**API**

We are using Flask API to send and get data from the application.

Initially we are having two functions which both requires data from the user.



We specify the paths here.

Base\_path which contains unannotated folder, required csv files.

Classes\_List.csv which contains class names. This is required when we load the model.

New\_Images\_List.csv can be a csv file we use to add user added images.

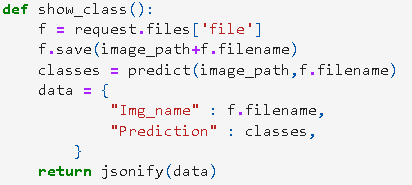
Information contained on that csv is dependent on the application as well.

We load the model weights using detecto load function.

The model weights are of extension ‘.pth’



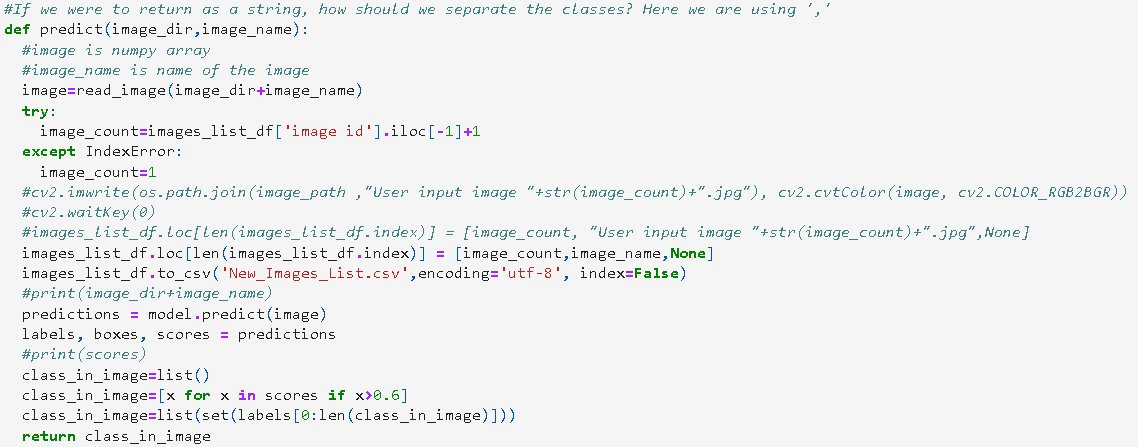
The first function requires an image from the user so that we can send back our prediction.



request.files -> requests for the file which here is the image

Then we save the image in image\_path which is unannotated folder.

Then we go on to predict function:



We read the image using detecto’s read\_image function. Image count is just id off of the csv file.

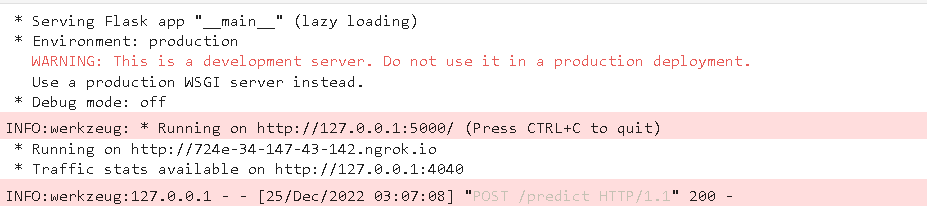
Model.predict(image) gives the labels, boxes, scores of the predictions.

Labels being class names, boxes being bounding box in array format, scores being the confidence of prediction.

Here we are having a threshold for score which is 60% [0.6]. We return unique classes having greater then 60% confidence.



The second function is to get actual classes if the prediction is wrong. The actual class will be input by the user. In the application, we will ask to the user what class was expected if the prediction is wrong. We save that in the csv file so that the annotators know what class the image is of.



When run on colab, ngrok creates a server each time it’s run. We test the URL generated when we run the code. The API Endpoints can be tested on postman.