Table of Contents

Project Overview

Title: Global AI Colab For Good

Members:
Rama Edlabadkar (ramasandeepedlabadkar@g.harvard.edu),
Shanzeh Batool (shanzehbatool@g.harvard.edu),
Labdhi Gandhi (labdhigandhi@g.harvard.edu),
Hinal Jajal (hjajal@g.harvard.edu)

Project: The scope of this project is to build a global platform that links AI research groups with organizations aiming to solve social issues using AI. The platform will have a search interface for organizations to look for AI research papers relevant to their social cause. A dashboard will provide a curated list of relevant research to the user prompt, the research groups, and how the research work relates to the user's problem prompt. The platform will be designed to support a growing number of research groups and global organizations. We process a large corpus of AI

research papers & social issue descriptions and train LLMs for information retrieval and matching between research and real-world problems.

Milestone 2

Dataset

We utilized textual data fetched from social impact-related papers obtained using the ArXiv API, then embedded it using the Hugging Face library and stored it in the vector database. Currently, we fetch and store the manuscripts (both raw and embedded) for around 30 papers.

Virtual Environment Setup Process

We use Docker containers to separate tasks like data scraping, preprocessing, embedding generation, and user prompt processing via the RAG model. This setup allows for easy management and scalability of different services. Each container runs specific processes, which can be updated, modified, or scaled independently based on requirements. We have set up the virtual environment using Docker to support containerized components. This will ensure that all project elements, from data scraping to large language model (LLM) processing, remain isolated and easy to integrate.

Screenshots of the running instances and logs

Container for data scraping/fetching (retrieve_papers)

```
Locking [dev-packages] dependencies...
Updated Pipfile.lock (073193008382ee006ce494452dfcf769de6e5f7849cdbb28bcbf0e1b27a69309)!
labdhigandhi@dhcp-10-250-43-94 retrieve_papers % docker build --no-cache -t retrieve_papers .
[+] Building 38.6s (12/12) FINISHED                                                                docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                              0.0s
 => => transferring dockerfile: 878B                                                                              0.0s
 => [internal] load metadata for docker.io/library/python:3.12-slim-bookworm                                      0.4s
 => [auth] library/python:pull token for registry-1.docker.io                                                     0.0s
 => [internal] load .dockerignore                                                                                 0.0s
 => => transferring context: 2B                                                                                   0.0s
 => CACHED [1/6] FROM docker.io/library/python:3.12-slim-bookworm@sha256:032c52613401895aa3d418a4c563d2d05f993bc3ecc065c8f4e2280978acd249   0.0s
 => [internal] load build context                                                                                 0.0s
 => => transferring context: 30.38kB                                                                              0.0s
 => [2/6] RUN set -ex;     apt-get update &&     apt-get upgrade -y &&     apt-get install -y --no-install-recommends build-essential git ffmpeg &&     pip install --no-cache-dir --upgrade pip &&   34.8s
 => [3/6] WORKDIR /app                                                                                            0.0s
 => [4/6] ADD Pipfile Pipfile.lock /app/                                                                          0.0s
 => [5/6] RUN pipenv sync                                                                                         2.2s
 => [6/6] ADD . /app                                                                                              0.0s
 => exporting to image                                                                                            1.1s
 => => exporting layers                                                                                           1.1s
 => => writing image sha256:bca678aa590daf9e965960b9ec65a4fbf9eacaedc1ead2b553105e7655a66aa1                      0.0s
 => => naming to docker.io/library/retrieve_papers                                                                0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/riyek3xoiuhxyujchcc6ne4np

 1 warning found (use docker --debug to expand):
 - SecretsUsedInArgOrEnv: Do not use ARG or ENV instructions for sensitive data (ENV "GOOGLE_APPLICATION_CREDENTIALS") (line 8)

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview
labdhigandhi@dhcp-10-250-43-94 retrieve_papers % docker run --rm -ti -v "$(pwd)":/app embed_papers
/bin/bash: ./docker-entrypoint.sh: No such file or directory
labdhigandhi@dhcp-10-250-43-94 retrieve_papers % docker run --rm -ti -v "$(pwd)":/app retrieve_papers
Launching subshell in virtual environment...
root@e419b27dca38:/app#  source /root/.local/share/virtualenvs/app-4PlAip0Q/bin/activate
(app) root@e419b27dca38:/app# python retrieve_papers.py
File arxiv_social_impact_papers.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/astro-ph/9704140v2...
Downloaded .tar file to downloads/astro-ph_9704140v2/astro-ph_9704140v2.tar
/app/retrieve_papers.py:77: DeprecationWarning: Python 3.14 will, by default, filter extracted tar archives and reject files or modify their metadata. Use the filter argument to control this behavior.
  tar.extractall(path=extract_path)
Extracted contents to downloads/astro-ph_9704140v2/extracted
Found .tex file: downloads/astro-ph_9704140v2/extracted/G347_5_0_5_accept_1.tex
Saved .tex file as manuscript_texts_to_retrieve/astro-ph_9704140v2.txt
File manuscript_texts_to_retrieve/astro-ph_9704140v2.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/cs/9903013v1...
Downloaded .tar file to downloads/cs_9903013v1/cs_9903013v1.tar
Extracted contents to downloads/cs_9903013v1/extracted
Found .tex file: downloads/cs_9903013v1/extracted/text.tex
Saved .tex file as manuscript_texts_to_retrieve/cs_9903013v1.txt
File manuscript_texts_to_retrieve/cs_9903013v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/quant-ph/0003132v1...
Downloaded .tar file to downloads/quant-ph_0003132v1/quant-ph_0003132v1.tar
Extracted contents to downloads/quant-ph_0003132v1/extracted
```

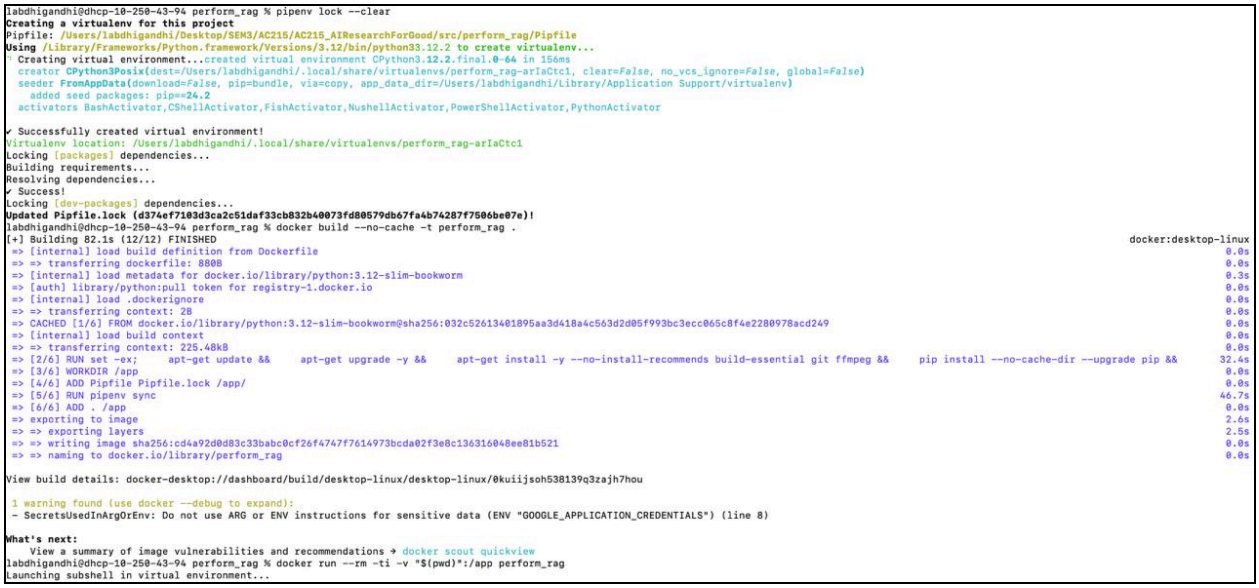
```
Saved .tex file as manuscript_texts_to_retrieve/cond-mat_0503607v2.txt
File manuscript_texts_to_retrieve/cond-mat_0503607v2.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/cs/0502056v2...
Downloaded .tar file to downloads/cs_0502056v2/cs_0502056v2.tar
Extracted contents to downloads/cs_0502056v2/extracted
Found .tex file: downloads/cs_0502056v2/extracted/ipm-coauthor.tex
Saved .tex file as manuscript_texts_to_retrieve/cs_0502056v2.txt
File manuscript_texts_to_retrieve/cs_0502056v2.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/physics/0508029v1...
Downloaded .tar file to downloads/physics_0508029v1/physics_0508029v1.tar
Extracted contents to downloads/physics_0508029v1/extracted
Found .tex file: downloads/physics_0508029v1/extracted/contrarians_v6.tex
Saved .tex file as manuscript_texts_to_retrieve/physics_0508029v1.txt
File manuscript_texts_to_retrieve/physics_0508029v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/cs/0409019v2...
Failed to download the file from http://arxiv.org/src/cs/0409019v2. HTTP Status Code: 403
Skipping http://arxiv.org/src/cs/0409019v2 due to download error.
Skipping upload for http://arxiv.org/src/cs/0409019v2 due to processing errors.
Processing http://arxiv.org/src/cs/0511011v1...
Downloaded .tar file to downloads/cs_0511011v1/cs_0511011v1.tar
Extracted contents to downloads/cs_0511011v1/extracted
Found .tex file: downloads/cs_0511011v1/extracted/link05impact.tex
Saved .tex file as manuscript_texts_to_retrieve/cs_0511011v1.txt
File manuscript_texts_to_retrieve/cs_0511011v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/cs/0601030v1...
Downloaded .tar file to downloads/cs_0601030v1/cs_0601030v1.tar
Extracted contents to downloads/cs_0601030v1/extracted
Found .tex file: downloads/cs_0601030v1/extracted/bollen06scimet.tex
Saved .tex file as manuscript_texts_to_retrieve/cs_0601030v1.txt
File manuscript_texts_to_retrieve/cs_0601030v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/q-bio/0601032v1...
Downloaded .tar file to downloads/q-bio_0601032v1/q-bio_0601032v1.tar
Extracted contents to downloads/q-bio_0601032v1/extracted
Found .tex file: downloads/q-bio_0601032v1/extracted/yusufeli.tex
Saved .tex file as manuscript_texts_to_retrieve/q-bio_0601032v1.txt
File manuscript_texts_to_retrieve/q-bio_0601032v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/physics/0603248v1...
Downloaded .tar file to downloads/physics_0603248v1/physics_0603248v1.tar
Extracted contents to downloads/physics_0603248v1/extracted
Found .tex file: downloads/physics_0603248v1/extracted/ising_jh.tex
Saved .tex file as manuscript_texts_to_retrieve/physics_0603248v1.txt
File manuscript_texts_to_retrieve/physics_0603248v1.txt uploaded to paper-rec-bucket.
Processing http://arxiv.org/src/cs/0002018v2...
Failed to download the file from http://arxiv.org/src/cs/0002018v2. HTTP Status Code: 403
Skipping http://arxiv.org/src/cs/0002018v2 due to download error.
Skipping upload for http://arxiv.org/src/cs/0002018v2 due to processing errors.
Processing http://arxiv.org/src/physics/0605129v1...
Downloaded .tar file to downloads/physics_0605129v1/physics_0605129v1.tar
downloads/physics_0605129v1/physics_0605129v1.tar is not a valid tar file.
Skipping http://arxiv.org/src/physics/0605129v1 due to extraction error.
Skipping upload for http://arxiv.org/src/physics/0605129v1 due to processing errors.
Processing http://arxiv.org/src/cs/0306128v3...
Downloaded .tar file to downloads/cs_0306128v3/cs_0306128v3.tar
downloads/cs_0306128v3/cs_0306128v3.tar is not a valid tar file.
Skipping http://arxiv.org/src/cs/0306128v3 due to extraction error.
Skipping upload for http://arxiv.org/src/cs/0306128v3 due to processing errors.
Processing http://arxiv.org/src/physics/0608296v1...
Downloaded .tar file to downloads/physics_0608296v1/physics_0608296v1.tar
Extracted contents to downloads/physics_0608296v1/extracted
Fo   .tex file: downloads/physics_0608296v1/extracted/LS.tex
S   tex file as manuscript_texts_to_retrieve/physics_0608296v1.txt
F   nuscript_texts_to_retrieve/physics_0608296v1.txt uploaded to paper-rec-bucket.
(a   root@e419b27dca38:/app#
```

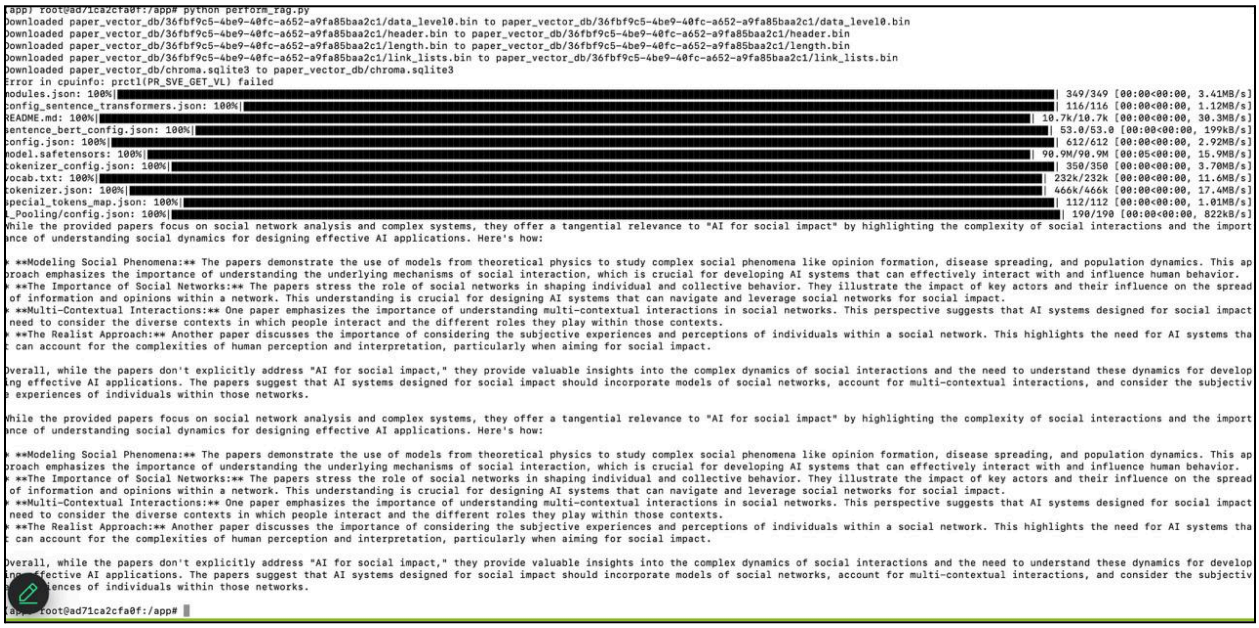
Container for data embedding, storing, and other preprocessing (embed_papers)

```
labdhigandhi@dhcp-10-250-43-94 embed_papers % docker build --no-cache -t embed_papers .
[+] Building 79.0s (11/11) FINISHED                                                                                                      docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                                                     0.0s
 => => transferring dockerfile: 1.60kB                                                                                                                   0.0s
 => [internal] load metadata for docker.io/library/python:3.12-slim-bookworm                                                                             0.2s
 => [internal] load .dockerignore                                                                                                                        0.0s
 => => transferring context: 2B                                                                                                                          0.0s
 => CACHED [1/6] FROM docker.io/library/python:3.12-slim-bookworm@sha256:032c52613401895aa3d418a4c563d2d05f993bc3ecc065c8f4e2280978acd249                0.0s
 => [internal] load build context                                                                                                                        0.0s
 => => transferring context: 262.35kB                                                                                                                    0.0s
 => [2/6] RUN set -ex;     for i in $(seq 1 8); do mkdir -p "/usr/share/man/man${i}"; done &&     apt-get update &&     apt-get upgrade -y &&     apt-get install -y --no-install-recommends build-essen  15.2s
 => [3/6] WORKDIR /app                                                                                                                                   0.0s
 => [4/6] ADD --chown=app:app Pipfile Pipfile.lock /app/                                                                                                 0.0s
 => [5/6] RUN pipenv sync                                                                                                                               60.2s
 => [6/6] ADD --chown=app:app . /app                                                                                                                     0.0s
 => exporting to image                                                                                                                                   3.4s
 => => exporting layers                                                                                                                                  3.3s
 => => writing image sha256:31c7b663d84daf4e524ab22a76a52a5626a0fe5deb3e0aa725cad5ea3abb73da                                                             0.0s
 => => naming to docker.io/library/embed_papers                                                                                                          0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/vxzhpl5shpfm0scf96pxpwr91

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview
labdhigandhi@dhcp-10-250-43-94 embed_papers % docker run --rm -ti -v "$(pwd)":/app embed_papers
Container is running!!!

Launching subshell in virtual environment...
app@9bb0012bb05a:/app$  source /home/app/.local/share/virtualenvs/app-4PlAip0Q/bin/activate
(app) app@9bb0012bb05a:/app$ python embed_papers.py
Error in cpuinfo: prctl(PR_SVE_GET_VL) failed
modules.json: 100%|██████████| 349/349 [00:00<00:00, 4.53MB/s]
config_sentence_transformers.json: 100%|██████████| 116/116 [00:00<00:00, 442kB/s]
README.md: 100%|██████████| 10.7k/10.7k [00:00<00:00, 21.8MB/s]
sentence_bert_config.json: 100%|██████████| 53.0/53.0 [00:00<00:00, 184kB/s]
config.json: 100%|██████████| 612/612 [00:00<00:00, 2.78MB/s]
model.safetensors: 100%|██████████| 90.9M/90.9M [00:05<00:00, 15.6MB/s]
tokenizer_config.json: 100%|██████████| 350/350 [00:00<00:00, 1.85MB/s]
vocab.txt: 100%|██████████| 232k/232k [00:00<00:00, 6.88MB/s]
tokenizer.json: 100%|██████████| 466k/466k [00:00<00:00, 16.1MB/s]
special_tokens_map.json: 100%|██████████| 112/112 [00:00<00:00, 567kB/s]
1_Pooling/config.json: 100%|██████████| 190/190 [00:00<00:00, 561kB/s]
Downloaded manuscript_texts_to_retrieve/astro-ph_9704140v2.txt from paper-rec-bucket to /tmp/astro-ph_9704140v2.txt
Created a chunk of size 1980, which is longer than the specified 1000
Created a chunk of size 1341, which is longer than the specified 1000
Created a chunk of size 1480, which is longer than the specified 1000
Created a chunk of size 2282, which is longer than the specified 1000
Created a chunk of size 1438, which is longer than the specified 1000
Created a chunk of size 2470, which is longer than the specified 1000
Created a chunk of size 2179, which is longer than the specified 1000
Created a chunk of size 1054, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/astro-ph_9704140v2.txt to manuscript_texts_done/astro-ph_9704140v2.txt in the cloud
Deleted manuscript_texts_to_retrieve/astro-ph_9704140v2.txt from Google Cloud Storage
Deleted local temp file /tmp/astro-ph_9704140v2.txt
Downloaded manuscript_texts_to_retrieve/cond-mat_0004026v1.txt from paper-rec-bucket to /tmp/cond-mat_0004026v1.txt
Moved manuscript_texts_to_retrieve/cond-mat_0004026v1.txt to manuscript_texts_done/cond-mat_0004026v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/cond-mat_0004026v1.txt from Google Cloud Storage
Deleted local temp file /tmp/cond-mat_0004026v1.txt
Downloaded manuscript_texts_to_retrieve/cond-mat_0111070v1.txt from paper-rec-bucket to /tmp/cond-mat_0111070v1.txt
Cr    d a chunk of size 1082, which is longer than the specified 1000
        a chunk of size 1528, which is longer than the specified 1000
        a chunk of size 1166, which is longer than the specified 1000
Cr    d a chunk of size 1190, which is longer than the specified 1000
```

```
Created a chunk of size 1705, which is longer than the specified 1000
Created a chunk of size 1018, which is longer than the specified 1000
Created a chunk of size 3409, which is longer than the specified 1000
Created a chunk of size 2955, which is longer than the specified 1000
Created a chunk of size 2476, which is longer than the specified 1000
Created a chunk of size 1235, which is longer than the specified 1000
Created a chunk of size 3222, which is longer than the specified 1000
Created a chunk of size 1179, which is longer than the specified 1000
Created a chunk of size 1085, which is longer than the specified 1000
Created a chunk of size 3287, which is longer than the specified 1000
Created a chunk of size 1912, which is longer than the specified 1000
Created a chunk of size 1134, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/physics_0502047v1.txt to manuscript_texts_done/physics_0502047v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/physics_0502047v1.txt from Google Cloud Storage
Deleted local temp file /tmp/physics_0502047v1.txt
Downloaded manuscript_texts_to_retrieve/physics_0508029v1.txt from paper-rec-bucket to /tmp/physics_0508029v1.txt
Created a chunk of size 1194, which is longer than the specified 1000
Created a chunk of size 1026, which is longer than the specified 1000
Created a chunk of size 1847, which is longer than the specified 1000
Created a chunk of size 1163, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/physics_0508029v1.txt to manuscript_texts_done/physics_0508029v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/physics_0508029v1.txt from Google Cloud Storage
Deleted local temp file /tmp/physics_0508029v1.txt
Downloaded manuscript_texts_to_retrieve/physics_0603248v1.txt from paper-rec-bucket to /tmp/physics_0603248v1.txt
Created a chunk of size 1339, which is longer than the specified 1000
Created a chunk of size 3539, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/physics_0603248v1.txt to manuscript_texts_done/physics_0603248v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/physics_0603248v1.txt from Google Cloud Storage
Deleted local temp file /tmp/physics_0603248v1.txt
Downloaded manuscript_texts_to_retrieve/physics_0608296v1.txt from paper-rec-bucket to /tmp/physics_0608296v1.txt
Created a chunk of size 1195, which is longer than the specified 1000
Created a chunk of size 1026, which is longer than the specified 1000
Created a chunk of size 1211, which is longer than the specified 1000
Created a chunk of size 1098, which is longer than the specified 1000
Created a chunk of size 1550, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/physics_0608296v1.txt to manuscript_texts_done/physics_0608296v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/physics_0608296v1.txt from Google Cloud Storage
Deleted local temp file /tmp/physics_0608296v1.txt
Downloaded manuscript_texts_to_retrieve/q-bio_0601032v1.txt from paper-rec-bucket to /tmp/q-bio_0601032v1.txt
Created a chunk of size 4180, which is longer than the specified 1000
Created a chunk of size 1443, which is longer than the specified 1000
Created a chunk of size 1092, which is longer than the specified 1000
Created a chunk of size 1249, which is longer than the specified 1000
Created a chunk of size 1200, which is longer than the specified 1000
Created a chunk of size 1492, which is longer than the specified 1000
Created a chunk of size 1727, which is longer than the specified 1000
Created a chunk of size 1954, which is longer than the specified 1000
Created a chunk of size 2349, which is longer than the specified 1000
Created a chunk of size 1103, which is longer than the specified 1000
Created a chunk of size 1333, which is longer than the specified 1000
Created a chunk of size 1198, which is longer than the specified 1000
Created a chunk of size 1315, which is longer than the specified 1000
Created a chunk of size 2407, which is longer than the specified 1000
Moved manuscript_texts_to_retrieve/q-bio_0601032v1.txt to manuscript_texts_done/q-bio_0601032v1.txt in the cloud
Deleted manuscript_texts_to_retrieve/q-bio_0601032v1.txt from Google Cloud Storage
Deleted local temp file /tmp/q-bio_0601032v1.txt
Uploaded paper_vector_db/chroma.sqlite3 to paper-rec-bucket
Uploaded paper_vector_db/36fbf9c5-4be9-40fc-a652-a9fa85baa2c1/data_level0.bin to paper-rec-bucket
Uploaded paper_vector_db/36fbf9c5-4be9-40fc-a652-a9fa85baa2c1/length.bin to paper-rec-bucket
Upl  ded paper_vector_db/36fbf9c5-4be9-40fc-a652-a9fa85baa2c1/link_lists.bin to paper-rec-bucket
U     d paper_vector_db/36fbf9c5-4be9-40fc-a652-a9fa85baa2c1/header.bin to paper-rec-bucket
A    uscripts have been embedded and uploaded to the cloud.
(a    app@9bb0012bb05a:/app$
```

Container for user prompt processing through a RAG model (perform_rag)





Containerized Components

Purpose of different containers used (for data scraping, data preprocessing, RAG model) and instructions for running them:

This milestone covers the development and setup of a virtual environment using containers for different stages of the project, including data scraping, data preprocessing, and running the Retrieval-Augmented Generation (RAG) model. Each container is designed to handle specific tasks efficiently, ensuring modularity, scalability, and ease of deployment. Broadly, containers (1) and (2) handle the creation and updating of the vector database of research papers, and container (3) handles the RAG part given a user query.

1. Container for data scraping/fetching:

Purpose:

- This container handles getting academic research papers from ArXiv. It retrieves AI-related research papers based on predefined categories or search queries provided.
- Currently, we only fetch research papers from ArXiv with the query “social impact” for the first baseline iteration.
- [Next steps] In the future, we will consider focusing on papers from specific conferences and tracks and limiting the domain to computer science to better filter the papers.
- [Next steps] We will scrape social issue descriptions from non-profit organizations and research databases.

Key Components:

- Libraries/Frameworks: Requests for API-based data retrieval (ArXiv API). In the future, we might use Python's BeautifulSoup, Selenium, or Scrapy for scraping research group descriptions or non-profit use cases.
- Tasks performed:
    1. Query ArXiv API for papers on “social impact” and fetch metadata for the top 30 results.
    2. For each of the 30 results returned by the API, perform string manipulation to get the link of the `.tex` file for the corresponding folder of the manuscript and download it. Untar the folder, obtain the `.tex` file and convert it to a `.txt` file.
    3. Save all manuscript .txt files to the Google Cloud bucket within `manuscript_texts_to_retrieve`.

Instructions for Running:

- `docker build -t retrieve_papers .`
- `docker run --rm -ti -v "$(pwd)":/app retrieve_papers`

2. Container for data embedding, storing, and other preprocessing:

Purpose:

- This container is responsible for processing the collected data and converting text into embeddings that can be used for matching research papers to social problems.

- It also handles tasks like text cleaning, tokenization, and storing embeddings in a vector database.

Key Components:

- Libraries/Frameworks: all-MiniLM-L6-v2 embedding from sentence-transformers HuggingFace library, ChromaDB for vector storage, pandas, and NumPy for data manipulation.
- Tasks performed:
    1. Load the manuscripts (which are in `.txt` format) from the `manuscript_texts_to_retrieve` directory.
    2. Perform chunking using the character text splitting strategy and 1000 characters.
    3. Embed each chunk using the sentence transformer model `all-MiniLM-L6-v2` from the HuggingFace library. We opt for this model because it is lightweight and still ranks relatively high on well-known benchmarks.
    4. Store the embeddings in a ChromaDB vector database for later retrieval. We initiate the database if it doesn't exist and, if it exists, the code adds to the existing database. We chose to upload the vector database to the cloud for scalability as we will continue to add more papers. Apart from scalability, adding to the cloud also helps avoid redundancy (as we do not have to repeat the embedding process for papers that were processed on another team member's local computer) and makes our data storage future-proof (since GCS automates backups).
    5. For the manuscripts that have been chunked and embedded, we move the manuscripts to `manuscript_texts_done` and delete them from `manuscript_texts_to_retrieve`. This makes sure that when we run the `embed_papers` container again for a new set of retrieved papers, the embedding process is not repeated for the older papers.
    6. Currently, the only metadata we save along with the chunk embedding to the vector database is the paper source. [Next steps] In the future, we will also work on attaching other metadata to the embedding such as the title of the paper, the abstract, the publish date, and authors.

Instructions for Running:

- `docker build -t embed_papers .`
- `docker run --rm -ti -v "$(pwd)":/app embed_papers`

3. Container for user prompt processing through a RAG model:

Purpose:

- This container manages the retrieval of relevant research papers and generates responses for user queries using a Retrieval-Augmented Generation (RAG) model.
- It integrates the stored embeddings and research papers, processes user prompts, retrieves relevant information, and generates human-readable output for users.

Key Components:

- Libraries/Frameworks: LangChain for RAG models, HuggingFace for embedding search, VertexAI for generating a response using the Gemini-1.5-Flash API, and (in the future) Flask/FastAPI for serving user prompts.
- Tasks performed:
    1. Load the existing ChromaDB from GCP onto our local instance. We believe that this is not necessarily the most efficient way to interact with a database on the cloud. However, ChromaDB currently has no support for deploying to GCP [source]. Another alternative we are considering is the VertexAI built-in vector search; however, we are evaluating whether this is the best use of our GCP credits.
    2. Accept user query, embed the query, and perform vector search in the vector database with the research papers created via container (2).
    3. Feed the retrieved research paper chunks with the user query and generate a response for the user explaining how the papers relate to their query.
    4. [Next steps] Provide a dashboard for displaying relevant research to the user prompt, the research groups, and how the research work relates to the user's problem prompt. Currently, we do not have a front-end dashboard but we plan to do this in the coming milestones.

Instructions for Running:

- `docker build -t perform_rag .`
- `docker run --rm -ti -v "$(pwd)":/app perform_rag`

Orchestration Using Docker-Compose

Since these containers need to work sequentially, we created a `docker-compose.sh`, file to orchestrate them. The shell script is included to automate the running of all containers sequentially.

Versioned Data Strategy (Planned, not yet done)

To ensure reproducibility and portability of data pipelines, we will implement a data versioning strategy using DVC (Data Version Control). This will allow us to track the data from different versions of research papers and social problems, ensuring consistency when embedding and storing data and fine-tuning models.

- **Strategy:** We will choose DVC for its seamless integration with Git, enabling tracking of data files alongside model code. This also provides an efficient way to manage large datasets. The versioned data pipeline will be containerized, ensuring easy integration into any environment.
- **Version Control History:** We will track dataset versions, commits, and logs to ensure the exact same datasets can be reproduced at any point.

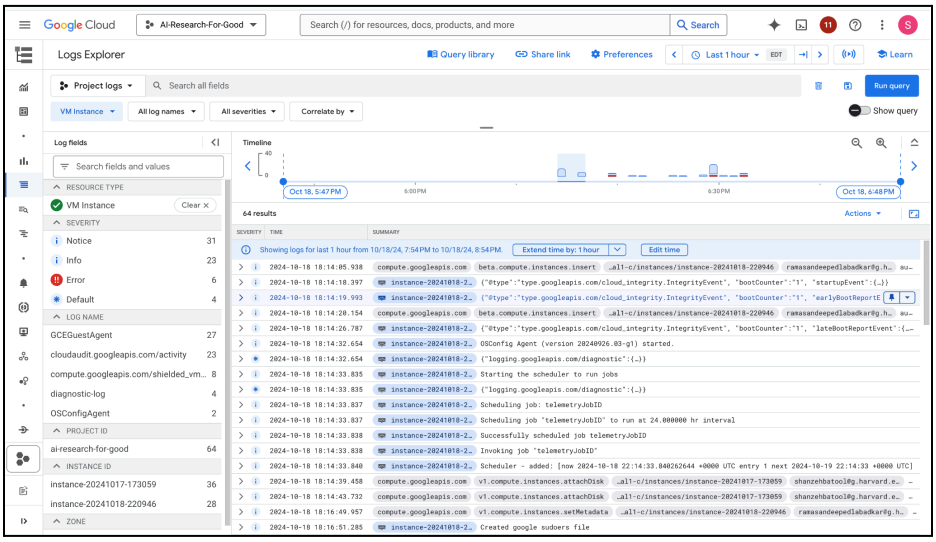
Utilizing LLM (Large Language Model)

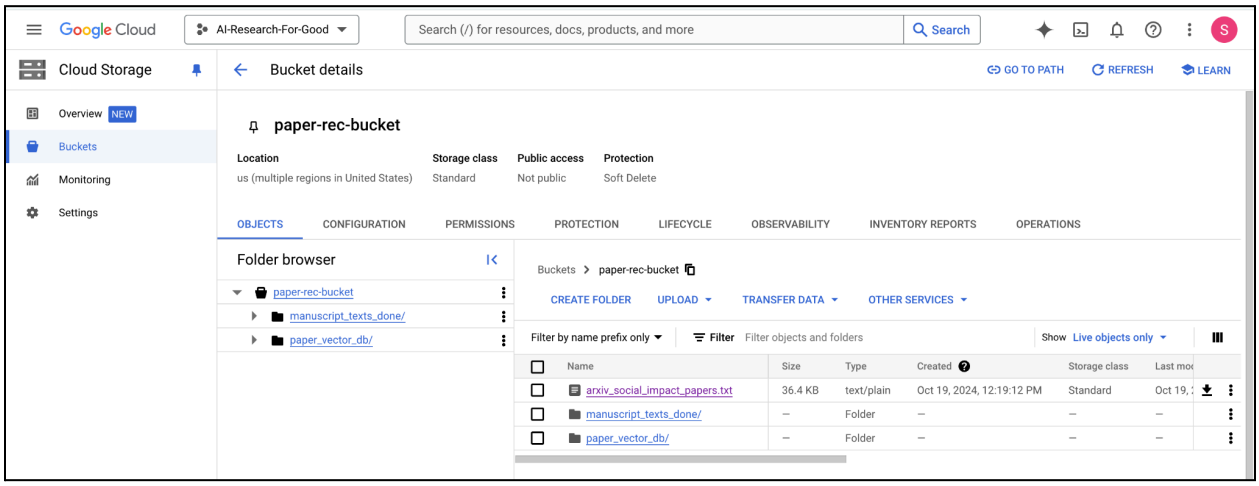
We utilize LLMs in our RAG model to match AI research with societal issues. To do so, we implemented a workflow that chunked the research papers and stored the vectors in a vector database. When a user enters a prompt, we retrieve relevant research papers from the database and generate a natural language response for the user query using an LLM.

1. **RAG Pipeline:**
    - **Retrieval:** Given a user query, we search our vector database of research papers and retrieve the top 5 chunks. Note that at this stage we only retrieve the relevant parts of the papers (and not the whole papers). Also, recall that we are using ChromaDB to store the chunked research papers.
    - **LLM Response Generation:** The user query and retrieved relevant chunks are fed into an LLM, which is prompted to explain how the retrieved papers are related to the user's query/use case.
2. **Fine-tuning:** We outline our plan for fine-tuning. We base our plan based on the existing efforts that address our use case, and how we can add unique value. We notice that there is existing research towards information retrieval from scientific literature (for e.g., embedding models fine-tuned on scientific abstracts). However, work towards matching these papers to relevant non-profit use cases is limited. Thus, we will fine-tune a reranker model that assigns relevance scores to papers based on their helpfulness in addressing the user-query. The steps for fine-tuning this model are below.
    - Annotate a set of 500 examples of (query, paper, relevance) triplets. We will likely perform this annotation by first generating examples of user non-profit queries, then finding relevant papers and ranking them high, and finding "irrelevant" papers and ranking them low. Thus, we also make sure to have hard negatives. Because of this style of annotation, there will likely be multiple samples per query.
    - Some candidate base models we consider for our reranker are `bge-reranker-base` and `CohereRerank`.

- Split the annotated dataset into 70-15-15 train, validation, and test sets. Use validation set for hyperparameter tuning and model selection, and test set for evaluation.
- Depending on how time-consuming data annotation is and how well our initial fine-tuning performs, we will consider generating synthetic data for this task based on our initial seed of annotated examples. Since the base reranker models are already highly performant, we believe that having only a few hundred training samples could be sufficient for teaching the model our task.

3. Experiment details/Logs:





Mock-up of the Application

We created a wireframe of the final platform, integrating the search interface, dashboard, and backend components for matching research papers to social issues.

- **Prototype:** The user interface will allow organizations to input prompts and receive relevant research papers / suggestions. The dashboard shows research groups working on similar problems and provides detailed insights.
- **Wireframe:** The wireframe below illustrates how the UI interacts with back-end components, displaying user prompts, search results, and research matches.