NutriNudge: Dietary Guidelines QnA - Powered by RAG and Ollama

Your Name

July 12, 2024

Abstract

This report presents a comprehensive question-answering system designed to provide accurate and contextually relevant information about dietary guidelines. The system leverages advanced natural language processing techniques, including document embedding, vector search, and language model-based generation. By combining these technologies, we create an interactive and efficient tool for accessing and understanding complex dietary information.

1 Introduction

The Dietary Guidelines Q&A System is developed to address the growing need for accessible and accurate nutritional information. By utilizing the latest advancements in natural language processing and machine learning, this system aims to bridge the gap between complex dietary guidelines and the general public's understanding.

2 System Architecture

The system is built on a multi-component architecture that integrates various technologies to provide a seamless question-answering experience.

2.1 Document Processing

The core document, "Dietary Guidelines for Americans 2020-2025," is processed using the following steps:

1. PDF Loading: The document is loaded using PyPDFLoader from the LangChain library.

2. Text Splitting: The loaded document is split into manageable sections using RecursiveCharacterTextSplitter, with a chunk size of 2000 characters and an overlap of 200 characters.

2.2 Vector Database Creation

To enable efficient semantic search, we create a vector database:

1. Embedding Generation: Each text section is converted into a dense vector representation using the SentenceTransformer model 'all-MiniLM-L6-v2'.
2. FAISS Index: We utilize FAISS (Facebook AI Similarity Search) to create an efficient index for these high-dimensional vectors.
3. Data Storage: The original text sections and their metadata are stored in a separate pickle file for retrieval during the question-answering process.

2.3 Question-Answering Pipeline

The Q&A process involves the following steps:

1. Query Embedding: The user's question is embedded using the same SentenceTransformer model.
2. Vector Search: The query embedding is used to search the FAISS index for the most similar document sections.
3. Context Retrieval: The top-k (default k=3) most similar sections are retrieved and combined to form the context.
4. Answer Generation: The context and question are passed to a language model (Llama3 via Ollama) to generate the final answer.

3 System Evaluation

The system's performance can be evaluated based on several criteria:

- Accuracy of answers
- Relevance of retrieved context
- Response time
- User satisfaction

A comprehensive evaluation would involve user studies and comparison with baseline systems.

4 Future Improvements

Potential areas for enhancement include:

- Fine-tuning the language model on dietary guideline data

- Implementing a more sophisticated ranking algorithm for context retrieval
- Adding support for multi-modal inputs (e.g., images of food)
- Incorporating user feedback to improve answer quality over time

5 Conclusion

The Dietary Guidelines Q&A System demonstrates the potential of combining vector search techniques with advanced language models to create an intuitive and informative tool for accessing complex information. By providing quick and accurate answers to dietary questions, this system has the potential to improve public understanding of nutrition guidelines and contribute to better health outcomes.

References

[1] U.S. Department of Agriculture and U.S. Department of Health and Human Services. (2020). *Dietary Guidelines for Americans, 2020-2025*. 9th Edition. Available at `https://www.dietaryguidelines.gov/`

[2] LangChain. (2023). *LangChain Documentation*. Retrieved from `https://python.langchain.com/docs/get_started/introduction`

[3] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

[4] Johnson, J., Douze, M., & Jégou, H. (2017). Billion-scale similarity search with GPUs. *arXiv preprint arXiv:1702.08734*.

[5] Streamlit. (2023). *Streamlit Documentation*. Retrieved from `https://docs.streamlit.io/`

[6] Ollama. (2023). *Ollama: Get up and running with large language models, locally*. Retrieved from `https://ollama.com/`

[7] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). Llama: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

[8] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems*, 33, 9459-9474.

[9] Bernstein, Y., & Zobel, J. (2019). A scalable system for identifying co-derivative documents. *In String Processing and Information Retrieval: 26th International Symposium, SPIRE 2019* (pp. 55-69). Springer International Publishing.