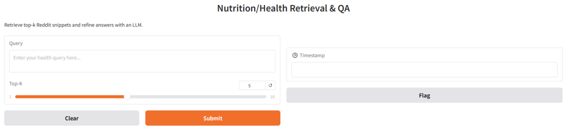
Nutrition and Health Chatbot Based on Retrieval-Augmented Generation

Hang Xu(hx38), Xinshu Zhao(xz120)

April, 2025

Abstract

In this project, we develop a nutrition and health information retrieval chatbot based on Retrieval-Augmented Generation (RAG). Leveraging Reddit health forums as our knowledge source, we employ dense retrieval techniques using semantic embeddings provided by SentenceTransformer models and integrate these with the GPT-4o generative model to produce coherent and relevant health-related answers. Our system significantly improves the efficiency of online health consultations by accurately retrieving contextually relevant information and generating reliable responses.



Nutrition/Health Retrieval & QA

Retrieve top-k Reddit snippets and refine answers with an LLM.

Top-K: 5

Figure 1: Example of the Health RAG.

1. Introduction

With the rapid growth of online health content, more and more people turn to digital platforms for answers to their health-related questions. However, the internet is filled with unstructured, inconsistent, and sometimes misleading information, making it hard for users to find accurate and helpful guidance. This creates a strong need for systems that can retrieve and generate reliable, trustworthy health information.

Retrieval-Augmented Generation (RAG) is a modern technique that combines information retrieval with powerful generative models to address this challenge. First proposed by Lewis et al. (2020), RAG systems retrieve relevant documents from a knowledge base and then generate answers based on the retrieved context. By using dense retrieval methods and Large Language Models (LLMs), these systems can produce more context-aware and accurate responses compared to traditional approaches.

In this project, we build a health-focused information retrieval and question-answering system based on Reddit data. Reddit provides a rich source of real-world discussions on a wide range of health topics, making it a valuable dataset for modeling user health queries. We clean and preprocess over 23,000 Reddit posts, then apply a dense retrieval method using Sentence-BERT to semantically match user questions with relevant documents.

To improve answer quality, we integrate OpenAI's GPT-4o model, which takes the top retrieved passages and generates a final answer in natural language. We also develop an interactive front-end using Gradio that allows users to enter their questions and receive real-time results.

Our main goal is to provide a simple and effective system that delivers trustworthy, readable, and relevant health information. By combining dense retrieval and generation, we aim to support better digital health assistance and reduce misinformation from informal online sources.

2. Prior Work

2.1. Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation, or RAG, is a method that helps language models give more accurate and relevant answers. Instead of just relying on what the model was trained on, RAG lets it search through a large set of documents—like Wikipedia—to find useful information while answering a question. It was introduced by Lewis et al. in 2020 and basically combines a search system with a text generator. This way, the model can include facts from outside sources, making its answers more trustworthy and specific.

2.2. Dense Passage Retrieval (DPR)

Dense Passage Retrieval (DPR) is a method introduced by Karpukhin et al. in 2020 to improve open-domain question answering. Instead of using traditional keyword-based search methods like

BM25, DPR turns both questions and documents into dense vectors using neural networks. These vectors are placed in the same space so the system can easily find the most relevant matches. Thanks to this design, DPR has shown much better performance on many question-answering tasks.

2.3. Sentence-BERT (SBERT)

Sentence-BERT (SBERT) is a version of BERT that's designed to make it easier to compare the meanings of sentences. It was introduced by Reimers and Gurevych in 2019 to solve the problem that regular BERT wasn't great at handling sentence similarity. SBERT uses special network setups called siamese or triplet networks to turn sentences into meaningful vector representations. This makes it really useful for things like semantic search and clustering, which are important for building good retrieval systems.

3. Model, Algorithm, and Method

Our system is designed following the Retrieval-Augmented Generation (RAG) paradigm, which integrates dense vector-based retrieval with generative large language models to produce accurate, context-aware answers. This section outlines the key components of our implementation, including corpus construction, semantic retrieval, response generation, and user interaction.

3.1. Corpus Preparation and Preprocessing

We begin by constructing a health-focused corpus from Reddit data, sourced from multiple CSV files containing user-generated posts. Each entry includes a title, main content, and associated comments. These were combined and processed into two parallel corpora:

- **Original Corpus:** Maintains the raw text for human-readable display.
- **Cleaned Corpus:** Optimized for retrieval through a multi-step text preprocessing pipeline:
  - Lowercasing and removal of punctuation and web links
  - Elimination of English stopwords and informal internet expressions (e.g., "lol", "omg")
  - Optional stemming and lemmatization using NLTK tools

This dual-corpus approach supports both effective semantic matching and user-friendly output.

3.2. Dense Semantic Retrieval

To enable semantic matching between user queries and Reddit entries, we employ dense retrieval based on sentence embeddings. Specifically, we use the `intfloat/multilingual-e5-small` model from the `SentenceTransformer` library, which encodes both queries and documents into fixed-size vectors in a shared semantic space.

We implement a custom retrieval class, `DenseRetrievalExactSearch`, which extends a generic `BaseSearch` interface. This retriever includes:

- **Encoding:** Queries and documents are converted to embeddings using the transformer model.
- **Similarity Scoring:** Pairwise similarity is computed using cosine similarity or dot product.
- **Chunked Corpus Processing:** To manage GPU memory, the corpus is processed in chunks (default size: 1000 documents).
- **Top-K Selection:** A min-heap is used to maintain the highest-scoring documents per query efficiently.

The retriever ranks results by semantic similarity, returning the top-k most relevant documents for each query. This architecture ensures scalability and modularity across different corpora and model backends.

3.3. Retrieval-Augmented Generation

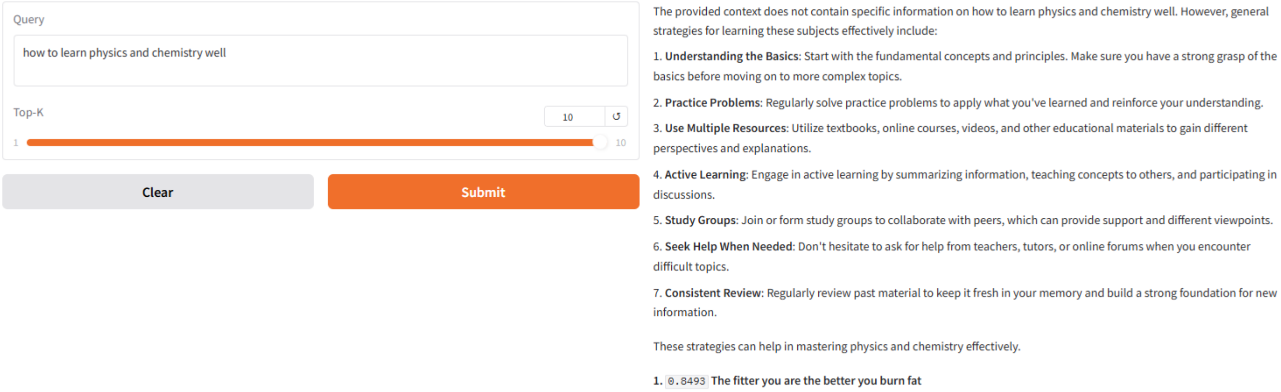
Once the relevant documents are retrieved, they are used to construct a prompt for OpenAI's GPT-4o model. The prompt includes:

- A **system message** defining the assistant's role as a reliable source of health information
- A **user message** combining the original query and a context block composed of the top-k retrieved Reddit posts (title and truncated text)

The prompt is passed to the GPT-4o model via OpenAI's chat-completion API, using parameters that prioritize factuality and coherence (`temperature=0.2`, `max_tokens=12800`). This integration allows the system to generate answers that are not only fluent but grounded in real-world health discussions.

Table 1: Retriever evaluation on four example health queries.

Query	P@10	Reciprocal Rank
What's a healthy diet?	1.00	1.00
How to lose weight?	0.80	1.00
How to build muscle naturally?	0.80	1.00
Tips for better sleep?	0.50	0.50
What foods help reduce inflammation?	0.70	1.00
What are the effects of melatonin supplements?	0.80	0.50
How often should I do cardio vs strength training?	0.60	0.33
What's the best home workout routine for beginners?	0.90	1.00
What causes sore muscles after working out?	1.00	1.00
Is intermittent fasting effective for weight loss?	0.50	0.25



Query: how to learn physics and chemistry well

Top-K: 10

The provided context does not contain specific information on how to learn physics and chemistry well. However, general strategies for learning these subjects effectively include:

1. **Understanding the Basics**: Start with the fundamental concepts and principles. Make sure you have a strong grasp of the basics before moving on to more complex topics.

2. **Practice Problems**: Regularly solve practice problems to apply what you've learned and reinforce your understanding.

3. **Use Multiple Resources**: Utilize textbooks, online courses, videos, and other educational materials to gain different perspectives and explanations.

4. **Active Learning**: Engage in active learning by summarizing information, teaching concepts to others, and participating in discussions.

5. **Study Groups**: Join or form study groups to collaborate with peers, which can provide support and different viewpoints.

6. **Seek Help When Needed**: Don't hesitate to ask for help from teachers, tutors, or online forums when you encounter difficult topics.

7. **Consistent Review**: Regularly review past material to keep it fresh in your memory and build a strong foundation for new information.

These strategies can help in mastering physics and chemistry effectively.

1. `0.8493` **The fitter you are the better you burn fat**

Figure 2: Example using irrelevant query

3.4. Interactive Interface with Gradio

We deploy our pipeline using Gradio. The interface allows users to:

- Enter a health-related question
- Select the number of retrieved documents (top_k)
- View retrieved Reddit snippets and the generated answer
- Monitor response time dynamically

The interface is powered by a generator-based `answer()` function that streams results incrementally. This enhances user experience by showing intermediate updates, such as processing status and partial retrievals.

4. Results & Evaluation

To assess the effectiveness of the dense retriever, we conducted a small-scale manual evaluation using 10 representative health-related queries, because in general our database is relatively subject, and it is very hard to find a labeled database where we could use to evaluate. For each query, we retrieved the top-10 documents and manually labeled each as either relevant or irrelevant.

4.1. Metrics

We report Precision@5 (P@10) and Reciprocal Rank (RR) as our evaluation metrics.

- **P@10**: Proportion of relevant documents in the top-10 results.
- **RR**: Reciprocal rank of the relevant documents.

4.2. Results

In addition to quantitative metrics like Precision@10 and Reciprocal Rank, we conducted a manual inspection of the retrieved and generated answers to better understand the strengths and limitations of the system.

First, we observed that for certain queries with ambiguous focus or multiple interpretations, such as *"Is intermittent fasting effective for weight loss?"*, the retriever tends to return passages broadly related to **healthy eating habits**, but not specifically targeting intermittent fasting. While this might limit direct relevance at the retrieval level, the LLM was often able to **refine the final answer accurately** by grounding on the available content and generating a more focused response.

Second, the system demonstrated **robustness in handling out-of-domain or weakly related queries**. For instance in the figure 2, when the user is asking a question like *"How to learn physics and chemistry well?"*—which is outside the nutrition or health domain—the LLM generated a well-structured response:

> *"The provided context does not contain specific information on how to learn physics and chemistry well. However, general strategies for learning these subjects effectively include..."*

This behavior indicates that the language model is able to **decline answering from hallucinated content** and instead offer a generic, informative suggestion.

These observations highlight the complementary strengths of the retriever and the generator: while the retriever provides **related facts**, the LLM improves **fluency and stucture**, especially when the retrieved snippets are only loosely relevant. The retriever demonstrates reasonable performance, retrieving relevant content for most queries within the top-10 positions. Future improvements may include domain-adapted reranking or hybrid dense–sparse retrieval strategies.

5. Discussion & Future Work

In this project, we successfully developed a retrieval-augmented system that combines semantic search over Reddit health data with a large language model (LLM) to generate relevant answers to user queries. The dense retriever based on SentenceTransformer demonstrated effective performance, especially in cases where lexical overlap was low.

Despite the system's strengths, we also observed several limitations. The reliance on Reddit as a data source means that retrieved passages may lack medical authority. Future versions of the system could incorporate more credible sources such as PubMed or official health organizations. Additionally, while response time (30s) is acceptable for a prototype, optimizing speed remains a priority. Techniques like embedding caching, reranking, or model compression can help reduce latency.

LLM-related issues, such as token limitations or API failures, occasionally led to incomplete responses. Compressing or summarizing retrieved context before feeding it into the model could improve robustness. Finally, our Gradio-based interface proved user-friendly, but further refinements such as dynamic status feedback and error handling would enhance usability.

Future work may also explore:

- Integrating hybrid retrieval (dense + sparse) and reranking for higher retrieval precision.
- Using open-source or fine-tuned domain-specific models to reduce API dependency.
- Incorporating user feedback to refine relevance over time.

Overall, this work demonstrates the potential of retrieval-augmented generation in the health domain and lays the groundwork for building more intelligent, trustworthy, and responsive assistant systems.

References

[1] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., et al. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *NeurIPS 2020*.

[2] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W. (2020). Dense passage retrieval for open-domain question answering. *EMNLP 2020*.

[3] Reimers, N., Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *EMNLP 2019*.