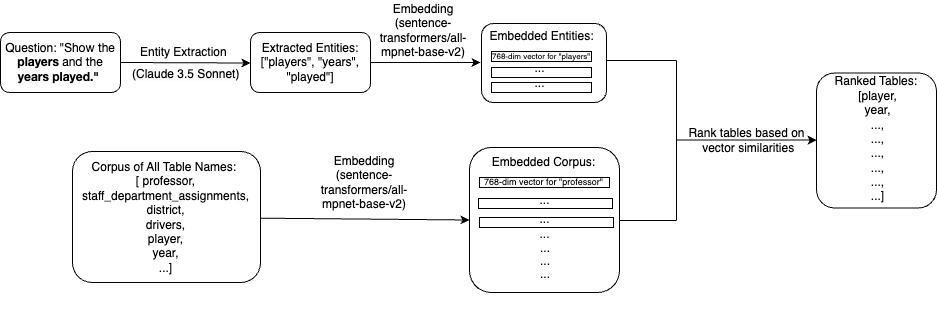
When generating SQL queries using GenAI, a typical approach is to include schemas for relevant tables as a part of the prompt. Due to the constraint in token number (e.g., 200k for Claude 3.5 Sonnet) and efficiency, it’s preferable to quickly eliminate a large number of irrelevant tables through retrieval.

Tech stack: Bedrock, Sagemaker, torch, pandas, json, numpy, sklearn, matplotlib.

Github: <https://github.com/fanglidayan/retrieval_for_text_to_sql_RAG>

The focus is solely on the retrieval component of RAG. The case study is based on the [spider](https://yale-lily.github.io/spider) dataset for text-to-SQL. Below is the retrieval system architecture.



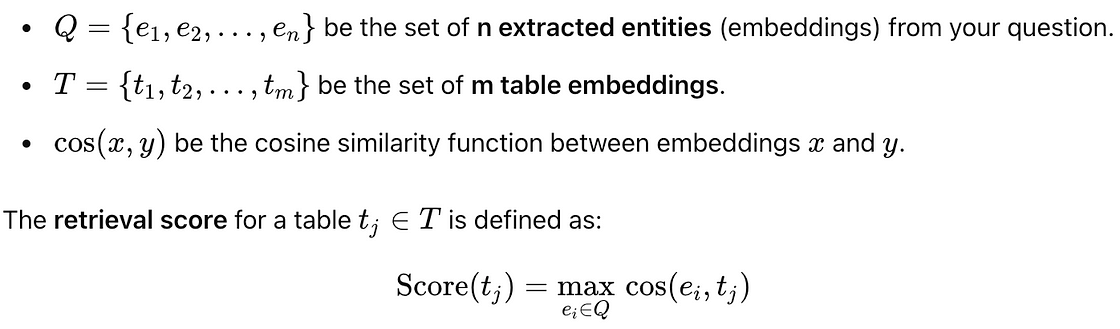
The dataset train\_spider.json consists of 7k question-SQL pairs. For each question, the goal is to retrieve all tables contained in its corresponding SQL query from the corpus of all tables. The retrieval is based on vector cosine similarity between the embeddings of queries and table names.

I ran three experiments: (1) Embedding the original questions without entity extraction, (2) embedding the questions’ extracted entities variation 1, and (3) embedding the questions’ extracted entities variation 2.

The entity extraction performed by calling the Claude 3.5 Sonnet API availble on Amazon Bedrock. I did prompt engineering to ask the GenAI to extract entities which could be used to identify (a) table names and (b) column names. In the paragraph above, experiment (2) only used extracted entities for table names; experiment (3) used all extracted entities.

The embedding for (i) original questions, (ii) extracted entities from questions, and (iii) table names were done by [sentence-transformers/all-mpnet-base-v2](https://huggingface.co/sentence-transformers/all-mpnet-base-v2) on a Sagemaker ml.g5.xlarge instance which includes an A10G GPU.

The retrieval score was determined by the equation below.



For the baseline experiment, one could use embeddings for questions instead of entities. For each question-table pair, and for each experiment, we have a well defined retrieval score and truth label (1 if the table is in the SQL query and 0 otherwise). We could calculate key evaluation metrics. The result is summarized below.



The baseline approach used the original questions for embedding; in entity extraction v1, only entities that could be used to identify table names were embedded; in entity extraction v2, additional entities that could be used to idenfity column names were embedded.

It turns out that only using extracted entities for tables can leave out some important information, however, if all extracted entities are included, the Claude 3.5 Sonnet entity extraction approach brings significant improvements in the retrieval performance.

For future development, one could generate a set of thresholds for retrieval scores. Another thing to try is to fine-tune a smaller GenAI model (e.g., Llama-3.2-3B) based on the dataset labelled by Claude 3.5 Sonnet for entity extraction.