Database Outline & Schema

Ethan Nechanicky

May 2023

Overview

In the design of this database, the following use case requirements have been defined. The database will be used to store the run-time analytics of the deep learning model. This will include entries for each execution of the model. Additionally, the database will be used to store the output and input of each model execution. The output will be a very large dataset representing the vector returned by the model, and the input will be a variably large sized representation of the initial list of input items. It is expected that the tables within the database will store hundreds of thousands of rows. Due to this expectation, the data within these rows will be optimized to contain mainly reference data, minimizing the space required by each row. This database will store the model information from each model execution. This will include, at minimum, a single row for each model execution. Lastly, information relevant to the Pinecone vector database usage within each program execution will be logged within the database. This information will include at least the Pinecone index name and namespace used for each upsertion.

Database Outline

- exec_info:∼ Stores the data related to program execution.
  - exec_id:∼ Unique auto-incrementing execution identifier.
  - exec_start:∼ Execution start time.
  - exec_stop:∼ Execution stop time.
  - input_size:∼ Quantity of people used as input.
  - output_size:∼ Quantity of people returned by similarity search.
  - name:∼ Name of the run set by the user.
- exec_input:∼ Stores the input data related to program execution.
  - input_id:∼ Unique auto-incrementing input identifier.
  - exec_id:∼ Foreign Key connecting input with its corresponding execution.

  - person_id:∼ Identifier for person in input item, id from external database used.

  Foreign Key from exec_info 1:M relationship

- exec_output:∼ Stores the output data related to program execution.
  - output_id:∼ Unique auto-incrementing output identifier.
  - exec_id:∼ Foreign Key connecting output with its corresponding execution.
  - person_id:∼ Identifier for person in output item, id from external database used.
  - k_value:∼ Similarity value produced by model with respect to input.

  Foreign Key from exec_info 1:M relationship

- exec_model:∼ Stores the model information related to program execution.
  - model_id:∼ Unique auto-incrementing model identifier.
  - exec_id:∼ Foreign Key connecting output with its corresponding execution.
  - tokenizer:∼ Pretrained tokenizer type used by transformer.
  - model:∼ Pretrained model type used for generating embeddings.
  - mod_start:∼ Model run-time start time.
  - mod_stop:∼ Model run-time stop time.
  - device:∼ Compute method used by model; generally GPU or CPU.
  - dim:∼ Dimension of the vectors in embedding.

  Foreign Key from exec_info 1:1 relationship

- exec_pinecone:∼ Statistics related to the pincone database connection used within the system.
  - pinecone_id:∼ Unique auto-incrementing pinecone connection identifier.
  - exec_id:∼ Foreign Key connecting output with its corresponding execution.
  - namespace:∼ Namepace used in pinecone upsert call.
  - index:∼ Index used to store embedding in pinecone.
  - upsert_start:∼Embedding upsert to pinecone index start time.
  - upsert_stop:∼ Embedding upsert to pinecone index stop time.
  - query_start:∼ Query call to pinecone index start time.
  - query_stop:∼ Query call to pinecone index stop time.

- kmin:∼ Minimum similarity value in output people.
- kmax:∼ Maximum similarity value in output people.
- kavg:∼ Average similarity value across all output people .

Foreign Key from exec_info 1:1 relationship

ER Diagram

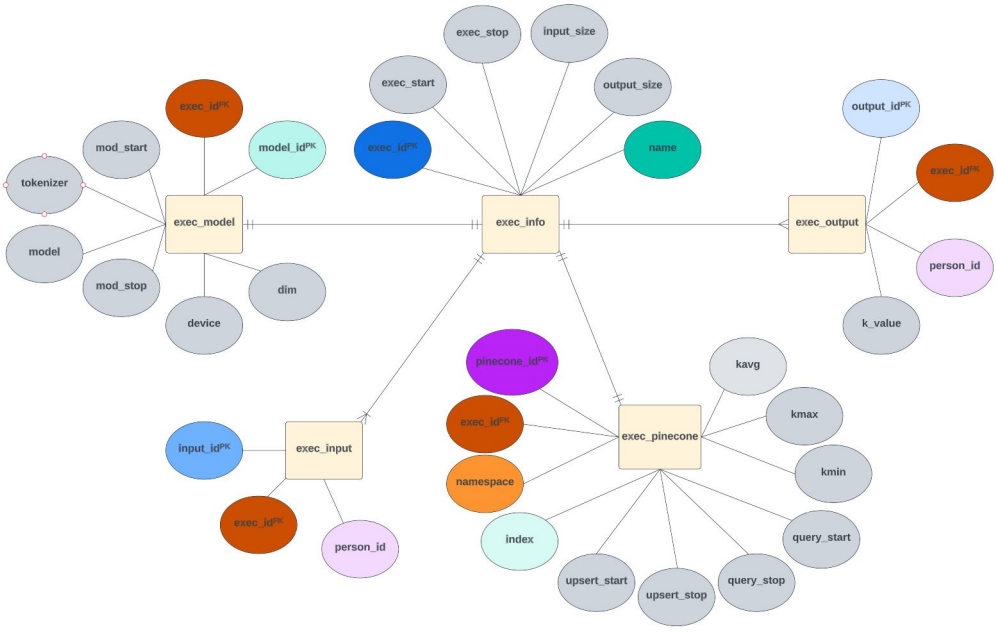

Figure 1: Database ERD

Schema

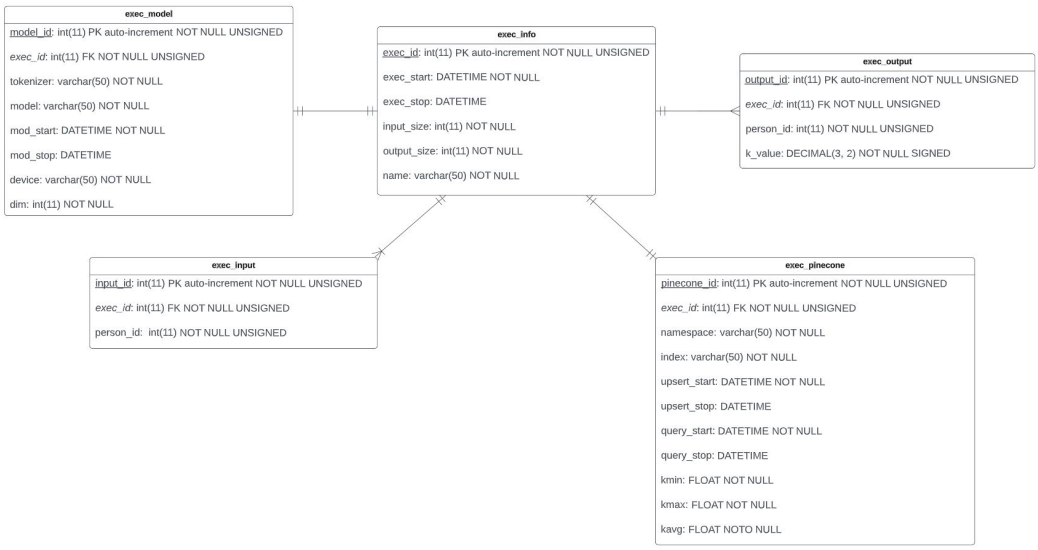

Figure 2: Database Schema

- exec_info:
  - exec_id:
    - int(11)
    - auto-increment
    - NOT NULL
    - UNSIGNED
    - PRIMARY KEY
  - exec_start:
    - DATETIME
    - NOT NULL
  - exec_stop:
    - DATETIME
  - input_size:
    - int(11)
    - NOT NULL
  - output_size:
    - int(11)

        * NOT NULL
    - name:
        * varchar(50)
        * NOT NULL

- exec_input:
    - input_id:
        * int(11)
        * auto-increment
        * NOT NULL
        * UNSIGNED
        * PRIMARY KEY
    - exec_id:
        * int(11)
        * FOREIGN KEY
        * NOT NULL
        * UNSIGNED
    - person_id:
        * int(11)
        * NOT NULL
        * UNSIGNED

- exec_output:
    - output_id:
        * int(11)
        * auto-increment
        * NOT NULL
        * UNSIGNED
        * PRIMARY KEY
    - exec_id:
        * int(11)
        * FOREIGN KEY
        * NOT NULL
        * UNSIGNED
    - person_id:
        * int(11)
        * NOT NULL
        * UNSIGNED

  - k_value:
    * DECIMAL(3, 2)
    * NOT NULL
    * SIGNED

- exec_model:
  - model_id:
    * int(11)
    * auto-increment
    * NOT NULL
    * UNSIGNED
    * PRIMARY KEY
  - exec_id:
    * int(11)
    * FOREIGN KEY
    * NOT NULL
    * UNSIGNED
  - tokenizer:
    * varchar(50)
    * NOT NULL
  - model:
    * varchar(50)
    * NOT NULL
  - mod_start:
    * DATETIME
    * NOT NULL
  - mod_stop:
    * DATETIME
  - device:
    * varchar(50)
    * NOT NULL
  - dim:
    * int(11)
    * NOT NULL

- exec_pinecone:
  - pinecone_id:
    * int(11)

    * auto-increment
    * NOT NULL
    * UNSIGNED
    * PRIMARY KEY
- exec_id:
    * int(11)
    * FOREIGN KEY
    * NOT NULL
    * UNSIGNED
- namespace:
    * varchar(50)
    * NOT NULL
- index:
    * varchar(50)
    * NOT NULL
- upsert_start:
    * DATETIME
    * NOT NULL
- upsert_stop:
    * DATETIME
- query_start:
    * DATETIME
    * NOT NULL
- query_stop:
    * DATETIME
- kmin:
    * FLOAT
    * NOT NULL
- kmax:
    * FLOAT
    * NOT NULL
- kavg:
    * FLOAT
    * NOT NULL