation, A.B. and H.C.; resources, A.B.; data curation, A.B.; writing—original draft preparation, A.B.; writing—review and editing, A.B. and H.C.; visualisation, A.B.; supervision, H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All datasets used in this paper are publicly available. See Sections 4.1 and 4.5 for pointers to those datasets.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

AI	Artificial Intelligence
NLP	Natural Language Processing
LLM	Large Language Model
BERT	Bidirectional Encoder Representations from Transformers
RAG	Retrieval-Augmented Generation
ICL	In-Context Learning

CoT	Chain-of-Thought
PEFT	Parameter Efficient Fine-Tuning
LoRA	Low-Rank Adaptation
PTQ	Post-Training quantization
QAT	quantization-Aware Training
MCQ	Multi-Choice Questions
MRR	Mean Reciprocal Rank
LaCC	Lenient Accuracy

Appendix A

Appendix A.1. System Requirements for the Research

The following requirements are needed to utilise the tools and libraries used in this research:

- A Hugging Face account to access models, datasets, and integrations.
- Hardware specifications:
  - Minimum 15 GB of RAM to ensure smooth operation of deep learning models and data processing.
  - At least 200 GB of free storage on an SSD or HDD for storing datasets, model checkpoints, and other necessary files.
  - An NVIDIA GeForce RTX 3060 or a higher GPU with at least 12 GB of GPU memory for efficient model training and inference.

Appendix A.2. Overview of Required Python Libraries

This research utilised a variety of Python libraries for deep learning, natural language processing, and efficient model optimisation. `PyTorch` is a fundamental library used for tensor computation and deep learning tasks, while `pypdf` handles PDF manipulations. Advanced NLP tasks are managed using `transformers[torch]` and `sentence_transformers` for embedding generation. Hugging Faces' `datasets` and `optimum` facilitate dataset management and model optimisation, respectively. Libraries like `accelerate` and `bitsandbytes` optimise distributed training and memory consumption, while `faiss_cpu` enables fast similarity search. Additionally, `langchain`, `llama_index`, and related plugins support building language model-powered applications, while tools like `rouge_score`, `bert_score`, and `sentencepiece` assist with model evaluation and tokenisation.

Appendix A.3. Example of Subjective Question Evaluation

This section provides an example result of a Subjective Question Evaluation response. The model dataset and classification choice for this example are the 'Meadow-MedQA' Dataset and Finetuned Model with RAG, respectively.

- Question: Who is at risk for Lymphocytic Choriomeningitis (LCM)?
- Ground Truth: Individuals of all ages who come into contact with urine, faeces, saliva, or blood of wild mice are potentially at risk for infection. Owners of pet mice or hamsters may be at risk for infection if these animals originate from colonies that were contaminated with LCMV, or if their animals are infected from other wild mice. Human fetuses are at risk of acquiring infection vertically from an infected mother. Laboratory workers who work with the virus or handle infected animals are also at risk. However, this risk can be minimised by utilising animals from sources that regularly test for the virus, wearing proper protective laboratory gear, and following appropriate safety precautions.

Table A1. Answers of LLMs to an example subjective query.

Model	Responses	Performance Metrics
Flan-T5-Large	LCMV infections can occur after exposure to fresh urine, droppings, saliva, or nesting materials from infected rodents.	BERT Score F1: 0.872 ROUGE-1 Score: 0.125 ROUGE-L Score: 0.078 BLEU Score: 0.0012 METEOR Score: 0.0393 Perplexity Score: 83.20
Llama-2-7B	LCMV infections can occur after exposure to fresh urine, droppings, saliva, or nesting materials from infected rodents. Transmission may also occur when these materials are directly introduced into broken skin, the nose, the eyes, or the mouth, or presumably, via the bite of an infected rodent. Person-to-person transmission has not been reported, with the exception of vertical transmission from infected mother to fetus.	BERT Score F1: 0.863 ROUGE-1 Score: 0.295 ROUGE-L Score: 0.1477 BLEU Score: 0.0346 METEOR Score: 0.140 Perplexity Score: 34.628
Mistral-7B	LCMV infections can occur after exposure to fresh urine, droppings, saliva, or nesting materials from infected rodents. Transmission may also occur when these materials are directly introduced into broken skin, the nose, the eyes, or the mouth, or presumably, via the bite of an infected rodent. Person-to-person transmission has not been reported, with the exception of vertical transmission from infected mother to fetus, and rarely, through organ transplantation. LCMV infections can also occur after exposure to infected farm animals, including cattle, sheep, goats, and pigs.	BERT Score F1: 0.863 ROUGE-1 Score: 0.3030 ROUGE-L Score: 0.1717 BLEU Score: 0.0412 METEOR Score: 0.1337 Perplexity Score: 23.0744

Appendix A.4. Example of MCQ Evaluation

This section provides example results of two types of MCQ Evaluation responses.

- Question: A 35-year-old man comes to the physician because of itchy, watery eyes for the past week. He has also been sneezing multiple times a day during this period. He had a similar episode 1 year ago around springtime. He has iron deficiency anaemia and ankylosing spondylitis. Current medications include ferrous sulfate, artificial tear drops, and indomethacin. He works as an elementary school teacher. His vital signs are within normal limits. Visual acuity is 20/20 without correction. Physical examination shows bilateral conjunctival injection with watery discharge. The pupils are 3 mm, equal, and reactive to light. Examination of the anterior chamber of the eye is unremarkable. Which of the following is the most appropriate treatment?
- Options: 'A': 'Erythromycin ointment', 'B': 'Ketotifen eye drops', 'C': 'Warm compresses', 'D': 'Fluorometholone eye drops', 'E': 'Latanoprost eye drops'
- Ground Truth: Ketotifen eye drops

Table A2. Methodological comparison of MCQ response evaluations.

Methodology	Output Response Format	Performance Metrics
Type A	Generated Answer: B': Ketotifen eye drops	Cosine Similarity = 1.0 Exact Match
Type B	Options and Confidence scores: Option A :: Erythromycin ointment - `Confidence: 0.4048, Similarity: 0.5229` Option B:: Ketotifen eye drops - `Confidence: 0.4423, Similarity: 0.5074` Option C:: Warm compresses - `Confidence: 0.2088, Similarity: 0.5516` Option D:: Fluorometholone eye drops - `Confidence: 0.3739, Similarity: 0.5061` Option E:: Latanoprost eye drops - `Confidence: 0.2895, Similarity: 0.5345` Generated Answer: The physician should prescribe topical antihistamines to relieve symptoms. Selected Answer: Option B:: Ketotifen eye drops Ground Truth: Ketotifen eye drops	MRR = 1.0 LaCC (k = 1) = 100% LaCC (k = 3) = 100%

References

1. Silvera-Tawil, D. Robotics in Healthcare: A Survey. *SN Comput. Sci.* **2024**, *5*, 189. [CrossRef]
2. Topol, E.J. High-performance medicine: The convergence of human and artificial intelligence. *Nat. Med.* **2019**, *25*, 44–56. [CrossRef] [PubMed]
3. Toukmaji, C.; Tee, A. Retrieval-Augmented Generation and LLM Agents for Biomimicry Design Solutions. In Proceedings of the AAAI Spring Symposium Series (SSS-24), Stanford, CA, USA, 25–27 March 2024.
4. Zeng, F.; Gan, W.; Wang, Y.; Liu, N.; Yu, P.S. Large Language Models for Robotics: A Survey. *arXiv* **2023**, arXiv:2311.07226.
5. Vaswani, A. Attention Is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
6. Jiang, A.Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B. *arXiv* **2023**, arXiv:2310.06825.
7. Ni, J.; Qu, C.; Lu, J.; Dai, Z.; Ábrego, G.H.; Ma, J.; Zhao, V.Y.; Luan, Y.; Hall, K.B.; Chang, M.-W.; et al. Large Dual Encoders Are Generalizable Retrievers. *arXiv* **2021**, arXiv:2112.07899.
8. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *arXiv* **2019**, arXiv:1908.10084.
9. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. BERTScore: Evaluating Text Generation with BERT. *arXiv* **2020**, arXiv:1904.09675.
10. Wolfe, C.R. LLaMA-2 from the Ground Up. 2023. Available online: https://cameronrwolfe.substack.com/p/llama-2-from-the-ground-up (accessed on 7 June 2024).
11. Driess, D.; Xia, F.; Sajjadi, M.S.M.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; et al. PaLM-E: An Embodied Multimodal Language Model. In Proceedings of the 40th International Conference on Machine Learning (ICML'23), Honolulu, HI, USA, 23–29 July 2023; Volume 202, pp. 8469–8488.
12. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *arXiv* **2023**, arXiv:2201.11903.
13. Béchard, P.; Ayala, O.M. Reducing hallucination in structured outputs via Retrieval-Augmented Generation. *arXiv* **2024**, arXiv:2404.08189.
14. Banerjee, S.; Agarwal, A.; Singla, S. LLMs Will Always Hallucinate, and We Need to Live with This. *arXiv* **2024**, arXiv:2409.05746.
15. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, M.; Wang, H. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv* **2024**, arXiv:2312.10997.
16. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *arXiv* **2021**, arXiv:2005.11401.
17. Gholami, A.; Kim, S.; Dong, Z.; Yao, Z.; Mahoney, M.W.; Keutzer, K. A Survey of Quantization Methods for Efficient Neural Network Inference. *arXiv* **2021**, arXiv:2103.13630.
18. Bajwa, J.; Munir, U.; Nori, A.; Williams, B. Artificial intelligence in healthcare: Transforming the practice of medicine. *Future Healthc. J.* **2021**, *8*, e188–e194. [CrossRef] [PubMed] [PubMed Central]
19. Pal, A.; Umapathi, L.K.; Sankarasubbu, M. MedMCQA: A Large-Scale Multi-Subject Multi-Choice Dataset for Medical Domain Question Answering. *arXiv* **2022**, arXiv:2203.14371.
20. Gu, Y.; Tinn, R.; Cheng, H.; Lucas, M.; Usuyama, N.; Liu, X.; Naumann, T.; Gao, J.; Poon, H. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Trans. Health Inform.* **2022**, *3*, 1–23. [CrossRef]
21. Bedi, S.; Liu, Y.; Orr-Ewing, L.; Dash, D.; Koyejo, S.; Callahan, A.; Fries, J.A.; Wornow, M.; Swaminathan, A.; Lehmann, L.S.; et al. A Systematic Review of Testing and Evaluation of Healthcare Applications of Large Language Models (LLMs). *medRxiv* **2024**, 2024.04.15.24305869. [CrossRef]
22. Ge, J.; Sun, S.; Owens, J.; Galvez, V.; Gologorskaya, O.; Lai, J.C.; Pletcher, M.J.; Lai, K. Development of a Liver Disease-Specific Large Language Model Chat Interface Using Retrieval Augmented Generation. *medRxiv* **2023**, 2023.11.10.23298364.
23. Ramjee, P.; Sachdeva, B.; Golechha, S.; Kulkarni, S.; Fulari, G.; Murali, K.; Jain, M. CataractBot: An LLM-Powered Expert-in-the-Loop Chatbot for Cataract Patients. *arXiv* **2024**, arXiv:2402.04620.
24. Liévin, V.; Hother, C.E.; Motzfeldt, A.G.; Winther, O. Can large language models reason about medical questions? *Patterns* **2024**, *5*, 100943. [CrossRef]
25. Jovanović, M.; Baez, M.; Casati, F. Chatbots as Conversational Healthcare Services. *IEEE Internet Comput.* **2021**, *25*, 44–51. [CrossRef]
26. Zhou, H.; Liu, F.; Gu, B.; Zou, X.; Huang, J.; Wu, J.; Li, Y.; Chen, S.S.; Zhou, P.; Liu, J.; et al. A Survey of Large Language Models in Medicine: Progress, Application, and Challenge. *arXiv* **2024**, arXiv:2311.05112.
27. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv* **2023**, arXiv:1910.10683.
28. Chung, H.W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, Y.; Wang, X.; Dehghani, M.; Brahma, S.; et al. Scaling Instruction-Finetuned Language Models. *arXiv* **2022**, arXiv:2210.11416.
29. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv* **2023**, arXiv:2307.09288.
30. Gao, Y.; Liu, Y.; Zhang, H.; Li, Z.; Zhu, Y.; Lin, H.; Yang, M. Estimating GPU Memory Consumption of Deep Learning Models. In Proceedings of the ACM, Virtual, 8–13 November 2020.

31. Jeon, H.; Kim, Y.; Kim, J.-J. L4Q: Parameter Efficient Quantization-Aware Fine-Tuning on Large Language Models. *arXiv* **2024**, arXiv:2402.04902.
32. Dettmers, T.; Pagnoni, A.; Holtzman, A.; Zettlemoyer, L. QLoRA: Efficient finetuning of quantized LLMs. *arXiv* **2023**, arXiv:2305.14314.
33. Xu, Y.; Xie, L.; Gu, X.; Chen, X.; Chang, H.; Zhang, H.; Chen, Z.; Zhang, X.; Tian, Q. QA-LoRA: Quantization-Aware Low-Rank Adaptation of Large Language Models. *arXiv* **2023**, arXiv:2309.14717.
34. Christophe, C.; Kanithi, P.K.; Munjal, P.; Raha, T.; Hayat, N.; Rajan, R.; Al-Mahrooqi, A.; Gupta, A.; Salman, M.U.; Gosal, G.; et al. Med42—Evaluating Fine-Tuning Strategies for Medical LLMs: Full-Parameter vs. Parameter-Efficient Approaches. *arXiv* **2024**, arXiv:2404.14779v1.
35. Han, T.; Adams, L.C.; Papaioannou, J.-M.; Grundmann, P.; Oberhauser, T.; Löser, A.; Truhn, D.; Bressem, K.K. MedAlpaca—An Open-Source Collection of Medical Conversational AI Models and Training Data. *arXiv* **2023**, arXiv:2304.08247.
36. Jin, Q.; Dhingra, B.; Liu, Z.; Cohen, W.W.; Lu, X. PubMedQA: A Dataset for Biomedical Research Question Answering. *arXiv* **2019**, arXiv:1909.06146.
37. Abacha, A.B.; Demner-Fushman, D. A Question-Entailment Approach to Question Answering. *BMC Bioinform.* **2019**, *20*, 511.
38. Hu, T.; Zhou, X.-H. Unveiling LLM Evaluation Focused on Metrics: Challenges and Solutions. *arXiv* **2024**, arXiv:2404.09135.
39. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.-J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; Association for Computational Linguistics: Philadelphia, PA, USA, 2002; pp. 311–318.
40. Lin, C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*; Association for Computational Linguistics: Barcelona, Spain, 2004; pp. 74–81.
41. Banerjee, S.; Lavie, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29 June 2005; pp. 65–72.
42. Zhou, J. QOG: Question and Options Generation based on Language Model. *arXiv* **2024**, arXiv:2406.12381.
43. Wu, J.; Zhu, J.; Qi, Y. Medical Graph RAG: Towards Safe Medical Large Language Model via Graph Retrieval-Augmented Generation. *arXiv* **2024**, arXiv:2408.04187.
44. Singhal, K.; Tu, T.; Gottweis, J.; Sayres, R.; Wulczyn, E.; Hou, L.; Clark, K.; Pfohl, S.; Cole-Lewis, H.; Neal, D.; et al. Towards Expert-Level Medical Question Answering with Large Language Models. *arXiv* **2023**, arXiv:2305.09617.
45. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A Survey of Large Language Models. *arXiv* **2023**, arXiv:2303.18223.
46. Mhatre, A.; Warhade, S.R.; Pawar, O.; Kokate, S.; Jain, S.; Emmanuel, M. Leveraging LLM: Implementing an Advanced AI Chatbot for Healthcare. *Int. J. Innov. Sci. Res. Technol.* **2024**, *9*. [CrossRef]
47. Singhal, K.; Azizi, S.; Tu, T.; Mahdavi, S.S.; Wei, J.; Chung, H.W.; Scales, N.; Tanwani, A.; Cole-Lewis, H.; Pfohl, S.; et al. Large language models encode clinical knowledge. *Nature* **2023**, *620*, 172–180. [CrossRef]