Project Diagram

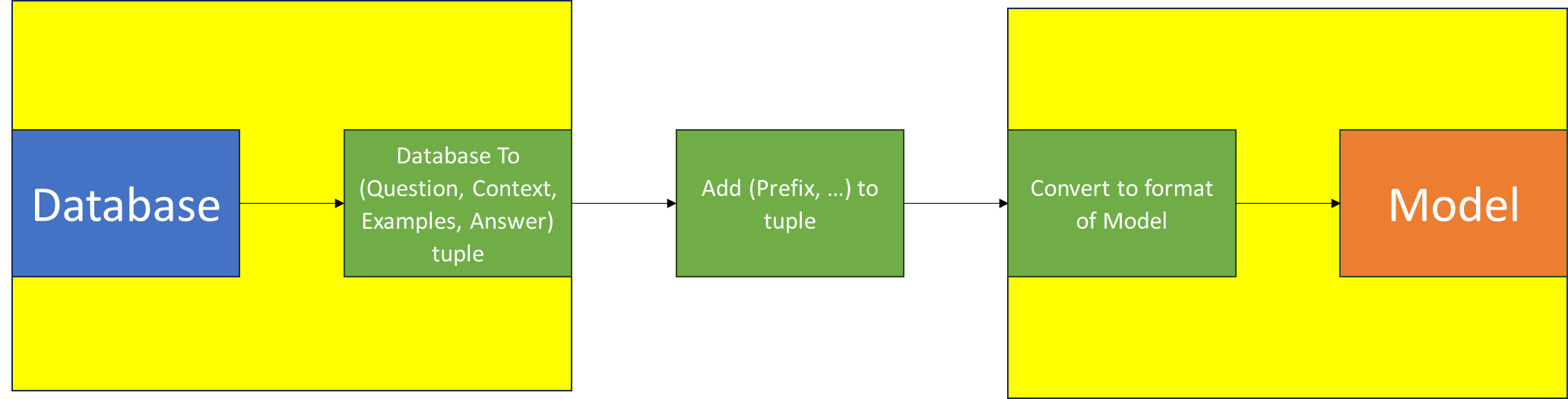
Blue Box – database

Green Box – function

Yellow box – tuple / struct

Orange box – NLP model

Basic structure



Plan:

1. Have a set of functions for each database which parse them into (Question, Context, Examples, Answer), given number of Examples (0+). Examples are Context-Answer pairs.
   1. Sentiment and Emotional Analysis Tweets Dataset
   2. ChatGPT App Reviews Dataset
   3. McDonald’s Reviews Dataset
   4. Fake / Real News Dataset
   5. Boolean Questions Code
      1. Transform into Dataset equivalent
   6. MathQA Dataset
   7. Parallel Translation Corpus
   8. English-French Translation Dataset
   9. English-Hindi Translation Dataset
2. Have a function that returns a (database, database-parser) tuple based on name
   1. Have a function that returns a database based on name. Long list of ‘if database X, call function Y to get X, then return X.
   2. Have a function that returns a database’s parser based on name
      1. Similar to 2a, needs 1 to be completed.
3. Have a function which receives a (database, database-parser) tuple, a number of Examples, and a Test Size. Return a vector of length <Test Size> of (Question, Context, Examples, Answer) with the given number of examples.
4. Have a set of functions that return a Prefix to questions. Return randomly sampled from list
   1. Polite
   2. Neutral
   3. Impolite
5. Have a set of functions which receive a vector of (Prefix, Question, Context, Examples) and translate it to a vector of model queries
   1. T5 / T5-flan
   2. Bloom
   3. ChatGPT API
   4. Falcon-instruct
   5. Macaw-answer
6. Have a function that returns a (model, model-parser) tuple based on name
   1. Have a function that returns a model based on name. As in 2a.
   2. Have a function that returns a model’s parser based on name, as in 2b.
7. Have a function that, given a (model, model-parser) tuple and a (Prefix, Question, Context, Examples, Answer) vector, evaluates the model on the vector
   1. Use parser to get (Model Queries) from (Prefix, Question, Context, Examples)
   2. Feed (Model Queries) to model to get (Model Answers)
   3. Compare to (Answers), return result
8. Have a function which receives a set of (dataset name, politeness, model name, test size, number of examples) and evaluates the performance of the given model on the politeness-edited dataset test queries, with a <number of examples>-Shot learning.
   1. Call (2) to get dataset + parser tuple
   2. Call (3) to get (Question, Context, Examples, Answer) vector
   3. Adds Prefix based on politeness level using (4) to get a (Prefix, Question, Context, Examples, Answer) vector
   4. Call (6) to get model + parser tuple
   5. Call (7) to evaluate model on the vector, and return the result
9. Use University Resources to run (8) on multiple different Test Sizes, Examples Number, and Politeness level on all possible combinations of model-dataset
   1. Translation is only for T5-flan, Bloom, and ChatGPT API