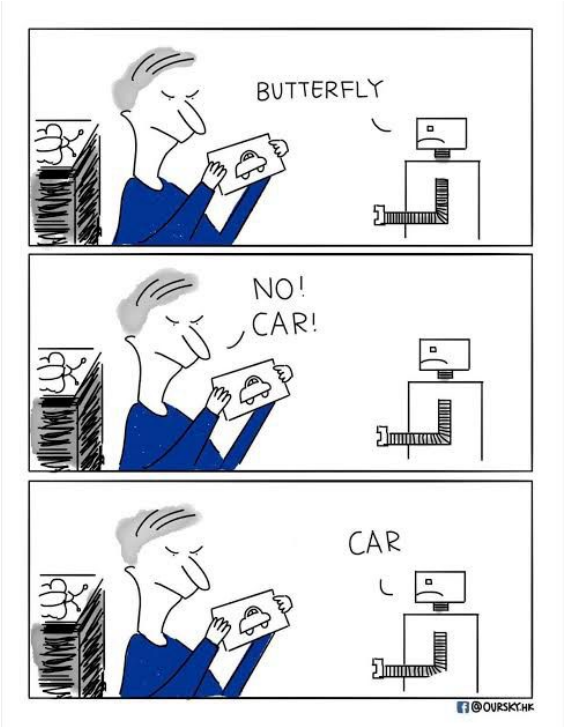
Demonstration session for Image Classification

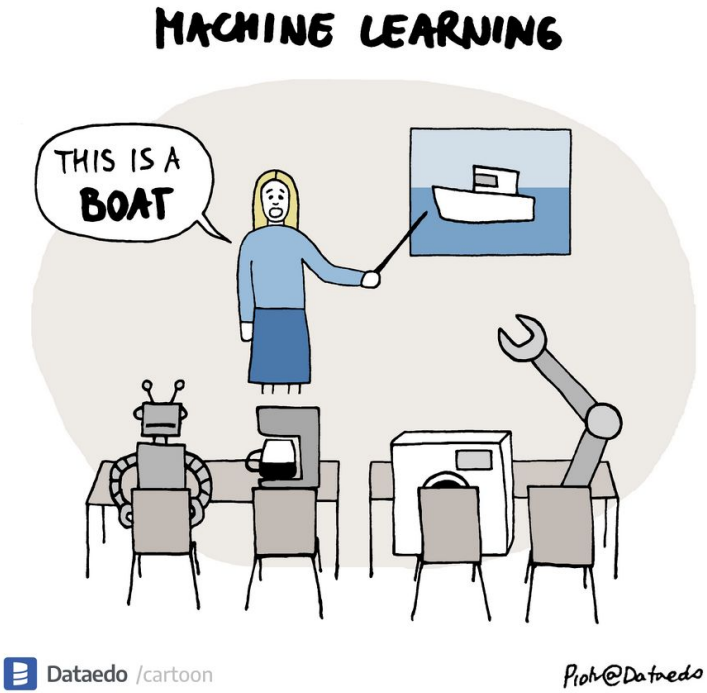
Nyi Nyi Soe

MBBS, MPH, PhD Candidate

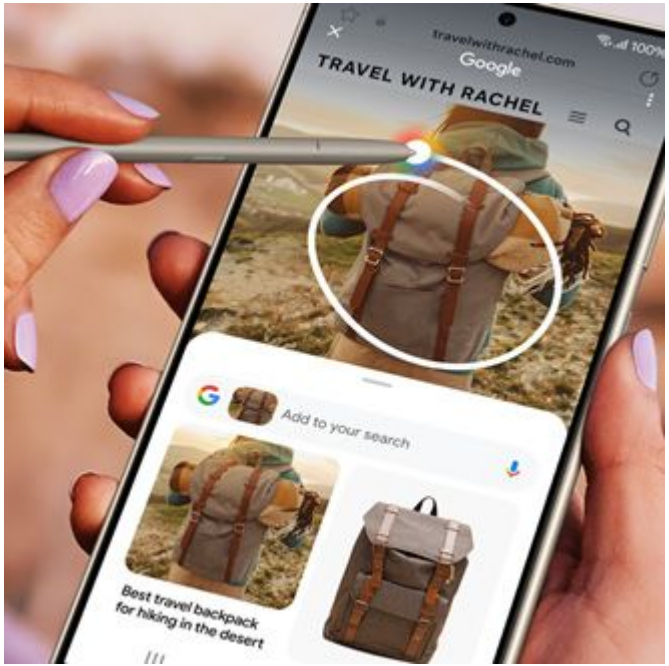
What is Image Classification?

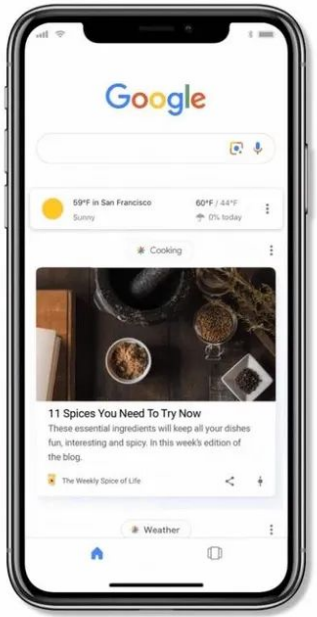
Let's start a teaching to a robot





What is Image Classification?





Write many lines of python codes

```
<div>Teachable Machine Image Model</div>
<button type="button" onclick="init()">Start</button>
<div id="webcam-container"></div>
<div id="label-container"></div>
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest/dist/tf.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@teachablemachine/image@latest/dist/teachablemachine-image.min.js"></script
<script type="text/javascript">
    // More API functions here:
    // https://github.com/googlecreativelab/teachablemachine-community/tree/master/libraries/image

    // the link to your model provided by Teachable Machine export panel
    const URL = "{{URL}}";

    let model, webcam, labelContainer, maxPredictions;

    // Load the image model and setup the webcam
    async function init() {
        const modelURL = URL + "model.json";
        const metadataURL = URL + "metadata.json";

        // load the model and metadata
        // Refer to tmImage.loadFromFiles() in the API to support files from a file picker
        // or files from your local hard drive
        // Note: the pose library adds "tmImage" object to your window (window.tmImage)
        model = await tmImage.load(modelURL, metadataURL);
        maxPredictions = model.getTotalClasses();

        // Convenience function to setup a webcam
        const flip = true; // whether to flip the webcam
        webcam = new tmImage.Webcam(200, 200, flip); // width, height, flip
        await webcam.setup(); // request access to the webcam
        await webcam.play();
        window.requestAnimationFrame(loop);

        // append elements to the DOM
        document.getElementById("webcam-container").appendChild(webcam.canvas);
        labelContainer = document.getElementById("label-container");
        for (let i = 0; i < maxPredictions; i++) { // and class labels
            labelContainer.appendChild(document.createElement("div"));
        }
    }
```

```
<div>Teachable Machine Image Model - p5.js and ml5.js</div>
<script src="https://cdn.jsdelivr.net/npm/p5@latest/lib/p5.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/p5@latest/lib/addons/p5.dom.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/ml5@latest/dist/ml5.min.js"></script>
<script type="text/javascript">
  // Classifier Variable
  let classifier;
  // Model URL
  let imageModelURL = '{{URL}}';

  // Video
  let video;
  let flippedVideo;
  // To store the classification
  let label = "";

  // Load the model first
  function preload() {
    classifier = ml5.imageClassifier(imageModelURL + 'model.json');
  }

  function setup() {
    createCanvas(320, 260);
    // Create the video
    video = createCapture(VIDEO);
    video.size(320, 240);
    video.hide();

    flippedVideo = ml5.flipImage(video);
    // Start classifying
    classifyVideo();
  }

  function draw() {
    background(0);
    // Draw the video
    image(flippedVideo, 0, 0);
```

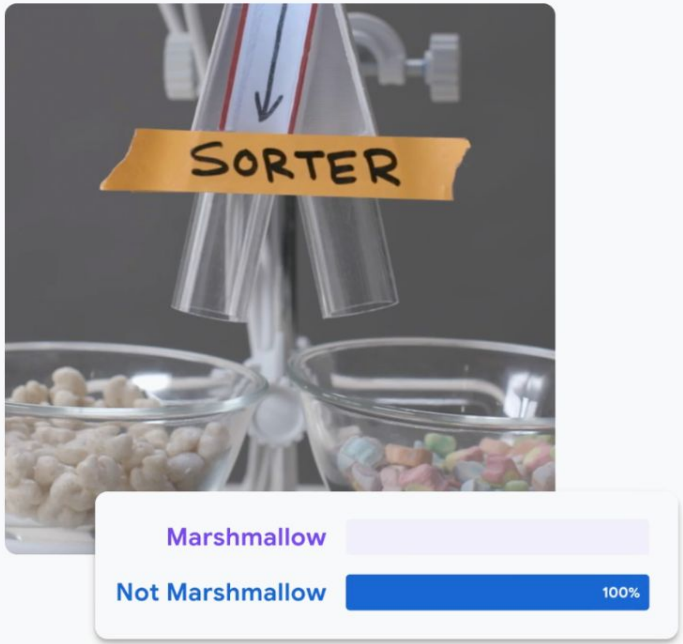
https://teachablemachine.withgoogle.com/

Teachable Machine

Train a computer to recognize your own images, sounds, & poses.

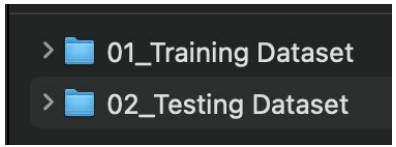
A fast, easy way to create machine learning models for your sites, apps, and more – no expertise or coding required.

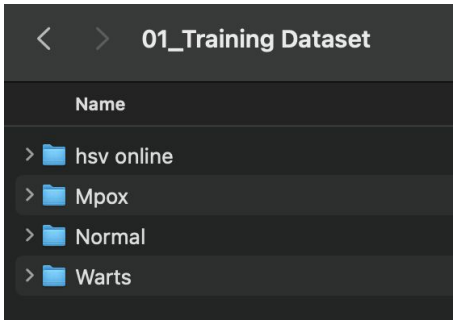
Get Started

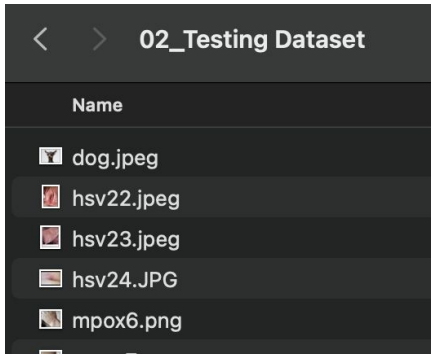


Prerequisite

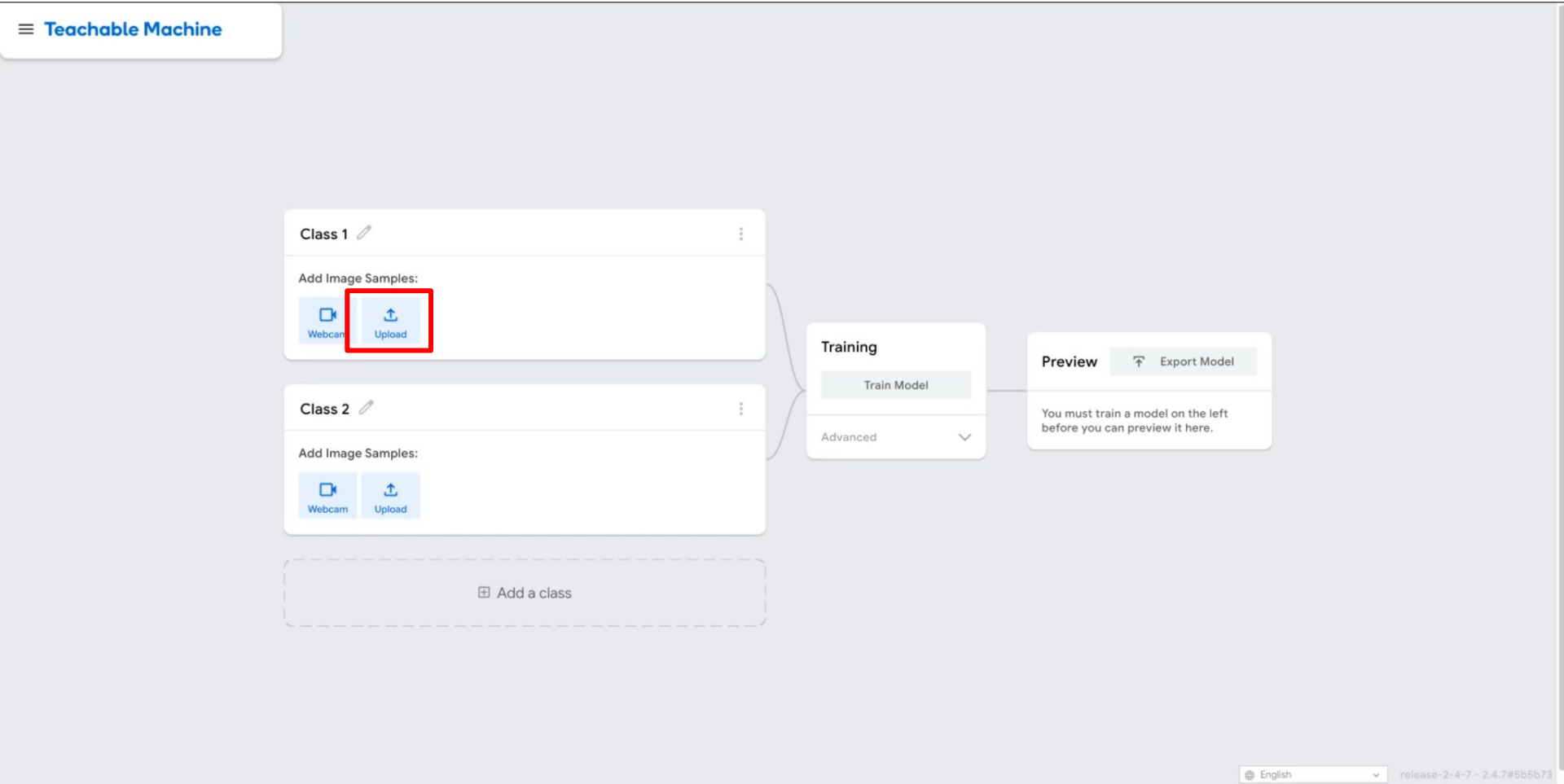
- Need Internet Access
- Download the Images to your device



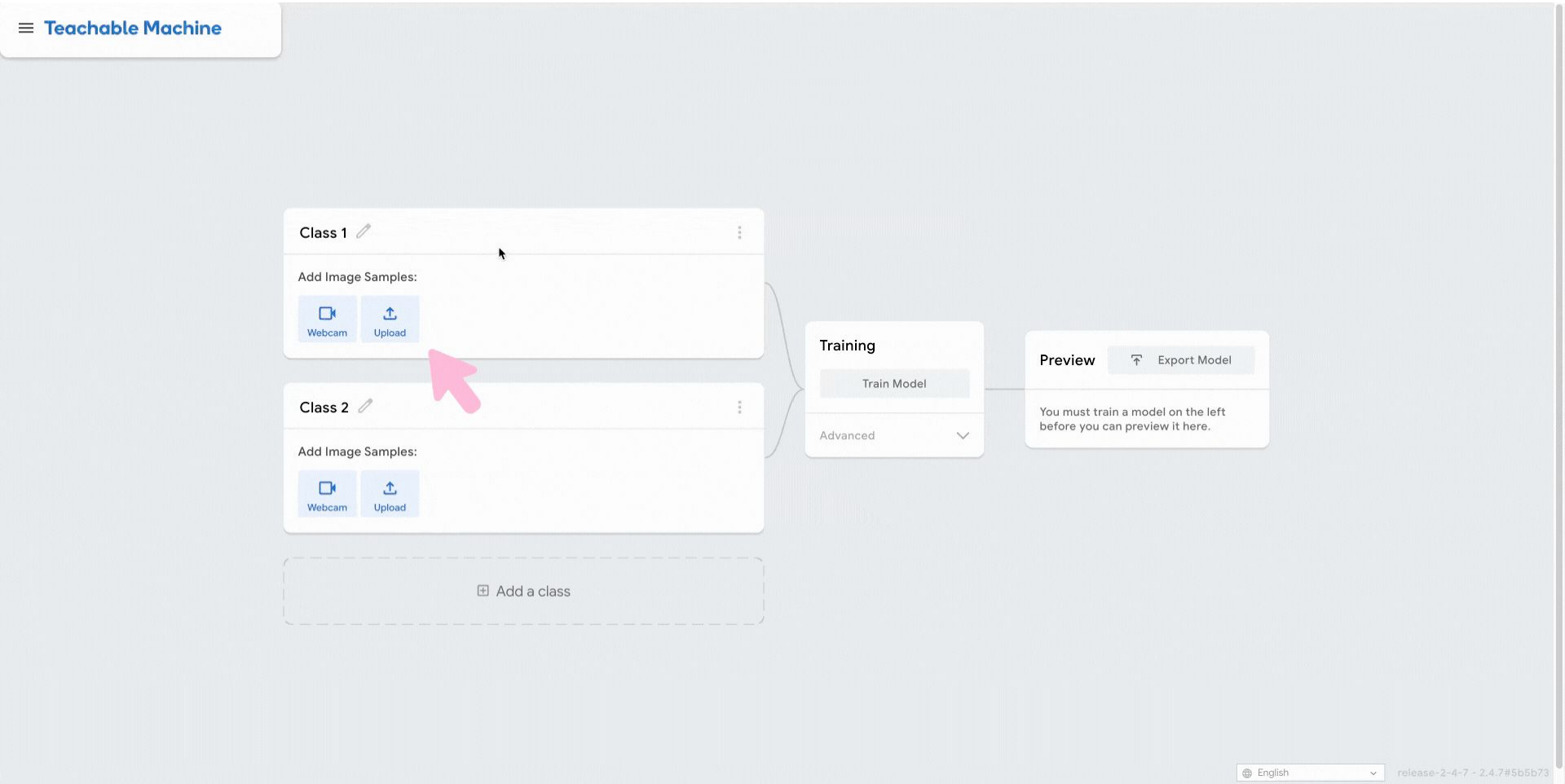




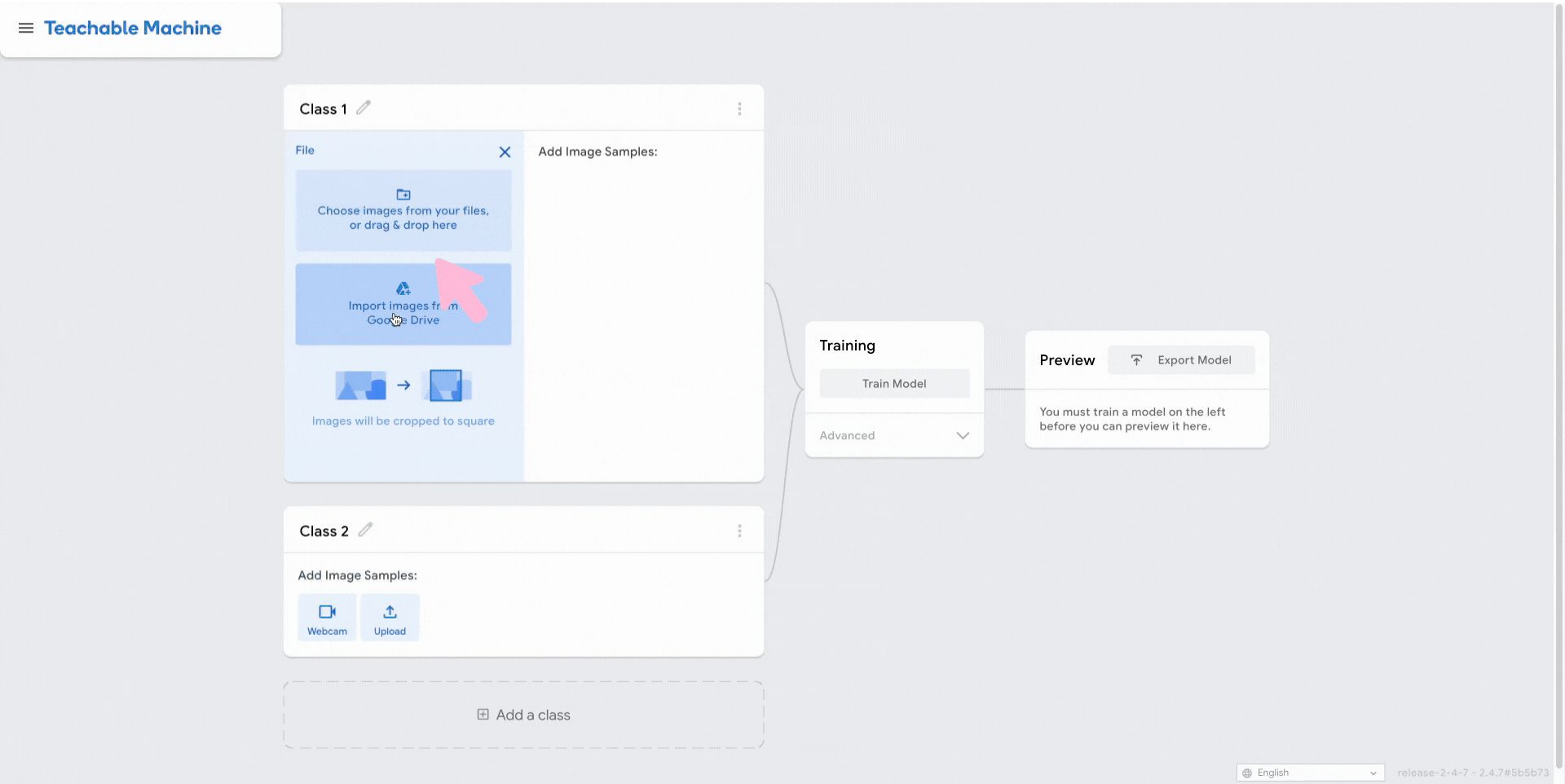
Step 1: Uploading Images



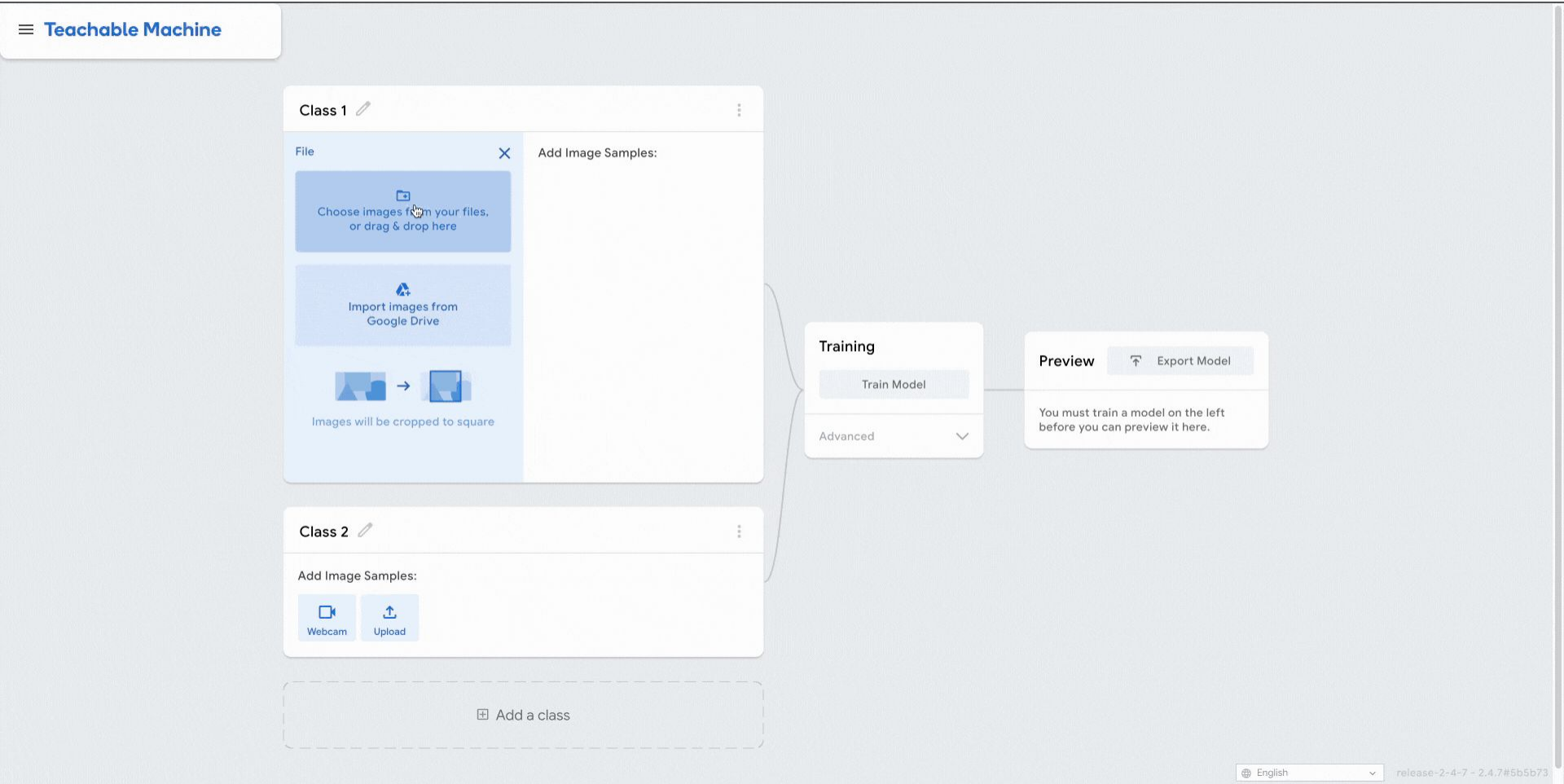
Step 1: Uploading Images



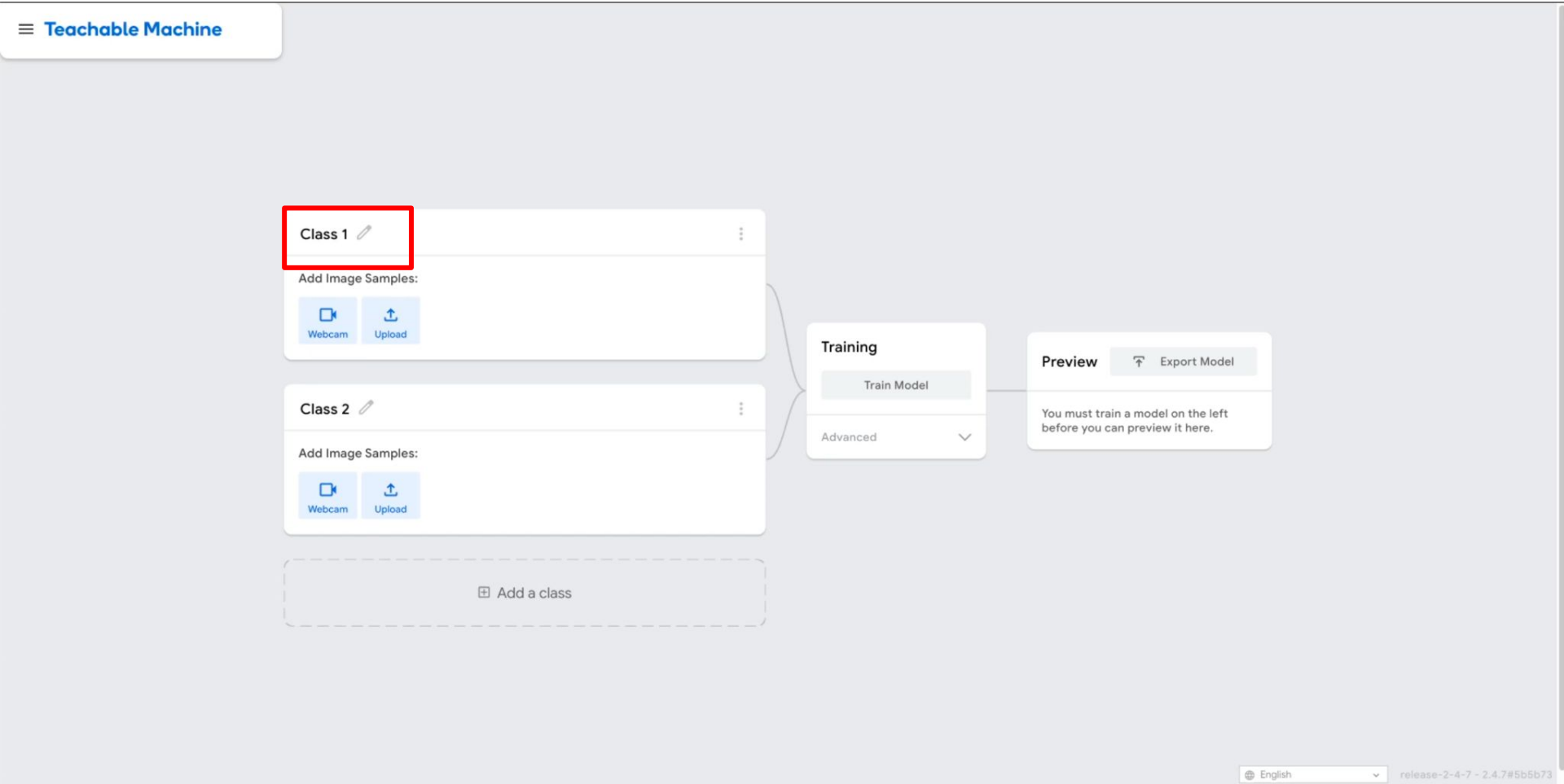
Step 1: Uploading Images



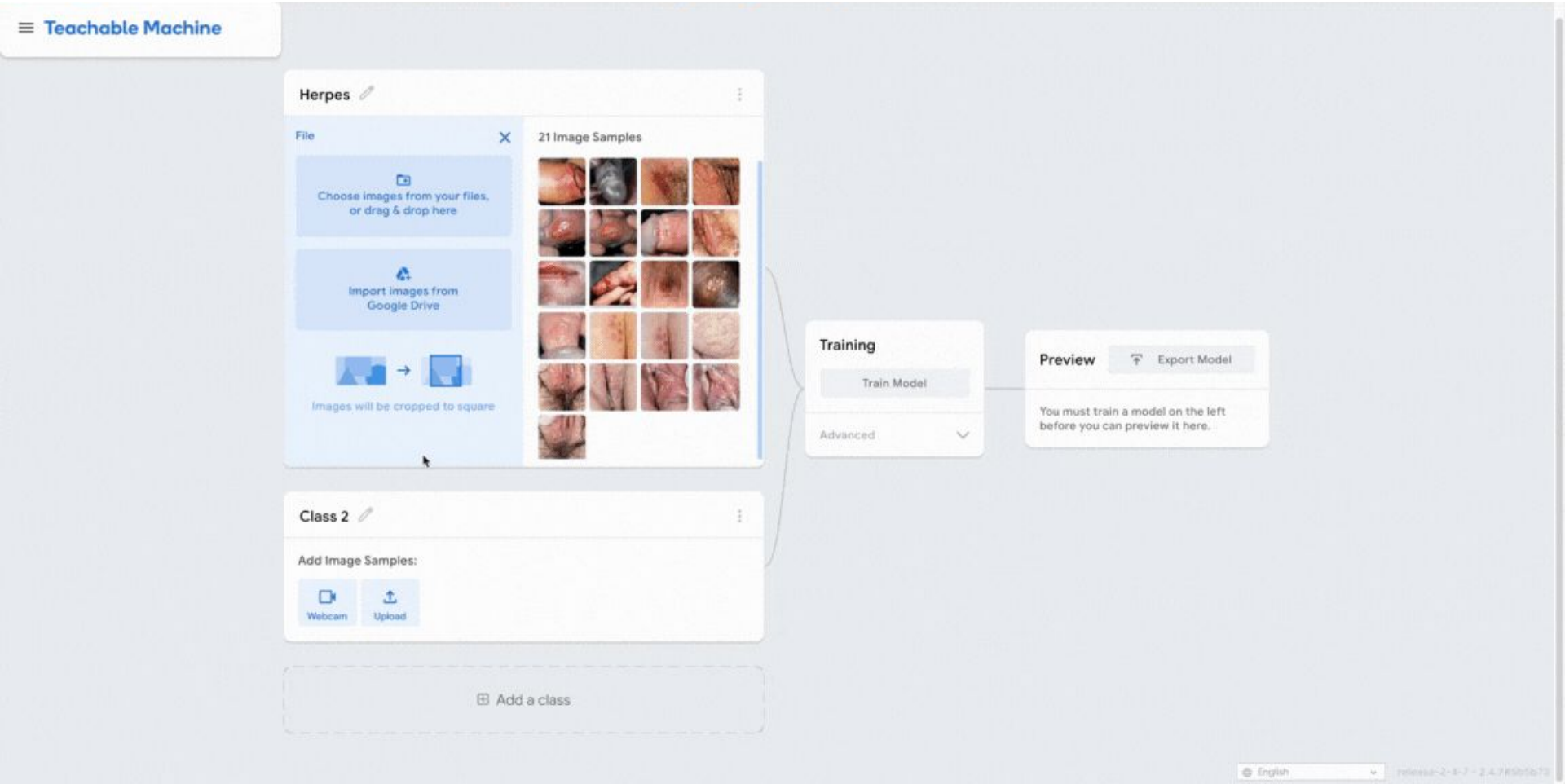
Step 1: Uploading HSV Images



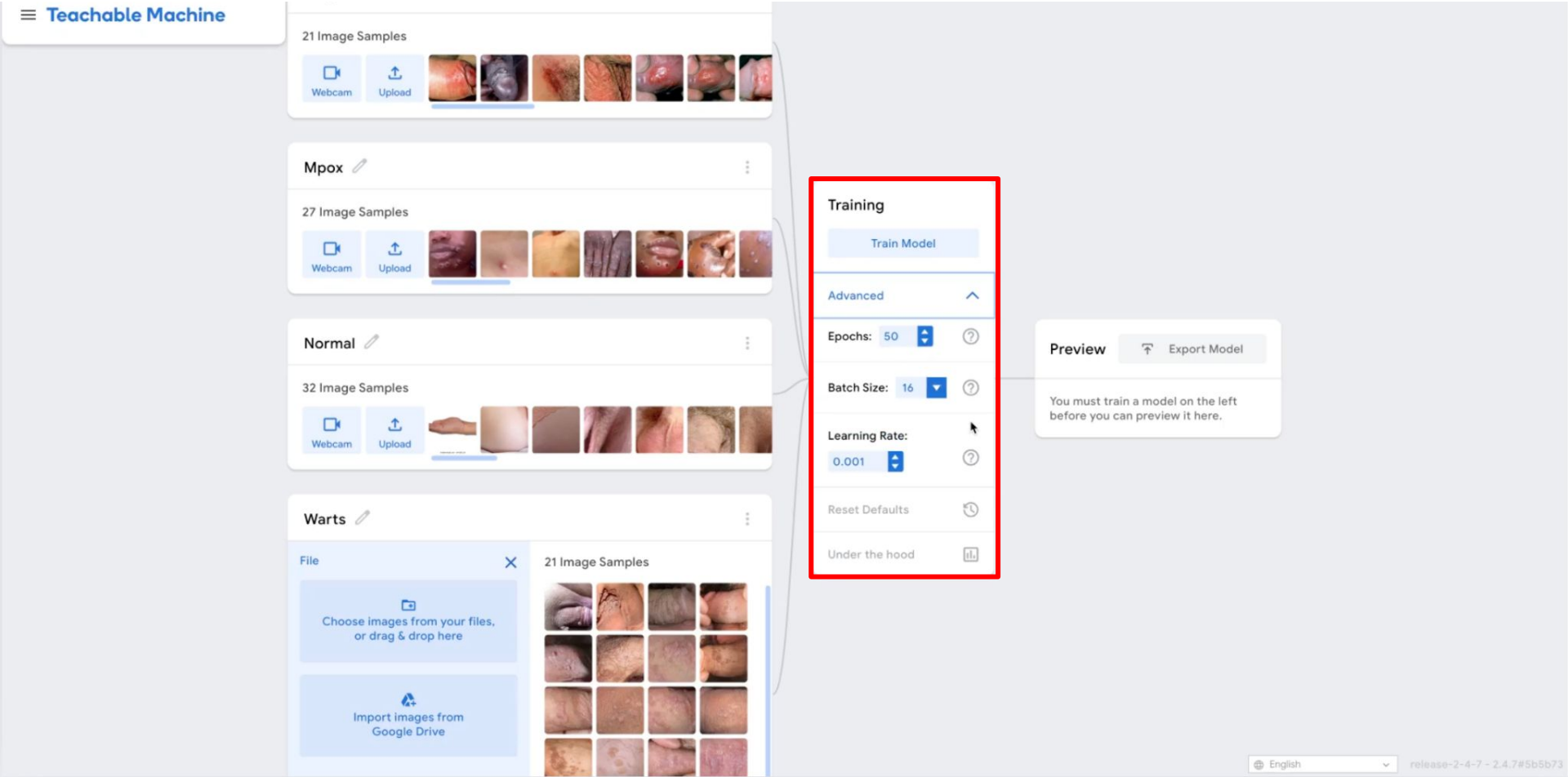
Step 1: Uploading Images



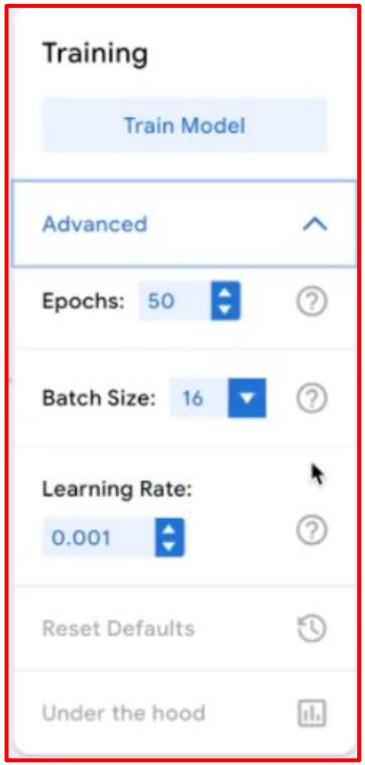
Step 1: Repeat uploading other disease classes



Step 2: Training



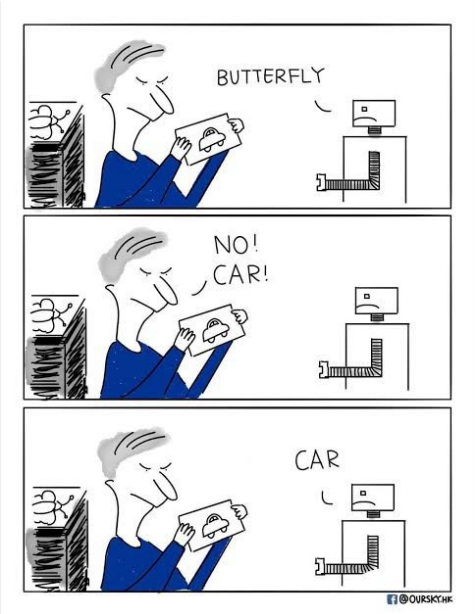
Step 2: Training



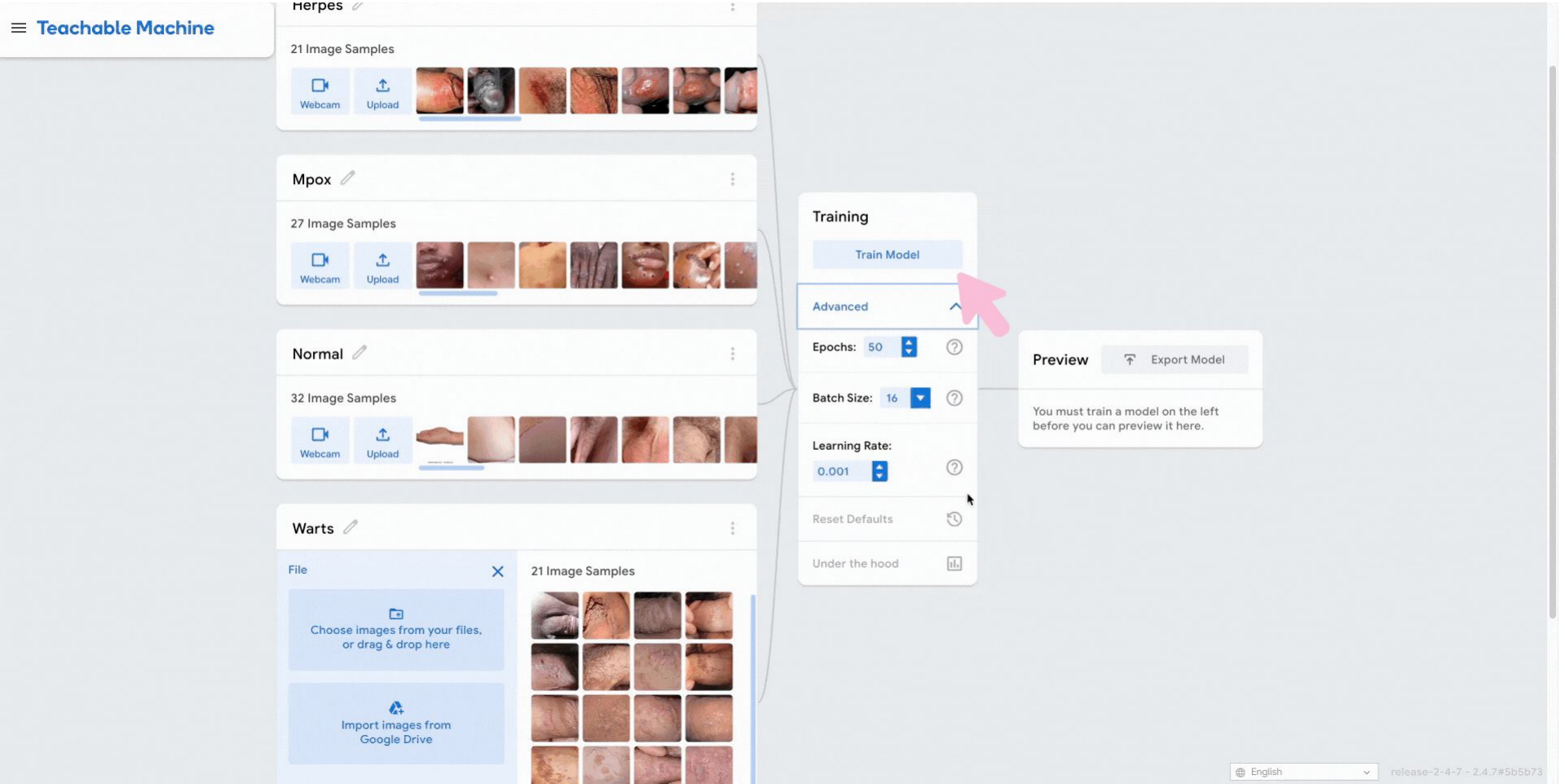
Epochs

One epoch: each and every images in the training has been used through training model model at least once.

50 epochs: training with entire dataset x 50 times

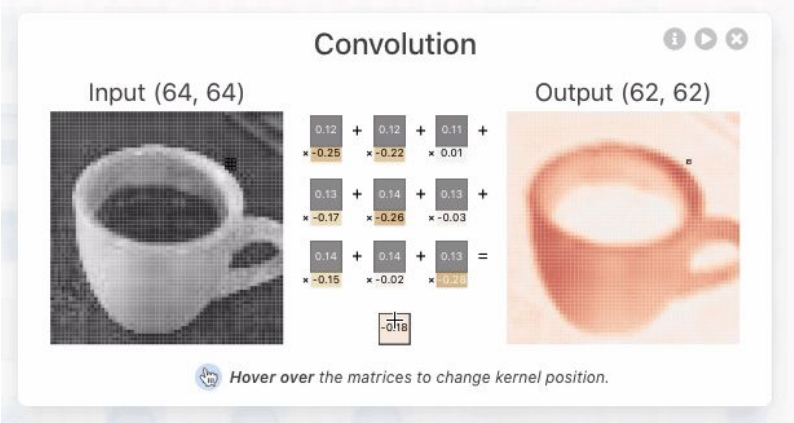


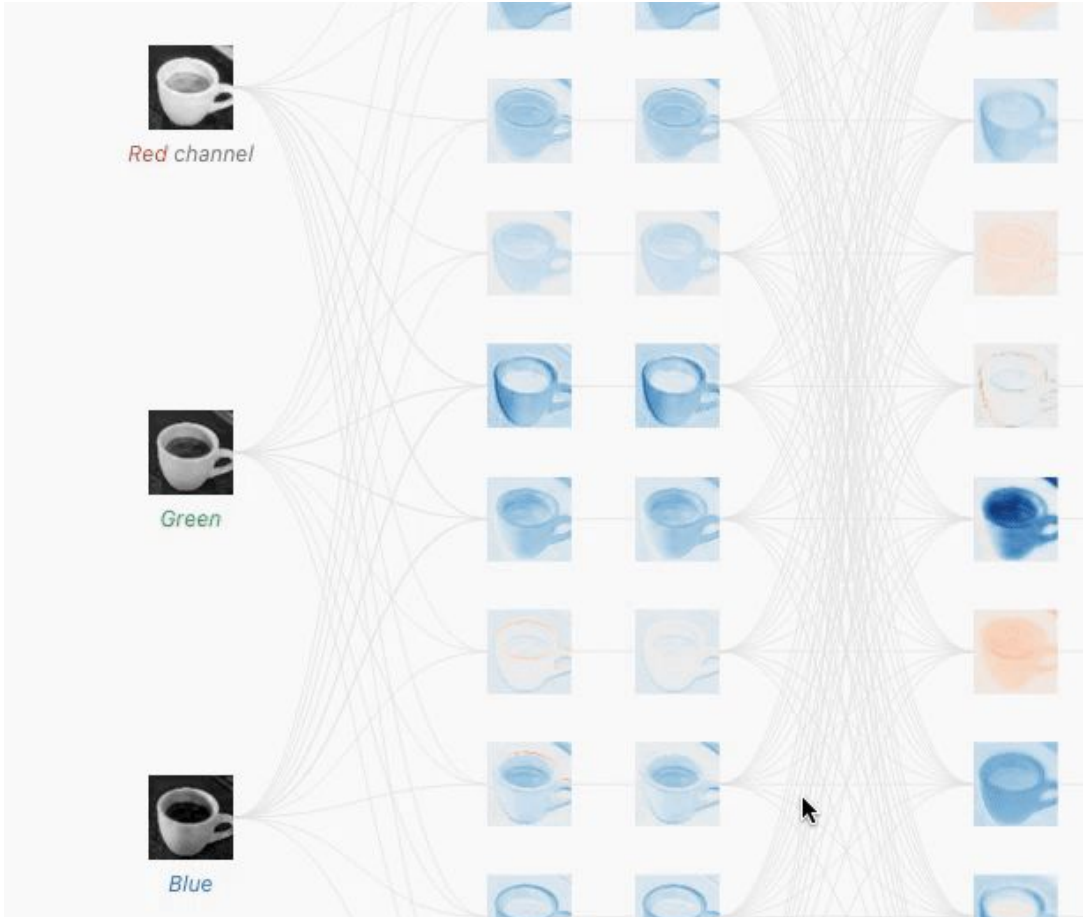
Step 2: Training



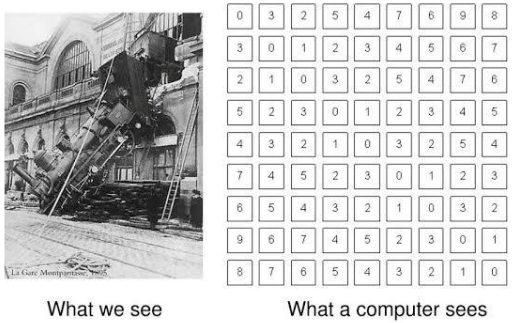
Step 2: How does it learn?

Step 2: Training





Computer vision vs human vision

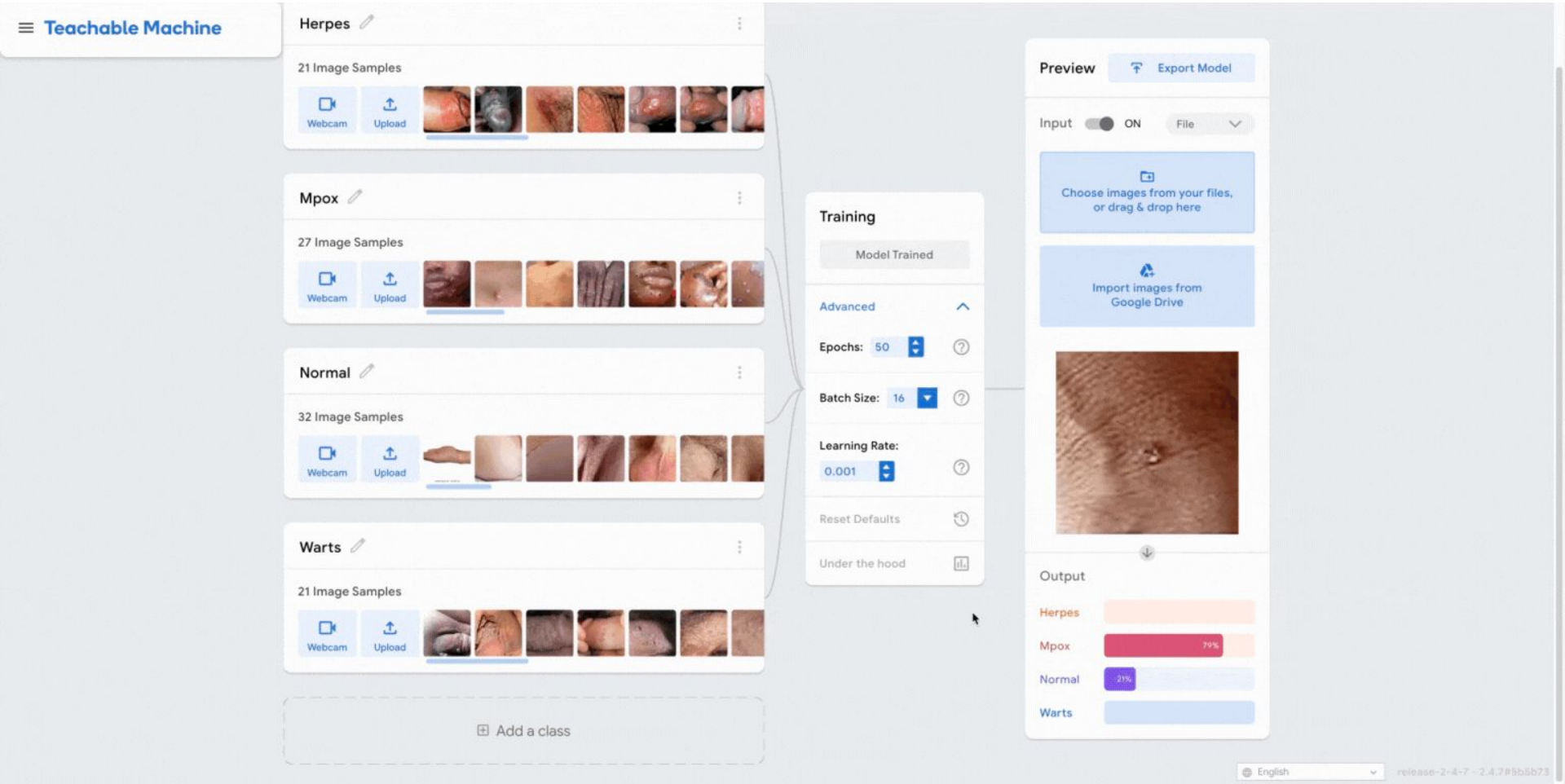


What we see

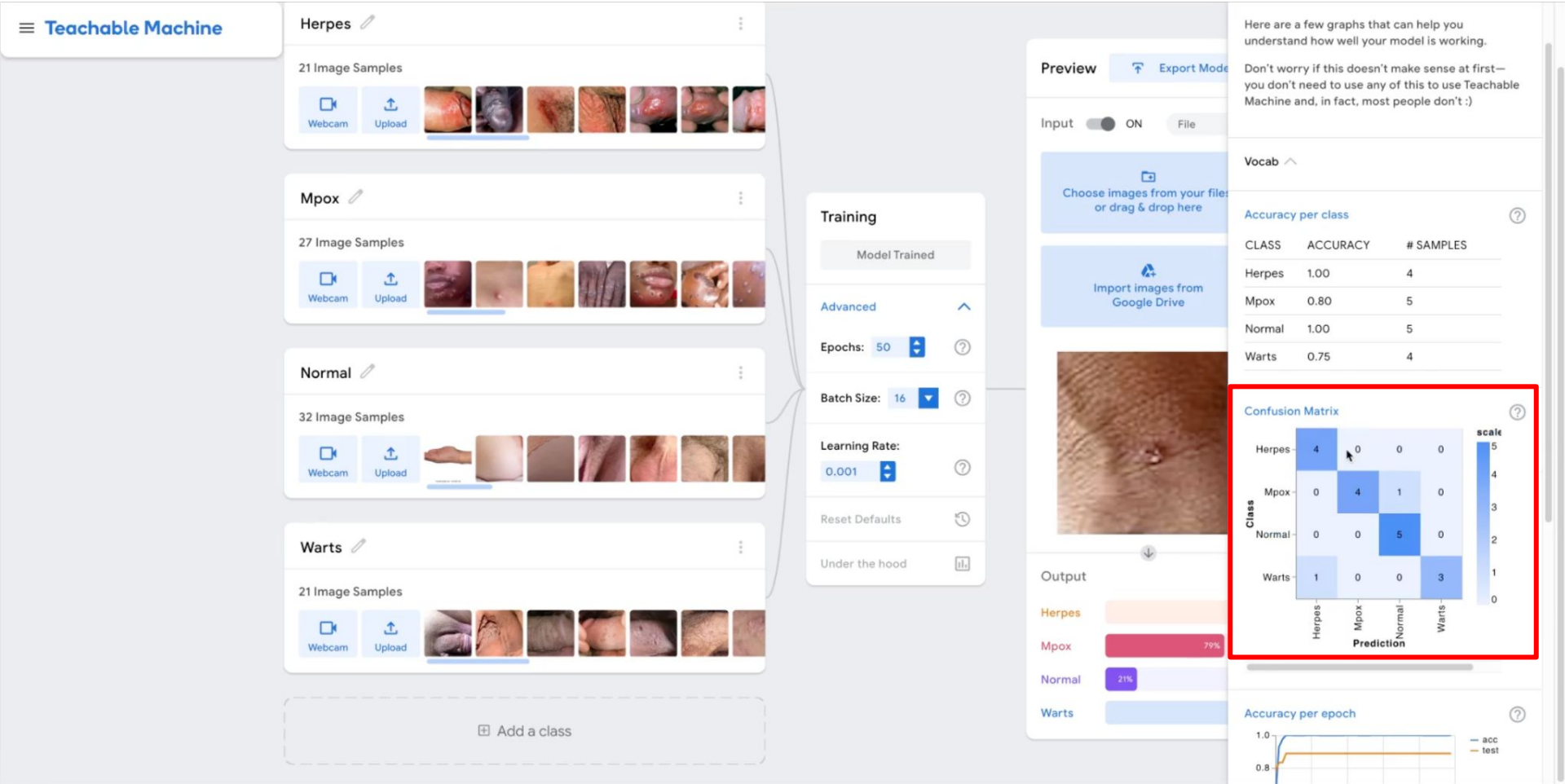
What a computer sees

https://poloclub.github.io/cnn-explainer/

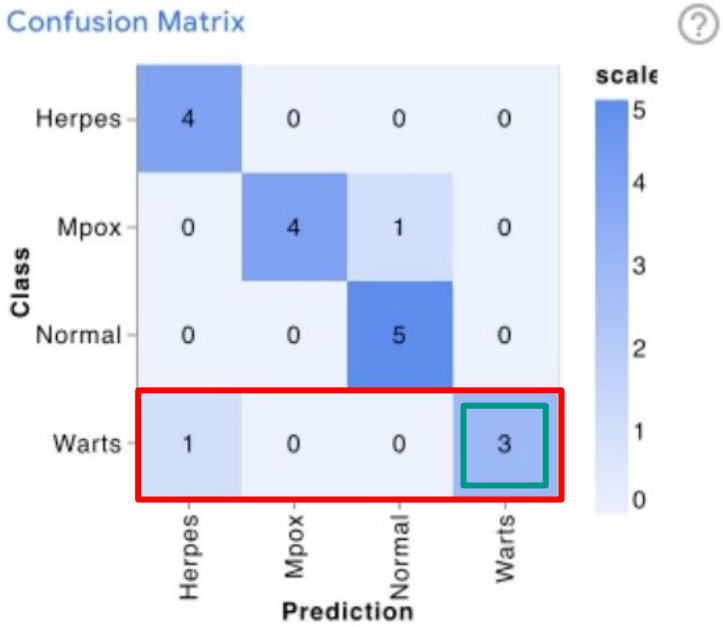
Step 3: Interpretations

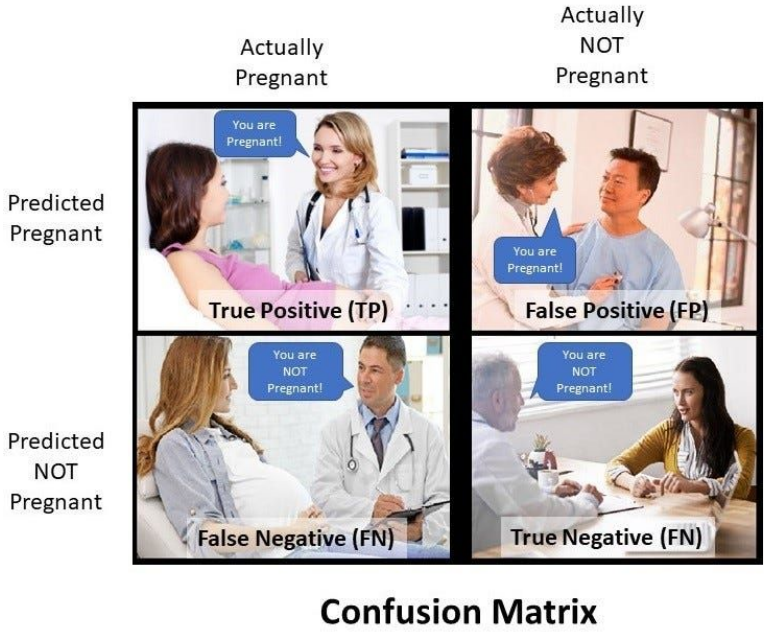


Step 3: Interpretations



Step 3: Interpretations





Confusion Matrix

Actual Warts = 4

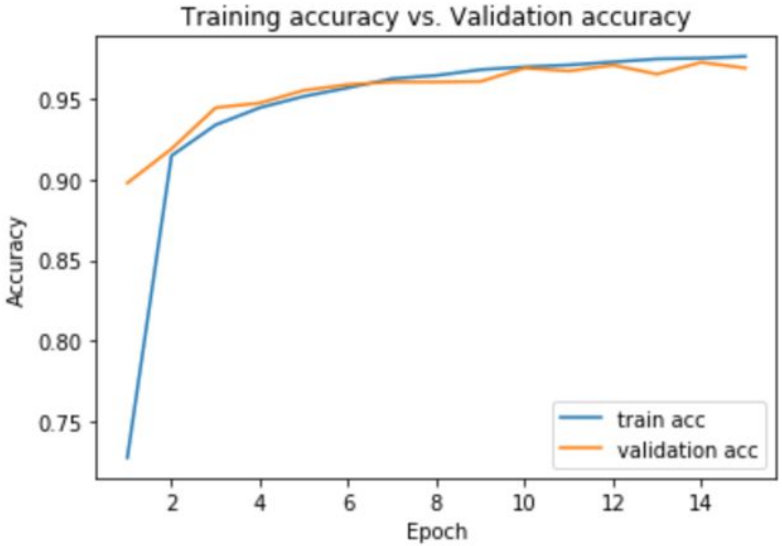
AI classification for Warts = 3 out of all tested images

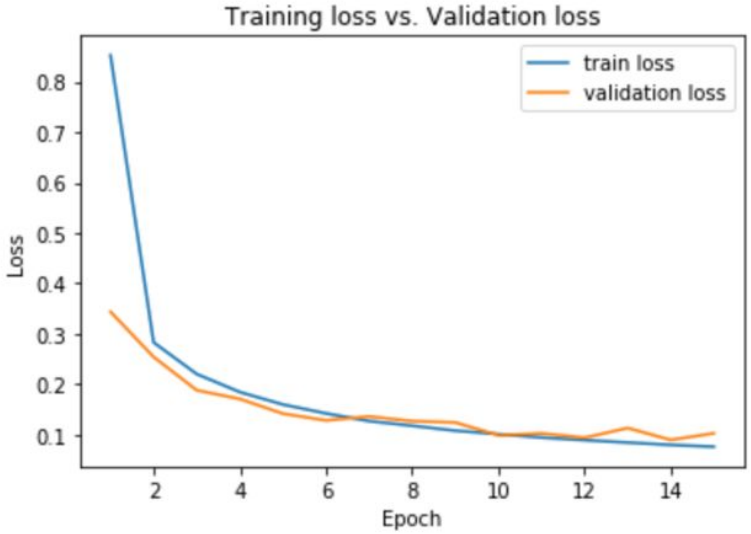
AI Accuracy for Warts = ¾ = 75%

Accuracy per class

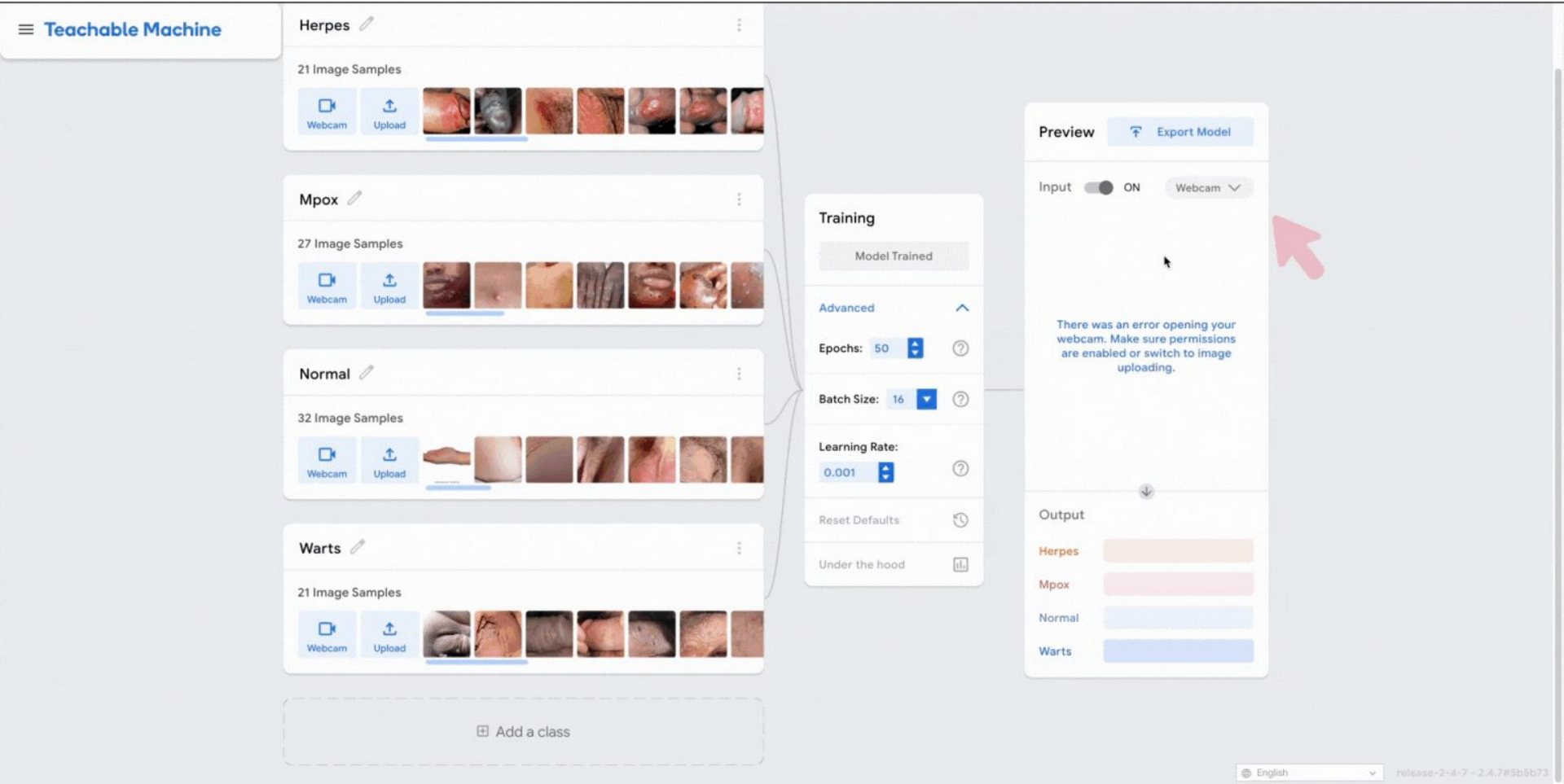
CLASS	ACCURACY	# SAMPLES
Herpes	1.00	4
Mpox	0.80	5
Normal	1.00	5
Warts	0.75	4

Step 3: Interpretations

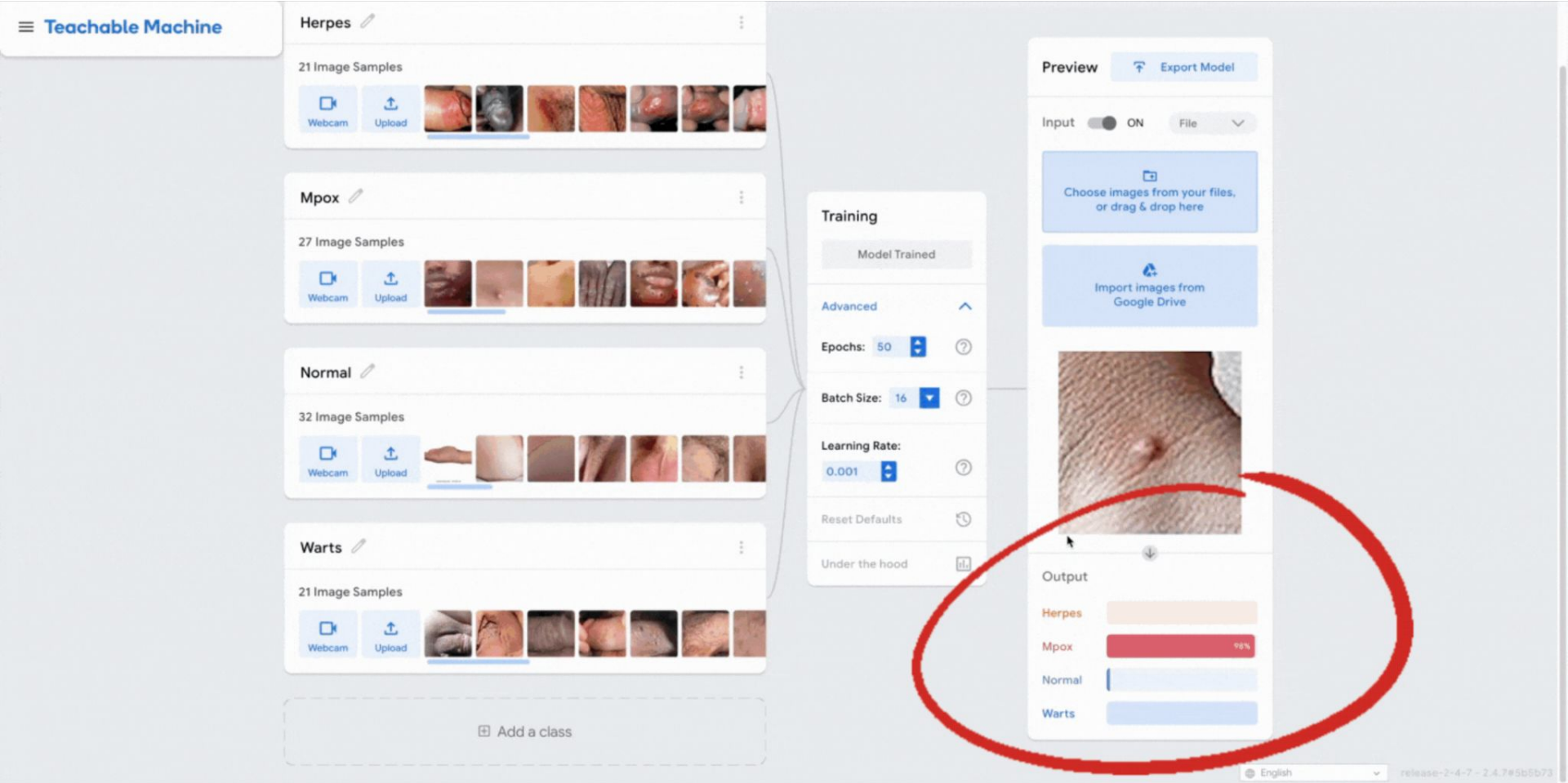




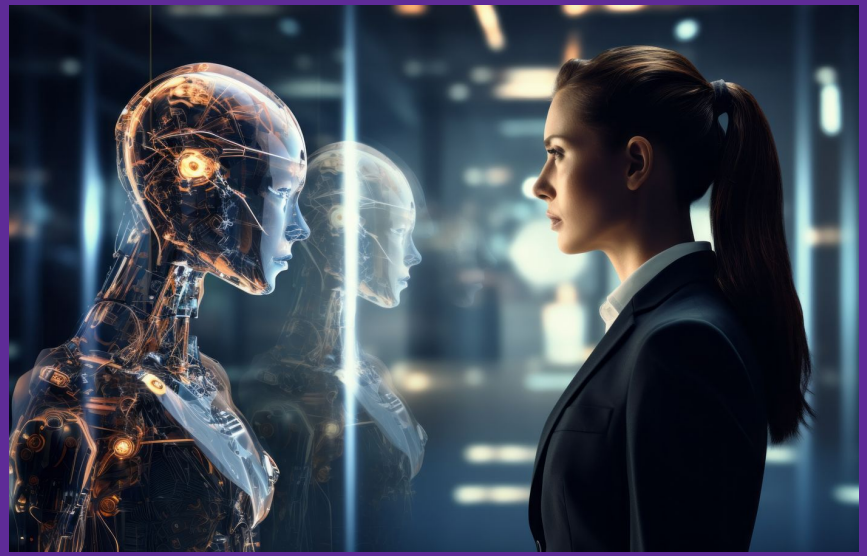
Step 4: Testing with unseen images



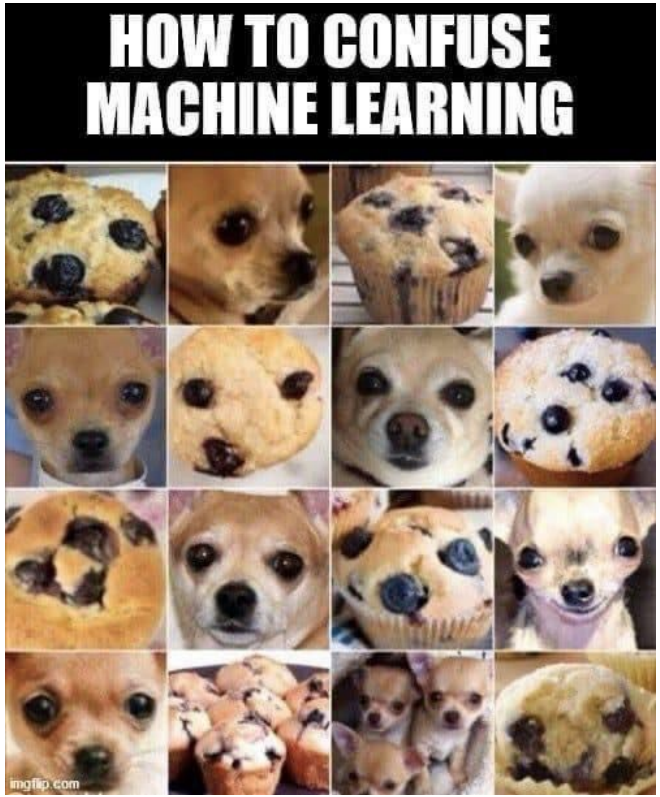
Step 4: Testing with unseen images

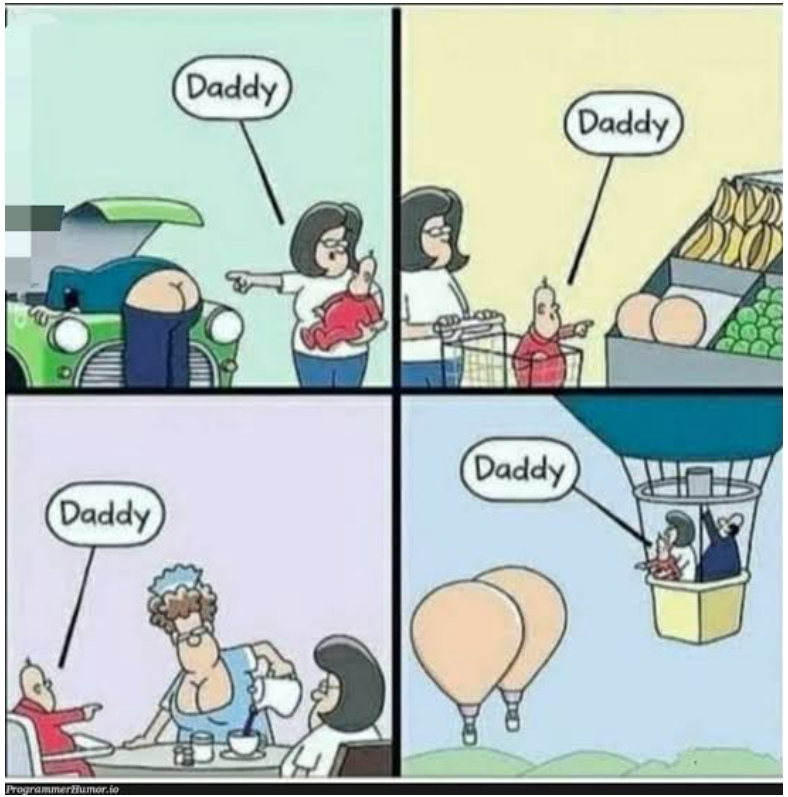


Will AI replace My job?

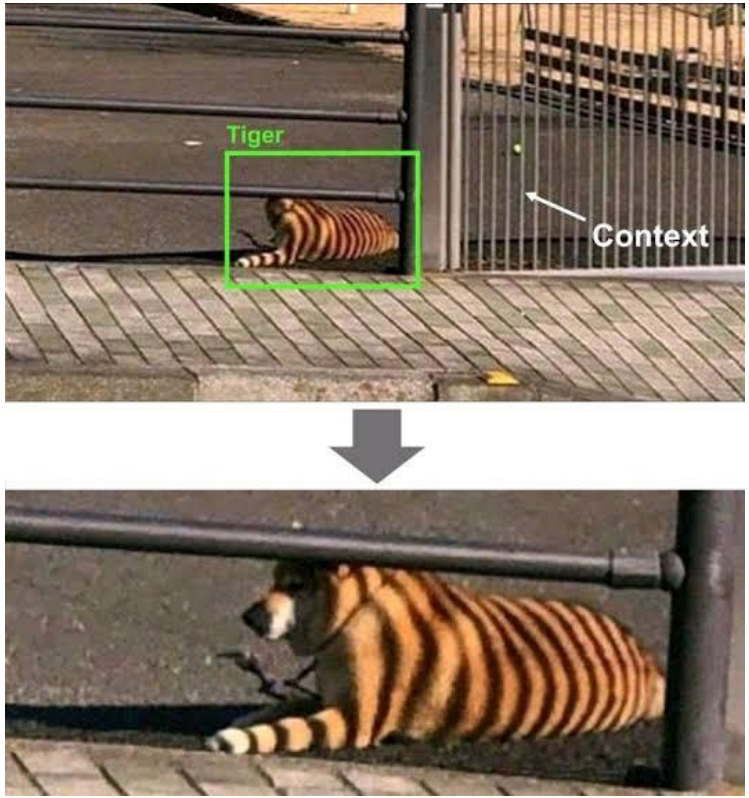


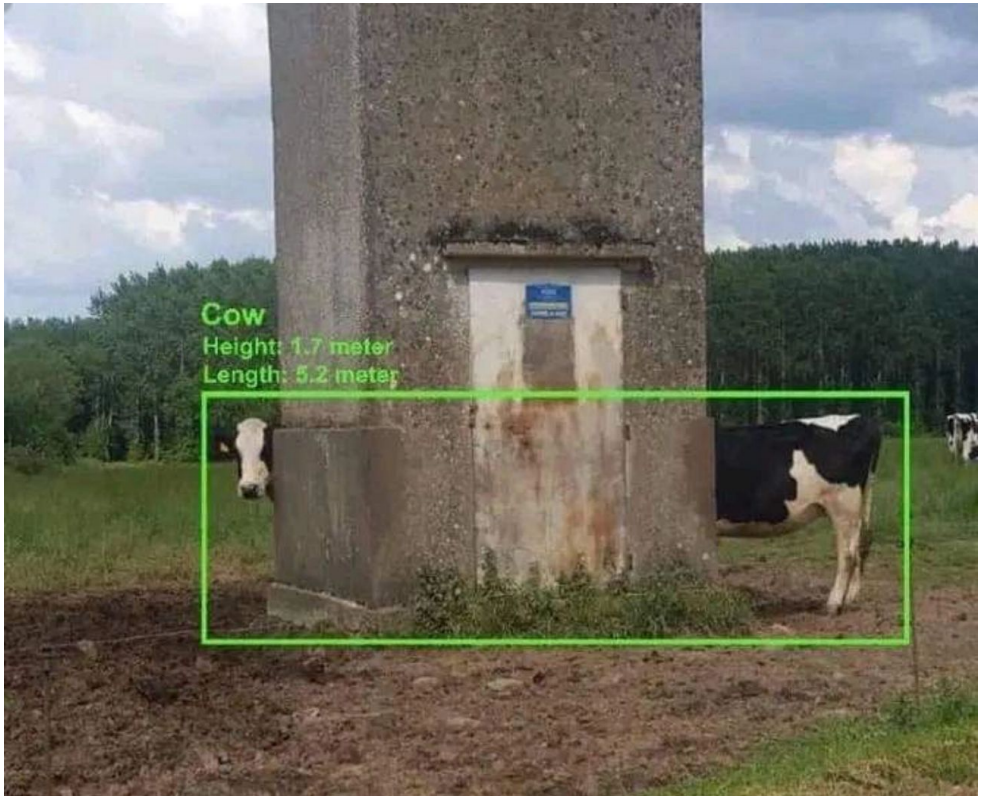
Challenges (Examples)



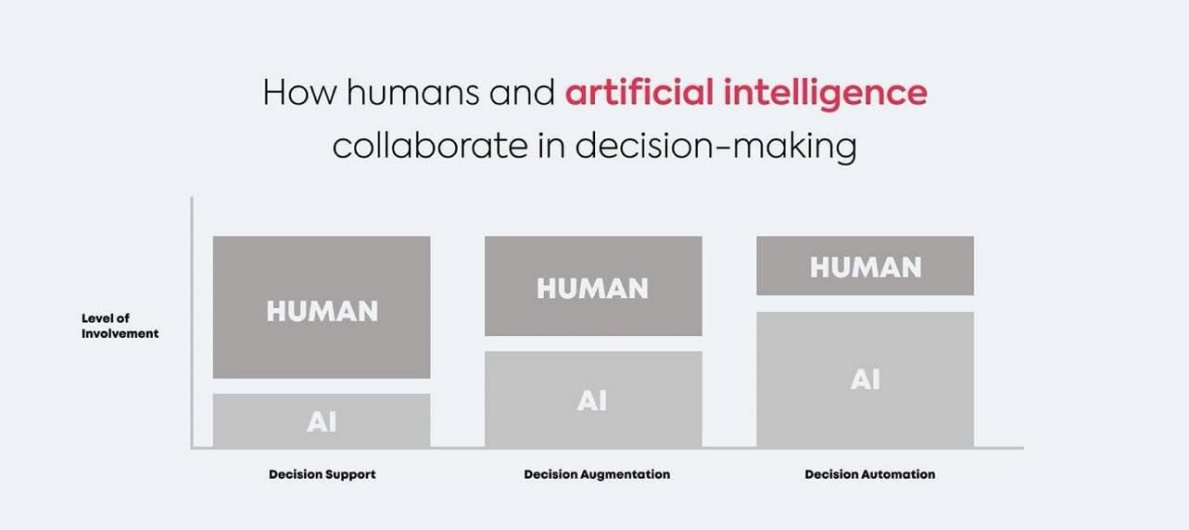


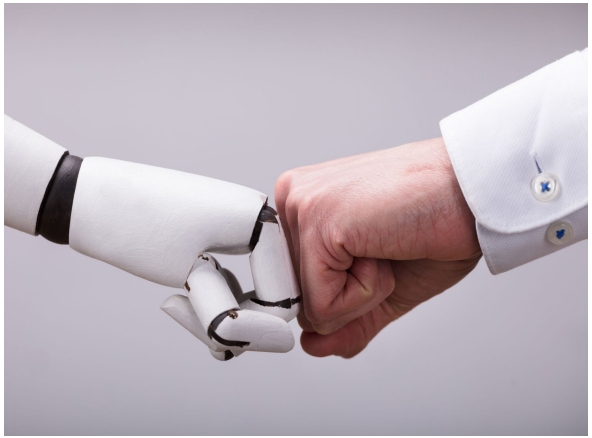
Challenges (Examples)





AI can assist in healthcare setting





Thank You