Information Retrieval vs Cache Augmented Generation vs Fine Tuning: A Comparative Study on Urdu Medical Question Answering

Ahmad Mahmood[1], Zainab Ahmad[1], Iqra Ameer[2], and Grigori Sidorov[1]

[1]Instituto Politécnico Nacional (IPN), Centro de Investigación en Computación(CIC), Mexico City, Mexico
[2]Division of Science and Engineering, The Pennsylvania State University, Abington, PA, USA

Abstract The development of medical question-answering (QA) systems has predominantly focused on high-resource languages, leaving a significant gap for low-resource languages like Urdu. This study proposed a novel corpus designed to advance medical QA research in Urdu, created by translating the benchmark MedQuAD corpus into Urdu using the Generative AI-based translation technique. The proposed corpus is evaluated using three approaches: (i) Information Retrieval (IR), (ii) Cache-Augmented Generation (CAG), and (iii) Fine-Tuning (FT). We conducted two experiments, one on a 500-instance subset and another on the complete 3,152-question corpus, to assess retrieval effectiveness, response accuracy, and computational efficiency. Our results show that JinaAI embeddings outperformed other IR models, while OpenAI 4o mini, FT achieved the highest response accuracy (BERTScore: 70.6%) but is computationally expensive. CAG eliminates retrieval latency but requires high resources. Findings suggest that IR is optimal for real-time QA, Fine-Tuning ensures accuracy, and CAG balances both. This research advances Urdu medical AI, bridging healthcare accessibility gaps.

Keywords: Information retrieval, retrieval-augmented generation, cache-augmented generation, fine-tuning, Urdu medical question-answering

1 Introduction

Medical question-answering (QA) systems [16] facilitate the healthcare practitioner and the general public by providing quick and reliable access to medical knowledge. Most of the professionals were using Google [7] to find evidence for their query instead of finding it through credible resources like PubMed[1] and UpToDate[2]. The main reason behind this was the time constraint. To address this problem, researchers have developed Medical QA systems for high-resource languages such as English [2, 12, 13, 17, 21], while ignoring low-resource languages such as Urdu. Some studies have developed a multilingual system to cater diverse range of languages like Spanish, Arabic, Chinese, German, French, etc. However, millions of Urdu-speaking individuals face barriers when seeking healthcare knowledge, as no study, to the best of our knowledge, has included the Urdu language.

The Major challenge in the development of a medical QA system for Urdu is the lack of a structured, and domain-specific corpus. Unlike high-resource languages like English, where corpora, such as MedQuAD, are beneficial in the development of medical QA models, whereas Urdu lacks equivalent corpora, making it a bit difficult to train and evaluate models effectively. Moreover, when performing direct translation from English to Urdu language often changes medical terminology, as Urdu incorporates Persian, and Arabic loanwords [20], which must be preserved accurately in technical contexts. Furthermore,

[1] `https://pubmed.ncbi.nlm.nih.gov/` Last visited: 02-26-2025
[2] `https://www.wolterskluwer.com/en/solutions/uptodate` Last visited: 02-26-2025

David C. Wyld et al. (Eds): NLCAI, AIFU, CCSEA, BIoT, SEA, SIPRO, BDML, CLOUD – 2025
pp. 35-50, 2025.CS & IT- CSCP 2025 DOI: 10.5121/csit.2025.151003

the morphological complexity and syntactic variations of Urdu language create challenges for information retrieval-based models, where ultimately the word order and script variations impact search accuracy. The existing Generative AI and large language models (LLMs) are not fine-tuned for Urdu medical QA, limiting their ability to generate precise and medically good responses. These challenges highlight the urgent need for a dedicated Urdu medical QA system, addressing both linguistic and computational constraints to improve healthcare accessibility for Urdu-speaking populations.

General QA [1, 15] systems are built for the Urdu language but the need for a dedicated Medical QA system is still there. Urdu is 10th[3] spoken language by approximately 238 million people worldwide. Urdu is spoken primarily in Pakistan, India, and among communities of the South Asian diaspora. English comprises more than 49% of the online content[4], on the contrary, Urdu constitutes less than 0.1%, making access to specialized knowledge highly limited. Urdu-speaking physicians and healthcare professionals struggle to access specialized medical literature, as most digital medical resources and QA systems are built for English-speaking users. This lack of medical AI tools for Urdu not only affects health literacy but also limits access to healthcare for large populations. Developing a dedicated Urdu medical QA system is essential to bridge this linguistic gap, ensuring equitable access to medical knowledge, and improving healthcare outcomes for Urdu-speaking communities.

To address this challenge, this study aims to enhance access to healthcare information for Urdu-speaking populations by developing and evaluating Urdu medical QA systems. The key research objectives are:

1. Development of an Urdu medical QA corpus, hereafter called Urdu-MedQuAD-25, by translating a subset of the MedQuAD benchmark corpus into Urdu. The translation process ensures the preservation of medical terminology through a combination of generative AI models and human verification. This corpus serves as the foundational resource for developing and evaluating Urdu medical QA systems.
2. Develop, evaluate, and compare different methodologies for Urdu medical QA, focusing on:
   - **Information Retrieval (IR) Approach**: Implementing a retrieval-based QA system that fetches relevant medical information using embedding models. The effectiveness of retrieval techniques is assessed based on Precision@k metrics.
   - **Cache-Augmented Generation (CAG) Approach**: Enhancing response efficiency by caching medical knowledge into model memory, reducing retrieval latency, and improving inference time.
   - **Fine-Tuning-based (FT) Approach**: Fine-tuning LLMs, such as OpenAI 4o Mini and LLaMa 3.2-1B, on the Urdu-MedQuAD-25 corpus to improve response generation accuracy and contextual understanding.
3. Benchmarking the effectiveness of these approaches by evaluating retrieval precision (Precision@k) for IR models and semantic similarity (BERTScore) for generative models. Additionally, we assess computational efficiency by measuring inference time across different methods.

This study sets a new benchmark for an under-resourced Urdu language, contributing to linguistically inclusive healthcare AI. Furthermore, the comparative analysis of three prominent methodologies identifies the most suitable, effective, and efficient approach for the task of medical QA, specifically and in general.

[3] `https://lilata.com/en/blog/most-spoken-languages-in-the-world/` Last Visited: 02-27-2025
[4] `https://en.wikipedia.org/wiki/Languages_used_on_the_Internet` Last Visited: 02-27-2025

The rest of this paper is structured as follows: Section 2 presents the related work for medical QA task using the MedQuAD corpus. Section 3 explains the complete corpus generation process from the selection of the corpus to the translation, and validation of our Urdu-MedQuAD-25 corpus. Section 4 provides a detailed explanation of the three approaches compared in this study, IR, CAG, and Fine-Tuned LLMs, and the metrics used to evaluate and compare these approaches. Section 5 presents the results and their analysis. Section 6 extends the study's limitations. Finally, Section 7 concludes the study by emphasizing the importance of medical QA for low-resource languages and also provides the future direction.

2 Related work

In recent studies, large language models (LLMs) are extensively used for the task of QA systems. The benchmark corpora for the task of medical QA are MedQuAD [2], PubMedQA [12], MASH-QA [13], BioASQ [17], and MedMCQA [21]. Some of the studies have developed multilingual corpora as well to cater to more than one language, such as Spanish, Chinese, and Arabic. The multilingual corpora include MedQA-USMLE [11], MMedBench [23], TM-PathVQA [25], and BiMed1.3M [22]. Researchers have proposed different techniques, such as traditional machine learning, clustering, Information Retrieval-based, and FT approaches, to improve the performance of QA systems in the healthcare domain.

Most of the studies have employed the MedQuAD benchmark corpus to develop and evaluate their proposed approaches. Harikrishnan et al. [6] presented a comparative analysis of BERT-based pipelines on the MedQuAD and SQuAD corpus [26]. The proposed approach achieved 60% accuracy on MedQuAD and performed significantly better on SQuAD by obtaining an accuracy of 79%. As SQuAD is a general-purpose corpus, the results depict that domain-specific QA systems are more challenging. The main reason for the lower performance of MedQuAD is complex medical terminologies and context.

Cho and Lee [4] introduced a retrieval-augmented technique to improve the performance of the QA system. Their proposed approach, called K-COMP, stands for "knowledge-injected compressor". This study presented experiments on the corpus including MedQuAD, MASH-QA, and BioASQ using the evaluation measure BertScore and UniEval. The proposed study outperformed traditional RAG methods. The study utilized general purpose (LLaMA, Mixtral, GPT-4) and medical purpose (MedAlpaca, Meditron) LLMs. The best BertScore achieved on MedQuAD, MASH-QA, and BioASQ are 85%, 83%, and 86%, respectively.

Kang et al. [14] proposed PRISM-Med, a parameter-efficient model designed to address domain adaptation challenges in medical QA. Their framework employs supervised domain classification and Low-Rank Adaptation (LoRA) modules to improve domain-specific knowledge retention while minimizing computational overhead. For the QA task, the study evaluated the technique on MedQuAD and MedMCQA using ROUGE-F1 scores. The best results obtained on MedQuADa, achieving the ROUGE-F1 of 42%, outperformed the traditional fine-tuning, achieving up to a 10.1% improvement, which reflects the effectiveness of parameter-efficient models in handling complex medical queries.

Lakatos et al. [18] investigated the effectiveness of RAG and Domain-Specific Fine-Tuning. This study compared these approaches using four LLM, including GPT-J-6B, OPT-6.7B, LLaMA, and LLaMA-2, using the MedQuAD and CORD-19 [32] corpora. In addition to evaluation measures such as ROUGE, BLEU, and METEOR, the techniques were evaluated and compared with the proposed Coverage Score (CS) based on cosine similarity. The findings indicated that RAG-based systems outperformed Fine-Tuning by 17%

in ROUGE, 13% in BLEU, and 36% in CS, highlighting the advantages of retrieval-based methods in reducing hallucinations and improving factual accuracy. However, Fine-tuning showed minor advantages in METEOR, suggesting slightly better creative flexibility in generating responses. The study not only compared both techniques but explored the integration of RAG and Domain-Specific Fine-Tuning. The results depicted the degradation in performance when the methods were combined. This indicates that there is a need for an optimized balance between RAG and fine-tuning.

Vazrala and Mohammed [31] proposed a hybrid gradient regression-based transformer model (RBTM), designed to enhance biomedical question-answering. The approach integrated gradient regression techniques with transformer architectures to improve contextual understanding and response accuracy in the biomedical domain. Feature extracted using LemmaChase Lemmatizer, for domain-specific concept identification utilized SNOMED-CT ontology, and to generate the vectors of input phrases used the concept2Vec approach. The proposed approach was evaluated on the MedQuAD corpus. The proposed RBTM approach achieved 99.09% of remarkable accuracy compared to the existing approaches.

Two recent studies have explored model uncertainty and hallucinations in LLMs for QA tasks. SycEval [5] evaluated sycophancy in LLMs when answering medical and mathematical questions. They used the MedQuAD for the medial QA. The study analyzed the top-notch LLMs, Claude, ChatGPT, and Gemini. Results depicted Gemini has the highest sycophancy rate of 62% and the lowest sycophancy rate 56% has been achieved by ChatGPT. Moreover, Vazhentsev et al. [30] introduced a density-based uncertainty quantification method using the Mahalanobis Distance, as the previous study [29] investigated sequence-level density-based methods proven ineffective. The study improved the performance across text summarization, Long and short QA, and MCQ tasks. The LLaMA and Gemma were used for selective generation while the study utilized the Mistral for fact-checking. The best mean rank for selective generation was 1.71, across all domains. Similarly, for fact-checking the best results obtained were an ROC-AUC of 0.750 and a PR-AUC of 0.410.

The Singhal et al. [27] presented the Med-PaLM 2 approach to develop an expert-level medical QA. Med-PaLM 2 worked by enhancing the base language model, fine-tuning it for the medical domain, and utilized techniques like ensemble refinement and chain of retrieval to improve reasoning and accuracy. Evaluated the proposed approach on different benchmark, and the MedQA achieved the best result by obtaining 86.5% accuracy.

Moreover, unlike previous studies that focused on general medical QA, Racha et al. [24] worked on mental health QA (MHQA). Developed two mental health QA corpora named as MHQA-Glod and MHQA-B, each comprising 2475 and 58.6k multiple-choice QA pairs. The corpora span four critical mental health domains including anxiety, depression, trauma, and obsessive-compulsive disorder (OCD). The domain integrated factoid, diagnostic, prognostic, and preventive questions, making it a comprehensive resource for evaluating LLM reasoning abilities in mental health contexts. LLaMA-3, GPT-3.5, and GPT-4o were fine-tuned and evaluated on the proposed corpora. The GPT-4o reported the highest F1 score of 79.8%.

As the literature illustrates, the MedQuAD corpus has shown significant progress in English medical QA, but there remains a gap in low-resource language applications. Existing studies primarily focus on English corpora, with limited attention to medical QA in low-resource languages such as Urdu. The hindrance to the development of a corpus for the Urdu language is the complex linguistic and medical terminology challenges. There is a need for domain-adapted models trained on the Urdu medical QA corpus. This study presents a corpus for Urdu-MedQuAD-25 to develop, evaluate, and compare state-

of-the-art methods for the medical QA system task to facilitate Urdu-speaking doctors and practitioners.

3 Corpus Generation Process

The primary goal of this study is to develop a corpus for the Urdu medical QA system. The process of developing the proposed Urdu-MedQuAD-25 corpus includes (i) selecting a source corpus, (ii) generative AI models for translation, and (iii) human verification.

3.1 Source Corpus

In developing a corpus for the medical QA system in the Urdu language, the benchmark MedQuAD was selected as a source corpus. A subset of 3152 question-answer pairs were extracted from the source corpus, covering a variety of focus areas including, glaucoma, high blood pressure, lung cancer, stroke, and heart attack (see Figure 1).

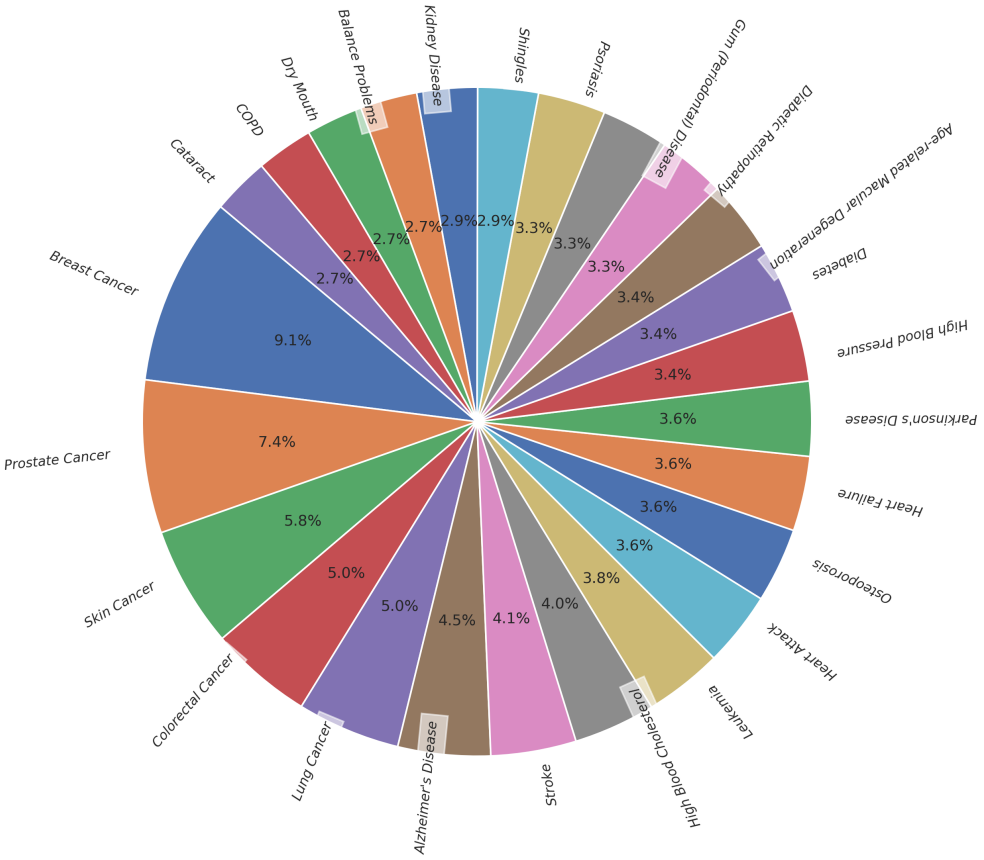

Figure 1. Focus Areas in Proposed Urdu-MedQuAD-25 Corpus

3.2 Generative AI: A Translation Tool

Accurate and reliable translation is crucial to ensure the quality of the corpus, especially for the domains like healthcare. Given the complexities of medical terminologies and the need for linguistic precision, a comparative analysis of multiple generative AI models such

as LLaMa [28], Mistral [10], and OpenAI [9] was performed to assess their effectiveness. The models were evaluated considering key factors such as translation accuracy, contextual understanding, preservation of medical terminology, and the overall fluency in the target Urdu language.

After analysis, OpenAI 4o Mini appeared as the most suitable AI translation tool for this research study. The choice was driven by its ability to produce high-quality translations while maintaining cost-effectiveness, making it a practical solution for large-scale data generation models.

3.3 Human Verification

While generative AI facilitated the translation process, ensuring linguistic accuracy and reliability requires human Inter-mediation to maintain the quality of the proposed corpus. To refine the translations, a manual review was conducted by native Urdu-speaking medical students. Their expertise allowed the identification and correction of translation errors that could impact the clarity and precision of medical information.

By integrating human verification into the translation pipeline, we enhanced the reliability of the proposed Urdu-MedQuAD-25 corpus. This step ensures that the translated content remains both medically sound and linguistically coherent, ultimately contributing to improved healthcare accessibility for Urdu-speaking populations. Figure 2 provides the sample instance of the proposed corpus.

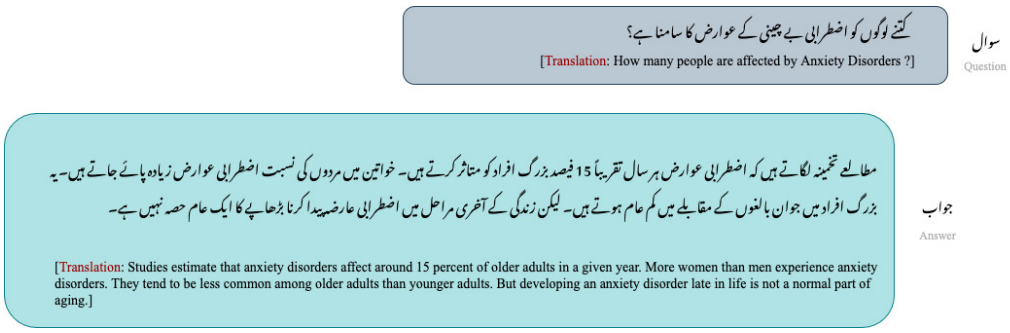

Figure 2. Sample question-answer pair form proposed Urdu-MedQuAD-25 Corpus

3.4 Corpus Characteristics

Table 1 shows the main characteristics of the proposed Urdu-MedQuAD-25 corpus. The corpus comprises 3,152 question-answer pairs. The question and answer have an average of 7 and 128 words, respectively. The total number of words in the question is 21,754 and the total number of words in the answer is 4,05,083. The maximum number of words in the question and answer are 21 and 199 words, respectively. The minimum number of words in the question and answer is 3 and 10 words, respectively.

4 Experimental setup

To demonstrate how the proposed Urdu-MedQuAD-25 corpus can be used to develop, evaluate, and compare the Urdu Medical QA system. This study applied different approaches including **(i)** IR (Information Retrieval), employed to evaluate the system at the

Table 1. Characteristics of Urdu-MedQuAD-25 Corpus

Attributes	**Urdu Question**	**Urdu Answer**
Total Count	3,152	3,152
Total Words	21,745	405,083
Avg. No. of Words	7	128
Minimum Words	3	10
Maximum Words	21	199

retrieval level. While utilizing a generation module to analyze qualitative outputs and assess results at a basic human level, this aspect was not included in the formal evaluation, **(ii)** integrated CAG (Cache Augmented Generation), a state-of-the-art approach, which has demonstrated significant improvements in accuracy and efficiency in prior studies, and **(iii)** FT (fine-tuned) pre-trained LLMs on the Urdu-MedQuAD-25 corpus to enhance domain-specific adaptation for medical QA in Urdu. FT allows the model to better understand medical terminology and contextual nuances.

4.1 Information Retrieval

Retrieval-augmented generation (RAG) [19] enhances QA systems by integrating an information retrieval (IR) component with generative modeling. Instead of relying solely on a pre-trained model, RAG retrieved relevant contextual information from the Urdu-MedQuAD-25 corpus before generating responses. This retrieval step is crucial to ensure that the model generates factually accurate and domain-specific answers, particularly in medical QA. However, the effectiveness of RAG is entirely dependent on the quality of its retrieval mechanism. If retrieval fails to fetch relevant information, the generative component will be unable to produce reliable answers, making the system ineffective for domain-specific tasks. Thus, retrieval quality is the key determinant of RAG's success in medical QA, and our evaluation focuses exclusively on the retrieval component.

To assess retrieval effectiveness, we stored and searched document embeddings using Faiss[5] (Facebook AI Similarity Search), a high-performance library optimized for large-scale vector retrieval. The choice of embeddings significantly impacts retrieval accuracy; therefore, we experimented with multiple multilingual embedding models, including BAAI/bge-m3[6] embeddings, JinaAI/jina-embeddings-v3[7] embeddings, All-MiniLM-L12[8] (a lightweight multilingual model), and DistilBERT-QA[9] (a distilled version of BERT optimized for question-answering tasks). By comparing these embedding models, we aim to determine which one provides the most effective document representations for medical QA in Urdu. We evaluated retrieval performance based on Precision@1, 3, 5, 7, and 10, ensuring that the highest-ranked retrieved documents align with expected medical knowledge.

To comprehensively analyze retrieval performance, we conducted 8 experiments using four different embedding models. The experiments were structured as:

1. **Experiment 1**: four experiments on a 500 question-answer subset of our proposed Urdu-MedQuAD-25 corpus, allowing for a time comparison with the CAG approach.

[5] `https://faiss.ai/` Last visited: 02-26-2025
[6] `https://huggingface.co/BAAI/bge-m3` Last visited: 02-26-2025
[7] `https://huggingface.co/jinaai/jina-embeddings-v3` Last visited: 02-26-2025
[8] `https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2` Last visited: 02-26-2025
[9] `https://huggingface.co/sentence-transformers/multi-qa-distilbert-cos-v1` Last visited: 02-26-2025

2. **Experiment 2**: four additional experiments on the complete Urdu-MedQuAD-25 corpus using four models.

Additionally, we further divided our Urdu-MedQuAD-25 corpus based on word count in the answers, where one group contained answers ranging from 1 to 99 words, and the other contained answers between 100 to 199 words. This categorization allowed us to evaluate the RAG retrieval component against itself, analyzing how retrieval effectiveness varies with different answer lengths and corpus sizes. This evaluation is critical because the entire RAG pipeline fails to deliver reliable results in a domain-specific setting if retrieval is suboptimal. By optimizing retrieval, we aim to maximize the effectiveness of RAG for Urdu-language medical QA and explore how retrieval performance scales with increasing document size and word count variations.

4.2 Cache Augmented Generation

Cache Augmented Generation (CAG) [3] is an alternative to RAG that eliminates real-time retrieval by pre-loading relevant knowledge into the model's extended context, unlike RAG, which retrieves external documents during inference, CAG stores and caches knowledge offline, ensuring that all necessary information is readily available for query processing without retrieval overhead.

RAG and IR systems often struggle with retrieval latency, incorrect document selection, and system complexity. These issues can degrade the quality of generated responses, especially in medical QA, where factual accuracy is paramount. CAG addresses these limitations by leveraging long-context LLMs capable of processing large amounts of preloaded text in a single inference step. This method ensures that the model has direct access to domain-specific knowledge, reducing dependency on real-time retrieval.

In proposed setup, we implemented CAG using the LLaMA, a quantized version (LLaMa3.2_1B_Instruct[10]) model, where we preloaded 500 question-answer pairs from the Urdu-MedQuAD-25 corpus into the model's extended context window. Each answer in the corpus contains between 1 to 99 words, ensuring a diverse range of response lengths while fitting within the model's context length. By pre-computing key-value (KV) caches, we allowed the model to generate responses directly based on the pre-loaded content, eliminating IR failures and reducing latency. During inference, the system only processed the user query, leveraging the stored knowledge without additional retrieval steps.

By bypassing the retrieval component, CAG simplifies the system architecture, improves response time, and ensures consistency across multiple queries. The trade-off is that it requires a manageable and well-structured knowledge base that fits within the model's context length. However, with advancements in LLMs offering larger context windows, CAG is emerging as a more efficient approach for knowledge-intensive tasks, particularly in specialized domains such as medical QA.

Experiment 1 evaluates CAG against IR and FT, assessing whether pre-loading domain-specific knowledge can outperform IR-based approaches in Urdu medical QA. While each approach is evaluated differently: IR is measured using retrieval precision while CAG and FT are assessed with BERTScore. However, we can still compare them based on efficiency, specifically the time taken for each method. A detailed analysis and results are provided in the Experiment section.

[10] `https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct` Last visited: 02-26-2025

4.3 Fine Tuning: Large Language Models

Fine Tuning (FT) enhances the performance of pre-trained LLMs by adapting them to domain-specific tasks. Unlike RAG and IR, which rely on retrieval, and CAG, which pre-loads knowledge, FT directly updates the model's weights, allowing it to internalize medical knowledge and generate responses without external retrieval or pre-loaded context. For our Urdu Medical QA system, we fine-tuned LLaMA(LLaMa3.2_1B_Instruct) and OpenAI (gpt-4o-mini[11]) models using the Urdu-MedQuAD-25 corpus. To optimize efficiency, we utilized a quantized version of LLaMA, reducing memory usage while maintaining strong performance. This allowed us to fine-tune on standard consumer-grade GPUs without compromising the model's capability. Additionally, we applied Low-Rank Adaptation (LoRA) [8] under the PEFT (Parameter-Efficient Fine-Tuning) framework, which selectively updates only a subset of parameters. This method significantly reduces memory consumption and accelerates training while ensuring effective domain adaptation.

The proposed FT process involved two stages: initially, we trained the model on 500 question-answer pairs, followed by fine-tuning on the complete Urdu-MedQuAD-25 corpus to maximize its understanding of medical terminology and QA patterns in Urdu. Since CAG also uses the 500 question-answer pairs, this enables a direct comparison between FT and CAG in terms of response quality and time to generate responses. To evaluate performance, we employed BERTScore, which measures semantic similarity between generated responses and ground-truth answers. FT offers a more adaptable solution compared to IR and CAG, as it learns from the corpus rather than depending on external retrieval. However, it requires higher computational costs, longer training times, and has a risk of memorization, which can affect generalization to unseen queries. A detailed comparison of FT, CAG, and IR, including both accuracy and efficiency, is discussed in the next section.

Two sets of experiments are designed, the first utilizes a subset of 500 instances from the corpus to provide a detailed analysis on a smaller scale, while the second leverages the complete corpus to evaluate performance across the entire corpus.

4.4 Evaluation Metrics

To evaluate the performance of our approaches on the proposed Urdu-MedQuAD-25 corpus, we employed two distinct evaluation measures tailored to the specific tasks. For IR-based models, we utilized Precision@1, 3, 5, 7, and 10 to assess how accurately the retrieved documents align with the expected relevant information, thereby quantifying the effectiveness of the retrieval process. In contrast, generation-based approaches like CAG and FT, relied on BERTScore [33] as the primary metric. BERTScore measures the similarity between generated and actual answers and provides an overall assessment of semantic alignment. This combined evaluation strategy allows us to conduct a comprehensive and fair comparison between retrieval and generation methods in the Urdu medical QA domain.

5 Result and Analysis

We designed two sets of experiments to evaluate the performance of retrieval-based and generation-based approaches for Urdu Medical QA. In the first experiment, we used a subset of 500 instances from the Urdu-MedQuAD-25 corpus to analyze retrieval effectiveness and response quality on a smaller scale, which allowed us to have a controlled

[11] `https://platform.openai.com/docs/models/gpt-4#gpt-4o-mini`Last visited: 02-26-2025

comparison between IR, CAG, and FT while focusing on precision, generation accuracy, and computational efficiency.

In Experiment 2, we extended the evaluation to the complete corpus, allowing us to have better insights into how retrieval and fine-tuned models perform and whether retrieval accuracy continues to influence generated response quality.

Tables 2 and 5 present the performance evaluation of IR in Experiment 1 and Experiment 2. The evaluation is based on Precision@1, Precision@3, Precision@5, Precision@7, and Precision@10, which measures how well the model can retrieve the relevant answers. "Word Count" categorizes the QA pairs based on the answer length, while "QA Pairs" represent the corpus instances used for evaluation. "Embedding Model" refers to the model used to extract the embeddings from the QA pairs. Finally, "Inference Time" provides the computational efficiency of each embedding model, the time taken to retrieve the relevant answers for the given question(s).

Similarly, Table 3 presents the performance evaluation of the CAG applied in Experiment 1. The Model column specifies the LLM used for generating the responses. Word Count categorizes the QA pairs (a.k.a Knowledge Cache) based on the length of the answers. QA Pairs represents the number of question-answer instances stored in the model's extended context for generation. The "Average BERT Score" measures the semantic similarity between the generated responses and the ground-truth answers, reflecting model's accuracy. Lastly, Inference Time provides the computational efficiency of the CAG approach, indicating the total time taken by the model to generate the response based on the preloaded knowledge.

Additionally, Tables 4 and 6 present the performance evaluation of the FT models applied in Experiment 1 and Experiment 2. The Model column lists the LLMs used for the fine-tuning, QA pairs represents the number of instances used for the fine-tuning, while Word Count specifies the length of the generated responses. The Average Bert Score evaluates the semantic similarity between the response generated and the ground truth, which serves as a measure of response quality. Finally, Inference Times represents the computational efficiency of the fine-tuned models, highlighting the time required for each model to generate the responses.

These experiments were conducted on a system that is equipped with an Intel(R)-i7-CPU@4.00 GHz, 67.34 GB of RAM and NVIDIA Quadro RTX 8000 GPU with 50.94 GB of VRAM. The system was configured with CUDA-enabled, ensuring the efficient execution of LLMs and Retrieval-based models.

5.1 Experiment 1: Retrieval Effectiveness (IR) vs Response Quality (CAG & FT)

To evaluate the impact of retrieval quality in IR compared to response generation in CAG and FT, we experimented on a subset of 500 question-answer pairs from the Urdu-MedQuAD-25 corpus. This subset allowed a direct comparison in terms of retrieval precision, response quality, and computational efficiency between the three approaches.

Retrieval Performance in IR: For the IR retrieval component, we tested multiple embedding models to assess how well they retrieve relevant medical knowledge. The performance of the retrieval method was evaluated using Precision@1, 3, 5, 7, and 10, to have a comparison of approaches in terms of retrieval, response quality, and computational efficiency. The BAAI and JinaAI embeddings were evaluated at different word count ranges, providing insights into their retrieval effectiveness. Results are represented in Table

2. BAAI achieved a Precision@1 of 76.2% for shorter answers (1-99 words), indicating strong retrieval accuracy when the target answer is concise. However, as answer length increased (100-199 words), Precision@1 dropped to 71.4%, suggesting a decline in retrieval effectiveness for longer responses. Similarly, JinaAI embeddings outperformed BAAI, achieving a higher Precision@1 of 77.8% for 1-99 word answers, demonstrating better retrieval capabilities for shorter responses. However, for longer responses (100-199 words), Precision@1 dropped to 72.8%, following a similar trend of reduced retrieval performance with increased answer length. Across models, retrieval accuracy was consistently higher for shorter answers, showing that retrieval models struggle as answers become more detailed and complex.

Table 2. IR Model Performance (Exp 1)

Embedding Model	**BAII**		**JINAAI**	
Word Count	1-99 words	100-199 words	1-99 words	100-199 words
QA Pairs	500			
Precision@1	76.2	71.4	77.8	72.8
Precision@3	87.0	80.8	89.0	83.2
Precision@5	90.4	84.8	91.0	87.4
Precision@7	91.4	87.4	92.4	88.2
Precision@10	93.2	89.0	93.2	90.0
Inference Time	18.1833	16.7779	63.3347	63.1169

Response Quality in CAG vs FT: Unlike IR, CAG, and FT directly generated responses, and their performance was evaluated using BERTScore, which measures the semantic similarity between generated and ground-truth answers. Results obtained for CAG and TF are represented in Table 3 and 4 respectively. CAG with LLaMa 3.2-1B-Instruct achieved a BERTScore of 50.85%, indicating moderate response quality. However, CAG was significantly faster compared to FT (LLaMa) methods, highlighting its efficiency advantage. Furthermore, FT on OpenAI 4o mini significantly outperformed CAG, achieving a BERTScore of 66.87%, demonstrating its ability to generate more accurate responses. However, the time required for inference was much higher, taking 916.11 seconds, compared to 63.98 seconds for IR-based models. LLaMa 3.2-1B-Instruct, when fine-tuned, achieved a lower BERTScore of 45.76%, underperforming compared to OpenAI's fine-tuned model. Additionally, its inference time was significantly longer, taking 11,021.99 seconds, making it the least efficient of all tested approaches.

Comparing Efficiency: IR vs CAG vs FT: Efficiency plays a crucial role in the feasibility of each approach. IR approach has proven to be the fastest, with response times ranging from 16.77s (BAAI, 100-199 words) to 63.33s (JinaAI, 1-99 words), making it suitable for real-time applications. On the other hand, CAG took significantly longer (9823.27s) but eliminated retrieval latency, making it a viable option when fast retrieval is not required. On the contrary, FT on OpenAI 4o mini offered the best response quality (BERTScore 66.87) but required 916.11s, making it computationally expensive. FT on LLaMa 3.2-1B-Instruct further increased inference time to 11,021.99s, making it impractical for real-time applications.

Table 3. CAG Model Performance (Exp 1)

Model	Word Count	QA Pairs	Average Bert Score	Inference Time
LLaMa 3.2-1B-Instruct	01-99 words	500	50.85	9823.2799

Table 4. FT Model Performance (Exp 1)

Model	QA Pairs	Words count	Average Bert Score	Inference Time
OpenAI 4o mini	500	0-99 words	66.87	916.1102
LLaMa 3.2-1B-Instruct	500	0-99 words	45.76	11021.9955

5.2 Experiment 2: IR vs FT

This experiment evaluated the performance of IR and FT on the complete Urdu-MedQuAD-25 corpus, focusing on their retrieval effectiveness, response quality, and efficiency. The corpus used in this evaluation includes 3,152 question-answer pairs, allowing us to analyze how retrieval and FT approaches scale with large corpora.

Retrieval Performance in IR: For IR, we conducted four retrieval experiments using different embedding models: BAAI, JinaAI embeddings, All-MiniLM-L12, and DistilBERT-QA. The retrieval effectiveness was measured using Precision@1, 3, 5, 7, and 10, providing insights into how accurately the models retrieve relevant medical knowledge. The results obtained from the full corpus for IR and FT are presented in Table 5 and 6, respectively. BAAI achieved a Precision@1 of 64.81% for the full corpus (3,152 questions), with retrieval accuracy improving as more documents were retrieved (Precision@10: 90.9%). While, JinaAI embeddings outperformed BAAI, achieving a Precision@1 of 73.3%, showing better retrieval accuracy across all levels. Conversely, All-MiniLM-L12 and DistilBERT-QA performed significantly worse, with Precision@1 dropped to 2.95% and 2.25%, indicating poor retrieval effectiveness for medical QA. As observed in Experiment 1, retrieval effectiveness declined as answer lengths increased. For instance, BAAI had higher precision on shorter answers (1-99 words) compared to longer responses (100-199 words).

Response Quality in FT: Unlike IR, FT directly generates responses, eliminating the need for retrieval. The fine-tuned models (OpenAI 4o mini and LLaMa 3.2-1B-Instruct) were evaluated using BERTScore, which measures the semantic similarity between generated responses and ground-truth answers. OpenAI 4o mini achieved the highest BERTScore of 70.6%, outperforming all Fine-tuning-based approaches. In contrast, LLaMa 3.2-1B-Instruct achieved a lower BERTScore of 62.3%, showing weaker alignment with expected answers compared to OpenAI 4o mini. FT, however, required significantly higher computation time, with OpenAI 4o mini taking 14,715.89 seconds and LLaMa 3.2-1B-Instruct requiring 72189.1795 seconds, making them much slower than RAG.

Efficiency Comparison: IR vs FT: While FT achieves superior response quality, its computational cost was significantly higher than IR. IR retrieval was much faster, with response times ranging from 83.74s (BAAI, 100-199 words) to 436.35s (JinaAI, full corpus). FT was considerably slower, requiring over 14,000s for OpenAI 4o mini and 72,000s for LLaMa 3.2-1B-Instruct. JinaAI embeddings consistently performed better than BAAI in retrieval precision, suggesting that choosing the right embedding model can significantly impact retrieval effectiveness.

Table 5. IR Model Performance (Exp 2)

Embedding Model	BAII	JINAAI	All-MiniLM-L12	DistilBERT-QA
QA Pairs	3125			
Word Count	1-199 words			
Precision@1	62.81	73.3	2.95	2.25
Precision@3	83.48	86.37	5.42	4.31
Precision@5	87.6	89.82	6.63	5.33
Precision@7	89.41	91.15	7.64	5.71
Precision@10	90.9	92.14	8.94	6.56
Inference Time	113.5865	436.3555	99.2165	80.9548

Table 6. FT Model Performance (Exp 2)

Model	QA Pairs	Word Count	Bert Score	Inference Time
OpenAI 4.0	3154	1-199 words	70.6	14715.8915
LLaMa 3.2-1B-Instruct	3154	1-199 words	62.3	72189.1795

6 Limitation

As this study represents a significant step toward developing a medical Urdu QA system, several limitations must be acknowledged. One of the primary challenges is the availability of limited high-quality Urdu medical corpora. Although Urdu-MedQuAD-25 serves as a valuable resource, its scope remains smaller compared to English medical QA corpora, ultimately affecting the model's generalisation capabilities. From the approaches perspective, IR performance declines with the increase in answer length, which indicates that the retrieval model struggles with long, complex responses. Whereas CAG requires large computational resources, making it impractical for systems with limited processing power. On the other hand, FT achieves the best response accuracy but is computationally expensive, requiring significant training and inference time, which may limit its deployment in real-time medical applications. Furthermore, evaluating responses in a medical QA system is inherently complex, as traditional automatic metrics like BERTScore may not fully capture factual correctness. A more robust evaluation, such as an LLM or a Human evaluation, is required to ensure the accuracy of the medical QA system.

7 Conclusion and Future Work

This study introduced Urdu-MedQuAD-25, a benchmark corpus for developing and evaluating Urdu medical QA systems. To assess the feasibility of different approaches for Urdu medical QA, we explored IR, CAG, and FT approaches. Our experiments were designed in two phases: (i) using a subset of 500 instances to evaluate retrieval and generation performance at a smaller scale and (ii) utilizing the entire corpus to analyze scalability and generalization across a larger corpus. Our findings highlight a clear trade-off between retrieval effectiveness, response accuracy, and computational efficiency. IR demonstrated strong retrieval performance with JinaAI embeddings, outperforming other retrieval models in terms of Precision@1. However, retrieval effectiveness declined as answer lengths increased, emphasizing the challenges in retrieving longer and more complex medical responses. CAG eliminated retrieval latency by pre-loading domain-specific knowledge, showing promising results but requiring substantial computational resources. FT on OpenAI 4o mini achieved the highest response accuracy (BERTScore: 70.6%) but at the cost of significantly higher inference time, making it less practical for real-time applications.

In future, we aim to optimize retrieval for more extended responses, increase corpus size, explore other Urdu embedding models, improve domain-specific Urdu embeddings, speed up the inference, and explore hybrid models that combine IR and FT for more efficient and accurate Urdu medical QA systems. By advancing research in low-resource medical AI, this work contributes toward bridging the gap in healthcare accessibility for Urdu-speaking populations.

8 Funding

The work was done with partial support from the Mexican Government through the grant A1-S-47854 of CONACYT, Mexico, grants 20241816, 20241819, and 20240951 of the Secretaría de Investigación y Posgrado of the Instituto Politécnico Nacional, Mexico. The authors thank the CONACYT for the computing resources brought to them through the Plataforma de Aprendizaje Profundopara Tecnologías del Lenguaje of the Laboratoriode Supercómputo of the INAOE, Mexico and acknowledge the support of Microsoft through the Microsoft Latin America PhD Award.

References

[1] Arif, S., Farid, S., Athar, A., and Raza, A. A. (2024). Uqa: Corpus for urdu question answering. *arXiv preprint arXiv:2405.01458*.

[2] Ben Abacha, A. and Demner-Fushman, D. (2019). A question-entailment approach to question answering. *BMC bioinformatics*, 20:1–23.

[3] Chan, B. J., Chen, C.-T., Cheng, J.-H., and Huang, H.-H. (2024). Don't do rag: When cache-augmented generation is all you need for knowledge tasks. *arXiv preprint arXiv:2412.15605*.

[4] Cho, J. and Lee, G. G. (2025). K-comp: Retrieval-augmented medical domain question answering with knowledge-injected compressor. *arXiv preprint arXiv:2501.13567*.

[5] Fanous, A., Goldberg, J., Agarwal, A. A., Lin, J., Zhou, A., Daneshjou, R., and Koyejo, S. (2025). Syceval: Evaluating llm sycophancy. *arXiv preprint arXiv:2502.08177*.

[6] Harikrishnan, V., Abinaya, N., Santhiya, S., Jayadharshini, P., Aarthi, B., and Nallamangai, K. S. (2025). Comparative analysis of bert-pipelines in squad and medquad question answering. In *Challenges in Information, Communication and Computing Technology*, pages 659–663. CRC Press.

[7] Hider, P. N., Griffin, G., Walker, M., and Coughlan, E. (2009). The information-seeking behavior of clinical staff in a large health care organization. *Journal of the Medical Library Association: JMLA*, 97(1):47.

[8] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. (2022). Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

[9] Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. (2024). Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

[10] Jiang, F. (2024). Identifying and mitigating vulnerabilities in llm-integrated applications. Master's thesis, University of Washington.

[11] Jin, D., Pan, E., Oufattole, N., Weng, W.-H., Fang, H., and Szolovits, P. (2021). What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

[12] Jin, Q., Dhingra, B., Liu, Z., Cohen, W. W., and Lu, X. (2019). Pubmedqa: A dataset for biomedical research question answering. *arXiv preprint arXiv:1909.06146*.

[13] Jin, Q., Yuan, Z., Xiong, G., Yu, Q., Ying, H., Tan, C., Chen, M., Huang, S., Liu, X., and Yu, S. (2022). Biomedical question answering: a survey of approaches and challenges. *ACM Computing Surveys (CSUR)*, 55(2):1–36.

[14] Kang, J., Ryu, H., and Sim, J. (2025). Prism-med: Parameter-efficient robust inter-domain specialty model for medical language tasks. *IEEE Access*.

[15] Kazi, S. and Khoja, S. (2021). Uquad1. 0: development of an urdu question answering training data for machine reading comprehension. *arXiv preprint arXiv:2111.01543*.

[16] Kell, G., Roberts, A., Umansky, S., Qian, L., Ferrari, D., Soboczenski, F., Wallace, B. C., Patel, N., and Marshall, I. J. (2024). Question answering systems for health professionals at the point of care—a systematic review. *Journal of the American Medical Informatics Association*, 31(4):1009–1024.

[17] Krithara, A., Nentidis, A., Bougiatiotis, K., and Paliouras, G. (2023). Bioasq-qa: A manually curated corpus for biomedical question answering. *Scientific Data*, 10(1):170.

[18] Lakatos, R., Pollner, P., Hajdu, A., and Joó, T. (2025). Investigating the performance of retrieval-augmented generation and domain-specific fine-tuning for the development of ai-driven knowledge-based systems. *Machine Learning and Knowledge Extraction*, 7(1):15.

[19] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

[20] Muneer, I., Saeed, A., and Adeel Nawab, R. M. (2025). Cross-lingual english–urdu semantic word similarity using sentence transformers. *The European Journal on Artificial Intelligence*, page 30504554241297614.

[21] Pal, A., Umapathi, L. K., and Sankarasubbu, M. (2022). Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

[22] Pieri, S., Mullappilly, S. S., Khan, F. S., Anwer, R. M., Khan, S., Baldwin, T., and Cholakkal, H. (2024). Bimedix: Bilingual medical mixture of experts llm. *arXiv preprint arXiv:2402.13253*.

[23] Qiu, P., Wu, C., Zhang, X., Lin, W., Wang, H., Zhang, Y., Wang, Y., and Xie, W. (2024). Towards building multilingual language model for medicine. *Nature Communications*, 15(1):8384.

[24] Racha, S., Joshi, P., Raman, A., Jangid, N., Sharma, M., Ramakrishnan, G., and Punjabi, N. (2025). Mhqa: A diverse, knowledge intensive mental health question answering challenge for language models. *arXiv preprint arXiv:2502.15418*.

[25] Rajkhowa, T., Chowdhury, A. R., Nagaonkar, S., and Tripathi, A. M. (2024). Tmpathvqa: 90000+ textless multilingual questions for medical visual question answering. *arXiv preprint arXiv:2407.11383*.

[26] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

[27] Singhal, K., Tu, T., Gottweis, J., Sayres, R., Wulczyn, E., Amin, M., Hou, L., Clark, K., Pfohl, S. R., Cole-Lewis, H., et al. (2025). Toward expert-level medical question answering with large language models. *Nature Medicine*, pages 1–8.

[28] Touvron, H., Lavril, T., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

[29] Vashurin, R., Fadeeva, E., Vazhentsev, A., Rvanova, L., Tsvigun, A., Vasilev, D., Xing, R., Sadallah, A. B., Grishchenkov, K., Petrakov, S., et al. (2024). Benchmarking uncertainty quantification methods for large language models with lm-polygraph. *arXiv preprint arXiv:2406.15627*.

[30] Vazhentsev, A., Rvanova, L., Lazichny, I., Panchenko, A., Panov, M., Baldwin, T., and Shelmanov, A. (2025). Token-level density-based uncertainty quantification methods for eliciting truthfulness of large language models. *arXiv preprint arXiv:2502.14427*.

[31] Vazrala, S. and Mohammed, T. K. (2025). Rbtm: A hybrid gradient regression-based transformer model for biomedical question answering. *Biomedical Signal Processing and Control*, 102:107325.

[32] Wang, L. L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Burdick, D., Eide, D., Funk, K., Katsis, Y., Kinney, R., et al. (2020). Cord-19: The covid-19 open research dataset. *ArXiv*, pages arXiv–2004.

[33] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Authors

Ahmad Mahmood received a Master of Computer Science from Comsats University Islamabad, Lahore Campus, and he did a Bachelor's degree in information technology from Bahria University, Lahore, Pakistan. Currently, he is pursuing his PhD in Computer Science from the Instituto Politécnico Nacional (IPN), Mexico City, Mexico. His research interests include Machine Learning, Natural Language Processing, Generative AI, LLM Quantization, and Agentic AI.

Zainab Ahmad received a Master of Computer Science from Comsats University Islamabad, Lahore Campus, and she did a Bachelor's degree in computer science from Lahore College for Women University, Lahore, Pakistan. Currently, she is pursuing her PhD in Computer Science from the Instituto Politécnico Nacional (IPN), Centro de Investigación en Computación (CIC), Mexico City, Mexico. Her research interests include Machine Learning, Natural Language Processing, and Generative AI.

Dr. Iqra Ameer is an Assistant Professor in the Division of Science and Engineering at The Pennsylvania State University at Abington, USA. Prior to joining Penn State, she was a Postdoctoral Researcher at Yale University and the University of Texas at Houston, where she focused on suicide ideation detection and named entity recognition using clinical text. Dr. Ameer earned her Ph.D. in Computer Science from Instituto Politécnico Nacional in 2022. During her doctoral studies, she worked on multi-label emotion classification for both code-mixed and monolingual text. Her research interests include Natural Language Processing, Machine Learning, Data Science, and GPT.

Dr. Grigory Sidorov received PhD in Science from Lomonosov Moscow State University, Russia, 1996. He is a professor-researcher at the Natural Language and Text Processing Laboratory of the Research Center in Computing at the National Polytechnic Institute (CIC-IPN), Mexico. He is a National Researcher of Mexico (member of the SNI) at Level 3, a member of the Mexican Academy of Sciences, and the editor-in-chief of the research journal Computación y Sistemas (indexed in ISI, Scopus, and others). His scientific interests include computational linguistics, automated text processing, and the application of machine learning methods to natural language processing tasks.