Vision for the Blind

Ayush, Kevin, Xiyi, Kay

Project Introduction

Close your eyes

Problem Introduction

Fundamental Aim in Computer Vision Research:

Machine Capable of Replication Human vision

Challenge:

Design technology that assists people who are blind, helping navigate visual challenges

Objective:

Assistive technology for the blind by developing an AI-powered visual query assistant, allowing visually impaired users to take photos, ask questions, and navigate surroundings for effectively. We enhance this experience by also helping them locate the placement of an exact image

Cognitive Problems

1. Semantic Scene Understanding and Contextual Question Answering

- Technology Used: CLIP and Vision-and-Language Transformer (ViLT) Model
- Function: Interprets complex scenes and contextual queries
- Key Features:
  - Provides nuanced understanding of both visual content and language queries
  - Grasps broader context of scenes and interactions between objects

2. Object Detection and Counting within Complex Scenes

- Technology Used: YOLO for Object Segmentation
- Function: Detects and counts objects within user-referenced queries
- Key Features:
  - Identifies and highlights specific items in an image
  - Assists users in locating and understanding objects in their environment

Dataset - VizWiz (2023 Edition)

Files include: *Annotations* (.json) , *train/val/test* (images)

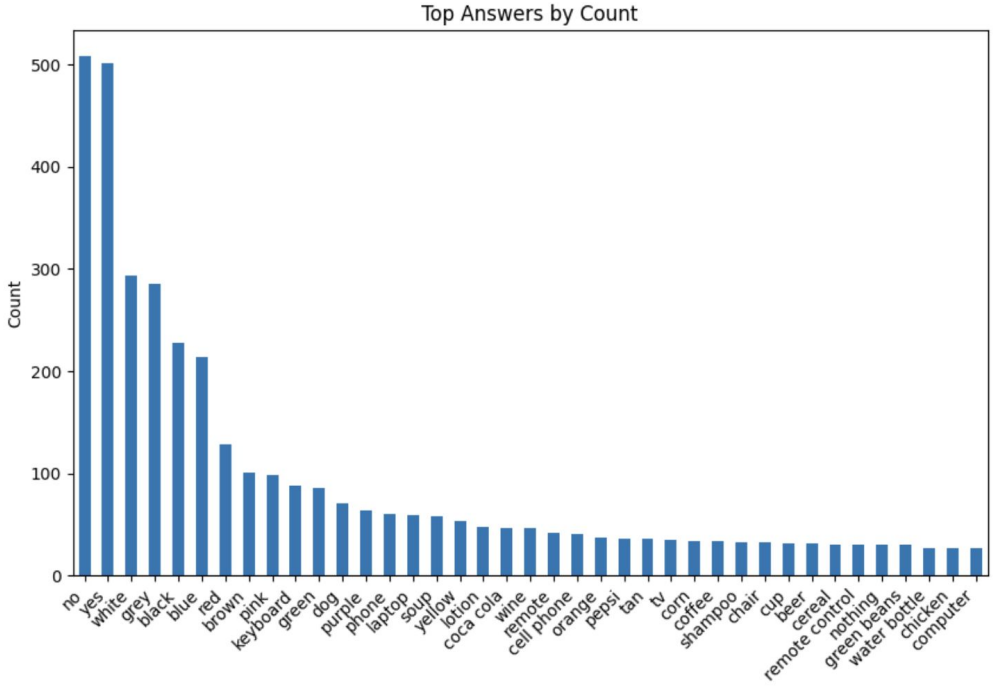
Structure:

- **answerable**: A binary indicator denoting if the question is answerable (1) or not (0).
- **image**: The filename of the image related to the question.
- **question**: The text of the question asked about the image.
- **answer_type**: The type of the answer, indicating whether it is answerable or unanswerable.
- **answers**: A list of answers provided by crowd workers. Each answer includes:
  - **answer**: The text of the answer.
  - **answer_confidence**: The confidence level of the answer (e.g., "yes", "no", "maybe").


```
"answerable": 0,
"image": "VizWiz_val_00028000.jpg",
"question": "What is this?"
"answer_type": "unanswerable",
"answers": [
    {"answer": "unanswerable", "answer_confidence": "yes"},
    {"answer": "chair", "answer_confidence": "yes"},
    {"answer": "unanswerable", "answer_confidence": "yes"},
    {"answer": "unanswerable", "answer_confidence": "no"},
    {"answer": "unanswerable", "answer_confidence": "yes"},
    {"answer": "text", "answer_confidence": "maybe"},
    {"answer": "unanswerable", "answer_confidence": "yes"},
    {"answer": "bottle", "answer_confidence": "yes"},
    {"answer": "unanswerable", "answer_confidence": "yes"},
    {"answer": "unanswerable", "answer_confidence": "yes"}
]
```


Source: https://www.kaggle.com/datasets/nga112/vizwiz-2023-edition/data

EDA: Training Data

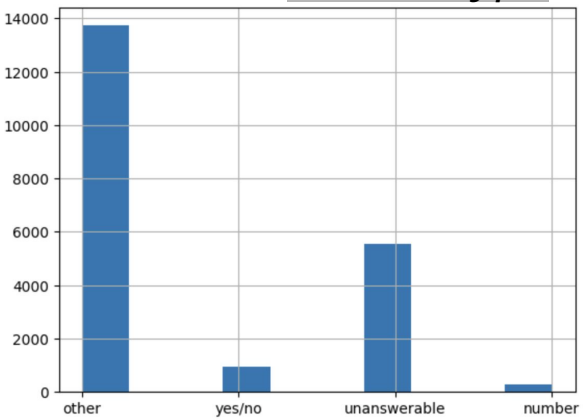


Top 40 Answers by Count

```
train_data.answer_type.hist()
```

```
<Axes: >
```

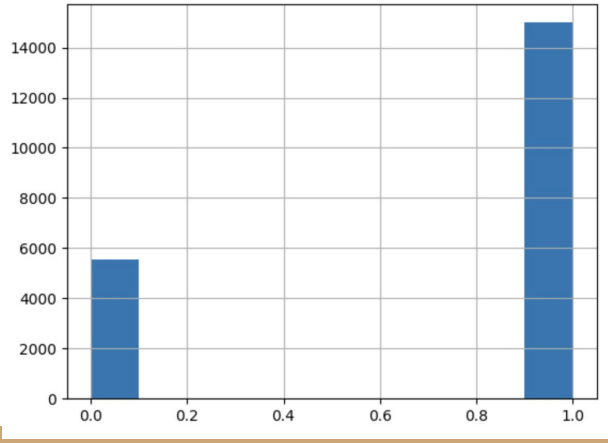
answer_type




```
train_data['answerable'].hist()
```

```
<Axes: >
```

answerable



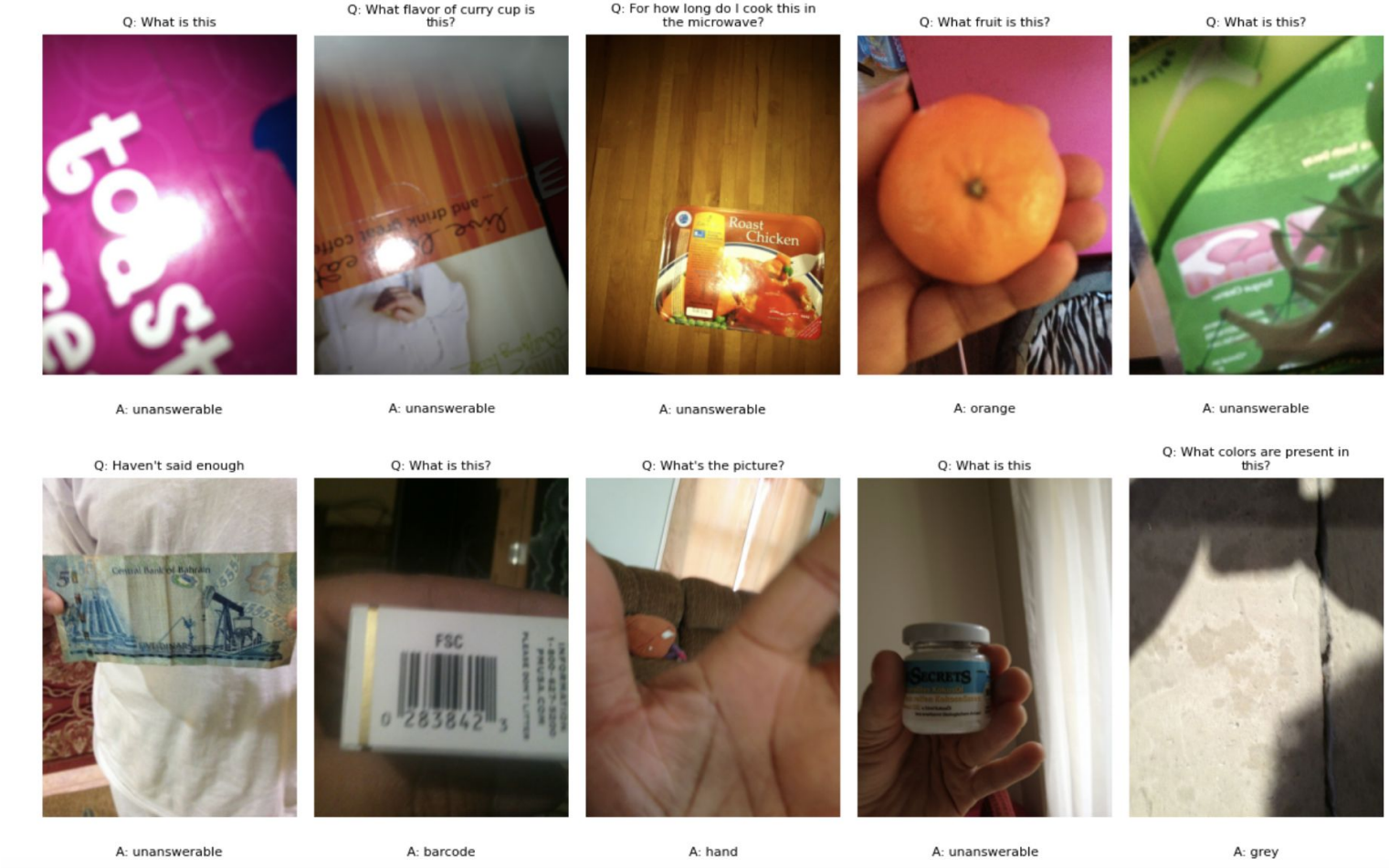
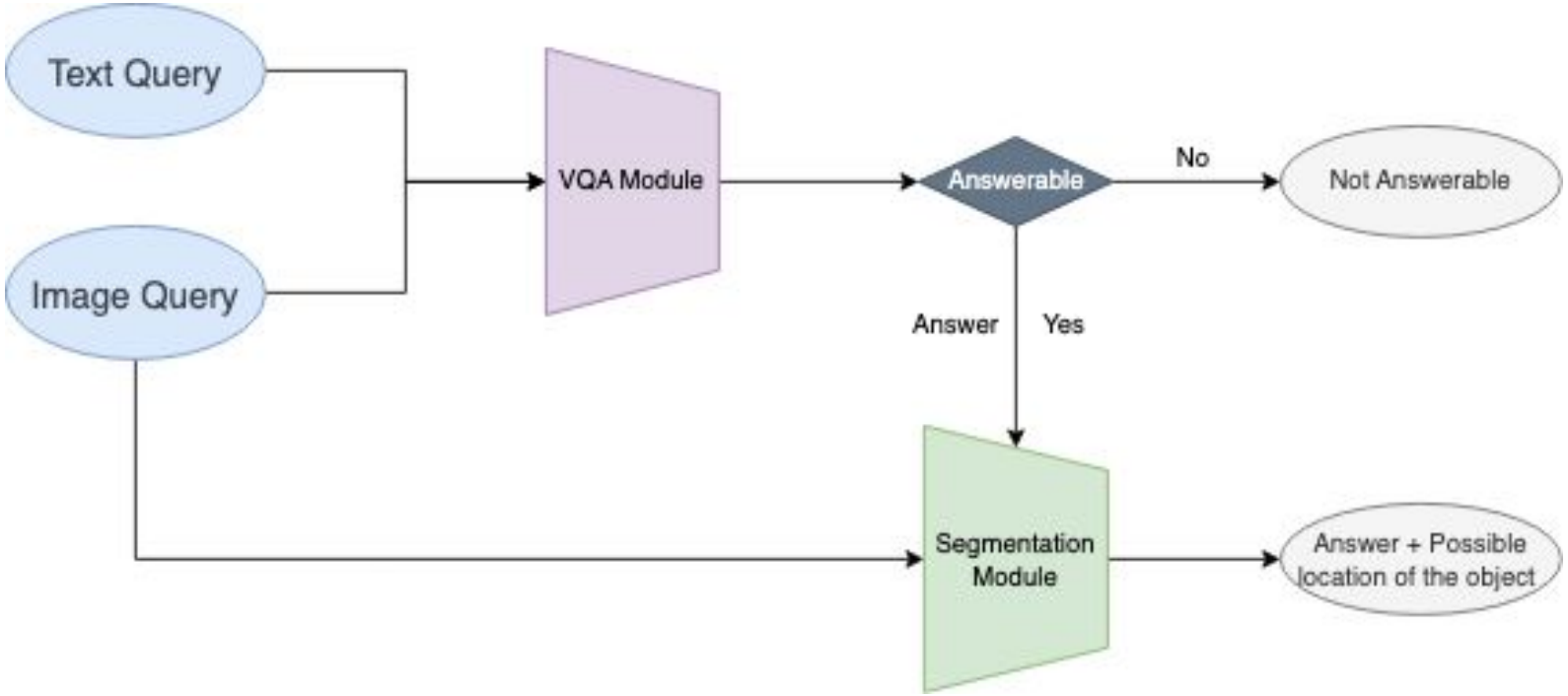


Figure. Sample Example from VizWiz Training Dataset

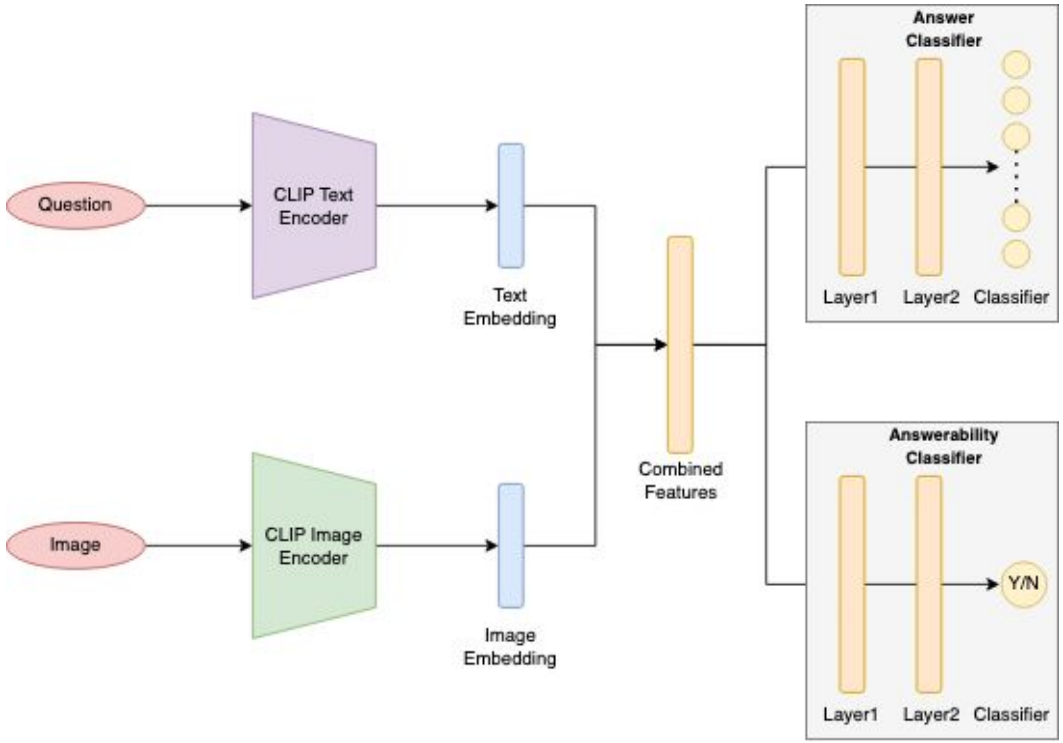
Project Flowchart



VisualAid VQA FlowChart

VQA

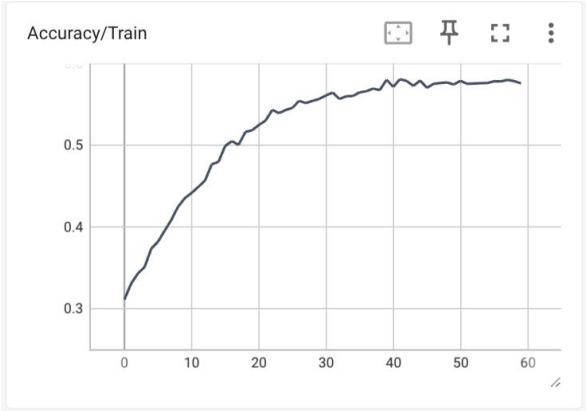
CLIP Based Visual Question Answering

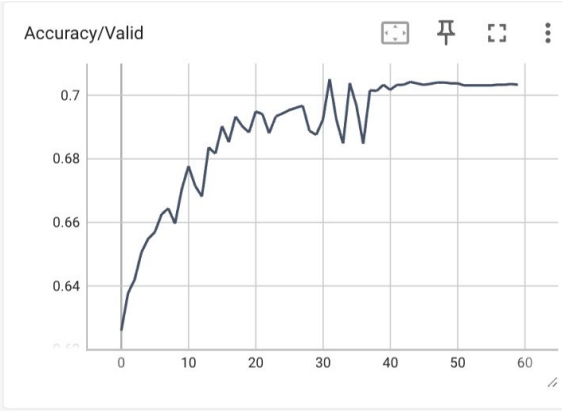


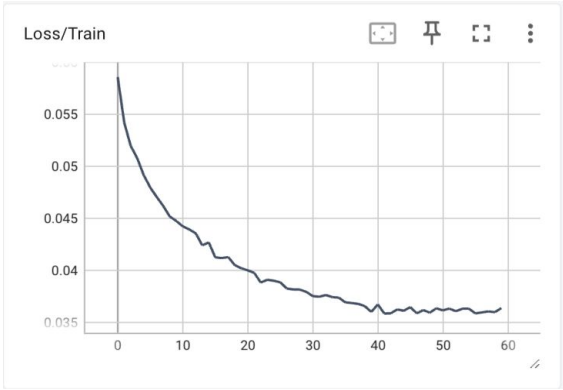
Key Components

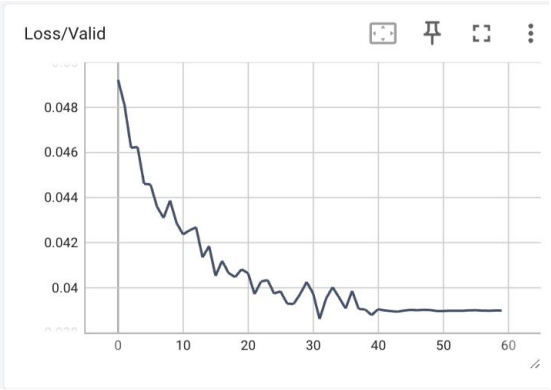
1. **Feature Extractors**
   a. CLIP Text Encoder : Transforms the question into a semantic text embedding.
   b. CLIP Image Encoder : Converts the image into a visual embedding.
2. **Answerability Detection**
   a. Determine if a given question can be answered with respect to the provided image
3. **Answer Classifier**
   a. Classifies each question into one of n predefined classes based on the combined image and text embeddings

CLIP VQA Training









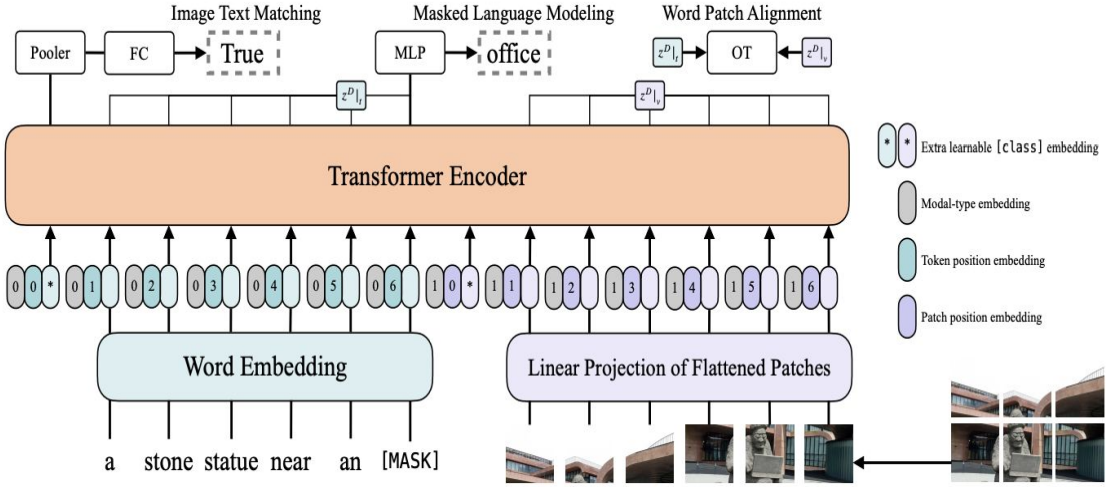
Hyperparameters

1. Epochs : 60
2. Batch_size : 128
3. Optimizer : Adam
4. Learning Rate : 0.0001
5. Clip Model : Clip RN50
6. Loss Function : Categorical Cross Entropy Loss

Network Architecture

1. CLIP Backbone Freezed
2. MLP Architecture
   a. FC (4096) -> LN -> Drop (0.5)
   b. FC (4096) -> LN -> Drop (0.5)

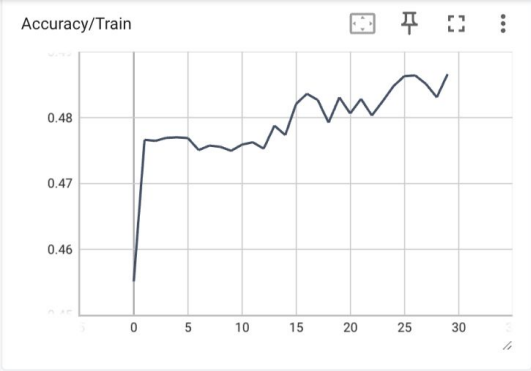
ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision

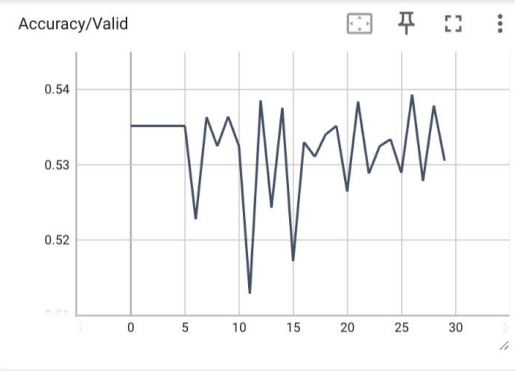


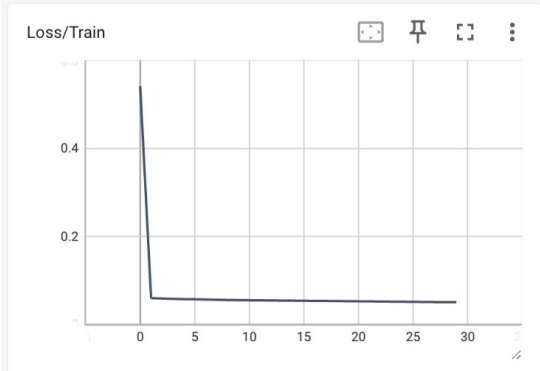
Key Components

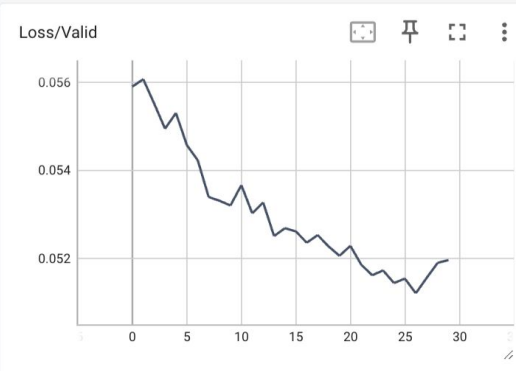
1. **Visual Encoder**: Utilizes Vision Transformers (ViT). Extracts visual features from images using linear projected patches.
2. **Textual Encoder**: Uses BERT to process the textual input
3. **Cross-Modal Interaction**: Uses a Transformer-based architecture to fuse visual and textual features.
4. **Pre-training Tasks**:
   a. Masked Language Modeling (MLM) and Image-Text Matching (ITM)
5. **Fine-Tuning for VQA**:
   a. Input: Combines image features and question embeddings.
   b. Output: Predicts answers from a predefined set of possible answers.

Vilt VQA Training









Hyperparameters

1. Epochs : 30
2. Batch_size : 160
3. Optimizer : SGD with Cyclic LR Scheduler
4. Base Lr : 0.0001
5. Max Lr : 0.005
6. Momentum : 0.9
7. Loss Function : Categorical Cross Entropy Loss

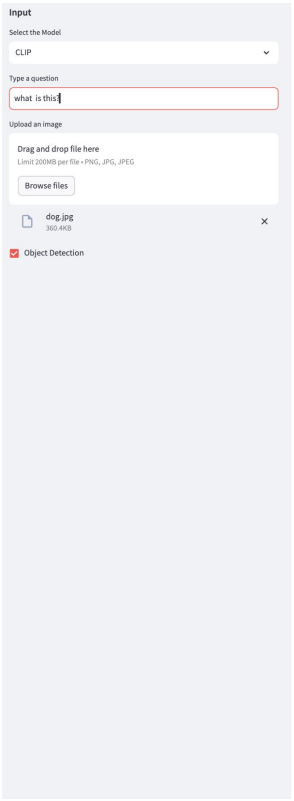
Results

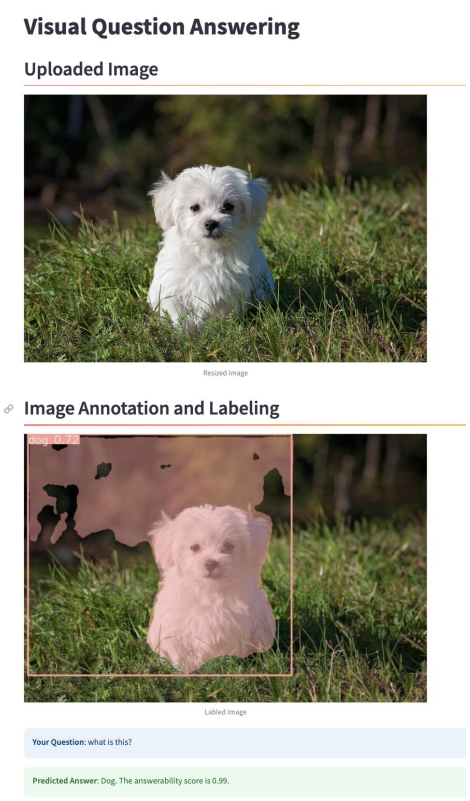
Performance Metrics

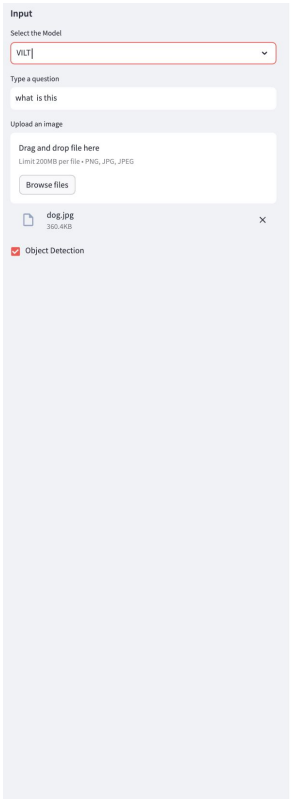
a. **accuracy** = min(# humans that provided that answer / 3, 1)

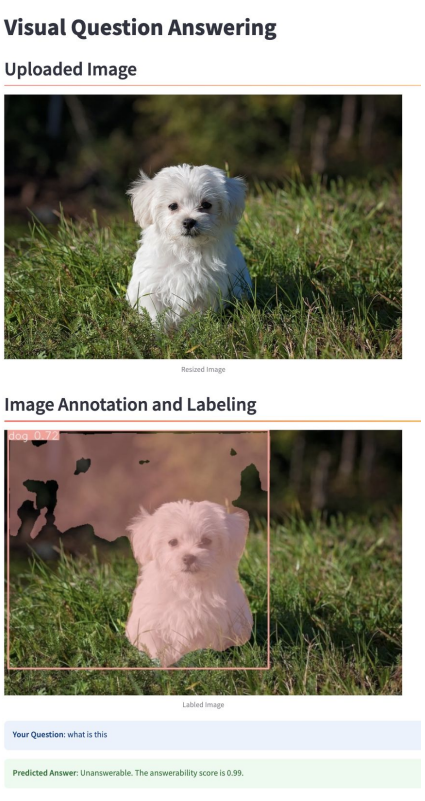
Model Name	Train Accuracy	Validation Performance
Answerability Module	98.72 %	94.37 %
CLIP VQA	56.44 %	70.51 %
Vilt VQA	47.49 %	53.85 %

Example Predictions

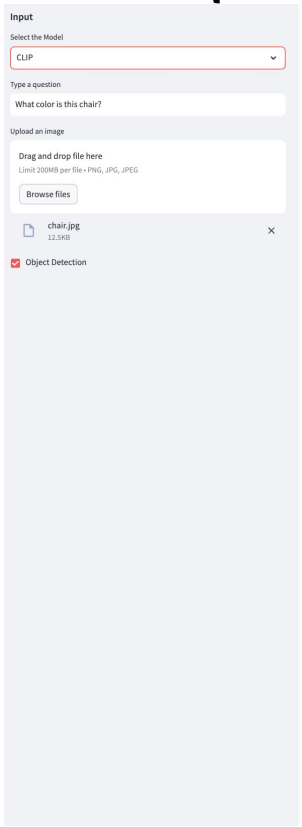


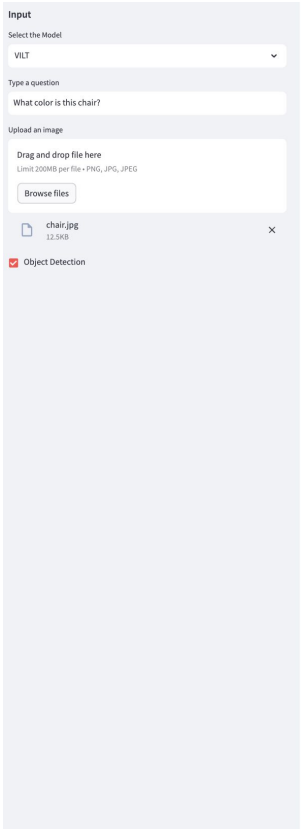






Example Predictions



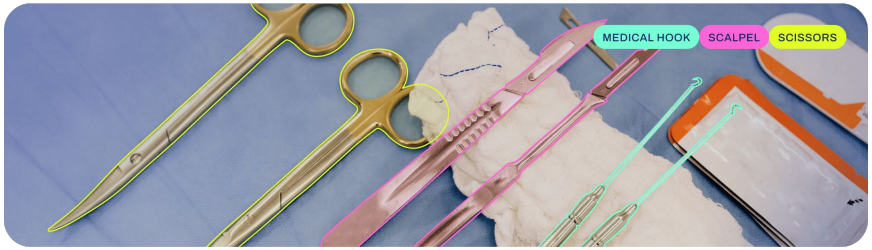


Instance Segmentation

Models

YOLOv8seg

- We trained on the top 5 classes in the dataset, which are *pen, dog, laptop, keyboard, wine*.
- The major data processing was conversion of annotations to YOLO format.



Mask RCNN

- The major data processing was the creation of masks from image annotations.
- Similarly trained from top 5 classes

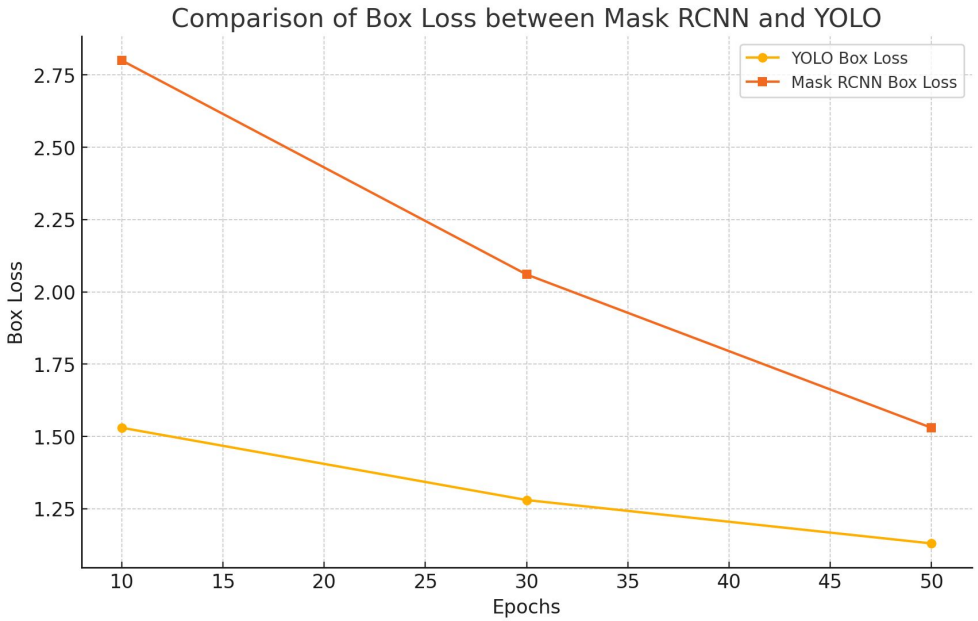
Model Definition

YOLOv8: operates as a single-stage detector, meaning it processes the entire image with a single network, which makes it faster.

Mask RCNN: Two-Stage Detector: first proposes candidate object regions using a Region Proposal Network (RPN), then refines these proposals and predicts masks.

Results

Evaluation metric: Visual inspection + Box loss



Box loss: how well the predicted bounding boxes match the ground truth bounding boxes

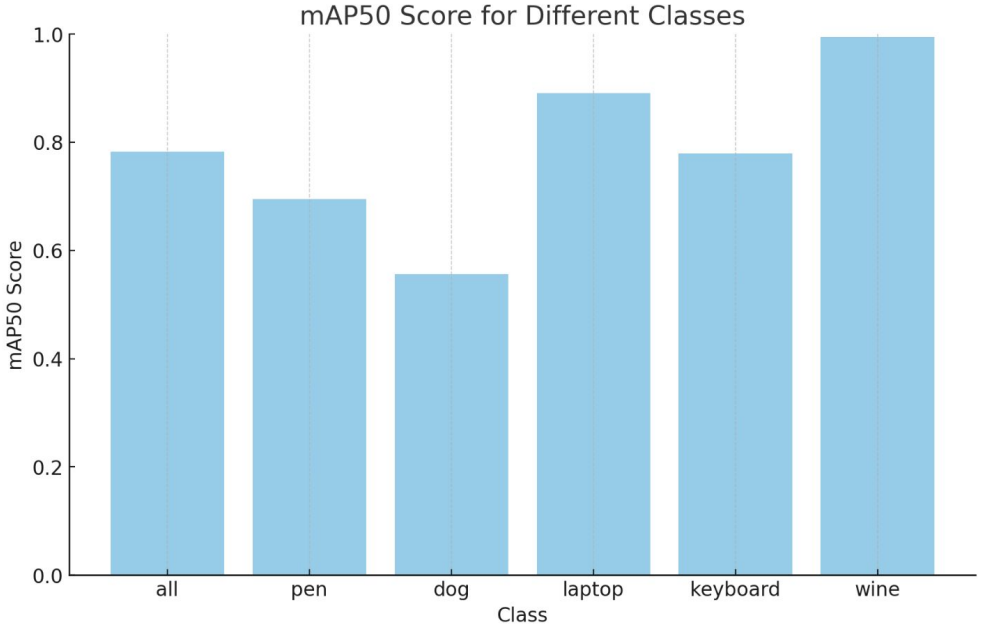
Both model hyperparameters:

- Epochs: 50
- Image size: 1280

YOLO MAP scores

What is MAP 50:

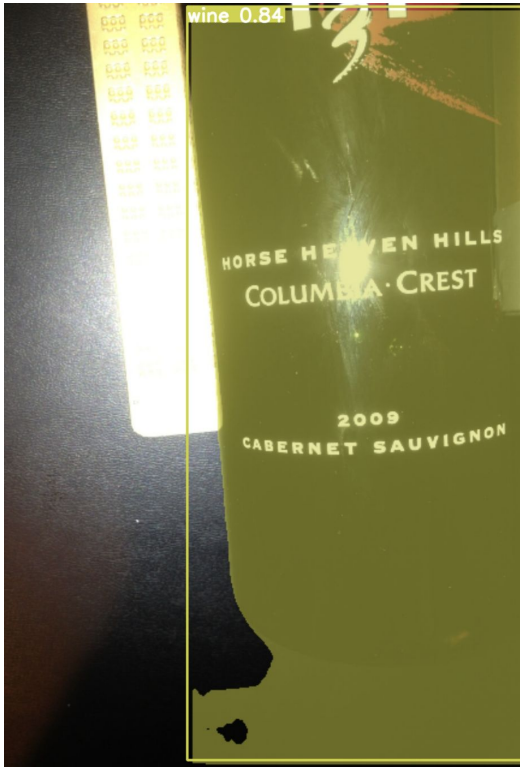
It accounts for both the precision and recall of the model's predictions at an IoU threshold of **0.50** (*predicted bounding box is considered a true positive if it overlaps with the ground truth bounding box by at least 50%*), offering a balanced measure of accuracy.

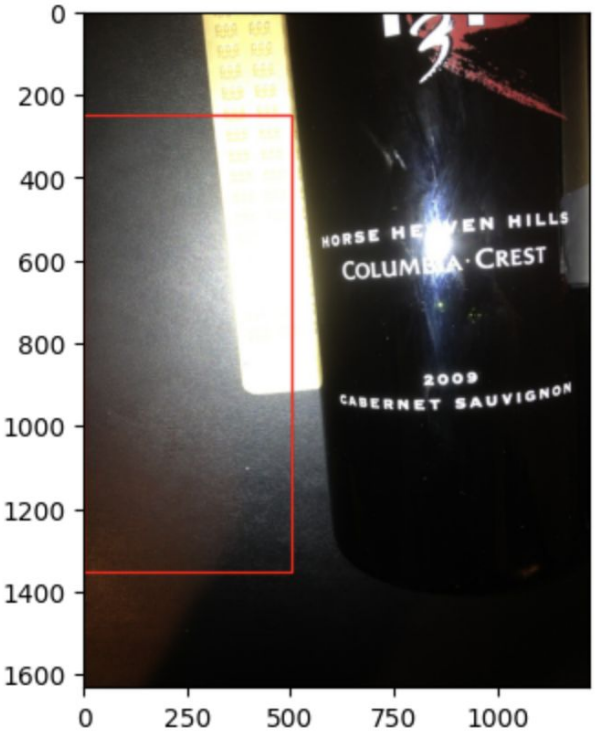


IoU: Intersection over Union

Sample Images

YOLOv8

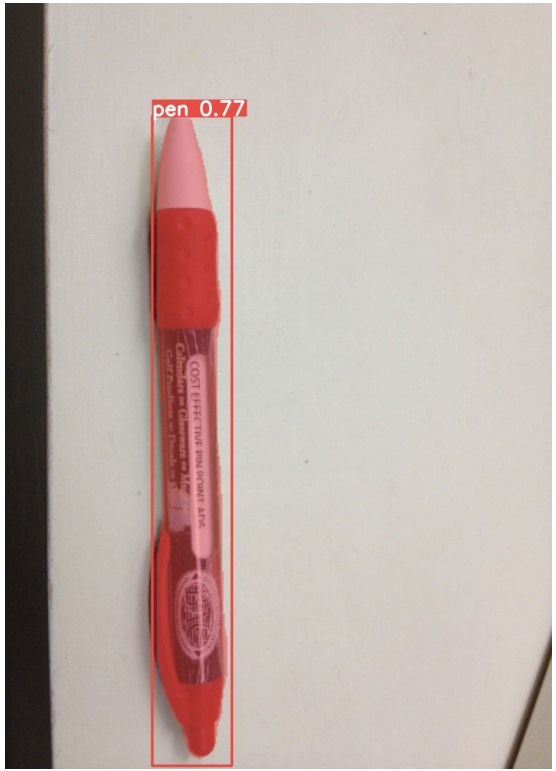


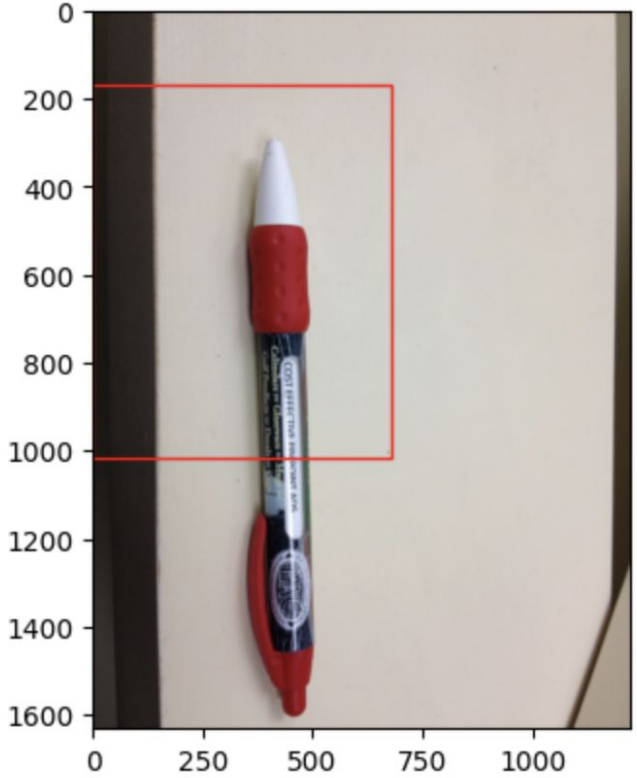


Mask RCNN

Sample Images

YOLOv8

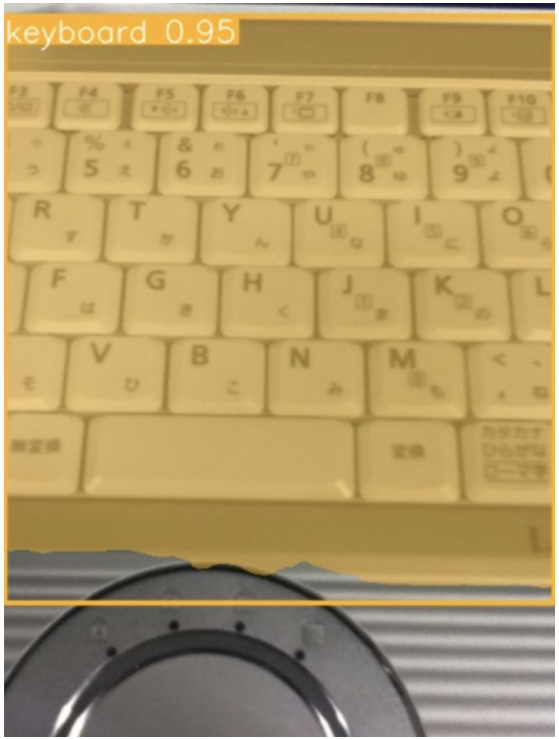


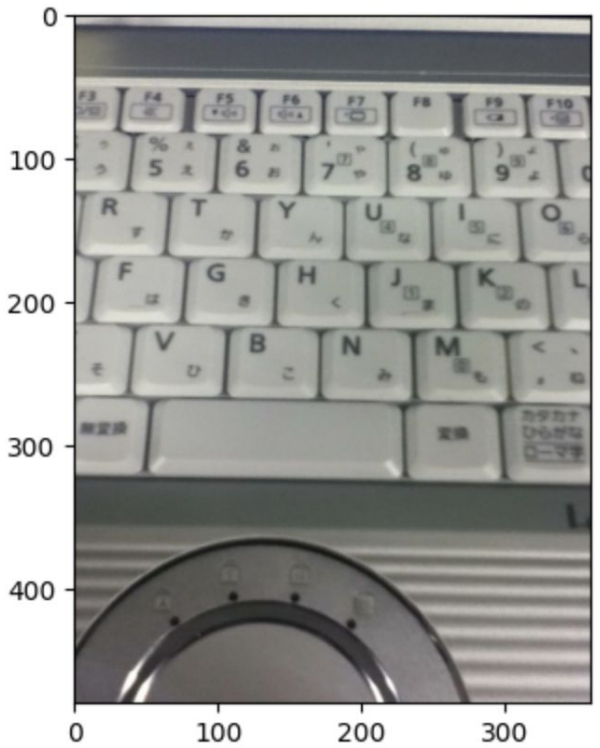


Mask RCNN

Sample Images

YOLOv8





Mask RCNN

Web Interface