Retrieval-Augmented Generation for Biomedical Question Answering

Team 22
Abdelrahman Mohamed Ahmed Abouelkheir 52-5388

May 2025

1 Introduction

This report documents the development and evaluation of a Retrieval-Augmented Generation (RAG) pipeline for biomedical question answering. The system integrates a dense retriever and a language model to synthesize answers based on retrieved biomedical contexts. The experiment was designed to evaluate the effectiveness of zero-shot and chain-of-thought prompting on a subset of the PubMedQA dataset.

2 Dataset: PubMedQA

The dataset used in this project is `PubMedQA`, specifically the `pqa_artificial` subset. PubMedQA is a benchmark for biomedical question answering, comprising real-world questions from PubMed abstracts. The artificial subset includes questions synthesized from existing biomedical knowledge and includes the following fields:

- **question**: A biomedical research question.
- **context**: A set of related biomedical text passages.
- **long_answer**: An in-depth answer to the question.
- **final_decision**: A short yes/no/maybe classification.

Only the first 30 samples were used to reduce computational load while still enabling meaningful evaluation.

3 Embedding Model: all-MiniLM-L6-v2

The retriever component of the RAG system relies on dense vector representations of text. For this purpose, the **sentence-transformers/all-MiniLM-L6-v2** model was selected.

3.1 Why all-MiniLM-L6-v2?

- **Efficiency**: MiniLM models are known for their compact size and fast inference time, making them ideal for rapid embedding generation, even on CPU-constrained environments.
- **Performance**: Despite being a smaller model (6 layers), it achieves strong performance on semantic similarity tasks and is widely used in production-grade applications.
- **Domain Adaptability**: Though not trained specifically on biomedical data, it offers good generalization and can capture semantic similarity effectively in domain-specific contexts.

The generated embeddings were stored using a FAISS vector store, which supports efficient similarity search over high-dimensional vectors.

4 Language Model: SmolLM-1.7B-Instruct

The language model used for answer generation was **SmolLM-1.7B-Instruct** hosted on HuggingFace.

4.1 Why SmolLM-1.7B-Instruct?

- **Instruction Tuning**: This model is fine-tuned to follow user instructions effectively, which makes it suitable for structured prompting, such as chain-of-thought (CoT) reasoning and zero-shot QA.
- **Lightweight Yet Capable**: With 1.7B parameters, SmolLM offers a balance between computational efficiency and language generation performance, especially for projects not requiring GPT-3/4 scale models.
- **Open Source and Reproducible**: The availability and permissiveness of the model under HuggingFace's framework make it ideal for academic and research applications.

The model was deployed using GPU acceleration, allowing for faster inference, especially when generating long, detailed responses.

5 System Architecture

The pipeline consists of three main components:

1. **Retriever**: Accepts a question and retrieves top-3 relevant passages using dense vector similarity (FAISS + MiniLM).
2. **Prompt Constructor**: Formats the question and context into a prompt using either a standard or chain-of-thought template.

3. **Generator**: Uses SmolLM to produce a natural language answer.

Two prompting strategies were employed:

- **Zero-shot Prompting**: Provides question and context directly to the model.
- **Chain-of-Thought Prompting**: Adds reasoning guidelines to encourage step-by-step explanation.

6 Evaluation Metrics and Methods

To evaluate the generated answers, a combination of lexical and semantic metrics were used:

6.1 ROUGE Scores

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) measures the overlap of n-grams between the predicted and reference answers. ROUGE-1, ROUGE-2, and ROUGE-L were calculated.

6.2 Observations

Chain-of-thought prompting generally produced more elaborate and structured responses, which captures sequence fluency. Zero-shot prompting was faster and simpler but occasionally missed critical reasoning steps.

7 Conclusion

This report presented a question-answering system tailored for biomedical queries using a combination of dense retrieval and large language generation. The system utilized:

- The **PubMedQA dataset**, which provided domain-specific, medically relevant QA samples.
- A compact yet powerful **MiniLM model** for semantic search and context retrieval.
- An instruction-tuned **SmolLM-1.7B LLM** to synthesize responses from the retrieved data.

Chain-of-thought prompting improved the reasoning capabilities of the LLM, especially in cases requiring complex logical deductions. The system demonstrates how smaller open-source components can be orchestrated into effective RAG pipelines for domain-specific tasks.