

1. Problem statement:

The aim of this project is to develop a webapp that allows a user to enter their log-in information for a metro card system, and then lets the user upload a picture of an item of trash to determine if it is recyclable. If it is, then the user will be prompted to recycle the item in the correct receptacle and will accrue points on their metro card. If the item cannot be recycled, then an informative message will be displayed to the user regarding recyclable material types and the benefits of recycling.

2. Data preprocessing:

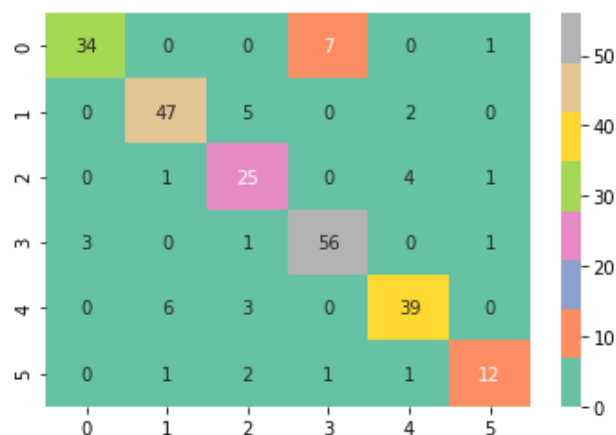
I am working with [this](#) garbage classification dataset with 6 classifications: cardboard (393), glass (491), metal (400), paper (584), plastic (482), and trash (127). To extract information from the image datasets and clean up the data before sending it to the model, I will preprocess the datasets by resizing the images to establish a base size for images that will be uploaded to the webapp, converting all colored images to grayscale to reduce computation complexity, removing background noise from the images, and augmenting the dataset with altered versions of the existing images.

3. Machine learning model:

I have changed my model from convolutional neural networks to transfer learning because it has been developed for image classification and I can avoid having to build my own model. I used a logistic regression model using 90% of the total trash classification images for my training data.

4. Preliminary Results:

This is the confusion matrix used to evaluate the performance of the classification.



5. Next Steps:

As of now, my preprocessed images are not properly saving into my Google Drive, so I will work on remediating this issue. I ran the training code on the original images of the 6 trash classes, but after properly saving the preprocessed images, I will run the model and confusion metric on this training set, instead.

This training model yielded an accuracy score of 33.28% for Rank 1 and 39.53% for Rank 5, which is a bit low. This may be due to the fact that I was not able to run my training code properly on my preprocessed images, so I will observe changes in the accuracy scores after I resolve this problem.