Scaling explanation

The FAQ Retrieval Assistant that I built works well with small datasets on a local setup. However, if we wanted to use it in a real product like Verba AI, there are several things to consider. In production, the system would need to handle very large FAQ databases, serve many users at the same time across different platforms and still respond quickly and accurately. The following sections outline how the system could be adapted for scalability.

Precomputing embeddings

In my solution, the app calculates embeddings for every question in the database each time it starts. For a large dataset , this process will take too long and will slow down the app. In a real setup the best approach is to compute the embeddings once offline and store them in a vector database. Then, the app can load them instantly, which improves performance and user experience.

Searching large datasets

Even with precomputed embeddings , checking every question one by one is too slow for large datasets. In a real setup, the system should use **smart indexing**. This allows the assistant to quickly narrow down the search to only the most relevant group of questions, instead of scanning the whole database.

Scalable multilingual support

In my app, after finding the best matching answer, the system uses Google Translator (external API) to translate it into the user's selected language. This works fine for a small dataset and a few users, it can become slow or unreliable as the FAQ database grows or multiple users submit questions simultaneously. A better approach is to pre-translate the most common answers and use a local translation model for less frequent ones, so the responses are faster and don't rely on an external service.

Conversational memory and user sessions

In my app, the assistant forgets everything after each question. In a real customer support scenario, users often ask follow-up questions. Implementing a per user session conversation buffer or using summarized context vectors allows the system to retain previous interactions.

Language detection

Automatically detect user's language would remove the need for manual selection.

Dynamic FAQ updates

Currently, the FAQs are stored in a static CSV file. A production system should allow adding or updating questions without the need of restarting the app. New FAQs would be embedded and added to the search index in real time, ensuring the knowledge base is always up-to-date.

Data storage and content support

FAQs and other knowledge base content should be stored in a scalable database or vector store instead of a static CSV file. This allows efficient retrieval, better data management, and support for richer content types such as images, PDFs, and structured documents.