From RAGS to Riches: Creating a CDISC Standards Customer Service Agent Using R and Generative AI

March 18, 2025

The Problem

- Standards teams are inundated with questions about how to implement various CDISC standards. It's hard to handle the volume of questions, much less navigate the regular updates to the standards and how they're implemented.

- How can we manage this unending deluge of questions?

Solution

- Create a virtual customer service agent powered by generative AI.
- Context can be included within a prompt that provides this added information.
- There's limits on what can be included in a prompt, as such more targeted retrieval can be leveraged to more efficiently provide context to the prompt.
- This class of solutions are called retrieval augmented generation or RAG.

Note About Data Privacy

- Data privacy remains a critical concern in AI-driven environments that handle sensitive clinical trial information.
- Rule of thumb: Do not put anything into a prompt that you couldn't share externally from your company.

Ingredients

- We're going to walk through the elements involved in creating this customer service agent.
- Few ingredients below.

For interacting with embeddings models

For accessing LLM APIs

Used to build the chat interface

Steps in the Process

1. Set up OpenAI API Key (or LLM of your choosing).
2. Basic query with gptstudio.
3. Standards document from CDISC.
4. Create an embedding
5. Roundtrip query.
6. Integrate with a chat interface

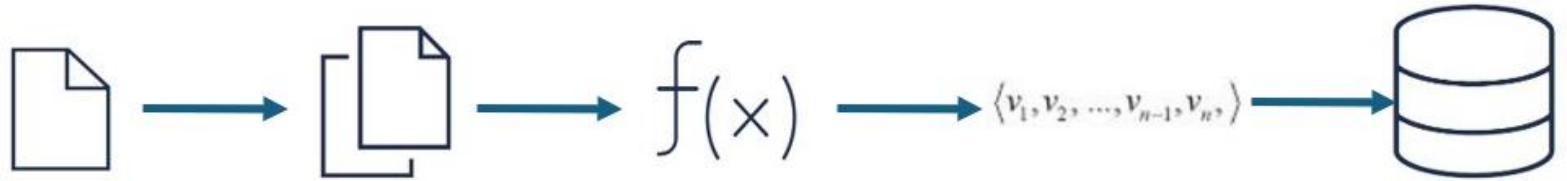
Basic Query

```
# Load dependencies
library(gptstudio)
library(gpttools)

# Set prompt
my_prompt <- "Using dplyr, how would I subset the adverse events dataset to
              Treatment Emergent Serious Adverse Events?"

simple_resp <- gptstudio::chat(prompt = my_prompt, model = "gpt-4o")
cat(simple_resp)
```

Basic Response

```
# Load the dplyr package
library(dplyr)

# Assuming your dataset is called 'adverse_events'
# and the column that specifies the event type is 'event_type'
# and the rows are characterized as "Treatment Emergent Serious Adverse Event"

# Subset the data
serious_adverse_events <- adverse_events %>%
  filter(event_type == "Treatment Emergent Serious Adverse Event")
```

RAG Workflow



CDISC ARDS Implementation Guide

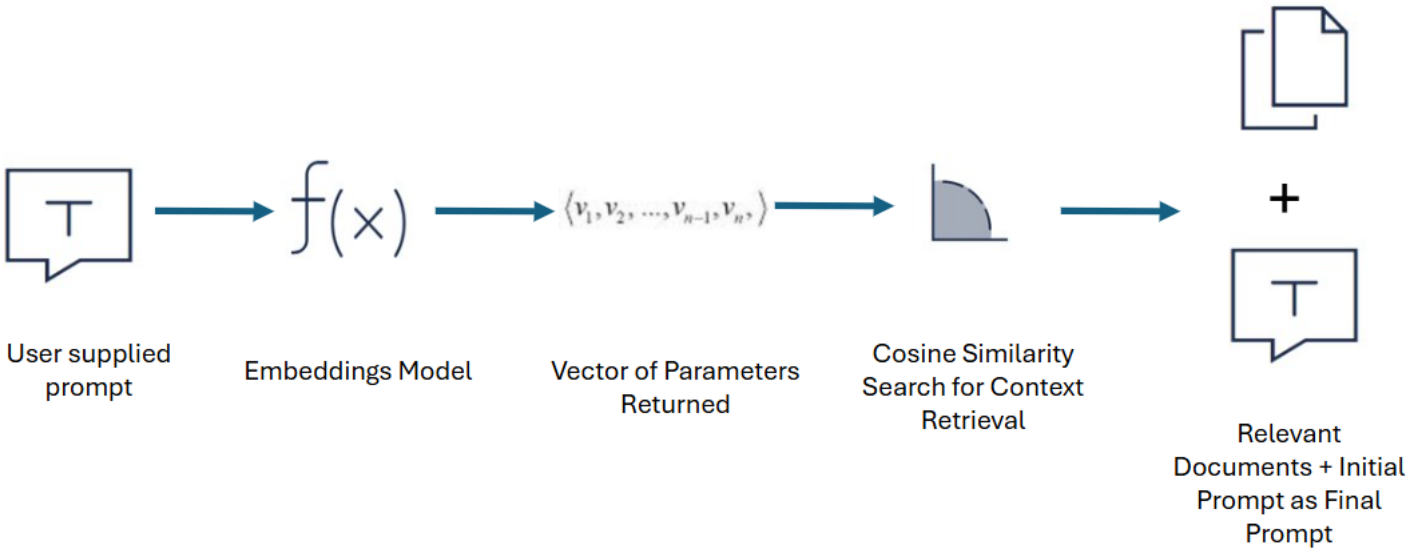
Chunked Into Smaller Text Fragments

Embeddings Model

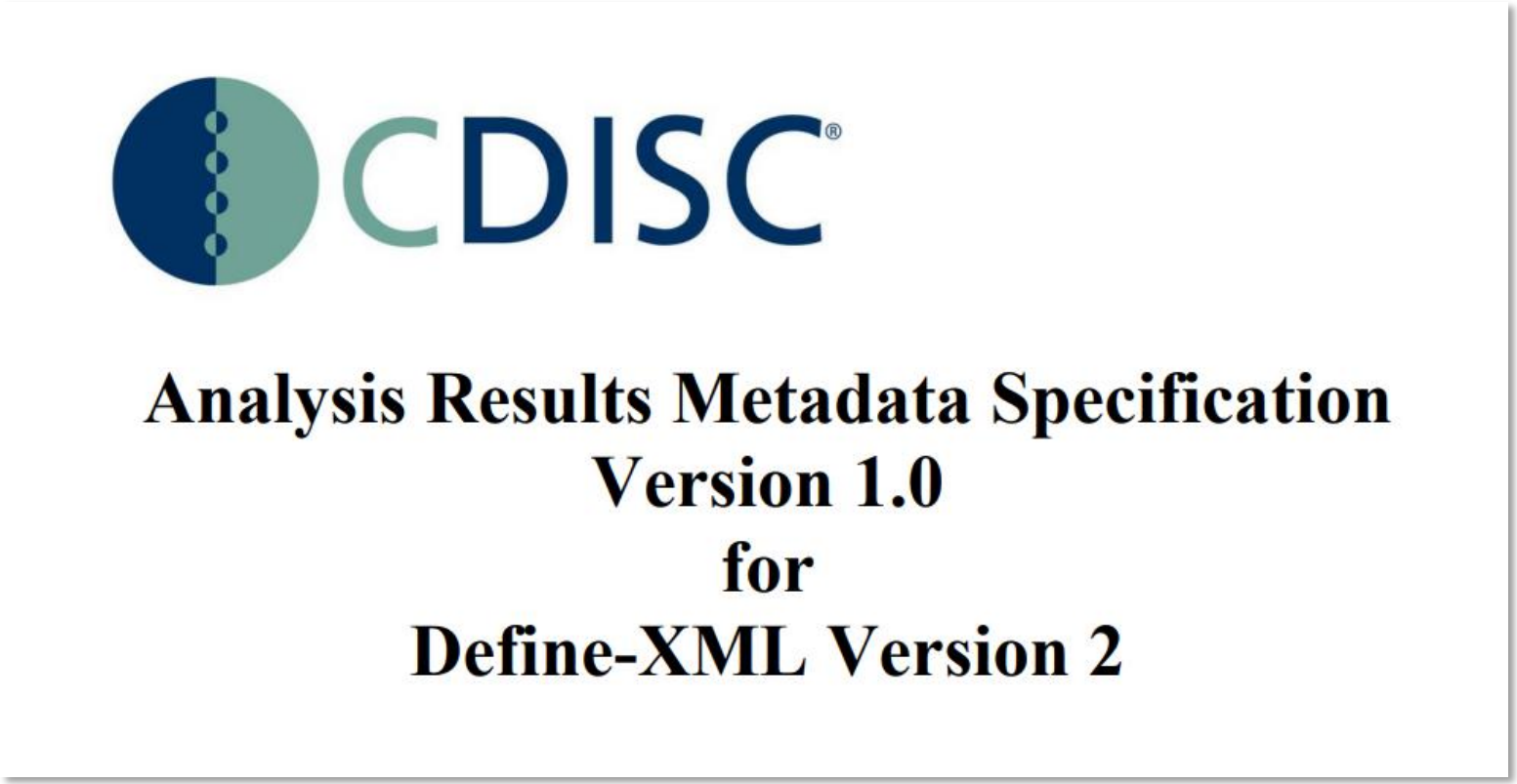
Vector of Parameters Returned

Vectors Stored in DB for Later Retrieval

RAG Workflow Continued



Standards Specification



Context Example

```
# Create index of embeddings
create_index_from_pdf(file_path = "Analysis Results Metadata v1.0 for Define-XML v2.pdf",
                      source = "cdisc",
                      link = "",
                      overwrite = FALSE)

# Load index
cdisc_index <- load_index(domain = "cdisc")

# Chat with index
context_resp <- chat_with_context(
  query = prompt,
  service = "openai",
  model = "gpt-4o",
  index = cdisc_index,
  k_context=10)[[3]]
cat(context_resp)
```

Context Reply

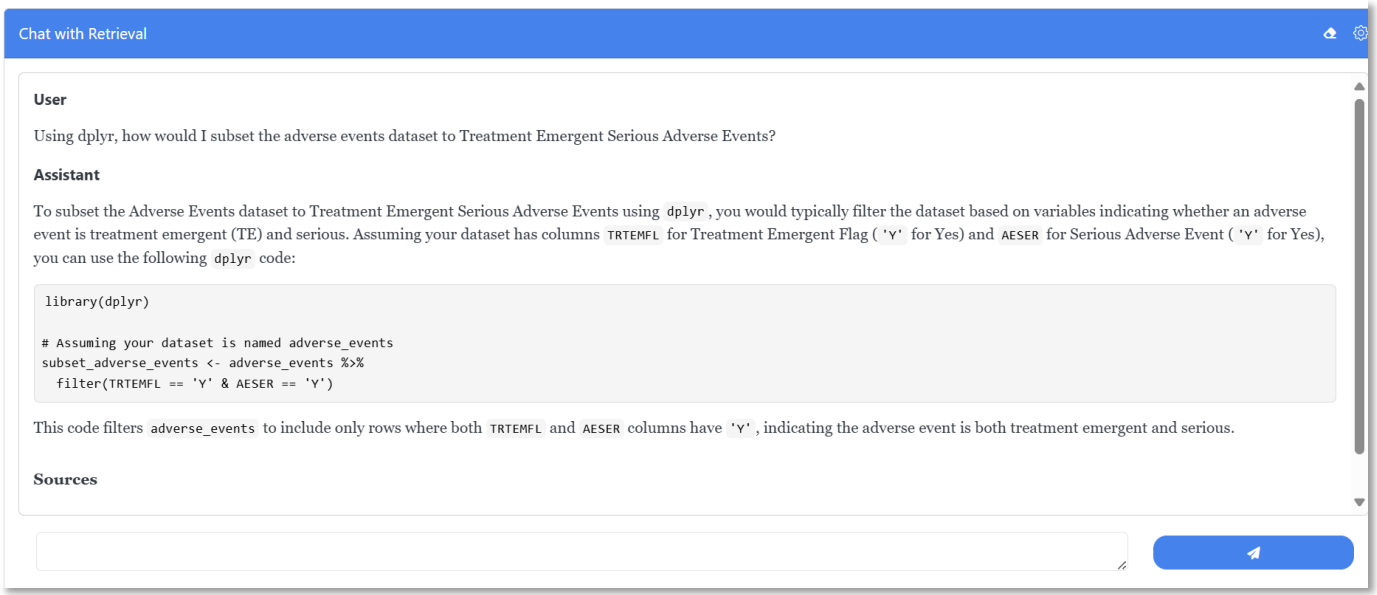
```
library(dplyr)

# Assuming your dataset is named 'ADAE'
ADAE_filtered <- ADAE %>%
  filter(TRTEMFL == 'Y' & AESER == 'Y')
```

What's in the payload?

```
# A tibble: 10 × 10
   source link  chunks                                                  n_tokens hash  usage embedding embedding_method version similarity
   <chr>  <chr> <chr>                                                      <dbl> <chr> <int> <list<li> <chr>            <chr>         <dbl>
 1 cdisc  ""    "will be taken from the Name attribute of the ItemDef wit…      663 7a3c…   630   [1,536] OpenAI           Not a …       0.402
 2 cdisc  ""    "> Subject - Level Analysis < / TranslatedText > < / Desc…      675 112f…   702   [1,536] OpenAI           Not a …       0.386
 3 cdisc  ""    "of the documentation depend on the level of detail requi…      667 6810…   607   [1,536] OpenAI           Not a …       0.373
 4 cdisc  ""    "ADSL is [ SAFFL = ’ Y ’ ] which subsets the Subject Leve…      660 1a8d…   679   [1,536] OpenAI           Not a …       0.372
 5 cdisc  ""    "an event occurred was recorded by Preferred Term , Syste…      716 f385…   570   [1,536] OpenAI           Not a …       0.351
 6 cdisc  ""    "/ www.cdisc.org / adam • ADaM Examples in Commonly Used …      714 3cfd…   635   [1,536] OpenAI           Not a …       0.323
 7 cdisc  ""    "> … < CodeListItem CodedValue = \" ACTOT \" OrderNumber …      655 77a2…   650   [1,536] OpenAI           Not a …       0.318
 8 cdisc  ""    "provided . Page 24 © 2015 Clinical Data Interchange Stan…      787 022a…   662   [1,536] OpenAI           Not a …       0.306
 9 cdisc  ""    ". . . . . . . . . . . . . . . . . . . . . . . . . . . . …      580 f092…   580   [1,536] OpenAI           Not a …       0.305
10 cdisc  ""    ". . . . . . . . . . . . . . . . . . . . . . . . . . . . …      278 fb0d…   540   [1,536] OpenAI           Not a …       0.305
```

Putting in a Shiny App

```
# Launch local Shiny app
chat_with_retrieval()
```



Questions?