AI-Powered Question-Answering Chatbot Using Retrieval-Augmented Generation (RAG)

To enhance chatbot accuracy by integrating external knowledge sources, addressing the limitations of LLMs like generating inaccurate responses. This improvement aims to provide more reliable customer support for Swisscom.

1. **How did we decide to do this?**

The decision to implement this project stemmed from the recognition that large language models (LLMs) often generate inaccurate or outdated responses when relying solely on the information embedded in their parameters. To overcome these limitations, we chose Retrieval-Augmented Generation (RAG) as the primary solution. RAG allows LLMs to retrieve up-to-date and accurate data from external knowledge sources, enhancing both the relevance and factual accuracy of responses. The project aim is to support the convenient question answering chatbot using RAG trained on datasets provided by Swisscom websites which will benefit the customers.

2. **Semantic chunking, why we use it?**

Normal Chunking

This typically involves splitting documents into fixed-size segments without considering the meaning or semantic value of each part.

Semantic chunking

Why normal is not best approach? The goal of semantic chunking is to divide documents into smaller, meaningful segments that retain the context necessary for accurate information retrieval.

Why we use it?

Semantic chunking ensures that the segmented data retains contextually relevant information, making it easier for the chatbot to accurately respond to user queries. Smaller chunks enable the retrieval of more precise information within the context window of the LLM, leading to better quality answers. This also improves the quality of embeddings for search queries, making retrieval more effective.

3. **Fine tuning of the embedding model, we find this model BAAI/ bge-m3. Why it is good?**

The embedding model that we chose is BAAI/BGE-M3. It is a multi-language efficient embedding model, making it well-suited for Swisscom's multilingual data. This enhances the chatbot's ability to answer the questions accurately in European languages.

Some Advantages:

- **Improved Accuracy**: Fine-tuning the model with specific data improves the relevance of search results.
- **Better Embeddings**: Produces higher-quality semantic embeddings that are crucial for effective information retrieval.

For fine tuning we used the following document.

https://docs.llamaindex.ai/en/stable/examples/finetuning/embeddings/finetune_embedding/

4. **Contextual Information To Chunks**

Contextual information involves attaching metadata or supplementary context to each chunk during the data segmentation phase. This has following advantages:

Improved Retrieval Efficiency: Smaller units allow for faster and more focused similarity searches in the vector database.

Enhanced LLM Context: Chunks provide the LLM with more specific information for generating relevant and informative responses.

Flexibility in Retrieval: Different chunks can be retrieved depending on the specific query, leading to more nuanced responses.

Reference: https://www.anthropic.com/news/contextual-retrieval

5. **FIASS appending small embeddings was not possible due to GPU memory constraints vs Chroma DB (append embeddings in it)**

FAISS is a powerful library for similarity search but it has limitations when it comes to handling dynamic updates or adding embeddings to the dataset. In our project, the inability to append embeddings continuously due to GPU limitations made FAISS less practical for scenarios that required frequent updates.

Unlike FAISS, ChromaDB is designed to handle dynamic updates more efficiently. It allows for seamless appending of embeddings without the need to re-index the entire database. Therefore, this makes it ideal for our project where new data needs to be frequently integrated into the existing database.

ChromaDB's ability to handle incremental changes without overburdening GPU memory makes it the preferable choice.

6. **Used similarity search together with search technique BM25 and merge the results with Reciprocal Rerank Fusion (RRF)**

BM25 (Best Matching 25): It is a ranking function that extends TF-IDF by considering term frequency saturation and document length, and used to handle keyword-based searches, capturing the relevance of each chunk by matching specific terms in user queries.

Reciprocal Rerank Fusion (RRF): It involves combining the results of different search algorithms (e.g. similarity search using Chroma DB and BM25), and uses a scoring mechanism to merge the search outcomes from various techniques, ensuring that results that rank well across multiple algorithms are prioritized.

Why Combine Them?

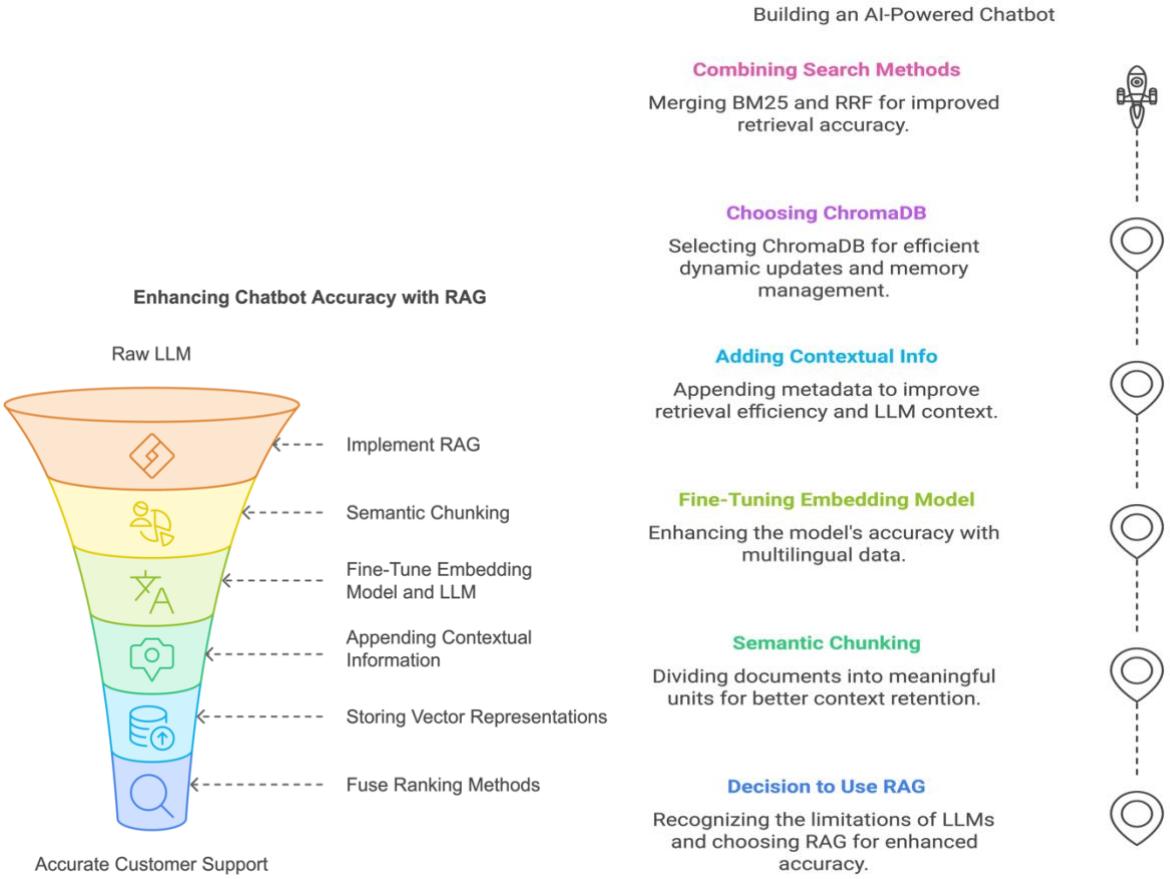
Using both similarity search with BM25, and merging them with RRF, allows the chatbot to take advantage of both semantic and keyword-based relevance. This hybrid approach increases the accuracy of retrieval by providing a comprehensive ranking system that balances the strengths of multiple search methodologies. Additionally, it minimizes the risks of any single method's biases or weaknesses impacting the search quality.

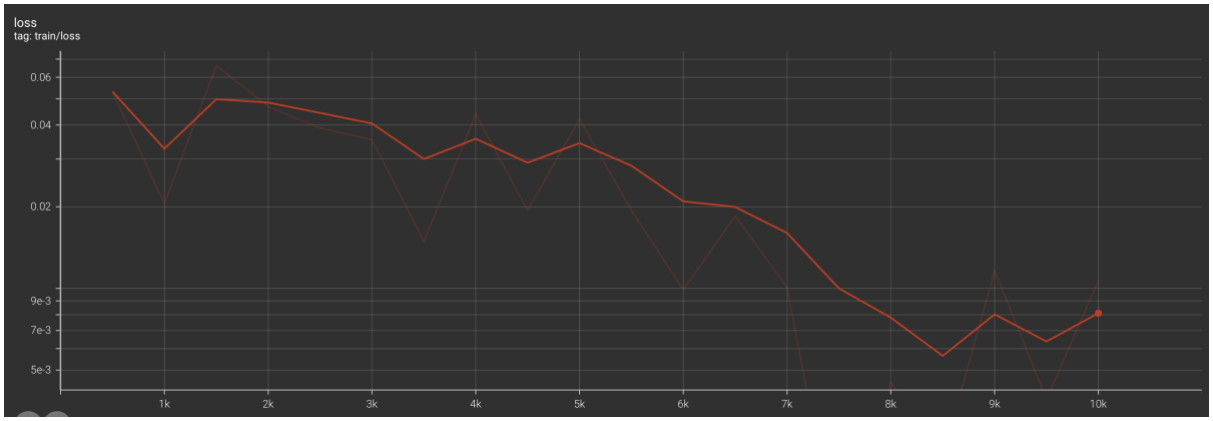
By integrating these advanced techniques and methodologies, the chatbot aims to deliver highly accurate, contextually relevant, and multi-language support for Swisscom's customer service needs.

Additionally, it minimizes the risks of any single method's biases or weaknesses impacting the search quality.

By integrating these advanced techniques and methodologies, the chatbot aims to deliver highly accurate, contextually relevant, and multi-language support for Swisscom's customer service needs.

Conclusion on Images





Fine tuning the BGE-M3
Loss curve is steadilly decreasing over epochs showing that it is getting better