# Visualizing demographics data

## Title

Visualizing demographics data

## Short Description

Utilized D3 and JavaScript to create advanced visualizations of complex data while meeting project requirements and deadlines.

## People

Fateme Rajabiyazdi – Supervisor

Gananatha Subrahmanyam – EDI Collaborator

## Problem Statement

Tracking diversity statistics can help universities to identify areas where they may need to improve in terms of recruiting, admitting, and retaining diverse students. Having proper tools to understand the data can assist Universities as they work to effect policy changes in relation to diversity, equity and inclusion.

Timeline

Description automatically generated The Carleton Original Data Cubes (CODC) contain information about the demographics of students. The diversity statistics include student counts by sex, age, and citizenship status. These may be compared against categories, including degrees, departments, STEM vs non-STEM, academic years, study status and/or the majors of students. Visualizing this data could patterns, trends or anomalies.

The primary goal was to design an interactive visualization on a web-based platform. It allows graphical comparison of student diversity in various academic categories. The secondary goal was to show the lack of demographic diversity data collected by the University.

## Implementation

Diagram

Description automatically generated

In phase one, I created low to medium-fidelity visualizations and collaborated with an EDI research specialist to discuss missing attributes and improve designs. Finally, I cleaned the CODC data using Python.

For the complete Design Rationale, see the GitHub repo readme.

Diagram

Description automatically generated with medium confidence In phase two, I implemented the designs using an interactive web-based platform and the d3 library.

We communicated with the Carleton Institutional Research & Planning Team about potential issues with the dataset, contributing to a complete system overhaul. We presented our project to them and suggested important demographic categories to collect.

We also wrote a paper detailing our project, for the Visualizing for Social Good workshop. Despite not being accepted, I gained valuable feedback for future paper submissions.

## Skills

JavaScript, HTML, CSS, D3, Requirements, Written Communication, Verbal Communication, Data wrangling, Tableau

## Links

Project page - https://kael558.github.io/EDIProjectPage/

Github - https://github.com/kael558/symmetrical-dollop

Visualization - https://kael558.github.io/symmetrical-dollop/

Paper download – drive link

# Research Paper Semantic Search and Clustering

## Title

Research Paper Semantic Search and Clustering

## Short Description

Developed a tool for semantic search and clustering of research papers that is adaptable to various data sets and can handle large volumes, resulting in significant time savings for research

## People

Sacha Gunaratne

## Problem Statement

This project was created for a AI Transformers hackathon hosted by LabLabAI.

Researchers generally find new papers by looking at the related research sections and references. The problem is that this could easily lead to getting stuck in a local cluster of research. Also in the case where a researcher might be trying to solve a difficult problem with novel research it might be hard to find applicable concepts.

Our goal was to help researchers and students find relevant research using a semantic search and clustering approach. This could potentially help with finding research concepts that are used in other fields.

## Implementation

We collected around 500 abstracts from various subjects on Arxiv and used Cohere's transformer to convert the summary of each paper into a n-dimensional vector. We then reduced these vectors to 2D and plotted them. We performed hierarchical clustering on the plotted points and analyzed the frequency of words within each cluster to identify the concepts present. Our interface allows users to interact with different cluster levels, input queries, and access the papers directly.

## Skills

Python, Cohere’s Transformers, Hierarchical Clustering, Streamlit, Annoy

## Links

Repo - <https://github.com/kael558/redesigned-spoon>

Demo - <https://kael558-redesigned-spoon-ui-em33xz.streamlit.app/>

# Stable Diffusion Video Creator Tool

## Title

Stable Diffusion Video Creator Tool

## Short Description

Created an interface that integrated Stable Diffusion models, allowing users to efficiently create videos with interpolation from specific images and prompts.

## People

Farid Hasseina -

## Problem Statement

This project was created for a Stable Diffusion hackathon hosted by LabLabAI.

Stable Diffusion is a technique for creating new images using prompts or initial images. While it is possible to use Stable Diffusion for video creation, existing methods are difficult to use and lack key features:

There is no tool that addresses the issue of limited options in video creation by allowing the selection of specific images for interpolation using Stable Diffusion. Other methods require rendering an entire video even if only certain images are desired, leading to a waste of time and resources. Our tool only generates, and re-renders changed frames, saving time, and making the process more efficient. This can be visually seen in the user interface.

The existing tools are complex to navigate and there is a need for a user-friendly interface for creating videos and timelines.

Finally, the tool needs to be able to connect to various models via API for different use cases. This allows for the flexibility to use different models and adapt to different projects.

Our goal was to create a tool to simplify and optimize the Stable Diffusion process, making it accessible to any user regardless of their experience level. It should be able to generate images based on styles, organize them frame-by-frame, and create videos by interpolating between keyframe images selected by the user.

## Implementation

Diagram

Description automatically generated

First the back-end was developed on Google Colab and we integrated it with Gradio and hosted the model on HuggingFace.

The front-end, created using Flask and D3, queries this API whenever a user requests it.

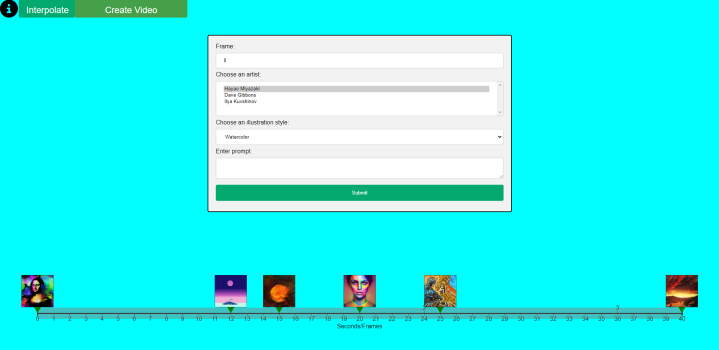
The API has two endpoints, one for generating an image and a conditioning tensor from a prompt, and another for receiving the conditioning tensors of both images to be interpolated and returning a list of interpolated images.

We then used FFMPEG to generate the video from the frames.

Graphical user interface, application

Description automatically generated

The user may select what model to use in the basic options.



The user may input prompts and generate images to be placed on the timeline as key frames. The user may generate the in-between frames by clicking the “Interpolate” button. They may convert the frames into a video by clicking the “Create Video” button which they may download.

If any of the key frames are changed after rendering, then only the changed frames will be re-generated. The status of each frame is represented by the hue on the timeline. Grey represents no frame and green represents a generated frame.

## Skills

Python, Stable Diffusion, API, Hugging Face Spaces, Gradio, Flask, HTML, JavaScript, d3, ffmpeg

## Links

Repo – <https://github.com/kael558/supreme-octo-tribble>

Demo - <https://kael558.github.io/supreme-octo-tribble/>

# Automating Annotations with a Zero-Shot Model

## Title

Automating Annotations with a Zero-Shot Model

## Short Description

Developed a tool to accelerate the data annotation process. Users may download images given a caption, predict annotations based on user input labels and verify them.

## People

## Definitions

* A zero-shot model is a machine learning model that can classify new objects or concepts that it has never seen before, based on prior knowledge of other objects or concepts.
* Bing Image Search provides relevant images based on a caption.
* Label Studio is a data annotation tool for labeling and exploring data.
* ZenML is a MLOps tool to automate the process of building, testing, and deploying models.
* Stack: a stack refers to the set of tools and technologies used for building, deploying, and managing machine learning models.
* Pipeline: a pipeline refers to a set of steps or stages that are executed in a defined order to build, test, and deploy machine learning models.

## Problem Statement

Annotating images can be a difficult and time-consuming task as it requires manual labor to accurately label and classify objects within the image. Many data annotation pipelines incorporate the model with predicted annotations to accelerate the data annotation process. Setting up this process is difficult to do and requires understanding of code and machine learning. However, there are cases where the data annotation process is beneficial as a separate step from model training.

This workflow can be applied if there is a pre-trained "supervised" model that fits your categories but needs fine-tuning for your own use case. This is because zero-shot models may mistake if:

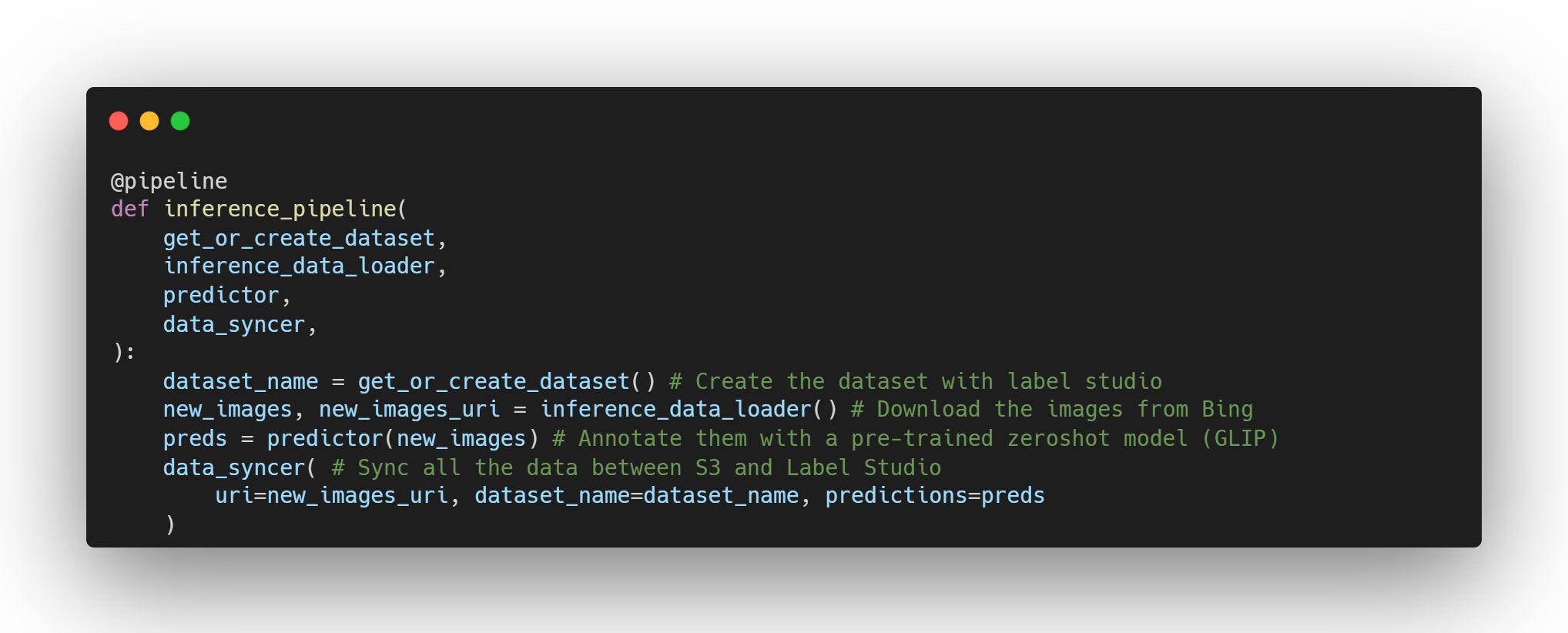
* The quality of image is not good.
* The object is too small.
* The objects are overlapping.

My goal for this project is to develop a tool to partially automate the data annotation process by allowing users to retrieve images, annotate them according to specified labels and verify the annotations.

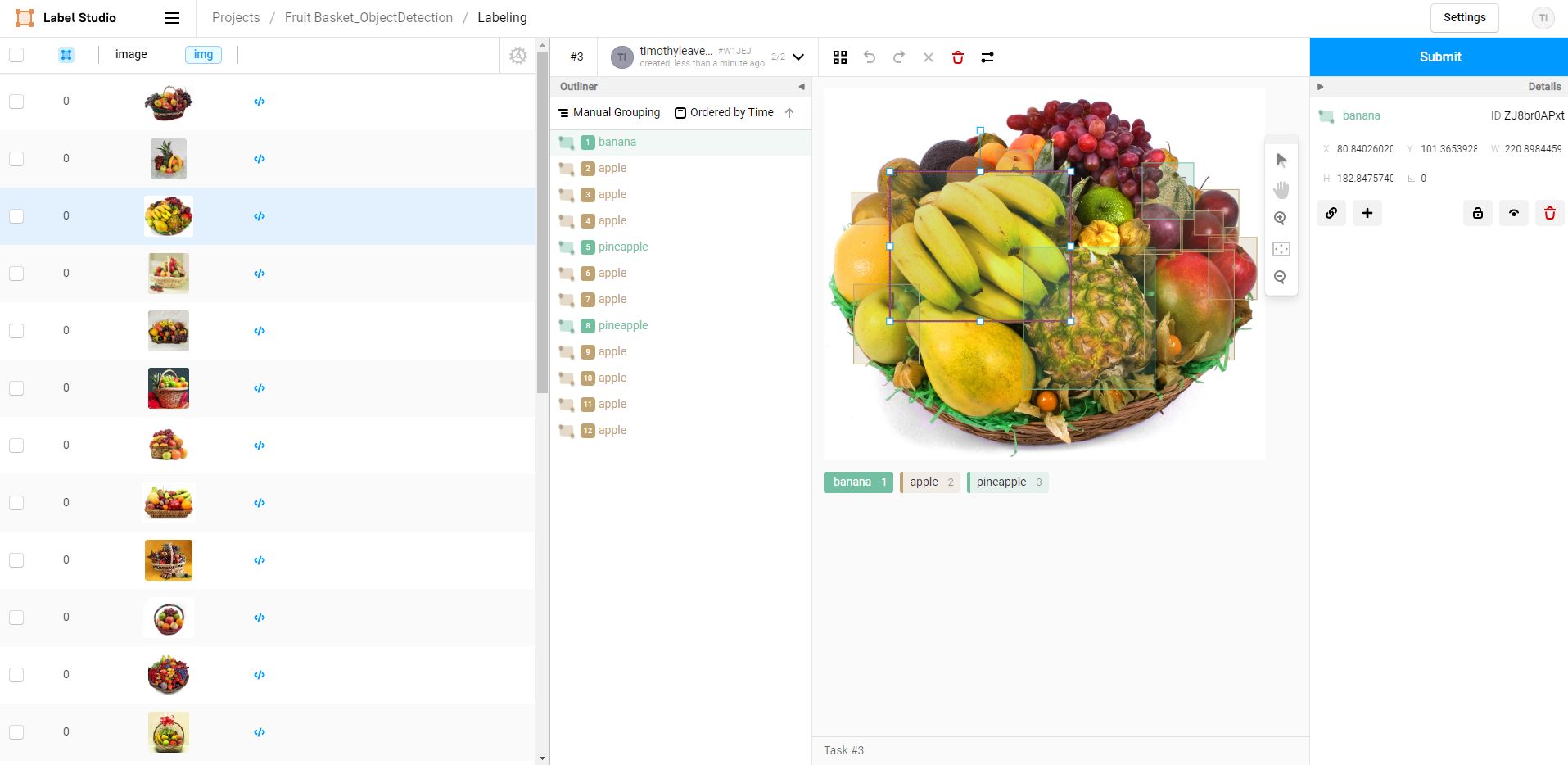
## Implementation



I created the stack with Amazon S3 as storage and Label Studio as the annotator. Access to the integrations is managed by a secrets manager with Amazon. The output from each step of the pipeline will be saved in an S3 bucket.



The pipeline is created to handle the dataset, to load the images, predict the annotations and finally save everything to the cloud. For more details on each step, the notebook is available with documentation.



These are the predicted annotations in Label Studio after running the pipeline with a caption of “fruit baskets” and labels of “pineapple, banana and apple”. Label Studio allows verification of the labels and addition of new labels.

## Skills

Bing Image Search API, ZenML, Zero‑Shot Model (GLIPv2), Label Studio, AWS S3, Google Colab

## Links

Notebook - <https://colab.research.google.com/drive/1bs9vufHWOIy84wIpP_155jUk61S6TvD4#scrollTo=U5a7CJMDxfN1>

# Web Indexer

## Title

Web Indexer

## Short Description

A question answering tool that can be easily integrated into any website, allowing users to find specific information, providing clear answers, and including sources and additional information as needed.

## People

## Definitions

## Problem Statement

This project was made for the AI21 labs hackathon organized by lablab.ai.

There is a prevailing issue with large language models. How can new information be

The search bar found on most websites typically only performs keyword searches, which can be a slow and tedious process for users as they must sift through a large amount of information before finding the specific piece of information they were looking for.

Developed to significantly improve the user experience by providing a service that ChatGPT, Google and standard search bars cannot.

## Implementation

## Skills

## Links

# Web Indexer

## Title

Web Index

## Short Description

## People

## Definitions

## Problem Statement

## Implementation

## Skills

## Links

# Web Indexer

## Title

Web Index

## Short Description

## People

## Definitions

## Problem Statement

## Implementation

## Skills

## Links

# Web Indexer

## Title

Web Index

## Short Description

## People

## Definitions

## Problem Statement

## Implementation

## Skills

## Links

# Web Indexer

## Title

Web Index

## Short Description

## People

## Definitions

## Problem Statement

## Implementation

## Skills

## Links

Persons 65 and older were more likely to have an ischemic stroke 21 days after receiving the Pfizer/BioNTech bivalent shot, compared with days 22-44. Reported by Reuters.

NYT said something similar but is less alarmist. It cites the officials saying it has not been backed by proper investigation.

For LTC, fourth dose VE was maintained even after 168 days after vaccination. 74% effective against severe outcomes.

For LTC, someone who takes the 4th dose after taking the 3rd dose 84 days prior will have the same immunity as someone who did not take the 4th after 168 days. Unsure whether this is 168 days after 3rd or 4th dose.

Effective protection for LTC residents in severe outcomes but not infection.

73 – 84% protection for vaccinated/boosted/unvaccinated? Which one.

90% RRR in hospitalizations in bivalent vs unvaccinated. 13.5-fold increased risk of hospitalization.

2.5-fold higher rate among seniors but without a bivalent booster.

Excess death is mentioned in younger Canadians under 45 years. ~20% more deaths in Jan-March 2022 and 10% more from April to June 2022.

Case Study Insights include:

* Need more data about VE and waning effectiveness in successive boosters.
* There is no data about VE in younger people in this case study. Is excess death linked to vaccines. If so, is it necessary to vaccinate them if the ARR is low.
* Waning effectiveness in infection -> High transmission
* 4th dose VE is same as 3rd dose protection after 168 days in all outcomes-> short term effectiveness means people should take it when they have a period of high risk (travel)
* Prof Eric Topol Scripps Research Institute – who is he, where is funding for Scripps Research Institute coming from.

KPI:

Effectiveness

Cost

Identify the positions of each company:

Why do they have that position?