Optimising RAG

Two innovative techniques for optimising the RAG model:

a. Embedding-Based Search Enhancement:

- In my project, I have utilized the Sentence Transformer model (a11-MiniLM-L6-v2) to generate embeddings for customer issues. To optimize this RAG model, fine-tune this embedding model on domain-specific data. For example, we could train it on a custom dataset of technical support queries to ensure the embeddings capture more nuanced relationships in your specific domain, improving retrieval accuracy in Pinecone.
- Additionally, using a hybrid retrieval strategy (combining dense retrieval using embeddings with sparse retrieval methods like keyword-based search) can significantly improve the retrieval process. This hybrid approach can help ensure that relevant documents are retrieved based on both semantic meaning and exact keyword matches, enhancing the response quality.

b. Contextual Re-ranking of Retrieved Documents:

- In the query index function, you retrieve documents from Pinecone based on the generated embeddings. A next step for optimization is applying contextual re-ranking to improve the relevance of the documents selected. This can be done by using a smaller, task-specific model to re-rank the documents based on how well they align with the current query or user context (e.g., previously asked questions or specific types of issues).
- This could be implemented by introducing a re-ranking model after retrieving the documents from Pinecone, where the top-k results are further assessed using semantic matching (e.g.. using cosine similarity or an additional fine-tuned transformer model) to improve the final selection before feeding them into Gemini Al for content generation.