CVPDL: Computer Vision Practice with Deep Learning

Homework #3 Object Detection + Data Augmentation

Wen-Huang Cheng (鄭文皇)

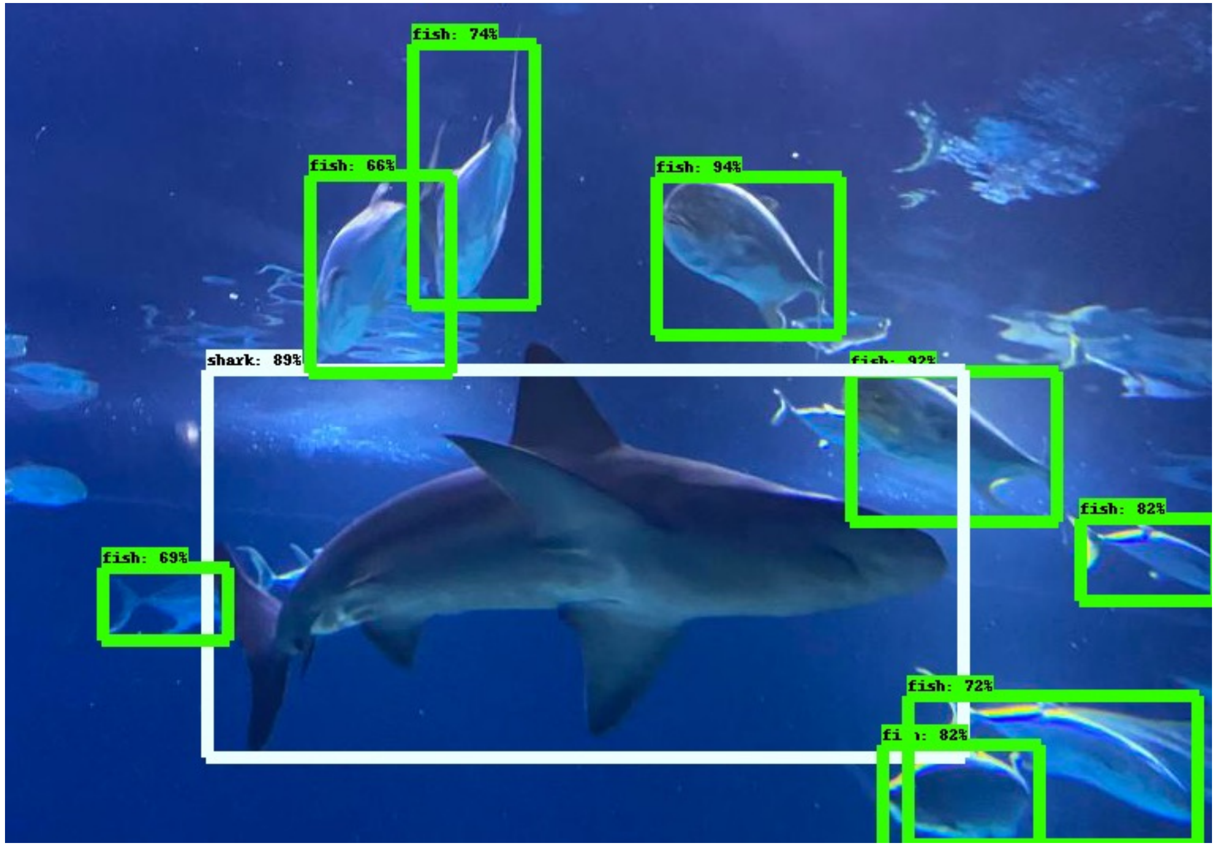
National Taiwan University

wenhuang@csie.ntu.edu.tw

Recap: Object Detection

- Object Detection
  - Input: 2D RGB image
  - Task: localization and classification
  - Output: N x [points, confidence]

- Dataset
  - Training: 448 images
  - Validation: 127 images
  - Testing: 63 images

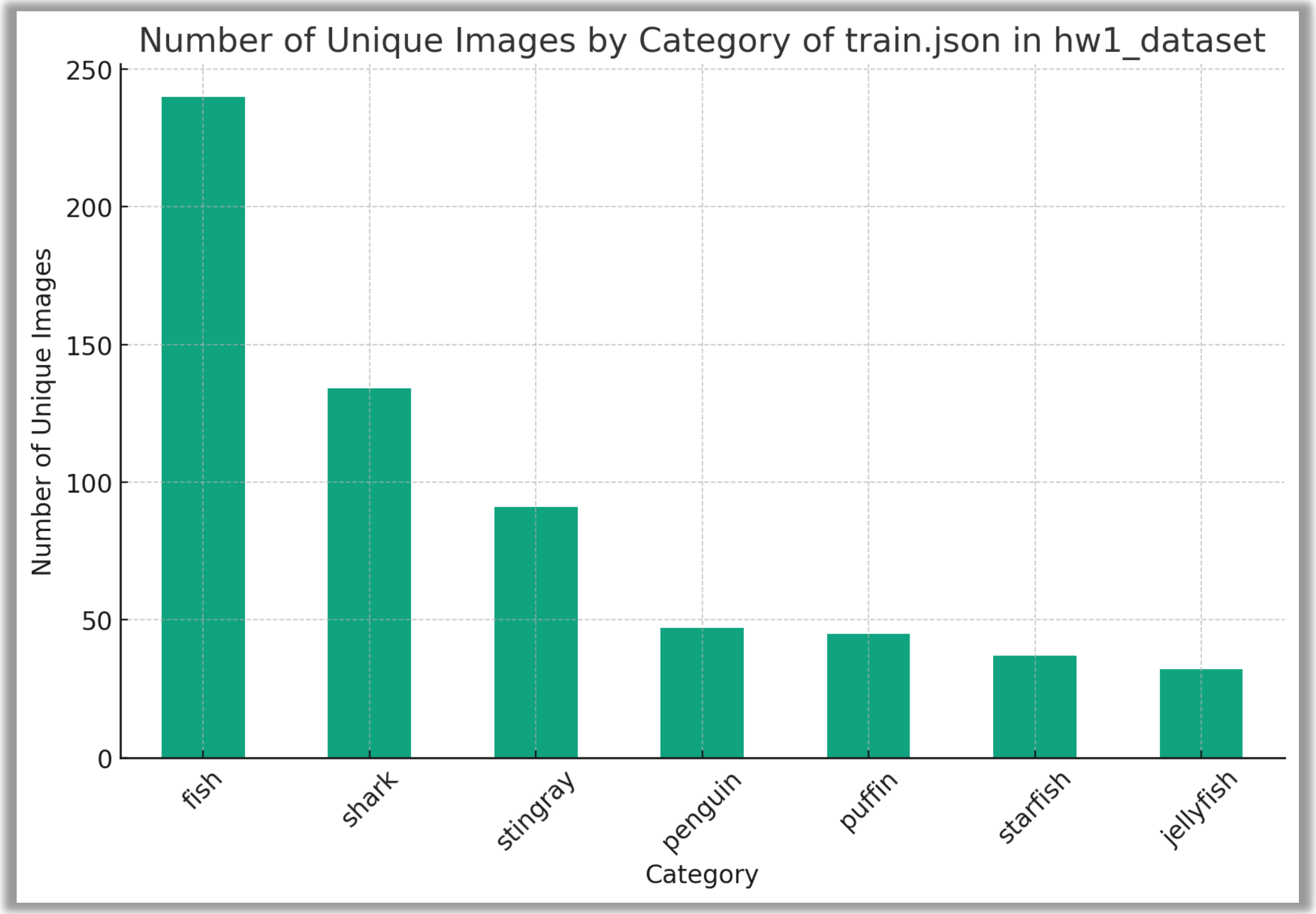


Dataset from HW1

- Data imbalance:

➠ After simple calculation, this graph clearly indicates a data imbalance across categories.

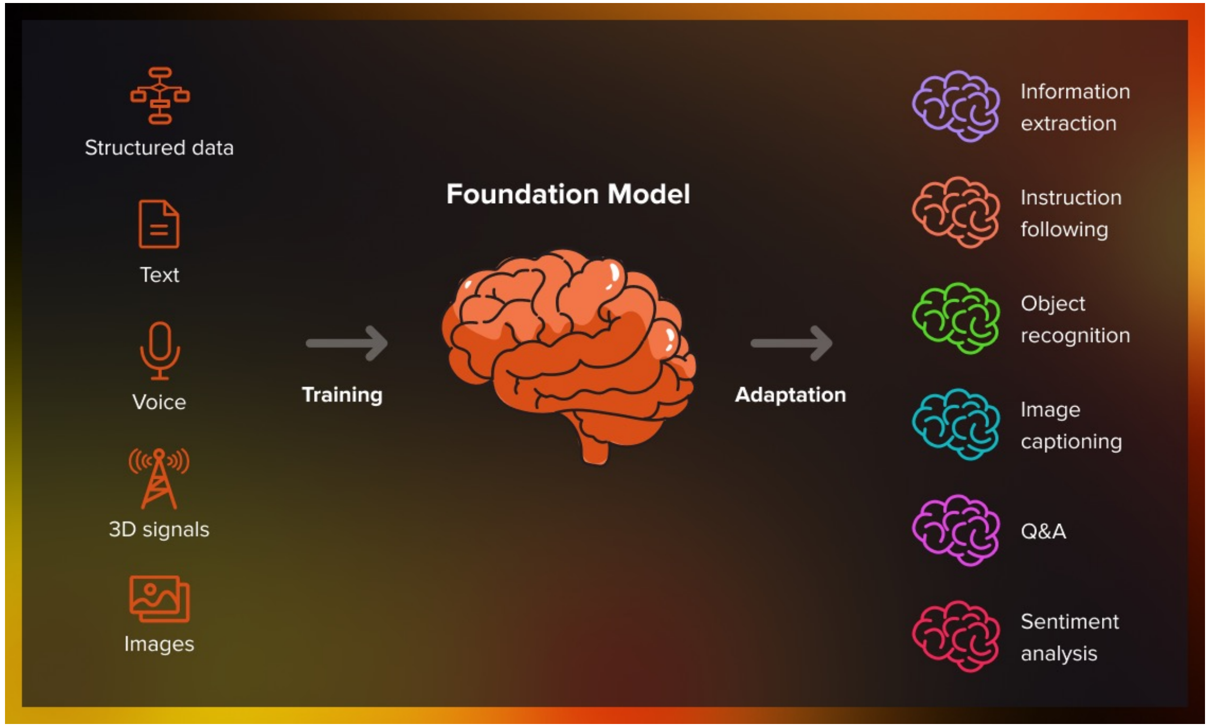
Addressing this imbalance is crucial for developing effective and unbiased models.



Background Information

- What are Foundation Models?
  - Large-scale, pre-trained models having been developed using vast amounts of data can be adapted to accomplish a broad range of tasks.
  - Examples:
    1. BERT (Question Answering, Translation)
    2. GPT (ChatGPT)
    3. Claude (Reasoning, Programming)
    4. Stable Diffusion (T2I Generation)
    5. BLIP2 (Visual Question Answering)
    6. …

https://serokell.io/blog/guide-to-foundation-models

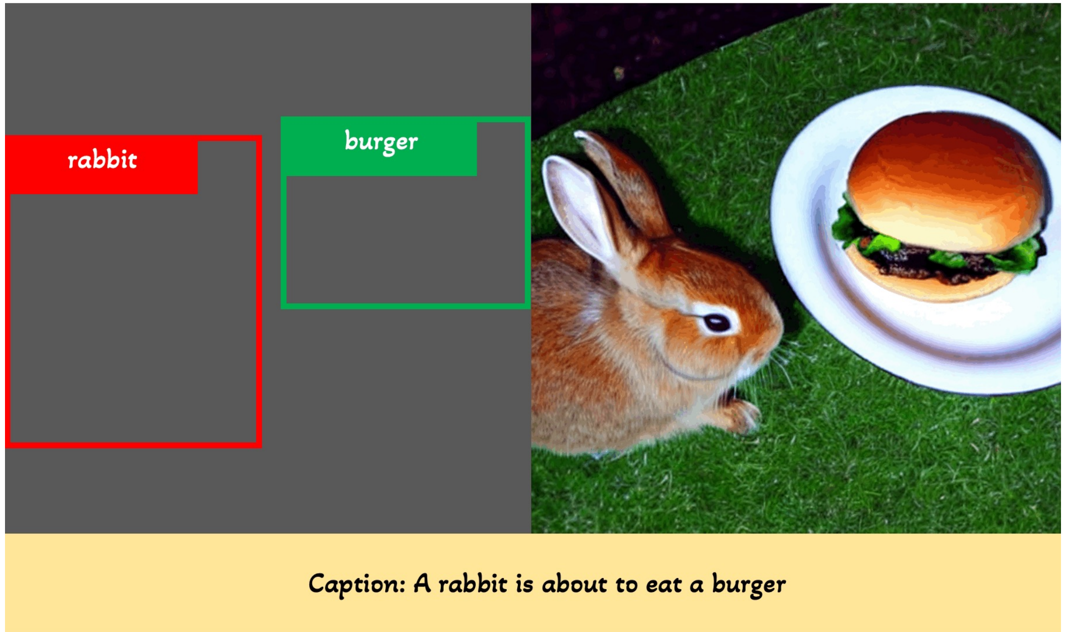


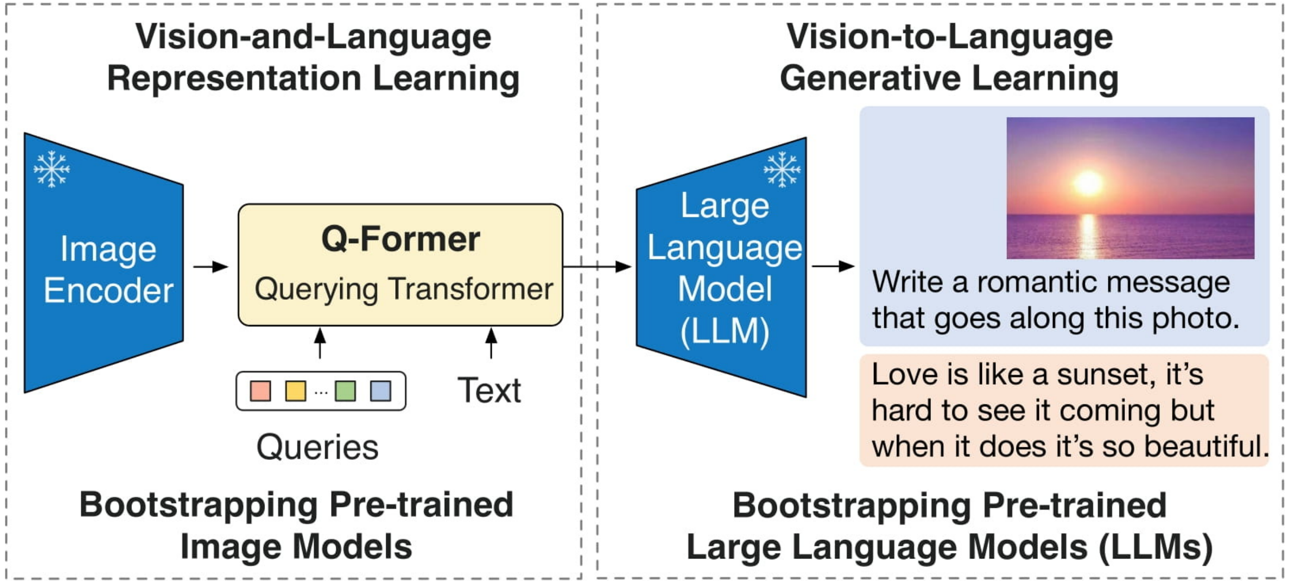
Goals of HW3

- We want to leverage two Foundation Models, BLIP2 and Stable Diffusion (GLIGEN), to solve the imbalance problem of HW1_dataset.
- Considering that Stable-diffusion-based methods require text prompts as inputs for generation, we can first generate prompts from the given dataset by the image captioning ability of BLIP2.
- After obtaining text prompts for later image generation, there is still one problem that needs to be solved. That is, object detection demands bounding boxes for training.
- Thus, we utilize GLIGEN to guide the Stable Diffusion model, so that we can generate objects at the regions defined by the bounding boxes from train.json.

Data Augmentation by Foundation Model

1. Image Captioning
   - ➢ BLIP2
2. Data Augmentation
   - ➢ GLIGEN

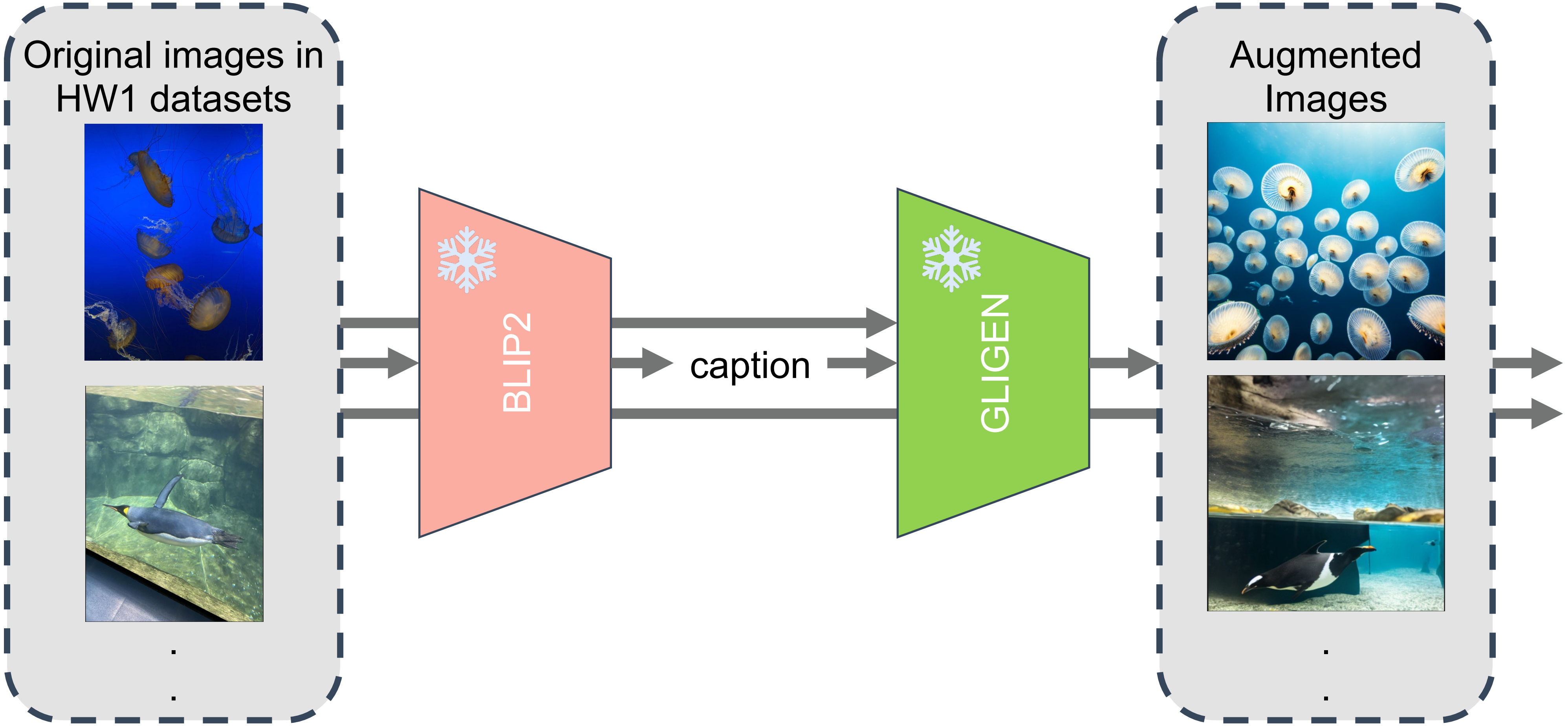




Write a romantic message that goes along this photo.

Love is like a sunset, it's hard to see it coming but when it does it's so beautiful.

Our pipeline

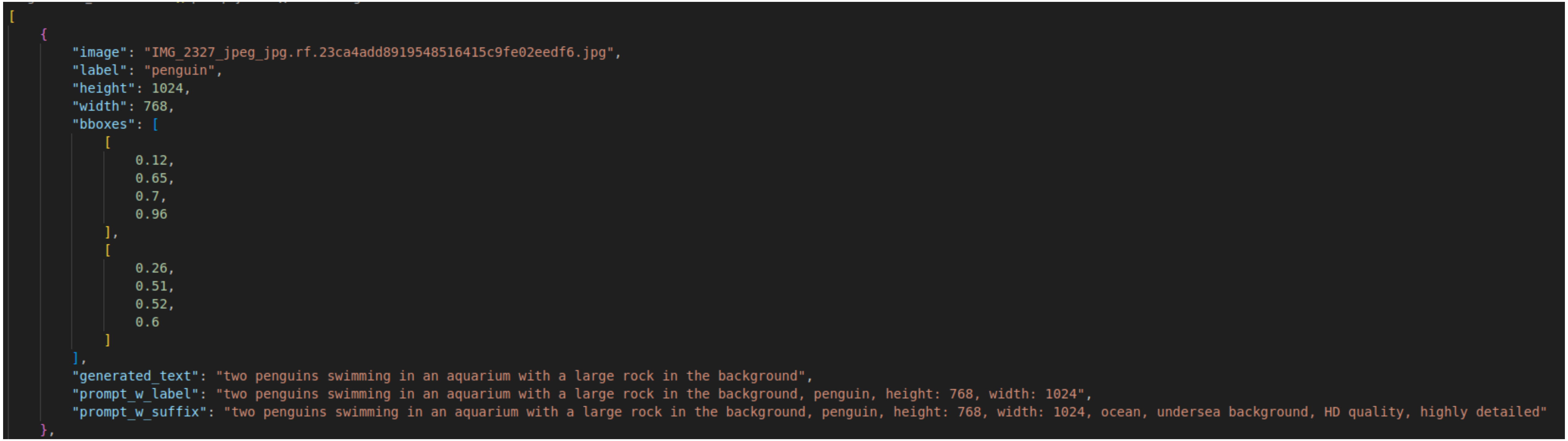


Use the original + augmented data (balanced data) to train a new object detector

Data Augmentation by Foundation Model

- Examples of Image Captioning by BLIP2

Your results after image captioning should be in a similar format for later T2I generation.
(bboxes should be normalized and saved in [x_min, y_min, x_max, y_max] format)



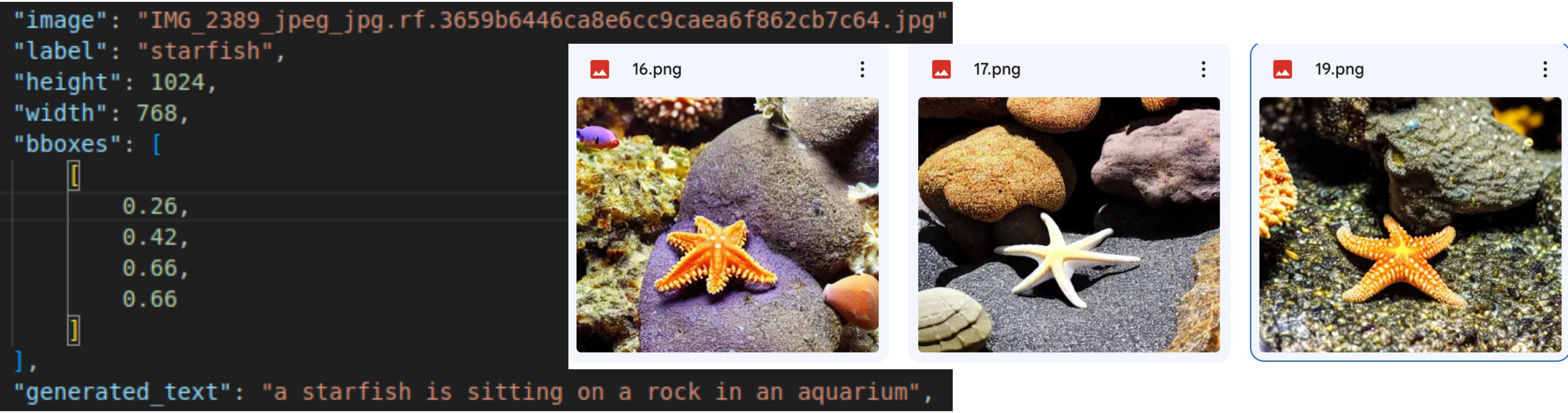
```
[
    {
        "image": "IMG_2327_jpeg_jpg.rf.23ca4add8919548516415c9fe02eedf6.jpg",
        "label": "penguin",
        "height": 1024,
        "width": 768,
        "bboxes": [
            [
                0.12,
                0.65,
                0.7,
                0.96
            ],
            [
                0.26,
                0.51,
                0.52,
                0.6
            ]
        ],
        "generated_text": "two penguins swimming in an aquarium with a large rock in the background",
        "prompt_w_label": "two penguins swimming in an aquarium with a large rock in the background, penguin, height: 768, width: 1024",
        "prompt_w_suffix": "two penguins swimming in an aquarium with a large rock in the background, penguin, height: 768, width: 1024, ocean, undersea background, HD quality, highly detailed"
    },
```

Notes:
The checkpoint used for the above example is `Salesforce/blip2-opt-6.7b-coco`
It's recommended to load half-precision weights by `torch_dtype=torch.float16`

Model Zoo
Sample Code

Data Augmentation by Foundation Model

- Examples of Text-to-Image Generation by GLIGEN



```
"image": "IMG_2389_jpeg_jpg.rf.3659b6446ca8e6cc9caea6f862cb7c64.jpg"
"label": "starfish",
"height": 1024,
"width": 768,
"bboxes": [
    [
        0.26,
        0.42,
        0.66,
        0.66
    ]
],
"generated_text": "a starfish is sitting on a rock in an aquarium",
```

Notes:
For better generation results, it's better to use images containing only one category.
Furthermore, it's recommended to discard images including more than 6 bounding boxes.
Otherwise, you may end up having some errors while generation or generating low-quality images.

Sample Code

Evaluation

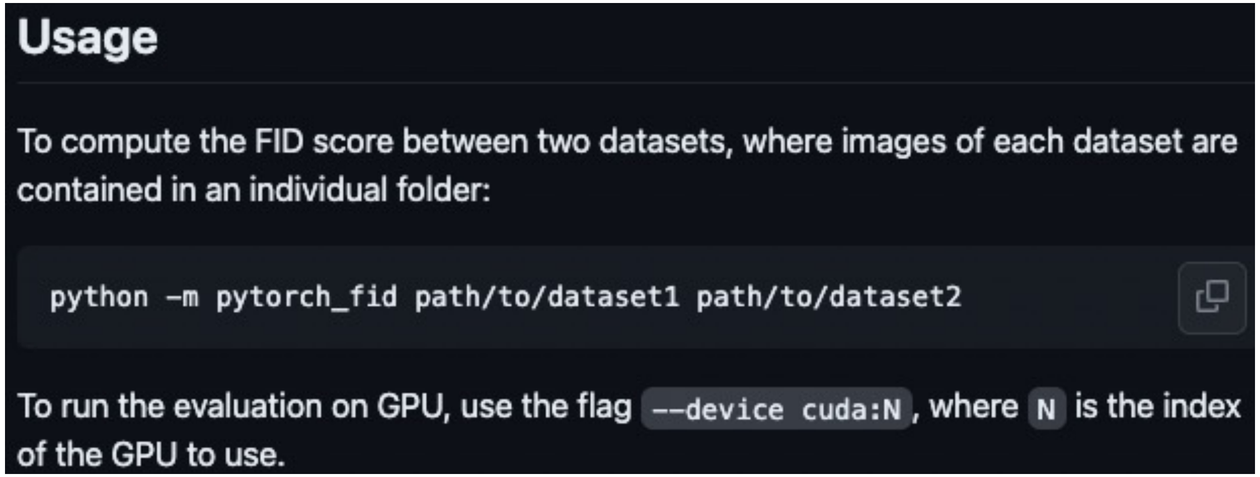
- Evaluation Metric
  - We'll use the metric – Fréchet inception distance (FID)
  - The quantitative evaluation should be evaluated by this

    (Notes:

    Resize the image to 512x512 first when computing FID.

    If you encounter ValueError: Imaginary component,

    just downgrade scipy to 1.11.1.)



Usage

To compute the FID score between two datasets, where images of each dataset are contained in an individual folder:

```
python -m pytorch_fid path/to/dataset1 path/to/dataset2
```

To run the evaluation on GPU, use the flag `--device cuda:N`, where `N` is the index of the GPU to use.

Evaluation

- Evaluation Metric
  - We’ll use the metric – Fréchet inception distance (FID)
  - The quantitative evaluation should be evaluated by this

In the submitted report, please manually select 20 images per category from the training dataset, and generate 20 images per class, that is, 7 x 20 = 140 images in total (140 real images and 140 synthesized images). Then, compute the FID between those two.

Report

1) Image Captioning
   a) Compare the performance of **2 selected** different pre-trained models in generating captions, and use the one you find the most effective for later problems. (Suggestion: choose models wisely based on VRAM size)
      - ✓ Salesforce/blip2-opt-2.7b
      - ✓ Salesforce/blip2-opt-6.7b-coco
      - ✓ Salesforce/blip2-opt-6.7b
      - ✓ Salesforce/blip2-flan-t5-xl

   b) Design 2 templates of prompts for later generating comparison (examples can be referred to p.8)
2) Text-to-Image Generation
   a) Use 2 kinds of generated prompts from Problem 1(b) to generate images. **(text only!)**
   b) Select the prompts for better-generating results, and perform image grounding generation. (**text + image**)

Report

3) Table of your performance based on FID

	Text grounding		**Image grounding**
prompt	Template #1	Template #2	Template #?
FID			

4) Table of the improvement of your detection model from HW1 after data augmentation

	Before Data Augmentation	**After Data Augmentation (Text grounding)**	**After Data Augmentation (image grounding)**
$AP_{[50:5:95]}$			

5) Visualization

- ➢ show the best 5 images for each category **(35 images in total!)**

Grading

- Report (100%)
  - 1 (30%)
    - (a) 10% (5% / model)
    - (b) 20% (10% / template)
  - 2,3 (30%)
    - 10% / column
  - 4 (30%)
    - Text grounding 15%
    - Image grounding 15%
  - 5 (10%)

Submission

- Deadline : 2023/12/24 (Sun.) 23:59
- Zip all files as hw3_<student_id>.zip
- Submit to NTU cool
- Your submission should include the following files
  - hw3_<student_id>.pdf
  - All codes for generation and training
  - Readme file
    - your environments
    - How to run your code

Rules

- Late Policy

  No late submission will be accepted!

- Plagiarism is a serious offense and will not be treated lightly.

Helps

- Mail
  - If you have any questions, contact TAs via this email
    cvpdl.ta.2023fall@gmail.com
  - Please note that emails sent to TA's personal email address will not receive responses.

Any Question

cvpdl.ta.2023fall@gmail.com