CCR Summer Internship

REU experience: May 28 - Aug 16 2024
Student: Timothy Scholtz - Computer Eng.
Mentor: Andrew Stoltman

University at Buffalo

Presentation contents

Code Generation (DAF)

- Purpose
- Usage
- Capabilities

Usage Tab Urls

- Overview
- Capabilities

XDMoD ChatBot

- Motivation
- Background
- Tests
- Performance

Presentation contents

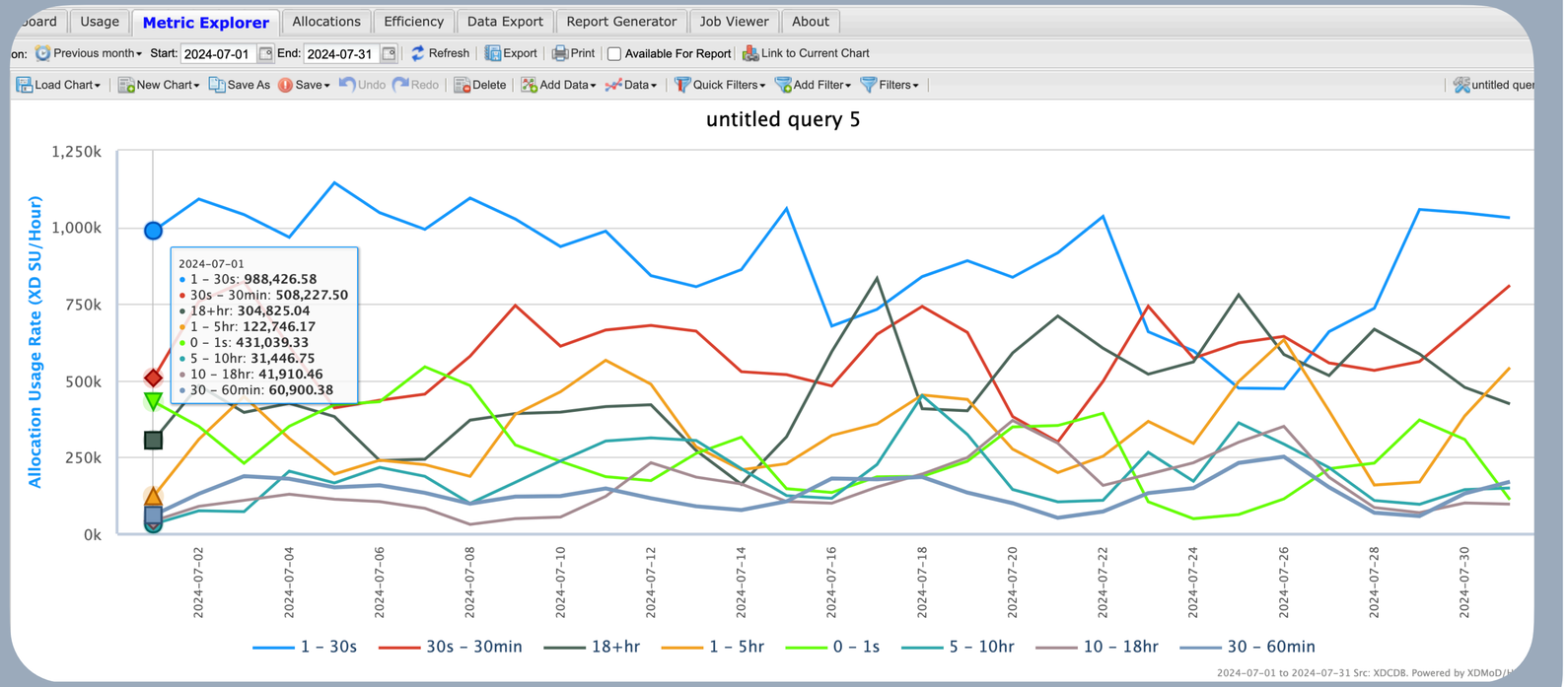
Code Generation (DAF)

- Purpose
- Usage
- Capabilities

Code Generation (DAF)

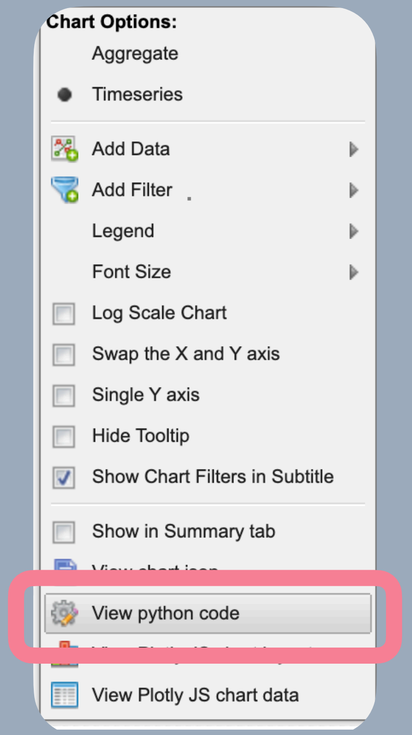
- Purpose

Enable users to re-create Metric Explorer charts in a coding environment for flexible data exploration



Code Generation (DAF)

- Usage



Button in options menu generates the code for the chart you are currently looking at, using the DAF (Data Analytics Framework)

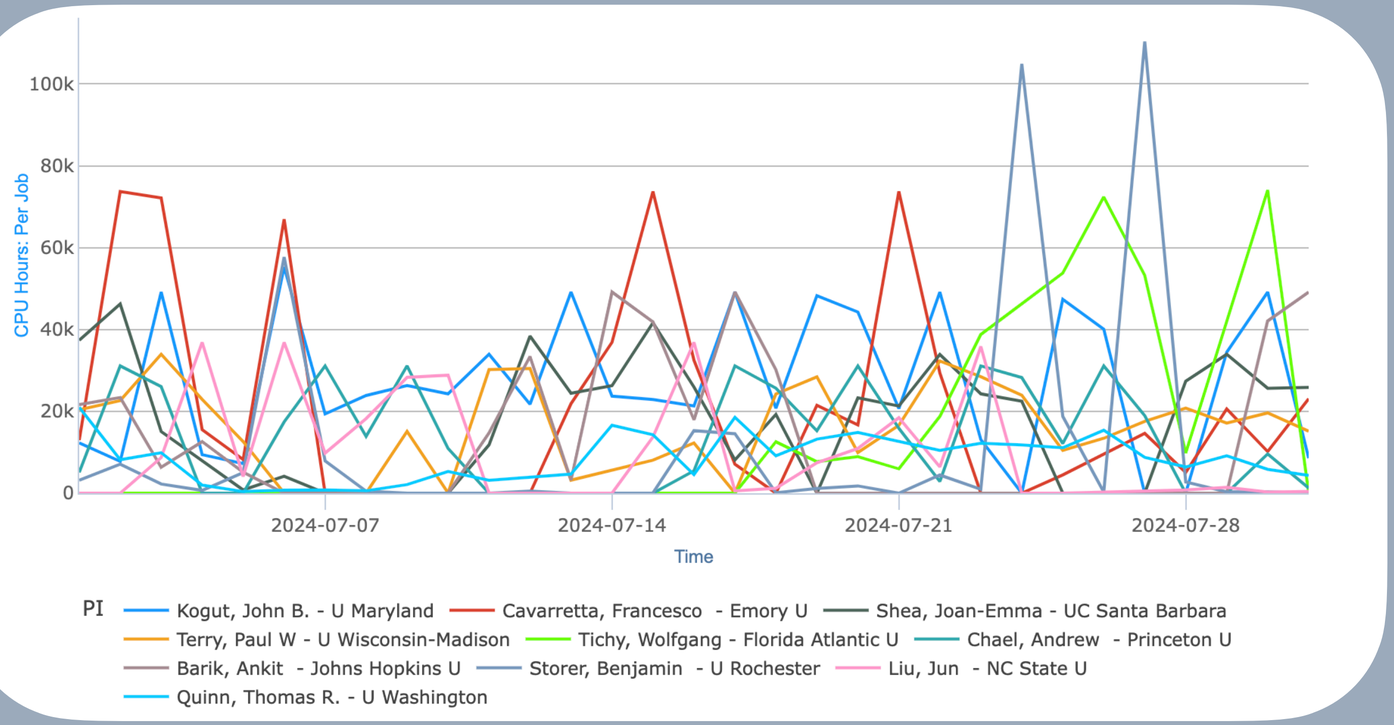
API Code

```
Python API code
***********************************************

import pandas as pd
# Call to Data Analytics Framework requesting data
with dw:
    data_0 = dw.get_data(
        duration=('Previous month'),
        realm='Jobs',
        metric='avg_cpu_hours',
        dimension='pi',
        filters={},
        dataset_type='timeseries',
        aggregation_unit='Auto',
    )



# Limit the number of data items/source to at most 10 and sort by descending';
    columns_list = data_0.columns.tolist()
    if (len(columns_list) > 10):
        column_sums = data_0.sum()
        top_ten_columns = column_sums.nlargest(10).index.tolist()
        data_0 = data_0[top_ten_columns]


# Format and draw the graph to the screen

    plot = px.line(
    data_0,
    labels={"value": dw.describe_metrics('Jobs').loc['avg_cpu_hours', 'label']},
    title='untitled query 1,'

    )
    plot.update_layout(
        xaxis_automargin=True,
    )
    plot.show()

***********************************************

The link to the data analytisc API can be found here

Infomation about the Plotly Express Libary can be found here
```

Code Generation (DAF)

- Usage

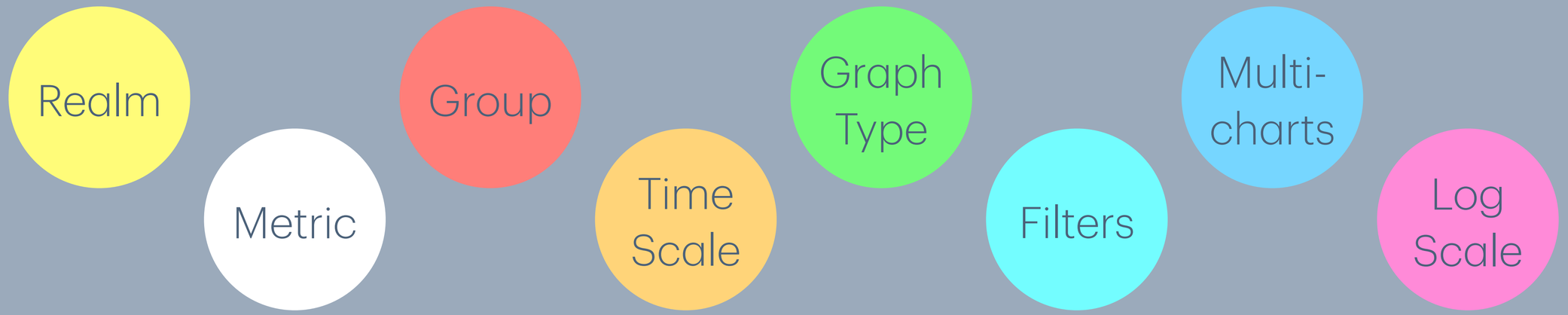
This code can be run in an XDMoD notebook thanks to the DAF, allowing users to explore and analyse the data using their own tools and discretion



Code Generation (DAF)

- ## Capabilities

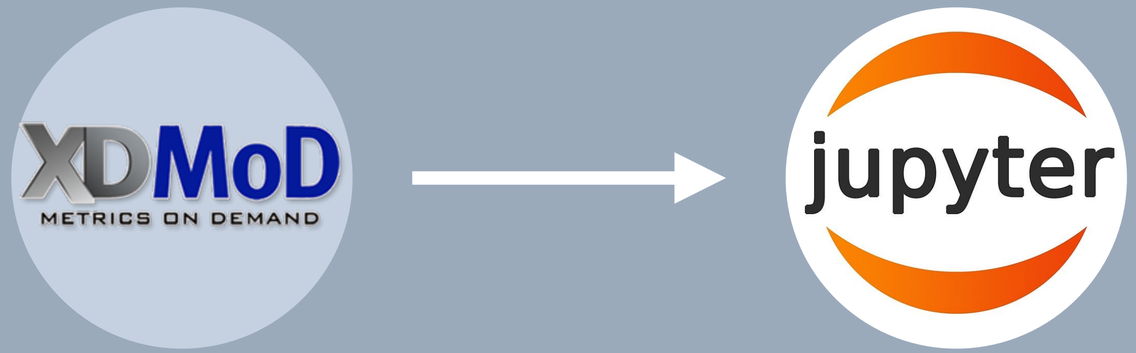
Supports many of the options available on the Metric Explorer to try and re-create the current chart the user is looking at as close as possible. Some of the features are:



Code Generation (DAF)

- Later Work

The ultimate goal is to enable users to launch a notebook on XDMoD resources with the chart preloaded, allowing for quick and easy interaction with the data



Presentation contents

Code Generation (DAF)

- Purpose
- Usage
- Capabilities

Presentation contents

Code Generation (DAF)

- Purpose
- Usage
- Capabilities

Usage Tab Urls

- Overview
- Capabilities

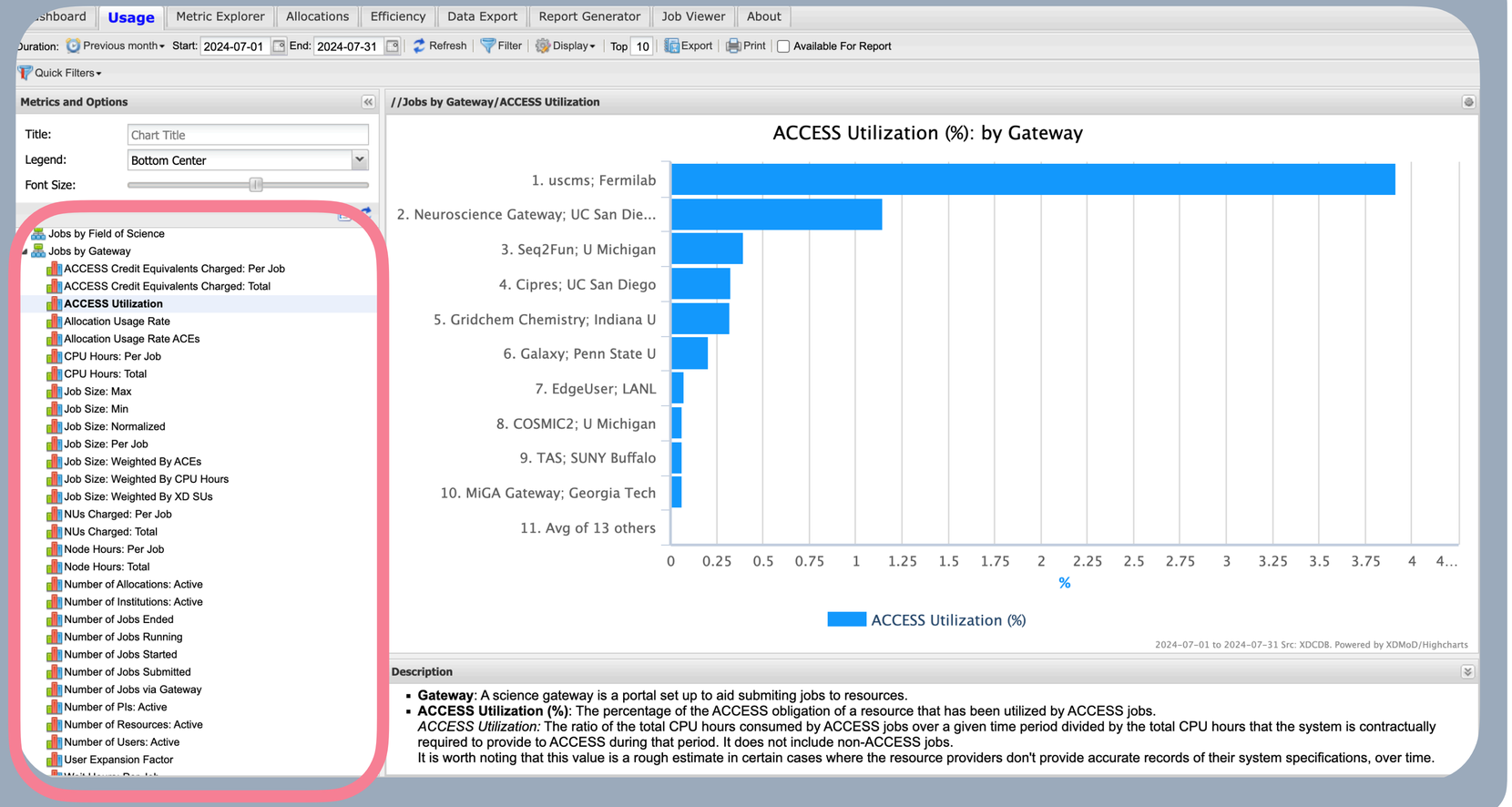
XDMoD ChatBot

- Motivation
- Background
- Tests
- Performance

Presentation contents

Usage Tab Urls

The Usage tab has a collection of pre-configured charts which can be viewed by anyone without logging in



Usage Tab Urls

These are Usage Tab urls. Each chart has an unique url, but when loading, it will default to a standard chart

<u>Until now!</u>

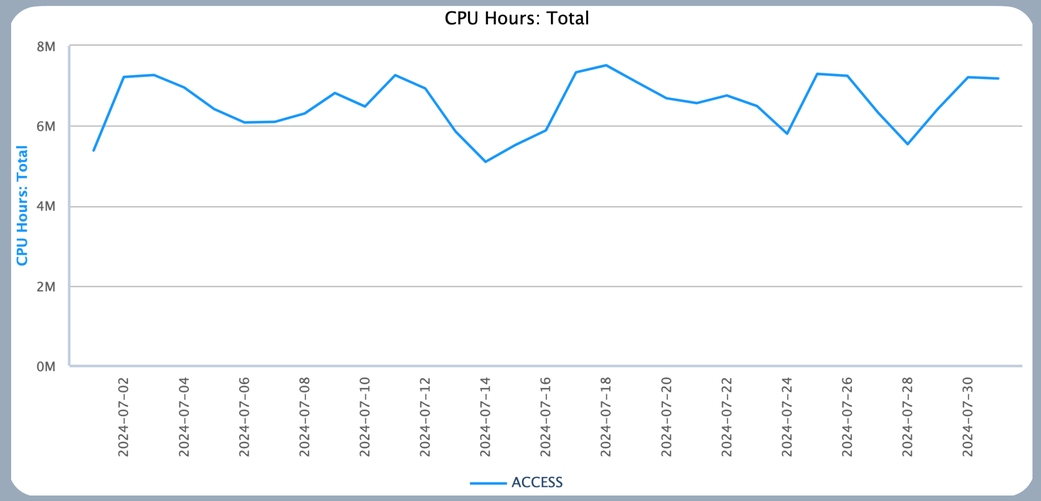
Updated so link take you to correct chart

xdmod.access-ci.org/index.php#tg_usage?node=statistic_Jobs_gateway_avg_ace

xdmod.access-ci.org/index.php#tg_usage?node=statistic_Gateways_nodecount_

xdmod.access-ci.org/index.php#tg_usage?node=statistic_SUPREMM_queue_total_ace

All lead to



Usage Tab Urls

Comparison of Usage tab url's to Metric tab url's

Usage Tab: https://xdmod.access-ci.org/#tg_usage?node=statistic_Jobs_resource_type_active_allocation_count

Metric Explorer: https://xdmod.access-ci.org/#main_tab_panel:metric_explorer?
config=eyJmZWF0dXJlZCI6ZmFsc2UsInRyZW5kX2xpbmUiOmZhbHNlLCJ4X2F4aXMiOnt9LCJ5X2F4aXMiOnt9LCJsZWdlbmQiOnt9LCJkZWZhdWx0RGF0YXNldENvbmZpZyI6e30sInN3YXBfeHkiOmZhbHNlLCJzaGFyZV95X2F4aXMiOmZhbHNlLCJoaWRlX3Rvb2x0aXAiOmZhbHNlLCJzaG93X3JlbWFpbmRlciI6ZmFsc2UsInRpbWVzZXJpZXMiOnRydWUsInRpdGxlIjoiQ29yZSBIb3VyIFV0aWxpemF0aW9uIChmellbGQgb2YgU2NpZW5jZSI6dHJ1ZSwic2hvd193YXJuaW5ncyI6dHJ1ZSwiZGF0YV9zZXJpZXMiOnsiZ3JvdXBfYnkiOiJmaWVsZ9mc2NpZW5jZSIsImNvbG9yIjoiYXV0byIsImxvZ19zY2FsZSI6ZmFsc2UsInN0ZF9lcnIiOmZhbHNlLCJ2YWx1ZV9sYWJlbHMiOmZhbHNlLCJkaXNwbGF5X3R5cGUiOiJhcmVhIiwiY29tYmluZV90eXBlIjoic2lkZSIsInNvcnRfdHlwZSI6InZhbHVlX2Rlc2MiLCJpZ25vcmVfZ2xvYmFsIjpmYWxzZSwibG9uZ19sZWdlbmQiOnRydWUsInhfYXhpcyI6ZmFsc2UsImhhc19zdGRfZXJyIjp0cnVlLCJ0cmVuZF9saW5lIjpmYWxzZSwibGluZV90eXBlIjoiU29saWQiLCJsaW5lX3dpZHRoIjoyLCJzaGFkb3ciOmZhbHNlLCJmaWx0ZXJzIjp7ImRhdGEiOltdLCJ0b3RhbCI6MH0sInpfaW5kZXgiOiAwLCJ2aXNpYmlsaXR5IjpudWxsLCJlbmFibGVkIjp0cnVlLCJtZXRyaWMiOiJjbG91ZF9jb3JlX3V0aWxpemF0aW9uIiwicmVhbG0iOiJDbG91ZCIsImNhdGVnb3J5IjoiQ2xvdWQiLCJpZCI6IjAwMDAwNTk3MzMifV0sInRvdGFsIjoxfSwiYWdncmVnYXRpb25fdW5pdCI6IkF1dG8iLCJnbG9iYWxfZmlsdGVycyI6eyJkYXRhIjogW10sInRvdGFsIjowfSwidGltZWZyYW1lX2xhYmVsIjoiMjAyMCIsInN0YXJ0X2RhdGUiOiIyMDIwLTAxLTAxIiwiZW5kX2RhdGUiOiIyMDIwLTEyLTMxIiwic3RhcnQiOjAsImxpbWl0IjoxMH0=

Presentation contents

Presentation contents

Code Generation (DAF)

- Purpose
- Usage
- Capabilities

Usage Tab Urls

- Overview
- Capabilities

XDMoD ChatBot

- Motivation
- Background
- Tests
- Performance

Presentation contents

XDMoD ChatBot

- Motivation
- Background
- Tests
- Performance

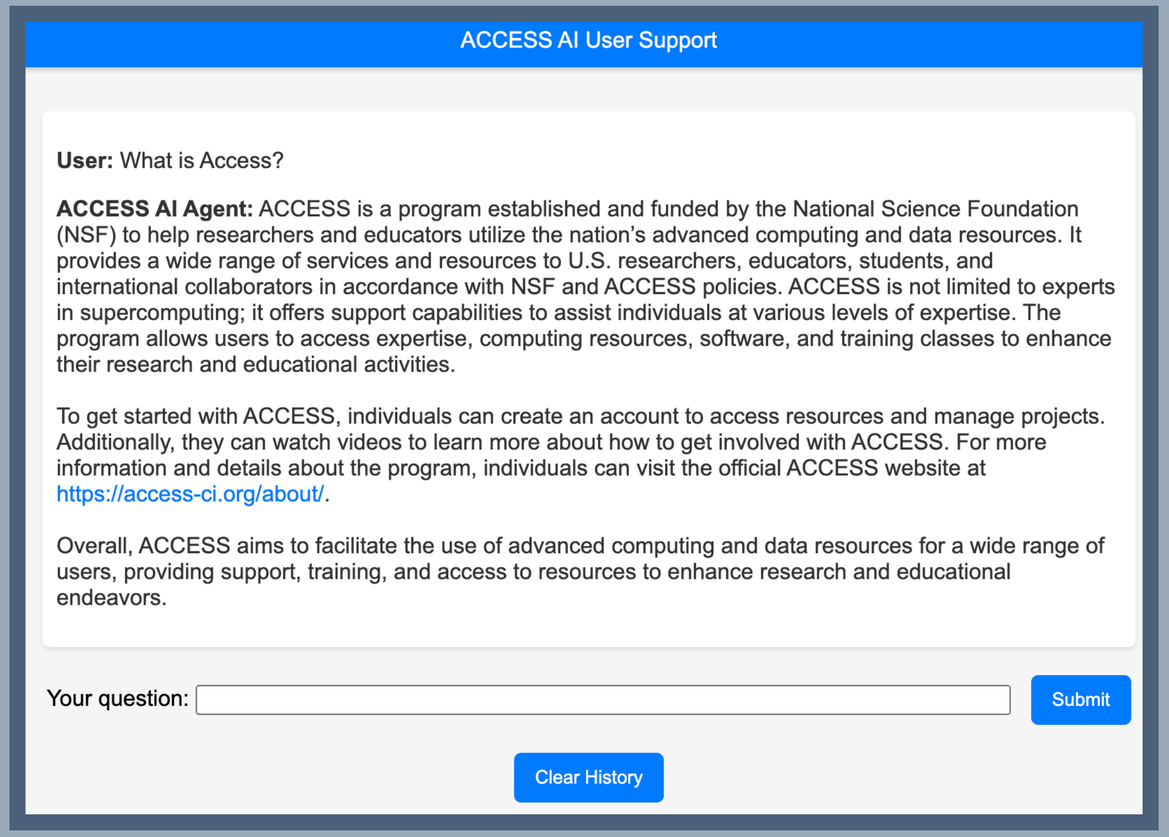
XDMoD ChatBot

Motivation

Background

The ACCESS team is working on a chat bot that can answer ACCESS related questions.

It can construct in-depth answers and is very flexible.



XDMoD ChatBot

Motivation

Goal

We want to update it so it can provide users with XDMoD charts to their questions when applicable.

Background Concepts

XDMoD Charts

Embeddings

LLM

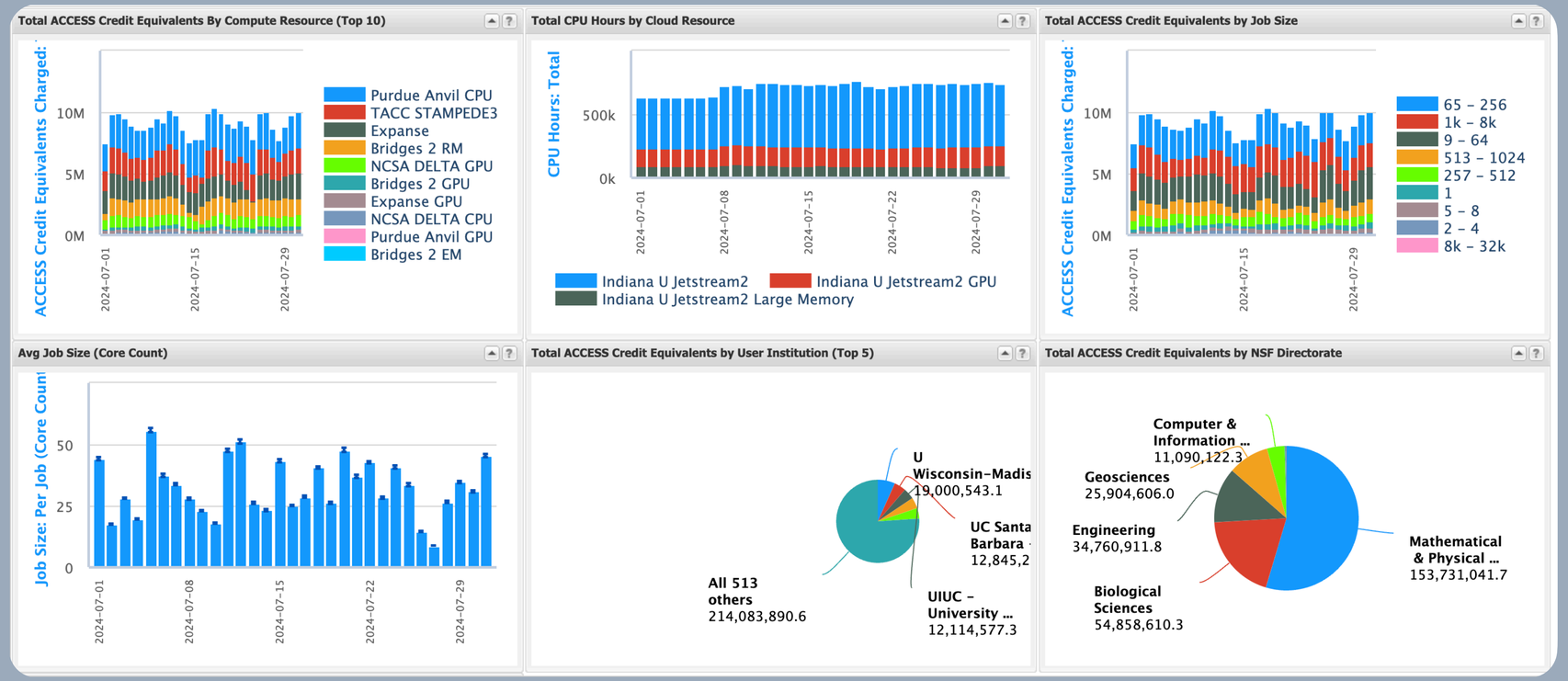
RAG

Background Concepts

XDMoD Charts

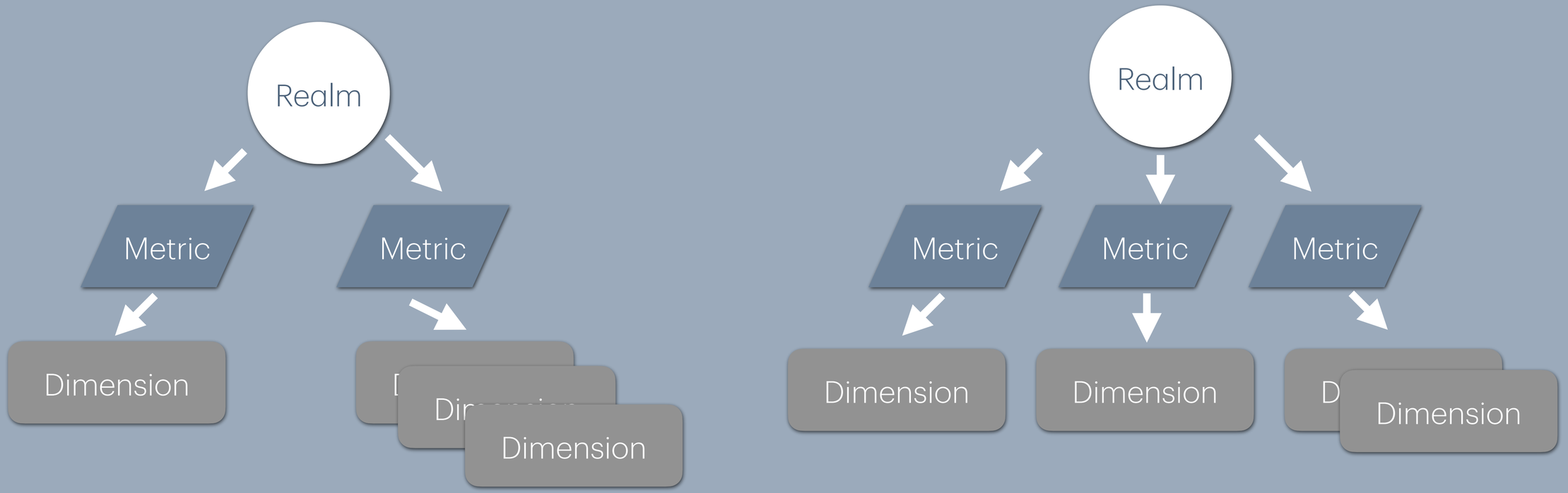
XDMoD Charts

XDMoD tracks ACCESS Supercomputers' usage and performance, providing data visualisation charts



XDMoD Charts

The data hierarchy consists of realms (data sources), each recording and grouping information in different ways



XDMoD Charts

Each chart is a JSON object with parameters like time series, group_by, graph type, metric, realm, and others

When customising a chart, your are updating the JSON object

Given the large number of parameters, the total number of chart combinations is easily in the millions

Background Concepts

XDMoD Charts

Background Concepts

XDMoD Charts

Embeddings

LLM

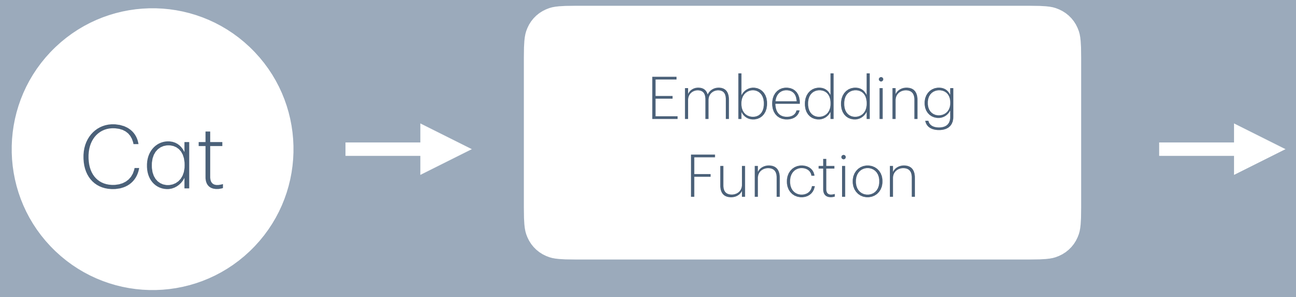
RAG

Background Concepts

Embeddings

Embeddings

An embedding is a high-dimensional vector which represents data in space. The distance between embeddings indicates how related they are.



Takes in a word / concept

Does weird Math Magic

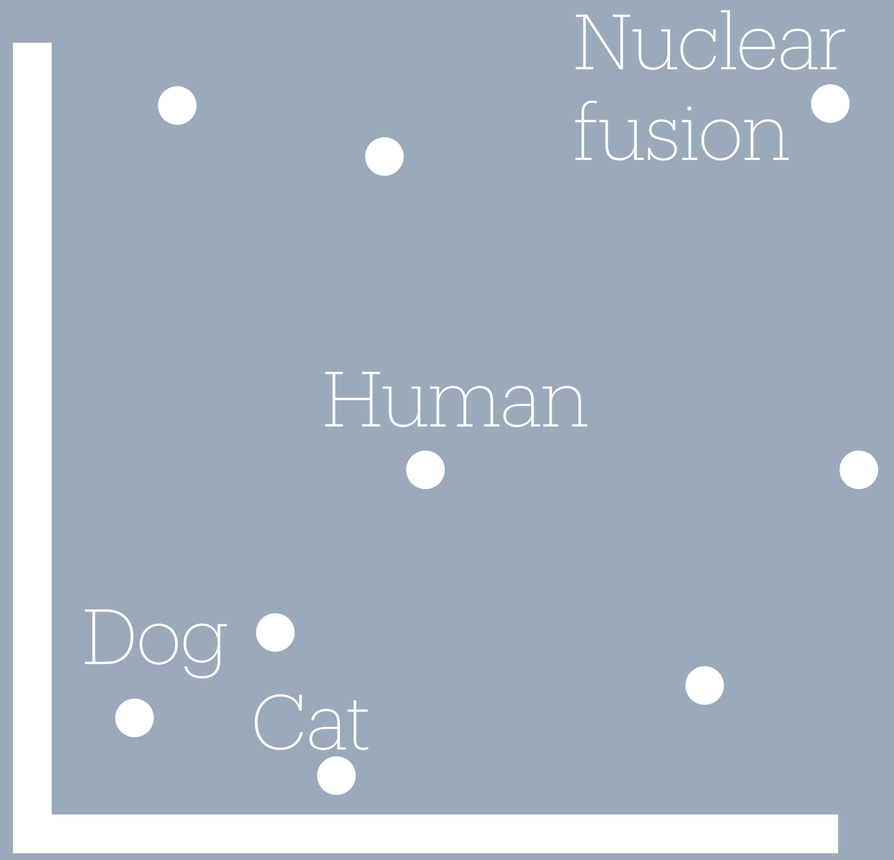
3.346	0.2435	34.1456
8.789	10.2345	6.3654
0.14321	6.376	28.254
0.4763	92.32	47.24
57.654	11.345	.....

Produces complex vector

Embeddings

Example

The embeddings of "cat" and "dog" would be closer to each other than to "nuclear fusion".



*Embeddings exist in 50 dimensional or higher space, no 2D

Background Concepts

Embeddings

Background Concepts

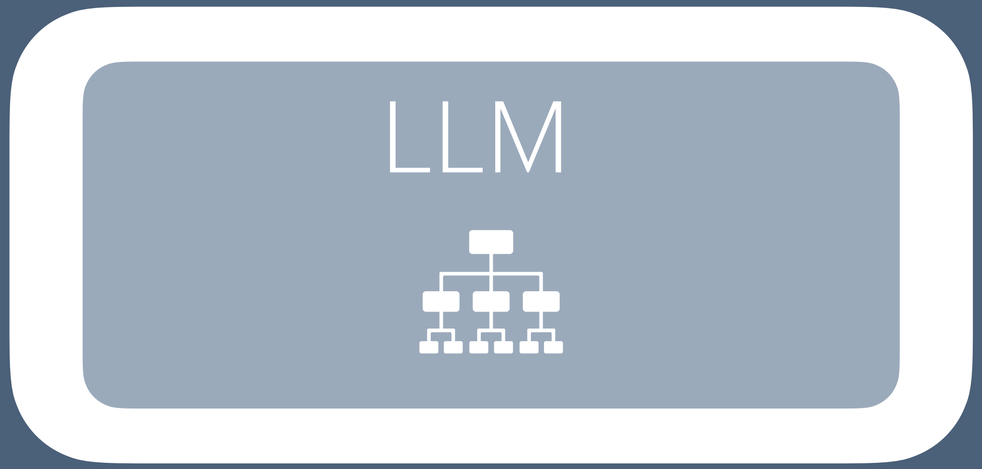
XDMoD Charts

Embeddings

LLM

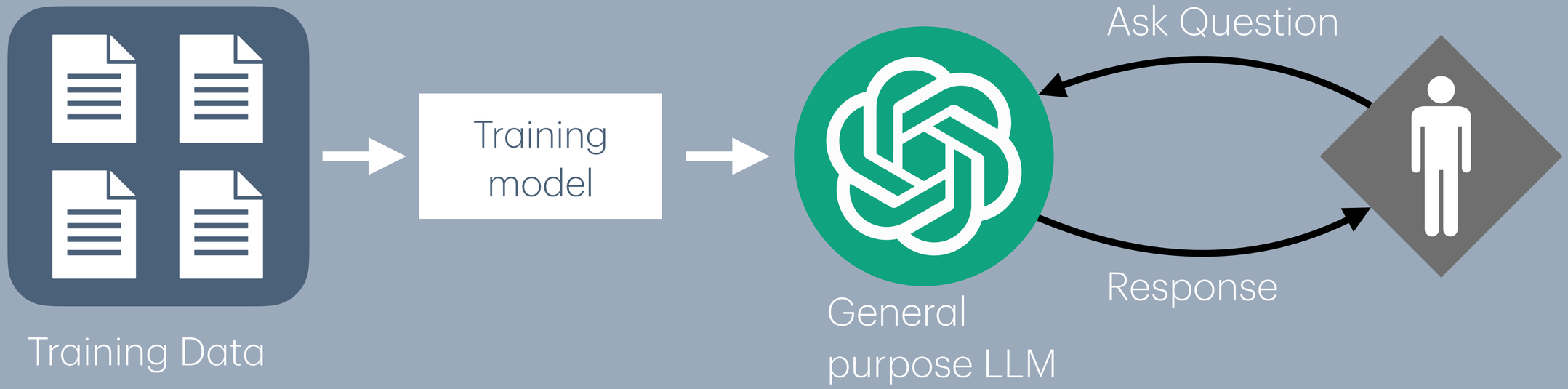
RAG

Background Concepts



LLM(Large Language Model)

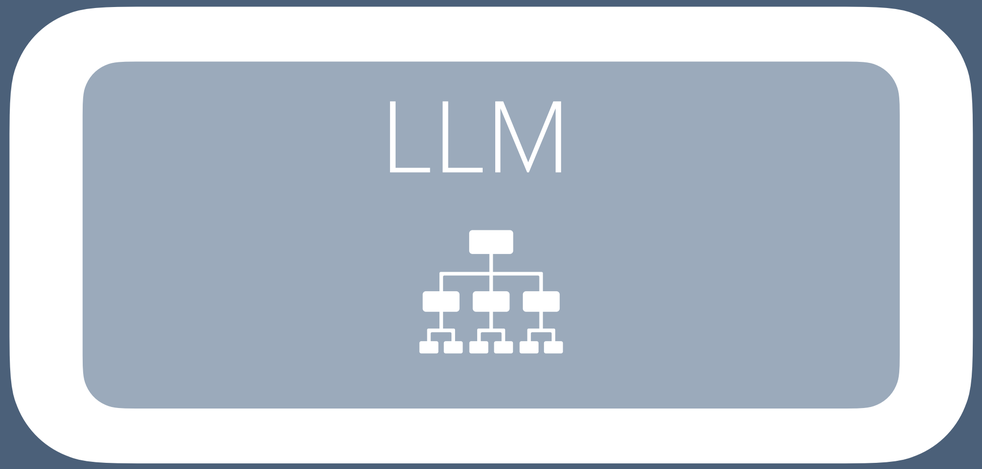
LLM are trained on huge data sets of language in order to learn. When responding their answers are an accumulation of everything they learned, and they can't point to a single source.



LLM(Large Language Model)

The ACCESS teams chat bot is built using the GPT API, which allows them to send user queries to OpenAI and get a response from them.

Background Concepts



Background Concepts

XDMoD Charts

Embeddings

LLM

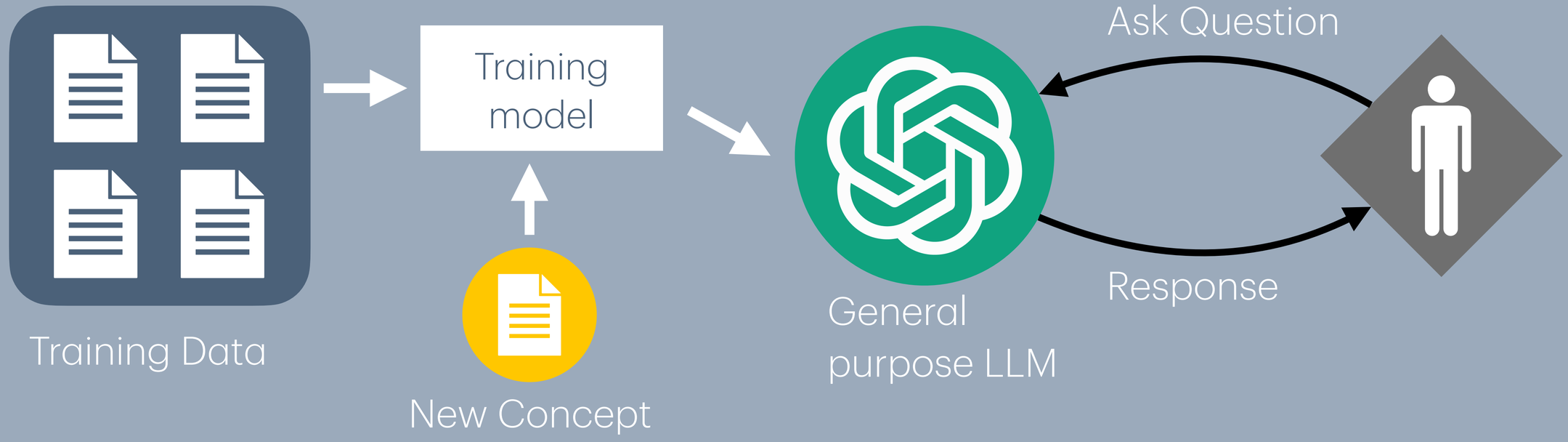
RAG

Background Concepts

RAG

Problems with LLM's

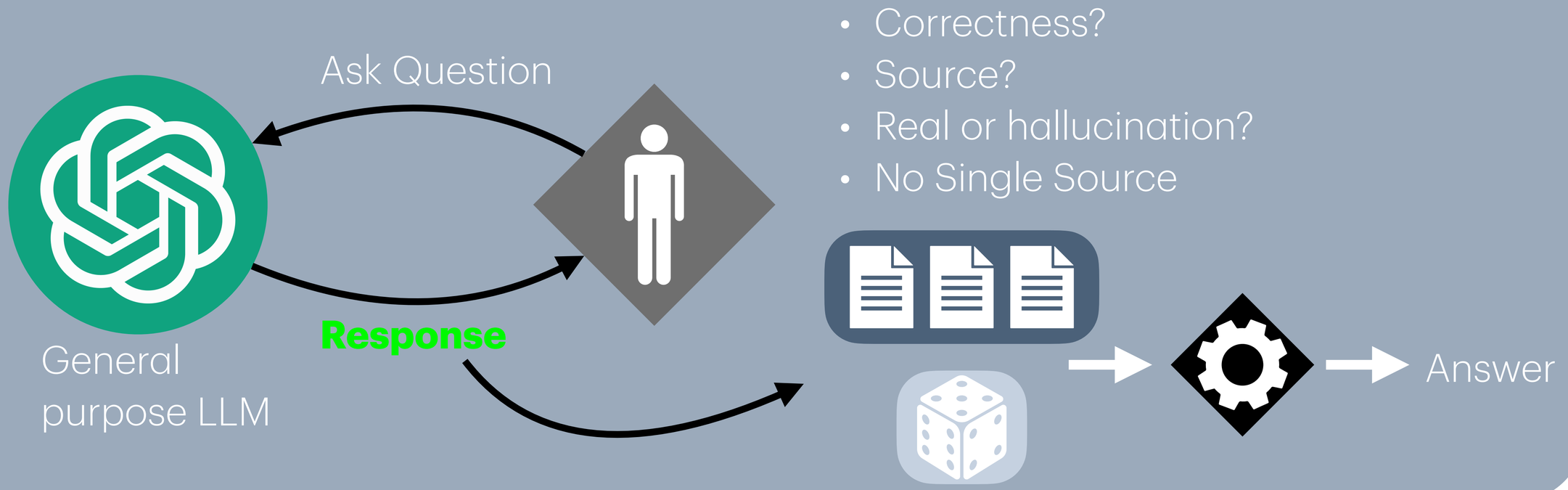
1) LLMs require *retraining* from scratch if you want to add new data into. Very resource and time intensive



RAG

Problems with LLM's

2) LLMs can't point where they got their information from, m they suffer hallucination and other problems

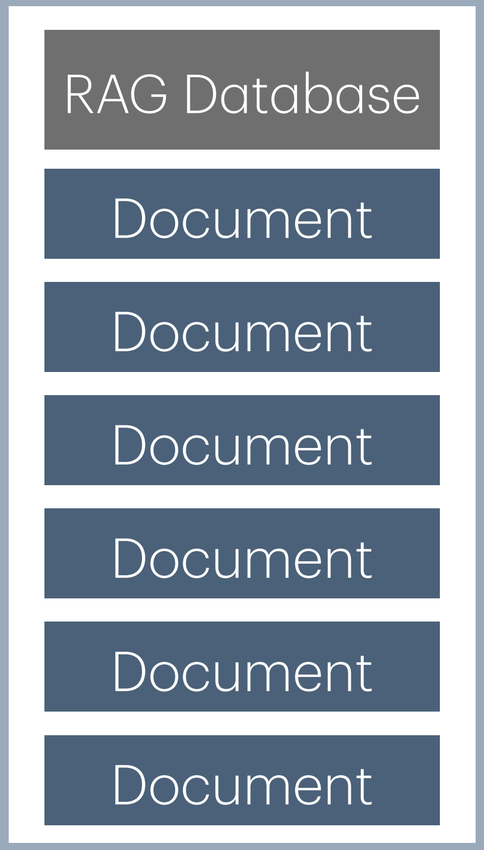


RAG(Retrieval Augmented Generation)

Solution - RAG

Provide the model at run time with a document which has the answer.

RAG(Retrieval Augmented Generation)

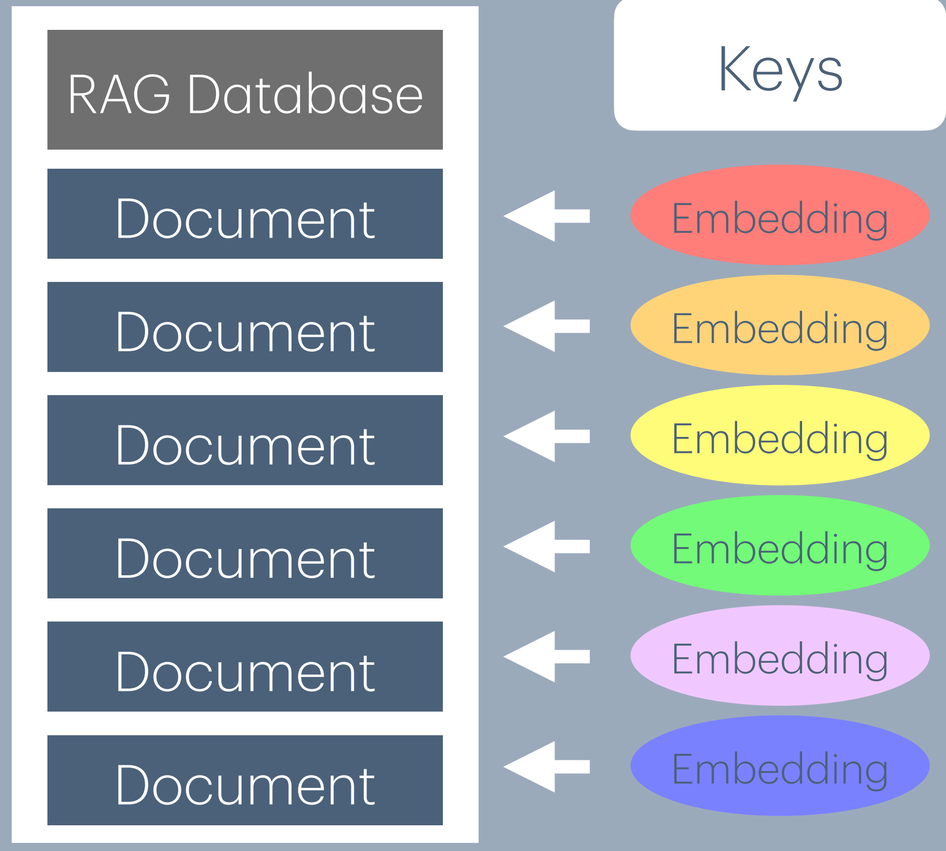


RAG Database

If you know the type of questions you want to answer, you can collect relevant documents and place them in a database.

Then you can grab the relevant document when you want to answer a questions

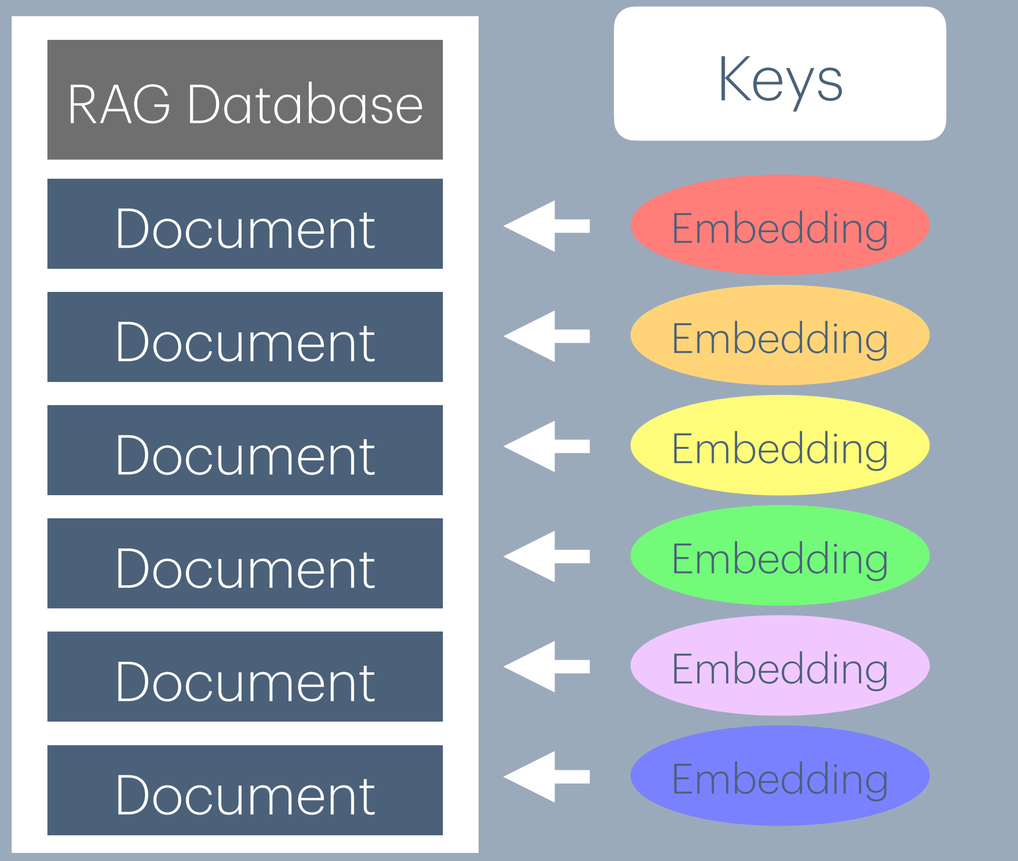
RAG(Retrieval Augmented Generation)



How to grab the correct document?

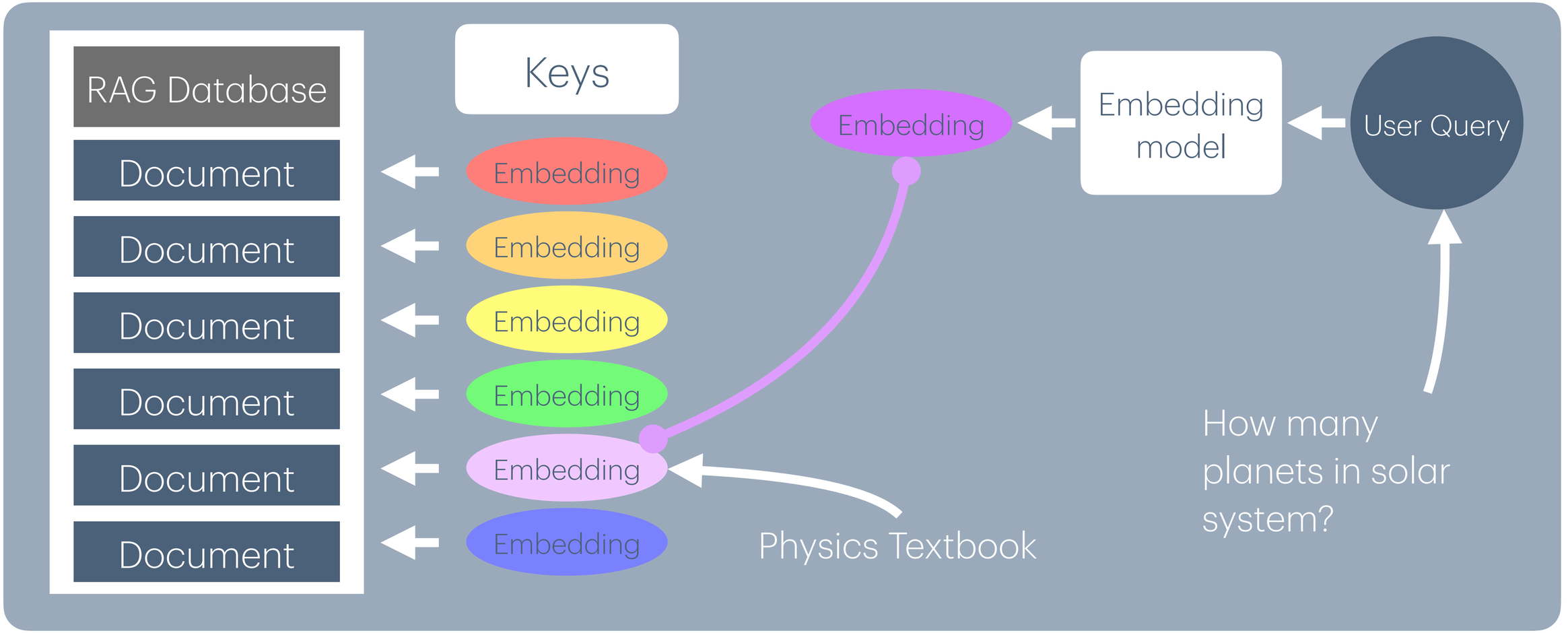
Every document has an embedding, which acts as a key. The embedding it created by passing a description of the document through an embedding model

RAG(Retrieval Augmented Generation)

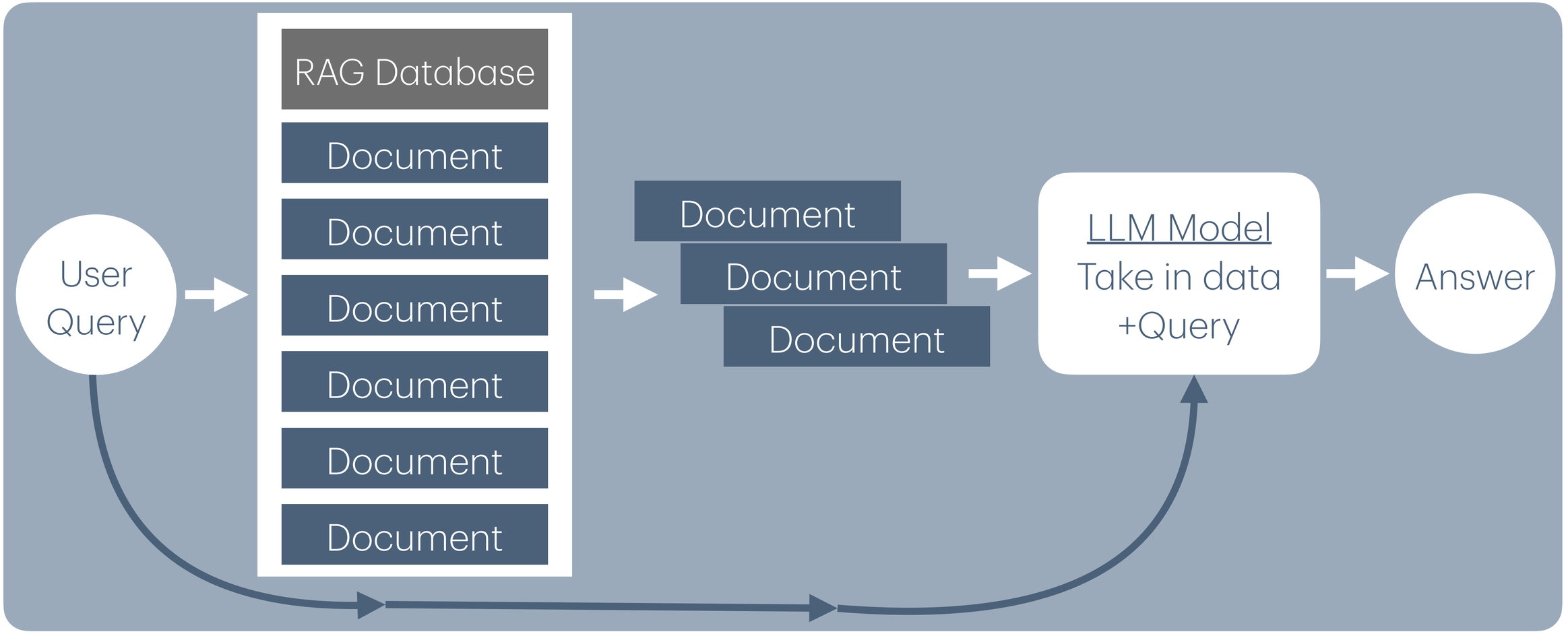


Then to retrieve a document, you can embed the users query, and look for the closet match between the document and question

RAG(Retrieval Augmented Generation)



RAG(Retrieval Augmented Generation)



Background Concepts

Background Concepts

XDMoD Charts

Embeddings

LLM

RAG

Presentation contents

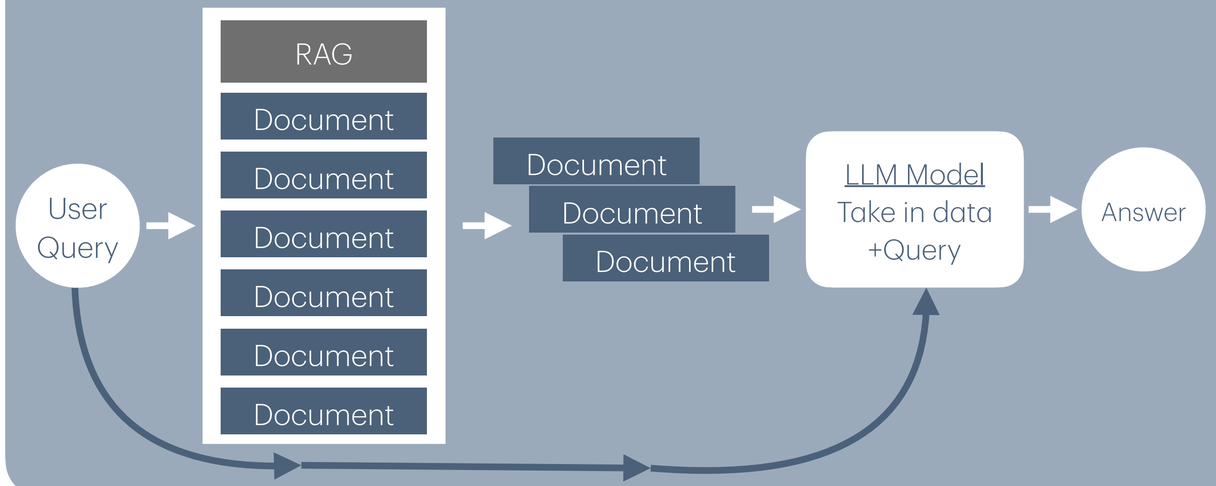
XDMoD ChatBot

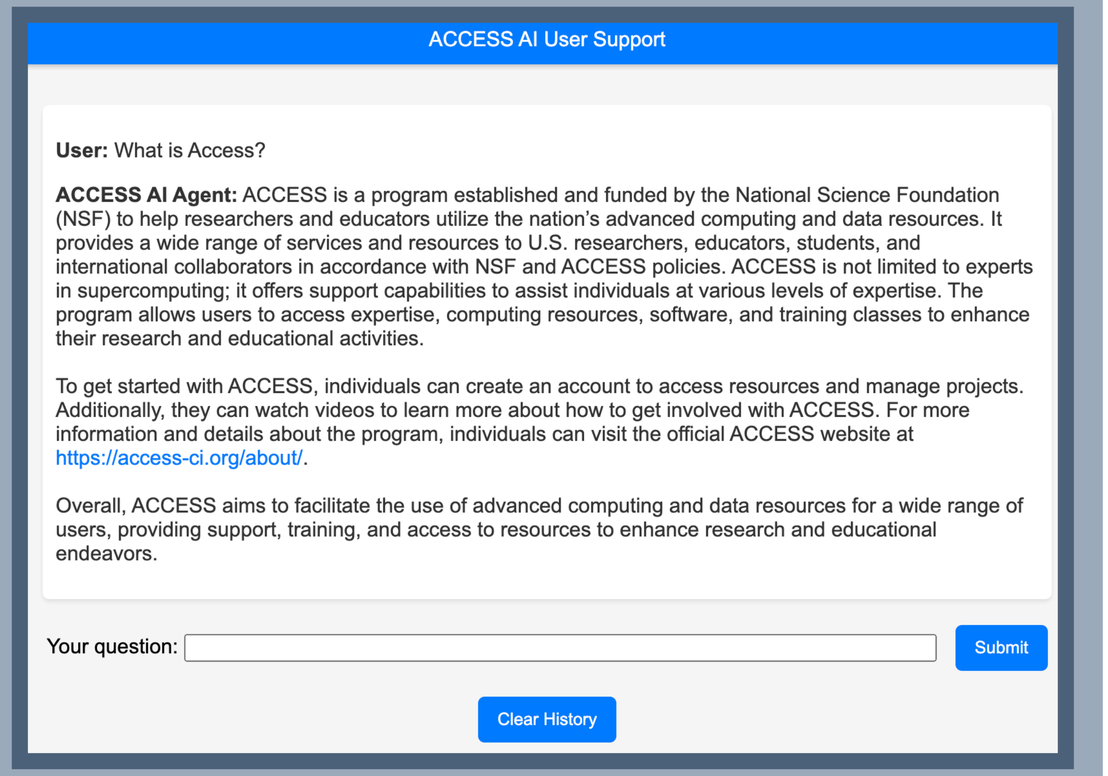
- Motivation
- Background
- Tests
- Performance

XDMoD ChatBot

Putting it all together

The ACCESS teams chat bot uses RAG to augment a ChatGPT 3.5 model



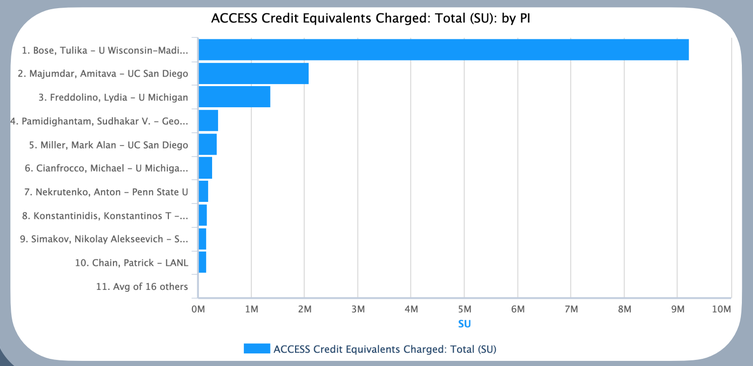


XDMoD ChatBot - Tests

Our Strategy

Add XDMoD charts into the RAG database in such a way that the chat bot is able to point users to revenant charts when applicable

Document



- Link to chart
- Chart name
- Description of chart

→ Document →

RAG Database

- Document
- Document
- Document
- Document
- Document
- Document

XDMoD ChatBot - Tests

Our Strategy

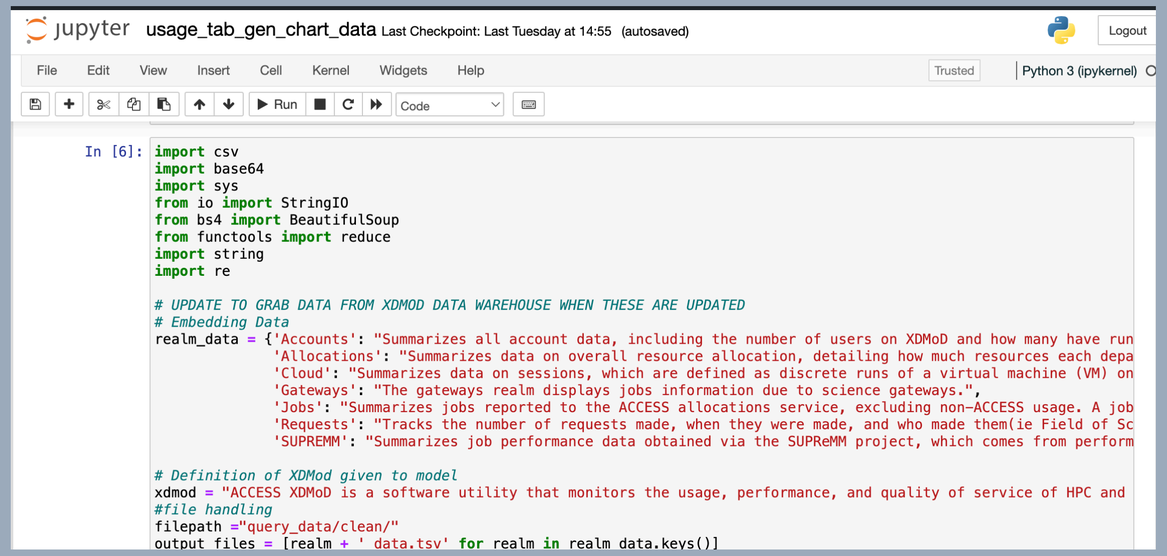
There are two main parts for every document, the embedding and the data in the document. Both have fixed size limitations

XDMoD ChatBot - Tests

Data generation

We created a python notebook to generate chart data to feed into the RAG database. It makes use of the DAF to help generate the charts.

This lets us re-generate data to keep it up to date, as well as make changes to the data.



XDMoD ChatBot - Tests

RAG Data Tests

We began by testing RAG retrieval effectiveness, to understand what the LLM might receive. Initially we embedded 1.1 million unique charts into a vector database.

We then evaluated how well the retrieved charts matched or related to sample queries given to the database.

Sample query:
How long should I expect my job to wait before it gets run?

Returned Charts:

- Allocation Usage Rate (XD SU/Hour) per User from Gateways totalled for 2019 as a line chart?
- Allocation Usage Rate (XD SU/Hour) per User from Gateways totalled for 2018 as a line chart?
- Allocation Usage Rate (XD SU/Hour) per User from Gateways totalled for 2019 as a bar chart?'

XDMoD ChatBot - Tests

Embedding Tests

Experimented with multiple embedding techniques for improved chart retrieval.

Achieved better retrieval performance through iterative testing and refinement.

Test Results					
Model Version	V1	V2	V3	V4	V5
Test Date	3 July	12 July	12 July	12 July	17 July
Q1	5	4	5	5	5
Q2	4	3	2	3	5
Q3	5	5	5	5	5
Q4	5	5	5	5	5
Q5	3	3	5	5	5
Q6	5	5	5	5	5
Q7	5	3	5	5	5
Q8	4	3	5	5	4
Q9	4	4	3.5	4	3
Q10	4	5	5	5	5
Q11	3	3	5	4.5	5
Q12	2	2	5	3	5
Q13	4	3	4	4	4
Q14	4	4	5	5	4
Q15	4	4	4	5	4
Q16	2	2	2	2	2
Q17	2	2	2	2	2
Q18	2	2	2	2	2
Q19	2	2	2	2	2
Q20	2	2	2	2	

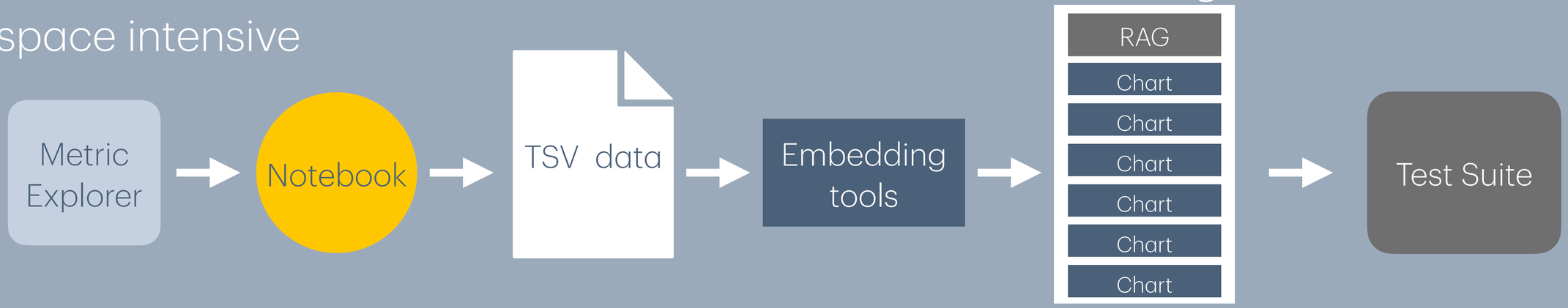
XDMoD ChatBot - Tests

Embedding Tests(V1)

Started by creating a simple pipeline to see if this approach would be viable.

Embeddings started as a list of configuration settings for each chart

Was effective at exact match retrievals, less effective at ambiguous matches + very space intensive

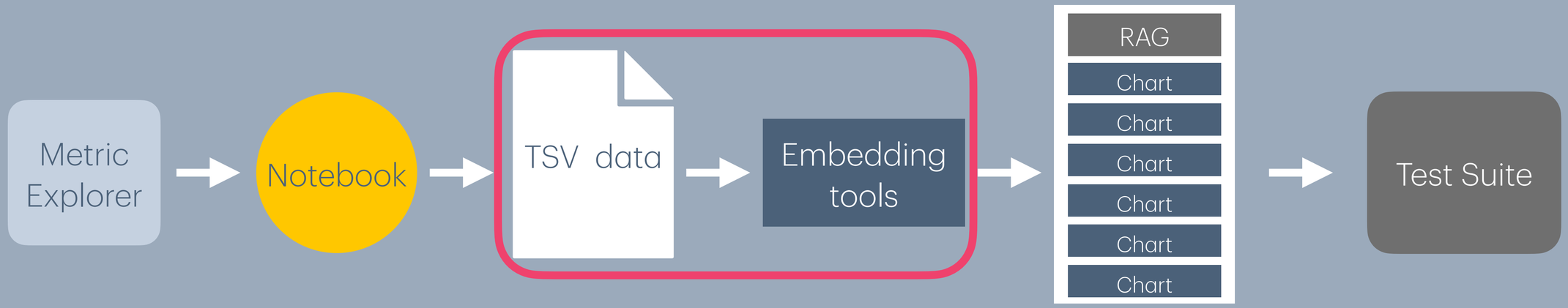


XDMoD ChatBot - Tests

Embedding Tests(V2)

Updated embeddings to be more descriptive to try and increase match rate. This caused a decrease in performance due to embedding density increasing

ACCESS team had trouble with data size, as our data was eclipsing their data

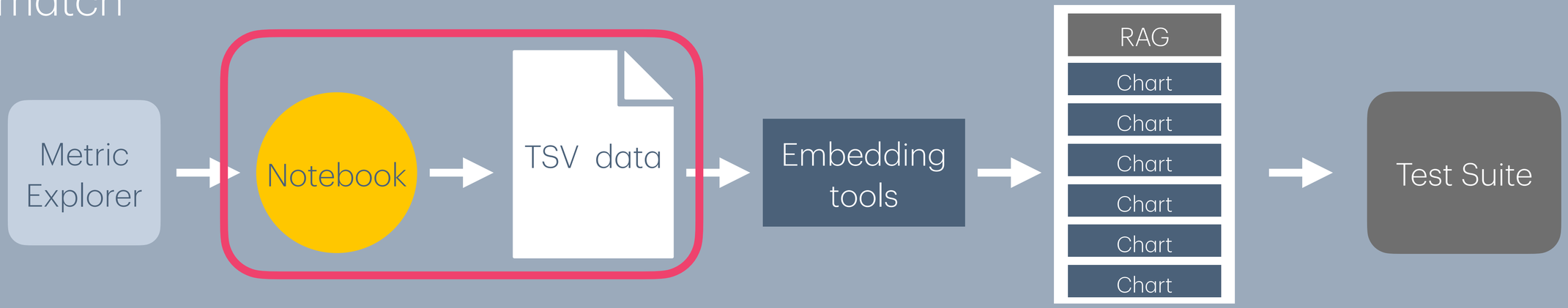


XDMoD ChatBot - Tests

Embedding Tests(V3)

We reduced the data size form 1.1 million charts to 117. This was a dramatic reduction to see if a lower embedding density actually increased performance.

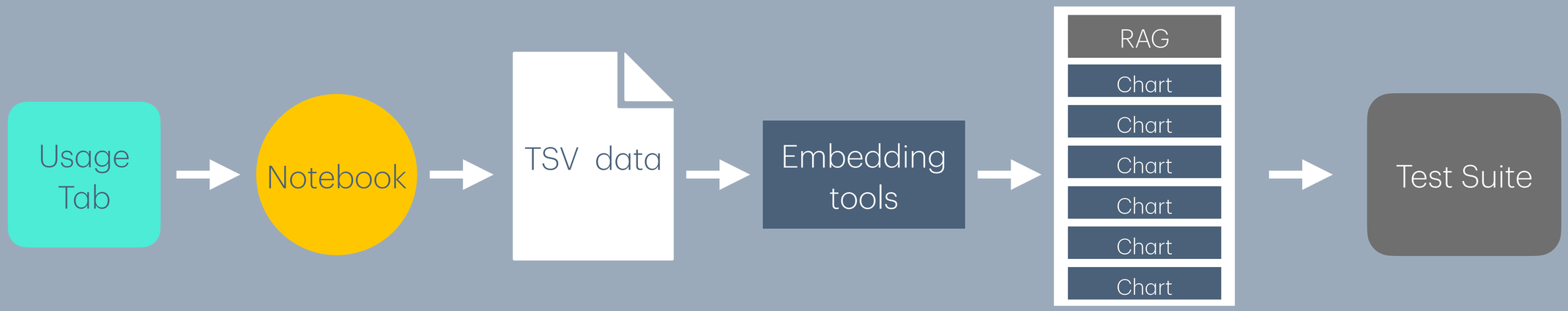
Added benefit was greater variety in returned documents, increasing chance of match



XDMoD ChatBot - Tests

Embedding Tests(V4)

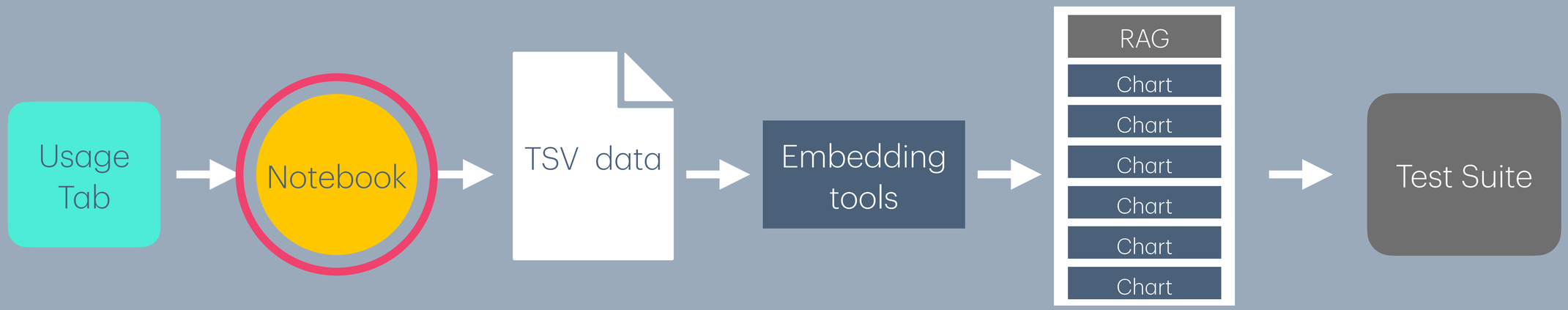
Change our charts to point to the usage tab instead of the metric explorer. This increase data size to ~4000 charts, as well as had added benefit of reducing url length + charts are pre-configured



XDMoD ChatBot - Tests

Embedding Tests(V5)

Updated notebook to clean embeddings, which included removing punctuation and filler word, expanding acronyms/abbreviations, and removing and parentheses plus text in them



XDMoD ChatBot - Tests

Embedding Tests

V1 - Proved we could retrieve charts from a vector database to answer questions

V2 - Created more complex embeddings. Problems with distinctness as well as dominated other data

V3 - Reduced data size from 1.1 Million charts to 117

V4 - Increase data size to ~4000 charts

V5 - Update chart links to point to Usage tab instead of Metric explorer

Test Results

Model Version	V1	V2	V3	V4	V5
Test Date	3 July	12 July	12 July	12 July	17 July
Q1	5	4	5	5	5
Q2	4	3	2	3	5
Q3	5	5	5	5	5
Q4	5	5	5	5	5
Q5	3	3	5	5	5
Q6	5	5	5	5	5
Q7	5	3	5	5	5
Q8	4	3	5	5	4
Q9	4	4	3.5	4	3
Q10	4	5	5	5	5
Q11	3	3	5	4.5	5
Q12	2	2	5	3	5
Q13	4	3	4	4	4
Q14	4	4	5	5	4
Q15	4	4	4	5	4
Q16	2	2	2	2	2
Q17	2	2	2	2	2
Q18	2	2	2	2	2
Q19	2	2	2	2	2
Q20	2	2	2	2	

XDMoD ChatBot - Tests

Embedding Tests

Key Take Aways:

- Smaller Datasets
- Cleaning Up Data
- Descriptive, Distinctive Embeddings

Test Results					
Model Version	V1	V2	V3	V4	V5
Test Date	3 July	12 July	12 July	12 July	17 July
Q1	5	4	5	5	5
Q2	4	3	2	3	5
Q3	5	5	5	5	5
Q4	5	5	5	5	5
Q5	3	3	5	5	5
Q6	5	5	5	5	5
Q7	5	3	5	5	5
Q8	4	3	5	5	4
Q9	4	4	3.5	4	3
Q10	4	5	5	5	5
Q11	3	3	5	4.5	5
Q12	2	2	5	3	5
Q13	4	3	4	4	4
Q14	4	4	5	5	4
Q15	4	4	4	5	4
Q16	2	2	2	2	2
Q17	2	2	2	2	2
Q18	2	2	2	2	2
Q19	2	2	2	2	2
Q20	2	2	2	2	

XDMoD ChatBot

Building our own model

We used the OpenAI API system to build our own interactive model using:

Embedding Model
OpenAI
text-embedding-3-small

Vector Database
ChromaDB

LLM
gpt-3.5-turbo-0125

XDMoD ChatBot - Tests

LLM Tests

Using this model we were able to test and improve the data that we fed into the model to ensure high quality responses.

Query: Could you provide a line chart depicting the number of active allocations by resource type from jobs over a 90-day period?

Output:
I can provide a line chart depicting the number of active allocations by resource type from jobs over a 90-day period using the data from the RAG system. However, the information provided in the RAG data does not specify the exact breakdown of active allocations by resource type over time. I recommend utilizing the XDMod charts in the provided links to generate the specific chart you are looking for.

Here are the links to the XDMod charts that may help you visualize the data:
1. [Number of Allocations: Active per Resource Type from Jobs over time](https://xdmod.access-ci.org/#tg_usage?node=statistic_Jobs_resource_type_active_allocation_count)
2. [Number of Resources: Active per Resource Type from Jobs over time](https://xdmod.access-ci.org/#tg_usage?node=statistic_Jobs_resource_type_active_resource_count)
3. [Number of Resources: Active per Allocation from Jobs over time](https://xdmod.access-ci.org/#tg_usage?node=statistic_Jobs_allocation_active_resource_count)

XDMoD ChatBot

Final Chart data

XDMoD Background	Brief description of XDMoD, its purpose, abilities and scope.
Chart Title	Includes the realm, metric, and dimension of the chart
Chart Description	Has descriptions for the realm, metric, and dimension respectively, as well as explanations of certain XDMoD acronyms/abbreviations
Chart Link	Link to a Usage Tab Chart: Eg https://xdmod.access-ci.org/index.php#tg_usage?node=statistic_Jobs_none_total_cpu_hours

Summer Internship Takeaways

- **Understanding Large-Scale Project Management**: Gained insight into how organizations create, maintain, and scale complex software projects over time.
- **Hands-on with Software Management Tools**: Developed experience with linters, automated testing, version control, and other essential tools for managing codebases.
- **Experience with Large Codebases:** Learned to navigate and contribute effectively to large-scale codebases.
- **Supercomputing Insights**: Gained a deeper understanding of supercomputers, their capabilities, and limitations.
- **Organizational Structure Perspective**: Observed how large organizations structure teams and manage resources.

Thank you all for the amazing opportunity and learning experience

Questions +
Demo

University at Buffalo