National College of Ireland

Evaluation of Large Language Models on MedQUAD Dataset

MSc Research Project
MSc. of Artificial Intelligence

Muhammad Hassan Shakeel
Student ID: 23215992

School of Computing
National College of Ireland

Supervisor: Dr. Anh Duong Trinh

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Muhammad Hassan Shakeel
Student ID:	23215992
Programme:	MSc. of Artificial Intelligence
Year:	2024
Module:	Practicum 2
Supervisor:	Dr. Anh Duong Trinh
Submission Due Date:	18/12/2024
Project Title:	Evaluation of Large Language Models (LLMs) on MedQUAD Dataset
Word Count:	4719
Page Count:	16

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Muhammad Hassan Shakeel
Date:	24th January 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Evaluation of Large Language Models (LLMs) on MedQUAD Dataset

Muhammad Hassan Shakeel
23215992

Abstract

In recent times, small-sized LLMs have outperformed bigger LLMs such as GPT-2 for domain-specific tasks after fine-tuning. This paper fine-tunes small-sized LLMs such as Gemma-2 (2 billion), Phi-2 (2.7 billion) and Llama-2 (7 billion) parameters for question-answering task on MedQUAD dataset. Among Gemma-2, Phi-2 and Llama-2, Llama-2 has outperformed others with ROUGE-1=0.455, ROUGE-2=0.289, ROUGE-L=0.373 and BLEU=0.275. On the dimensions of informativeness, relevance, grammaticality, naturalness and sentiment for human evaluation, the three models produced similar performance, however Llama-2 outperformed with the average score of 7.492. This paper observed a pattern observed a correlation between model parameter size and model performance, big model gives better performance compare to small models.

1 Introduction

Healthcare sector is generating data at massive rate of 1 million articles per year on PubMed database (Lamichhane and Kahanda; 2023). Healthcare professionals spend one hour in one database and 3 minutes to review a document on average (Kaddari et al.; 2020). To reduce time spent in exploration of relevant of information in the database, it is critical to devise methodology that extract information from unstructured database efficiently.

Natural Language Processing (NLP) tasks such as Named Entity Recognition (NER), Question-Answering and Natural Language Understanding (NLU) are employed in healthcare domain to process unstructured text. Deep Learning (DL) models such as Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), Convolutional Neural Network are extensively used for text processing. However models involving transformer architecture, such as BERT exceeded performances of its predecessor models (Yang et al.; 2022). Moreover in recent times, Large Language Model (LLM) such as GPT-2 outshined previous state-of-the-art models in NLP domain (Beattie et al.; 2024).

Generalized LLMs such as GPT-3 have demonstrated promising results however they do not replicate similar results for domain-specific tasks such as medical related tasks (Basit et al.; 2024). That is why many researchers pre-trained domain specific LLMs (Yang et al.; 2022). However, pre-training of an LLM requires high computational and memory resources. (Schick and Schütze; 2021) acheived competitive results with significantly small-sized LLMs to cater computational constraints. This paper fine-tunes LLMs such as Gemma-2, Phi-2 and Llama-2 that are significantly smaller in terms of model parameters for question-answering on MedQUAD dataset.

Research Question: *How does the performance of Large Language Models (LLMs) such as Gemma-2, Phi-2 and Llama-2 compare to each other for automatic answer generation of 37 types of question types such as treatment, diagnosis and side-effects associated with disease and drugs in terms of metrics such as ROUGE, BLEU and Human Evaluation on the MedQUAD dataset?*

Next section discusses related work and then section 3 of the paper discusses methodology. Section 4 discusses design specifications while section 5 go through implementation. Afterwards, this paper touch on evaluation in section 6 and section 7 discusses conclusion and future work.

2 Related Work

Evaluation of question-answering task for the medical domain is critical because the domain requires error free responses from the system. Although this section covers literature review evaluation of a wide variety of techniques for question-answering task using MeDQUAD dataset. This review focuses on approaches such as Deep Learning and LLMs for medical specific question-answering tasks. This literature review will mostly focus on evaluation of LLMs using MedQUAD dataset.

2.1 Question Answering using Deep Learning

(Abdallah et al.; 2020) used datasets from online resources such as eHealthForum and MedQUAD dataset for the training an end-to-end DL model using RNN and encoder-decoder architecture. Upon experiments with GRU, LSTM, Bi-LSTM and Attention mechanism, Attention outperformed on generation of 50 characters, 75 characters and 100 characters with BLEU scores of 83.11, 78.80 and 59.31 respectively. As an alternative approach, (Mutabazi et al.; 2023) suggested DL architecture with CNN layer for feature extraction and BiLSTM layer for answer generation. They trained the model using a subset of MedQUAD dataset that were formatted in their desired format and achieved precision, accuracy and F1-score of 93.33% each. (Athira et al.; 2024) suggested an approach of finding a similar query from available discussion thread and generated a similar of the discussion thread. They used the dataset available at Breastcancer.org for training of the Siamese network model and then they applied transfer learning using the MedQUAD dataset. For query similarity, BiLSTM using BioBERT embeddings achieved the highest Precision (0.85), Recall (0.86) and F1-score (0.855) after Transfer Learning using MedQUAD dataset. Without Transfer Learning, the same model gave the best performance among others with Precision (0.63), Recall (0.66) and F1-score (0.65). They also stored summarized in the database using BERT, and BioBERT embeddings. The BioBERT embedding approach gave the best performance with ROUGE-1 (0.491) and ROUGE-2 (0.297).

2.2 Question Answering using Large Language Models

(Haghighi et al.; 2024) pretrained EYE-Llama model and compared performance with ChatGPT. In the results, EYE-Llaama model outperformed ChatGPT with F1 score of 0.5758 compared to 0.5757 score of ChatGPT. Similarly, (Yang et al.; 2022) pretrained GatorTron model with different number of parameters i.e. 345 million, 3.9 billion and 8.9 billion. The GatorTron model significantly improved performance with F1 score of

0.7408 medical question-answering dataset. In contrast, (Abdul et al.; 2024) suggested Knowledgeable Diagnostic Transformer (KDT) and trained the model on MedQUAD dataset and MDQA dataset, on MedQUAD dataset KDT outperformed GPT-2, GPT-3.5-turbo with F1-scores of 99.2, 92.7 and 98.1 respectively. Alternatively, (Luo et al.; 2022) pretrained BioGPT model on PubMed dataset. BioGPT is a GPT-2 based architecture. For question-answering task, BioGPT outperformed GPT-2 and other models such as PubMedBERT with accuracy of 78.2%.

Using prompt-engineering, researchers manipulate the LLMs to extract desired output from the model. (Ahmad et al.; 2024) proposed a prompt-engineering technique called CanPrompt on Mistral 7x8B, Falcon 40b, and Llama 3-8b LLMs. In BERTScore metrics, Mistral 7x8b leads the other two models with average accuracy of around 84%. On Rouge metrics as well, Mistral outperformed other models with ROUGE-1 (26%), ROUGE-2 (8.2%) and ROUGE-L (15%). Comparatively, (Nguyen et al.; 2023) evaluated T5, Flan-T5, Alpaca and Llama models with zero-shot prompt-engineering. Alpaca model outperformed other models with Rouge-1 score of 0.168.

Fine-tuning is used to gain better performance out of LLM for domain-specific tasks. Considering that (Yagnik et al.; 2024) fine-tunned LLMs such as T5, GPT-2 and Bloom using MedQUAD dataset. On Bleu1 and Bleu4 scores, T5 scored the highest with the scores of 7.117 and 0.321 respectively but on Rouge-1 and Rouge-L evaluation metrics, Bloom outperformed with scores of 0.226 and 0.212 respectively. Similarly, (Subramanian et al.; 2024) finetunned Flacon (7B), Flan-T5 (3B), LLaMA 2 (7B), MPT (7B) models with 8-bit quantization to improve efficiency. On MedQUAD dataset, LLaMA 2 outperformed other models with ROUGE-L, BertScore and METEOR scores of 15.7, 58.5 and 16.6 respectively. In contrast, (Yu et al.; 2024) used precision as evaluation metric of T5, Phi-3, Gemma-2b and Mistral 7B with scores of 0.702, 0.718, 0.721 and 0.762 respectively. In their experiment, Mistral 7B model outperformed other models. Similarly, (Lamichhane and Kahanda; 2023) used composition of BM25 and Transformer models such as BERT and RoBERTa. Their approach achieved the F1 score of 0.260 and 0.449 for BERT and RoBERTa respectively. In the same way, (Das and Nirmala; 2022) fine-tuned BERT model for medical domain using PubMed and PMC dataset and proposed BioBERT model. However XLNet model showed slightly better performance with F1-score of 89.89%. (Nguyen et al.; 2023) also did similar experiment however with different dataset. They experimented with fine-tuning T5, Flan-T5, Alpaca, Llama. Upon evaluation using ROUGE-1, Alpaca gave the best score of 0.172. To improve the model performance, (Ismail et al.; 2024) integrated Reinforcement Learning with Human Feedback (RLHF) on BART model. However, they used Levenshtein distance (0.293) and cosine similarity (0.582) as evaluation metrics. In similar fashion (Gu et al.; 2021) fine-tuned a BERT model on MedQUAD dataset that achieved BELU score of 44.21%. In parallel, (Wang et al.; 2024) suggested a Joint Medical LLM and Retrieval Training for Enhancing Reasoning and Professional Question Answering Capability (JMLR) architecture. The proposed architecture surpassed all other state-of-the-art architectures with accuracy of 62.5%

The research community focus is mostly towards fine-tuning smaller sized LLMs compare to huge LLMs such as GPT-3. The smaller sized LLMs have given good results when applied to domain-specific problems such as medical domain. Other than LLMs fine-tuning, other Deep Learning models such as Bi-LSTM also give good performance, however LLMs have pushed the performance even further. Also, there are number of evaluation metrics that the researcher utilized to evaluate the model performance and

every paper did not use the same set of evaluation metrics. Even though many different evaluation metrics are used, application is human evaluation for the medical-domain using MedQUAD is not experimented with. Therefore, to extend on evaluation of LLMs, this paper will do human evaluation of the LLM models in addition to ROUGE and BLEU scores.

3 Methodology

3.1 Data Collection

For purpose of this study, publicly available medical question-answering dataset is chosen i.e. the MedQUAD dataset. The MedQUAD dataset consists of 47,457 medical question-answers from number of NIH resources such as cancer.gov, MedlinePlus Health Topics etc. The dataset consists of questions of types such as treatments, side-effects, and diagnosis related to diseases and medical procedures (Abacha; 2024).

3.2 Data Pre-processing

3.2.1 Data Cleaning

The data is transformed from the xml format to a favorable format i.e. csv format. Afterwards, data rows with empty answers are discard (some data rows are not available in the dataset due to compliance issues).

3.2.2 Text Summarization

Text Rank algorithm is a graph-based unsupervised learning summarization algorithm hence the algorithm has edge and vertices. The algorithm is a ranking-based algorithm where important vertex gets higher rank. One vertex cast vote to another, when they have a edge between them, when a vertex has more connections it is more important. For natural language data, words are taken as vertices and their adjacent appearance is considered as an edge (Mihalcea and Tarau; 2004).

To make model training computationally feasible, this paper uses Text Rank algorithm to generative extractive summary of answers in the MedQUAD dataset. Afterwards, the dataset is transformed into a suitable prompt template shown in Table 1.

LLM	Prompt
Google/Gemma-2	Instruct: ⟨question ⟩\nOutput: ⟨answer ⟩
Microsoft/Phi-2	Instruct: ⟨question ⟩\nOutput: ⟨answer ⟩
Llama-2	Instruct: ⟨s ⟩[INST] ⟨question ⟩[/INST] ⟨⟨s ⟩

Table 1: Prompts for LLMs

3.2.3 Tokenization

In NLP, tokenization is the process of splitting text sequences into smaller components i.e. words. or characters. All the sequence generation models such as RNN, LSTM and Transformers, processes one token at a time and they generate one token at a time. Therefore, tokenization is a necessary pre-processing step for training of an LLM. For

that reason, after generation of summary in the last step, this paper perform the step of tokenization. Since this paper finetunes a pretrained model, using LLM's own tokenizer is a suitable choice.

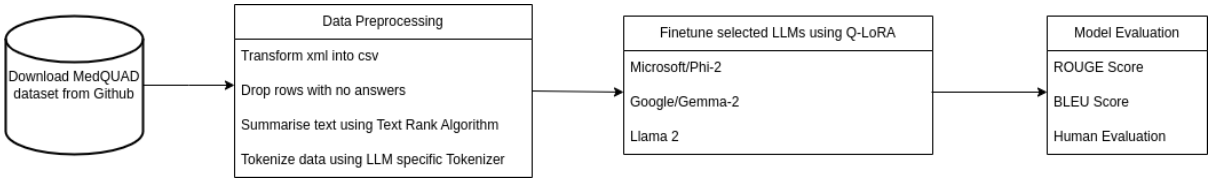


Figure 1: Research Methodology

3.3 Large Language Models Fine-tuning

General purpose LLMs such as GPT-3 gives good results on general purpose tasks but they fail to meet similar performance benchmark on specialized tasks such as medical tasks. Research community have experimented with pretraining an LLM on specialized dataset. However pretrainng billions of parameters requires huge computational power, memory and data availability. Fine-tuning an LLM on the other hand allows the LLM to adapt to specialized tasks. However, full fine-tuning also requires huge amount of memory and data availablity computational resources. Considering these limitations, (Dettmers et al.; 2023) introduces Quantized Low Rank Adapters (Q-LoRA). It is an efficient fine-tuning approach that allows researchers to train 65B parameters LLM on single 48 GB GPU. This approach is not only memory efficient but computationally efficient as well. It stores floating point values in 4-bit memory instead of 32-bit memory which not only saves memory but improves computational efficiency as well. (Dettmers et al.; 2023) reported with Q-LoRA fine-tuning, Guanaco model acheived 99.3% similar performance to GPT-4 on OASST1 dataset. Similar to (Dettmers et al.; 2023), this paper also uses 4-bit quantization with LoRA adapters. Moreover, to improve further computational efficiency this paper update weights of only last layer of each LLM. For example, this study update weights of 25^{th} layer of the Gemma-2 model while 31^{st} layer fo the Microsoft's Phi-2 model. Among other configurations, this study uses the paged_adamw_32bit optimizer and learning rate of $2\text{x}10^{-4}$.

Training data	15832
Test data	575

Table 2: Data shape

3.4 Evaluation of Large Language Models

Evaluation is a necessary step to compare the performance of models. This paper evaluates LLM models on ROUGE-1, ROUGE-2, ROUGE-L, BLEU score and also on the basis of Human Evaluation.

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) measures the quality of generated summary by comparing it to a golden summary (human written summary). ROUGE-N score is calculated by counting the overlapping n-grams or word sequences and ROUGE-L is the longest common subsequence in the generated summary and reference

summary (Lin; 2004). As its name suggests, ROUGE is a recall-related measure.

$$\text{ROUGE-N} = \frac{\sum_{\text{S} \in \text{ReferenceSummaries}} \sum_{\text{gram}_n \in \text{S}} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{\text{S} \in \text{ReferenceSummaries}} \sum_{\text{gram}_n \in \text{S}} \text{Count}(\text{gram}_n)}$$

where:

- n stands for n-gram size.
- $\text{Count}_{\text{match}}(\text{gram}_n)$ represents maximum number of n-grams co-occurring in a candidate summary and reference summaries.
- $\text{Count}(\text{gram}_n)$ is the total number of n-grams in a reference summary.

$$\text{ROUGE-L} = F_{\text{score}} = \frac{(1 + \beta^2) \cdot \text{P} \cdot \text{R}}{\beta^2 \cdot \text{P} + \text{R}}$$

where:

- $\text{P} = \frac{\text{LCS}(X,Y)}{\text{length of candidate summary (X)}}$, the precision based on the Longest Common Subsequence (LCS).
- $\text{R} = \frac{\text{LCS}(X,Y)}{\text{length of reference summary (Y)}}$, the recall based on the LCS.
- $\text{LCS}(X, Y)$ is the longest sequence of words that appear in the same order in both the candidate summary X and the reference summary Y.
- β is a weight factor that defines the relative importance of precision and recall (commonly set to $\beta = 1$).

The BLEU is an evaluation metric to rank machine translation. It compares the machine generated translation by comparing it with several human-generated translations (reference translations). It counts how many n-gram from machine-generated translation appear in any of the reference translations. Afterwards, the frequency of each n-gram is clipped with maximum frequency of that n-gram in any of the reference translations. Afterwards, these values are averaged and used as a weighted factor.

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Where:

- $p_n = \frac{\sum_{\text{gram}_n \in \text{candidate}} \min(\text{count}(\text{gram}_n), \max_{\text{ref}} \text{count}(\text{gram}_n))}{\sum_{\text{gram}_n \in \text{candidate}} \text{count}(\text{gram}_n)}$, the precision for n-grams of size n, calculated by clipping the frequency of n-grams to the maximum count in any reference.
- $w_n = \frac{1}{N}$ are the weights for each n-gram precision, where N is the maximum n-gram size.
- BP is the brevity penalty, calculated as:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{1-\frac{r}{c}} & \text{if } c \leq r \end{cases}$$

where c is the length of the candidate translation and r is the length of the reference translation (typically the reference closest to the candidate).

Text generation can go wrong in many ways while automatic evaluation scores such as ROUGE and BELU can still be very high. Also, the reliability of the reported scores also depends on the implementation done in the libraries used to calculate these scores. Automatic evaluation metrics fails to calculate the information quality presented in the generated answer. To measure grammatical accuracy, relevance of the answer to the question, informativeness, naturalness of the generated answer and sentiment (bias towards certain sex/community/religion), this paper uses human evaluation as an evaluation metric. All human evaluation parameters used in this paper are listed in Table 3 (van der Lee et al.; 2021).

Human Evaluation Dimensions
Informativeness
Relevance
Grammaticality
Naturalness
Sentiment

Table 3: Human evaluation dimensions

4 Design Specification

4.1 Transformer Architecture

LLMs are based on the transformer proposed by (Vaswani et al.; 2023). The transformer is based on the encoder decoder architecture. The encoder takes the sequence $(x_1, ..., x_n)$ as an input sequence and outputs the sequence $(z_1, ..., z_n)$. The encoder output is taken as an additional input by the decoder that produces the sequence $(y_1, ..., y_n)$ as its output. An overview of the transformer model architecture is shown in Fig. 2.

The proposed architecture consists of an encoder and a decoder stacks. The encoder stack consists of N=6 identical layers and each layer is divided into two sub-layers. The first sub-layer is a multi-head self-attention mechanism layer that takes the whole input sequence $(z_1, ..., z_n)$ as input at once. The second sub-layer is a position-wise Fully Connected Network (FCN). Each sub-layer is followed by a layer-normalization that has a residual connection as shown in Fig. 2. The decoder is similar to the encoder layer, however it has an additional multi-head attention sub-layer that takes encoder output as its input. The decoder layer also has layer-normalization with residual connection. Additionally, to ensure subsequent sequence only depends on encoder output and previous subsequence, decoder layer employs masking in its multi-head attention sub-layer.

Attention mechanism is at the core of transformer architecture. Attention can be described as function that takes three vectors query (Q), key (K) and value (V). These vectors are learned via model training. The output of the function calculated as per eq. 1. Dimensions for vectors Q and K are d_k and the dimension for V is d_v. In transformer architecture, attention scores are calculated for h times in parallel with different set of Q, K and V. After calculating attention scores for each heads, these scores are concatenated and normalized using softmax function as shown in eq. 3

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{2}$$

$$\text{where head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V). \tag{3}$$

Where the matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}, W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}, W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}, W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$.

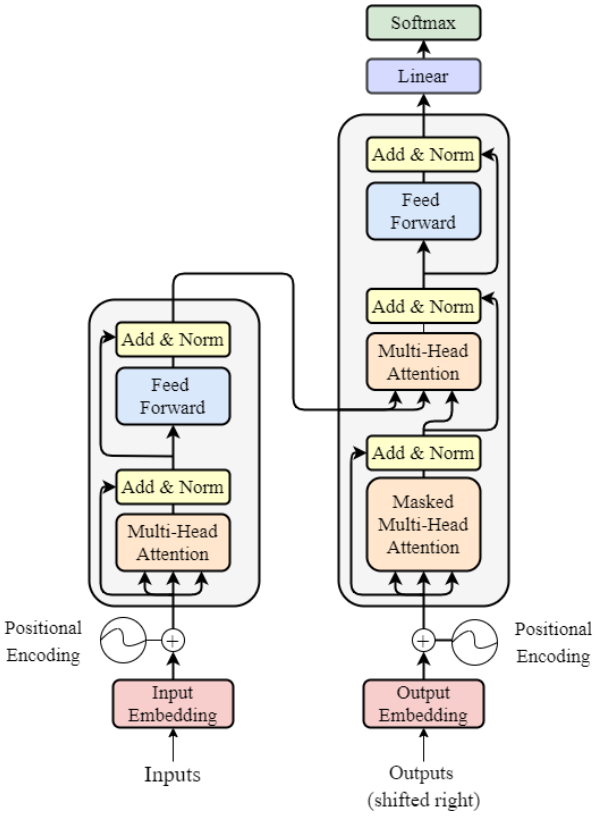


Figure 2: Architecture of Transformer model

4.2 Large Language Models Specifications

This paper fine-tunes 3 LLMs i.e. Phi-2, Gemma-2, Llama-2. These LLMs are freely available to download from HuggingFace website.

Phi-2 is a transformer model with 2.7 billion parameters, trained on python codes from *The Stack* (n.d.), question-answering content from (*Stack Overflow*; n.d.), programming contest dataset from (*google-deepmind/code_contests*; n.d.) and python content generated by GPT-3.5-turbo. The Phi-2 model is trained for the purpose of question-answering related to computer programming. Although Phi-2 model has achieved nearly state-of-the-art performance among models with 13 billion parameters or less. But Phi-2 model generates inaccurate code, another limitation is that Phi-2 is mainly trained on the python programming language based data. Also, Phi-2 is trained to understand the english language only (*microsoft/phi-2 · Hugging Face*; n.d.).

Gemma-2 is another light-weight decoder-only LLM built by Google. Gemma-2 is well suited for tasks such as question-answering, reasoning and summarization. This paper uses variant of Gemma-2 model that has 2 billion parameters. This variant of the LLM is trained english web documents to adopt variety of patterns, mathematical text

to strengthen the model with logical reasoning and computer programs to embed ability of code-generation ability in the LLM. As part of data preprocessing, extensive filtering of child-abusive-content was applied to eliminate the possibility of harmful content generation (*google/gemma-2-2b · Hugging Face*; 2024).

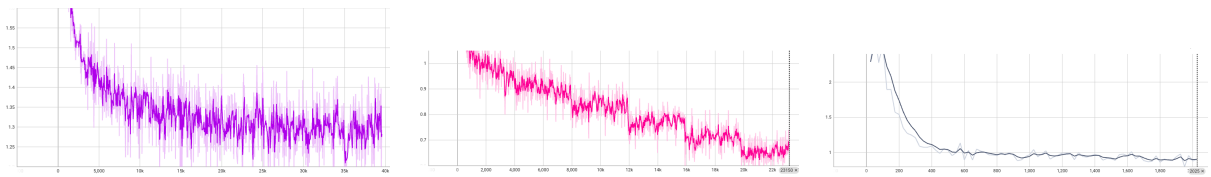
Llama-2 model, developed by Meta, is a collection models ranging from 7 billion to 70 billion parameter ranges. Llama-2 model is developed to perform number of NLP tasks however this paper uses Llama-2 model specialized for question-answering task. On human evaluation, Llama-2 model gives similar performance to models like ChatGPT and PaLM. For training of Llama-2 model, 3.3 million GPU hours on A100-80GB hardware is used (*NousResearch/Llama-2-7b-chat-hf · Hugging Face*; n.d.). Meta used publicly available data on the web to pretrain the Llama-2 model. For ethical reasons, they avoided using data from sources that contained private data (Touvron et al.; 2023).

5 Implementation

After acquiring the MedQUAD dataset from Github, data-rows with no answers are discarded and Text Rank algorithm is applied to generate extractive summary of the answers in the dataset. Later, prompts are formed in the LLM-specific format shown in Table 1. Afterwards, for each LLM, their specialized tokenizer is used to tokenize the prompts, the tokenized prompts is used in fine-tuning of each LLM. This paper generate extractive summary of maximum 100 words. A sample of prompts used for training of each LLM model is presented in Table 4. The summarized answer is much smaller in size in comparison to the actual answers available in the datasets. Summarization of text improves the computational efficiency for finetuning and model evaluation.

For fine-tuning, this paper uses LoRA rank of 64, drop-out probability of 0.1 for LoRA layers, learning rate of 2×10^{-4} and paged_adamw_32bit optimzer. For implementation of fine-tuning algorithm, this paper uses transformers, trl, torch, accelerate, peft, datasets, bitsandbytes and einops library available in python. In transfer learning, (Cireşan et al.; 2012) suggested finetuning of only final layers in Deep Neural Network (DNN) also produces good results. Therefore, to improve computational efficiency this paper freezes earlier layers of LLM models and finetune final attention layer only. After finetuning, LLM models generates text in similar pattern as prompts it was trained on.

Fig. 3 shows comparison of training loss. Small batch-size of 4 has caused fluctuation in the trainng loss however overall training loss is decreasing. Due to small number of parameters and less training data, plot shows stagnant progression in training for Phi-2. However, same is not true for Gemma-2 model. For Gemma-2 model, fluctuation in training loss is also less compare to Phi-2 model. For Llama-2 model, training loss became stagnant early just like Phi-2 model but fluctuation is extremely low despite of same batch-size as Gemma-2 and Phi-2 models.



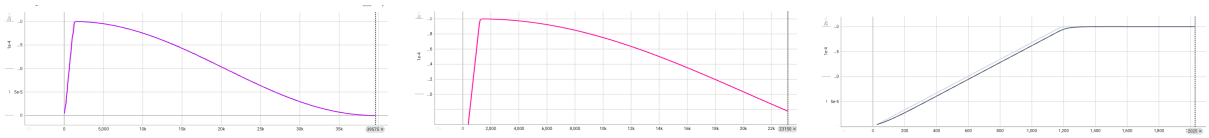
(a) Phi-2 model (b) Gemma-2 model (c) Llama-2 model

Figure 3: Training Loss Comparison

LLM	Prompt
Microsoft/Phi-2	Instruct: Do you have information about Ebola\nOutput: Summary : Ebola hemorrhagic fever is caused by a virus. It can affect humans and other primates. Symptoms of Ebola may appear anywhere from 2 to 21 days after exposure to the virus. However, if a person has the early symptoms of Ebola and there is reason to suspect Ebola, the patient should be isolated. Centers for Disease Control and Prevention
Google/Gamma-2	Instruct: What are the symptoms of Pediatric ulcerative colitis ?\nOutput: You can use the MedlinePlus Medical Dictionary to look up the definitions for these medical terms. Signs and Symptoms Approximate number of patients (when available) Abdominal pain - Diarrhea - Growth delay - Heterogeneous - Intestinal obstruction - Multifactorial inheritance - Recurrent aphthous stomatitis - Ulcerative colitis - Weight loss - The Human Phenotype Ontology (HPO) has collected information on how often a sign or symptom occurs in a condition. The frequency of a sign or symptom is usually listed as a rough estimate of the percentage of patients who have that feature. The first number of the fraction is how many people had the symptom, and the second number is the total number of people who were examined in one study. In these cases, the sign or symptom may be rare or common.
Llama-2	⟨s ⟩[INST] What are the symptoms of Pediatric ulcerative colitis ? [/INST] You can use the MedlinePlus Medical Dictionary to look up the definitions for these medical terms. Signs and Symptoms Approximate number of patients (when available) Abdominal pain - Diarrhea - Growth delay - Heterogeneous - Intestinal obstruction - Multifactorial inheritance - Recurrent aphthous stomatitis - Ulcerative colitis - Weight loss - The Human Phenotype Ontology (HPO) has collected information on how often a sign or symptom occurs in a condition. The frequency of a sign or symptom is usually listed as a rough estimate of the percentage of patients who have that feature. The first number of the fraction is how many people had the symptom, and the second number is the total number of people who were examined in one study. In these cases, the sign or symptom may be rare or common. ⟨\s ⟩

Table 4: Sample of Prompts

Fig. 4 depicts rate of change in learning-rate with progress in training. In theory, learning-rate changes to optimize training-time of models. For Phi-2 and Gemma-2, change in learning-rate is almost similar, they are increased earlier so loss can reach to optimal value quickly. Interestingly, for Llama-2 model, learning-rate increased linearly until learning-rate became optimal. The change in learning-rate is done within the library this paper used.



(a) Phi-2 model (b) Gemma-2 model (c) Llama-2 model

Figure 4: Learning-rate Comparison

6 Evaluation

This paper used ROUGE, BLEU and human evaluation for evaluation LLM models. ROUGE and BLEU scores are evaluated three times to ensure consistency in the scores. Tables 5, 6 and 7 report ROUGE and BLEU scores for three iterations. The reports show consistency in models performance on the dataset in terms of BLEU and ROUGE scores as reported values for each metrics are similar. Table 8 highlight average scores of three iterations for each models. Phi-2 and Gemma-2 models have achieved similar performance, however Llama-2 model has outperformed Phi-2 and Gemma-2 models on ROUGE and BLEU scores.

Evaluation Iteration	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BLEU**
1	0.127	0.115	0.127	0.071
2	0.144	0.132	0.144	0.086
3	0.140	0.128	0.140	0.081

Table 5: Phi-2 - ROUGE and BLEU scores for each evaluation iteration

Evaluation Iteration	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BLEU**
1	0.121	0.110	0.121	0.067
2	0.123	0.112	0.123	0.069
3	0.121	0.109	0.121	0.067

Table 6: Gemma-2 - ROUGE and BLEU scores for each evaluation iteration

Evaluation Iteration	ROUGE-1	ROUGE-2	ROUGE-L	BLEU
1	0.459	0.295	0.378	0.281
2	0.454	0.285	0.370	0.272
3	0.453	0.286	0.372	0.272

Table 7: Llama-2 - ROUGE and BLEU scores for each evaluation iteration

LLM	ROUGE-1	ROUGE-2	ROUGE-L	BLEU
Phi-2	0.137	0.125	0.137	0.079
Gemma-2	0.122	0.101	0.121	0.068
Llama-2	0.455	0.289	0.373	0.275

Table 8: Average ROUGE and BLEU Scores

(Snell et al.; 2024) discussed scaling model parameter size improves the model performance. Evidently, the result shown in the table 8 proves that increase in model size improves model performance as Gemma-2, Phi-2 and Llama-2 has 2 billion, 2.7 billion and 7 billion parameters respectively. Other than size of the model parameters, personal and non-ethical data was filtered out from the training dataset used for pre-training of Llama-2 model while similar pre-processing step was not taken for Phi-2 and Gemma-2 models. Quality of data could also have played role in better performance of the Llama-2 model, however more investigation needs to be carried out on this aspect.

Beside ROUGE and BLEU scores, this paper also uses human evaluation as a metric. The paper used Microsoft forms to get LLM-generated answers evaluated from 7 medical experts on parameters listed in Table 3. The Microsoft forms contained answers of five questions from the test dataset generated by each LLM and five medical experts assigned scores from 0 to 10 (10 being the best score) to the each answers. Table 9 presents average scores of five questions for each dimension. Similar to quantitative evaluation metrics, on average basis Llama-2 outperformed Phi-2 and Gemma-2 models. However on each dimension, each models are given similar scores. All three models are pre-trained on different datasets, but because they are fine-tuned on domain-specific datasets, they perform at similar scale in human evaluation. Further investigation needs to be done to analyze the effect of fine-tuning on human evaluation. It is possible that after fine-tuning, each model performs gives similar performance as they are trained curated dataset such as MedQUAD.

Automatic evaluation metrics do not evaluate in terms information quality. But human evaluation metric also has limitation of human judgment. Each evaluator can assign different range of scores for same answers. Human evaluation also suffers from the risk of random scoring by evaluators and subjective bias of evaluators. Therefore, neither automatic evaluation metrics nor the human evaluation metric alone can be used with reliability as an evaluation metric. However, both can be helpful in evaluating a Deep Learning as well as Large Language Models.

Human Evaluation Dimensions	Gemma-2	Phi-2	Llama-2
Informativeness	7.68	8.04	7.98
Relevance	7.76	7.36	7.24
Grammaticality	7.96	8.08	8.16
Naturalness	7.44	7.36	7.6
Sentiment	6.2	6.4	6.8
Average	7.408	7.448	7.492

Table 9: Human Evaluation of LLMs

7 Conclusion and Future Work

This paper evaluated the performance of Gemma-2, Phi-2 and Llama-2 on the MedQUAD dataset using ROUGE, BLEU and human evaluation as evaluation metric. Llama-2 outperformed Gemma-2 and Phi-2 models on the basis of these evaluation metrics. The human evaluation metric used for the evaluation has drawbacks of its own, such as limited evaluators used for evaluation of models and possibility of random evaluation by human evaluators.

As future work, data-efficient fine-tuning techniques can be used to improve computational efficiency for fine-tuning of LLM models. Also, a research methodology can be designed to integrate hallucination classifier to reduce the possibility of hallucination in these LLMs. Further investigation needs to be done analyze the correlation between model performance from the aspect of human evaluation and fine-tuning on the MedQUAD dataset.

References

Abacha, A. B. (2024). Medquad: A medical question answering dataset, `https://github.com/abachaa/MedQuAD`. A comprehensive dataset for medical question answering.

Abdallah, A., Kasem, M., Hamada, M. A. and Sdeek, S. (2020). Automated question-answer medical model based on deep learning technology, *Proceedings of the 6th International Conference on Engineering & MIS 2020*, ICEMIS'20, Association for Computing Machinery, New York, NY, USA.
URL: *https://doi.org/10.1145/3410352.3410744*

Abdul, A., Chen, B., Phani, S. and Chen, J. (2024). Improving preliminary clinical diagnosis accuracy through knowledge filtering techniques in consultation dialogues, *Computer Methods and Programs in Biomedicine* **246**: 108051.
URL: *https://www.sciencedirect.com/science/article/pii/S0169260724000476*

Ahmad, N., Mamatjan, E., Wali, T. and Mamatjan, Y. (2024). The development of canprompt strategy in large language models for cancer care, *2024 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–6.

Athira, B., Idicula, S. M., Jones, J. et al. (2024). An answer recommendation framework for an online cancer community forum, *Multimedia Tools and Applications* **83**: 173–199.
URL: *https://doi.org/10.1007/s11042-023-15477-9*

Basit, A., Hussain, K., Hanif, M. A. and Shafique, M. (2024). Medaide: Leveraging large language models for on-premise medical assistance on edge devices.
URL: *https://arxiv.org/abs/2403.00830*

Beattie, J., Neufeld, S., Yang, D., Chukwuma, C., Gul, A., Desai, N., Jiang, S. and Dohopolski, M. (2024). Utilizing large language models for enhanced clinical trial matching: A study on automation in patient screening, *Cureus* **16**(5): e60044.

Cireşan, D. C., Meier, U. and Schmidhuber, J. (2012). Transfer learning for latin and chinese characters with deep neural networks, *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6.

Das, B. and Nirmala, S. J. (2022). Improving healthcare question answering system by identifying suitable answers, *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, pp. 1–6.

Dettmers, T., Pagnoni, A., Holtzman, A. and Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms, *in* A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt and S. Levine (eds), *Advances in Neural Information Processing Systems*, Vol. 36, Curran Associates, Inc., pp. 10088–10115.
URL: *https://proceedings.neurips.cc/paper*$_f$*iles/paper*/2023/$file$/1feb87871436031bdc0f2$beaa$62a04
$Paper - Conference.pdf$

google-deepmind/code_contests (n.d.). original-date: 2022-01-31T09:48:14Z.
URL: *https://github.com/google-deepmind/code*$_c$*ontests*

google/gemma-2-2b · Hugging Face (2024).
URL: *https://huggingface.co/google/gemma-2-2b*

Gu, Z., Wang, Q., Li, F. and Ou, Y. (2021). Design of Intelligent QA for Self-learning of College Students Based on BERT, *Computer Technology and Transportation ISCTT 2021; 6th International Conference on Information Science*, pp. 1–5.
URL: *https://ieeexplore.ieee.org/document/9738905/?arnumber=9738905*

Haghighi, T., Gholami, S., Sokol, J. T., Kishnani, E., Ahsaniyan, A., Rahmanian, H., Hedayati, F., Leng, T. and Alam, M. N. (2024). Eye-llama, an in-domain large language model for ophthalmology, *bioRxiv [Preprint]* p. 2024.04.26.591355.

Ismail, A. R., Aminuddin, A. S., Nurul, A., Zakaria, N. A. and Fadaaq, W. H. N. (2024). A fine-tuned large language model for domain-specific with reinforcement learning, *2024 3rd International Conference on Creative Communication and Innovative Technology (ICCIT)*, pp. 1–6.

Kaddari, Z., Mellah, Y., Berrich, J., Bouchentouf, T. and Belkasmi, M. G. (2020). Biomedical question answering: A survey of methods and datasets, *2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS)*, pp. 1–8.

Lamichhane, P. and Kahanda, I. (2023). Enhancing health information retrieval with large language models: A study on medquad dataset, *2023 International Conference on Machine Learning and Applications (ICMLA)*, pp. 2147–2152.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries, *Text summarization branches out*, pp. 74–81.

Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H. and Liu, T.-Y. (2022). BioGPT: generative pre-trained transformer for biomedical text generation and mining., *Briefings in Bioinformatics* **23**(6): 1–11. Publisher: Oxford University Press / USA.
URL: *https://research.ebsco.com/linkprocessor/plink?id=71bea0f6-e2f3-3557-9682-b71ce58178fc*

microsoft/phi-2 · Hugging Face (n.d.).
URL: *https://huggingface.co/microsoft/phi-2*

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text, *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 404–411.

Mutabazi, E., Ni, J., Tang, G. and Cao, W. (2023). An improved model for medical forum question classification based on cnn and bilstm, *Applied Sciences* **13**(15).
URL: *https://www.mdpi.com/2076-3417/13/15/8623*

Nguyen, V., Karimi, S., Rybinski, M. and Xing, Z. (2023). MedRedQA for Medical Consumer Question Answering: Dataset, Tasks, and Neural Baselines, *in* J. C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti and A. A. Krisnadhi (eds), *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Nusa Dua, Bali, pp. 629–648.
URL: *https://aclanthology.org/2023.ijcnlp-main.42*

NousResearch/Llama-2-7b-chat-hf · Hugging Face (n.d.).
URL: *https://huggingface.co/NousResearch/Llama-2-7b-chat-hf*

Schick, T. and Schütze, H. (2021). It's not just size that matters: Small language models are also few-shot learners.
URL: *https://arxiv.org/abs/2009.07118*

Snell, C., Lee, J., Xu, K. and Kumar, A. (2024). Scaling llm test-time compute optimally can be more effective than scaling model parameters.
URL: *https://arxiv.org/abs/2408.03314*

Stack Overflow (n.d.).
URL: *https://stackoverflow.com/*

Subramanian, A., Schlegel, V., Kashyap, A. R., Nguyen, T.-T., Dwivedi, V. P. and Winkler, S. (2024). M-qalm: A benchmark to assess clinical reading comprehension and knowledge recall in large language models via question answering.

The Stack (n.d.). `https://huggingface.co/datasets/bigcode/the-stack`. Accessed: Dec. 11, 2024.

Touvron, H., Martin, L. and Stone, K. (2023). Llama 2: Open Foundation and Fine-Tuned Chat Models.

van der Lee, C., Gatt, A., van Miltenburg, E. and Krahmer, E. (2021). Human evaluation of automatically generated text: Current trends and best practice guidelines, *Computer Speech & Language* **67**: 101151.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I. (2023). Attention is all you need.
URL: *https://arxiv.org/abs/1706.03762*

Wang, J., Yang, Z., Yao, Z. and Yu, H. (2024). Jmlr: Joint medical llm and retrieval training for enhancing reasoning and professional question answering capability.
URL: *https://arxiv.org/abs/2402.17887*

Yagnik, N., Jhaveri, J., Sharma, V. and Pila, G. (2024). Medlm: Exploring language models for medical question answering systems.

Yang, X., Chen, A., PourNejatian, N. et al. (2022). A large language model for electronic health records, *npj Digital Medicine* **5**: 194.
URL: *https://doi.org/10.1038/s41746-022-00742-2*

Yu, H., Yu, C., Wang, Z., Zou, D. and Qin, H. (2024). Enhancing healthcare through large language models: A study on medical question answering.