

WRITTEN PROJECT REPORT

I. Suggested Cover Page:

Data Mining for Business (BUDT758T)

Project Title: __Waste Classification_____

Team Members: Adarsh Challa - 117394641,
Anshika Kapoor - 117444285,
George Puthean - 117447941,
Sowmya Murukuti - 118461809,
Varshitha Yalakatori - 118454573

ORIGINAL WORK STATEMENT

We the undersigned certify that the actual composition of this proposal was done by us and is original work.

	Typed Name	Signature
Contact Author	Adarsh Challa	<i>Adarsh Challa</i>
	Anshika Kapoor	<i>Anshika Kapoor</i>
	George Puthean	<i>George Puthean</i>
	Sowmya Murukuti	<i>Sowmya Murukuti</i>
	Varshitha Yalakatori	<i>Y.Varshitha</i>

II. Executive Summary

On an everyday basis we see different kinds of waste or garbage being segregated into containers in the form of recyclable, non-recyclable, dry, wet or metallic waste. Putting the right waste into the container requires very less effort but in contrast, separating back from a mix of waste requires more effort. In most cases this happens and it eventually increases the effort of segregation. Moreover, the prior effort of collecting the right waste in the right container also goes in vain. Our project will specifically focus on identifying the images of waste into the right class which increases the efficiency using machine learning techniques. This study can be used for environmental purposes and helps in reduction of global warming.

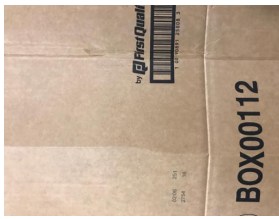
III. Data Description

- Data Source: Collected the data from Kaggle dataset. Attached the link here below:

<https://www.kaggle.com/asdasdasdasdas/garbage-classification?datasetId=81794&sortBy=dateRun&tab=bookmarked>

- The data used for classification is of garbage images.
- The waste classification dataset contains 6 classifications:
 1. Glass
 2. Metal
 3. Plastic
 4. Trash
 5. Paper
 6. Cardboard
 - We have collected images of garbage, pre-processed and performed exploratory analysis.
 - Data is pre-processed by transforming all the images into a single size and converting the raw .jpg file to machine-readable form. Data pre-processing checks whether the dataset is consistent by having the same size and scale.
 - With this step, the overall efficiency and accuracy of the model can be improved.
 - So the steps involved are:
 1. Gathering the images and dividing them Into different classes.
 2. Resizing all the images into standard size 256 by 256.
 3. Reading the image and the label.
 4. And setting the batch size to 48 with 16 in each row.
 - We have a total of 2467 images. Out of 2527 images, 491 are glass, 584 are paper, 472 are plastic, 400 are metal, 393 are cardboard and 127 are trash.
 - Data is not balanced. Data consists of a different number of images of each classification.
 - Data is randomly split into Training, Validation and Test.
 - Training data has a total of 1900 images of 6 classes: 452 glass images, 373 paper images, 359 plastic images, 311 metal images, 305 cardboard images and 100 trash images.
- Sample Data:

Attached the images here for reference:



Cardboard



Glass



Metal



Plastic



Paper



Trash

- Data set of interest link:

Click here: [Waste Images Dataset](#)

III. Research Questions

A few questions we wanted to research on and find solutions to are listed below.

1. How many classes can we divide garbage found from a household to effectively classify waste?

To identify what kinds of garbage is found in a household on a day to day basis and to classify them into categories to help improve garbage disposal methods and help in the recycling process.

2. Can images of garbage be identified and divided into their respective classes using deep learning?

To identify if images collected are sufficient to help train models and be identified by their respective classes, and find libraries or transformation techniques that can be used for our images to be readable for training and improve the classification.

3. How to improve accuracy for waste classification models?

To perform classification using the most optimal model and achieving the best accuracy.

IV. Methodology

Test Train Split: 80:20

Model Used: Resnet34

Framework Used: Python

Convolutional layers have an advantage over fully connected layers as they reflect a restricted range of features. A neuron in a completely connected layer is connected to every neuron in the layer before it, and as a result, it can change if any of the neurons in the layer before it changes.

We decided to choose convolutional layers. The convolution layers are mostly filters and follow two simple rules:

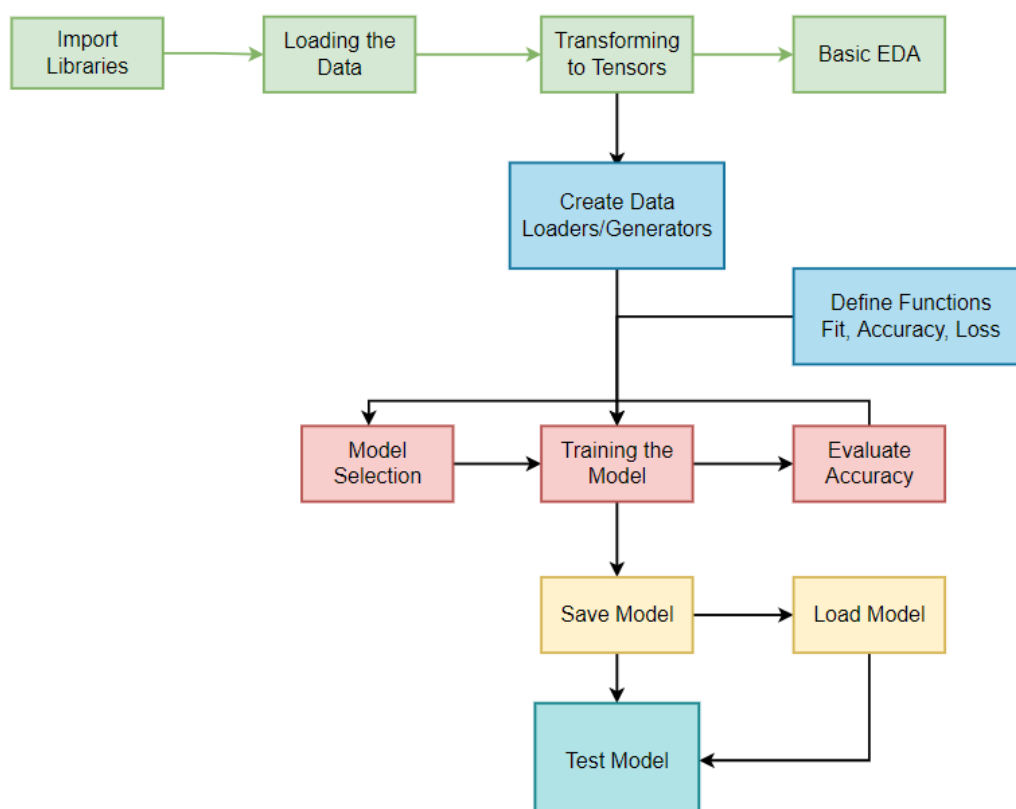
1. Same output feature map and the layers have the same number of filters.
2. If the size of the features map is halved, the number of filters is doubled to preserve the time complexity of each layer.

Plain Network: The plain baselines are mainly inspired by the philosophy of VGG nets. It is worth noticing that the ResNet model has fewer filters and lower complexity than VGG nets.

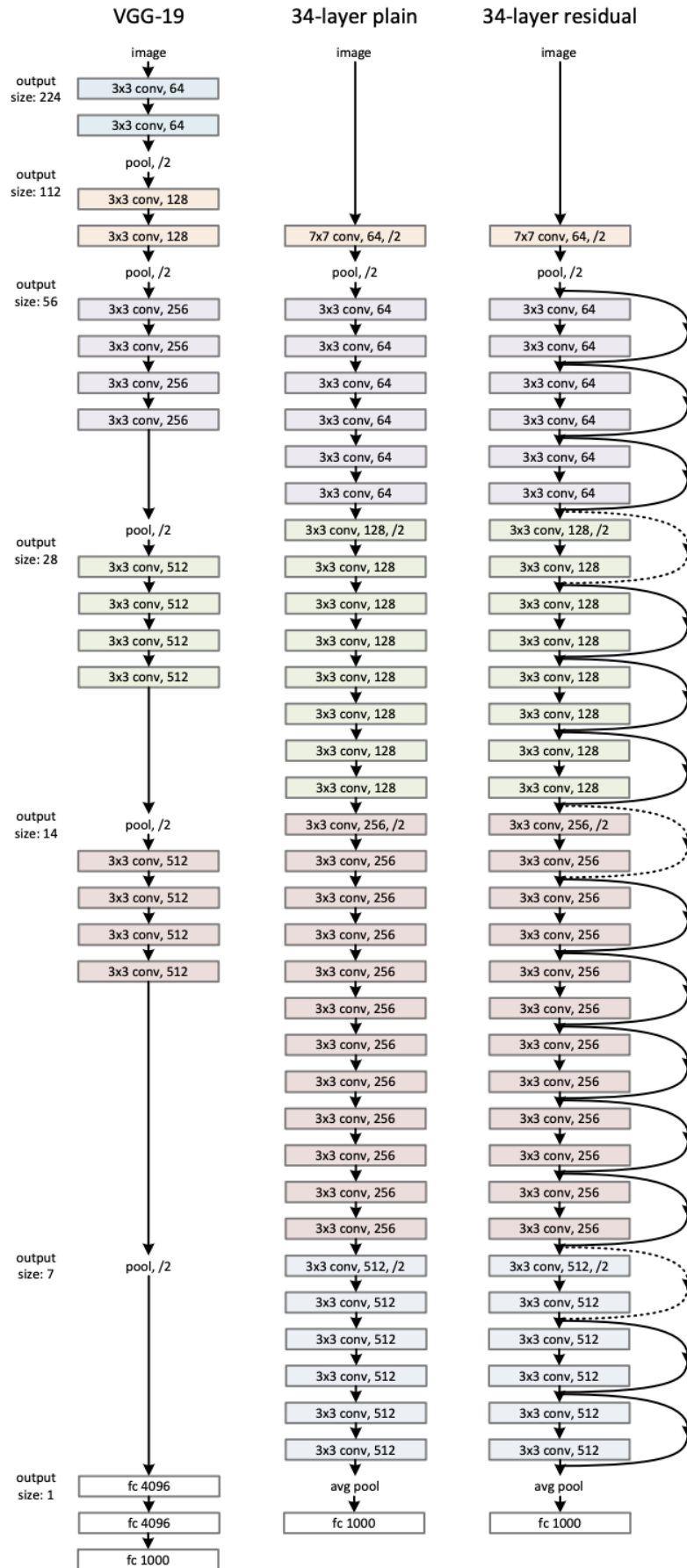
Residual Network: Based on the above plain network, a shortcut connection is inserted which turns the network into its counterpart residual version. The training times are much higher on 50 layers. The 34 layer network is just the subspace in the 50 layer network, and it still performs better. ResNet outperforms with a significant margin in case the network is deeper.

This is the reason why we chose ResNet34 over plain and ResNet50. You can find results on the ResNet50 and ResNet34 below.

Process Flow Diagram: The Diagram below shows how the code executed for the purpose of this project.

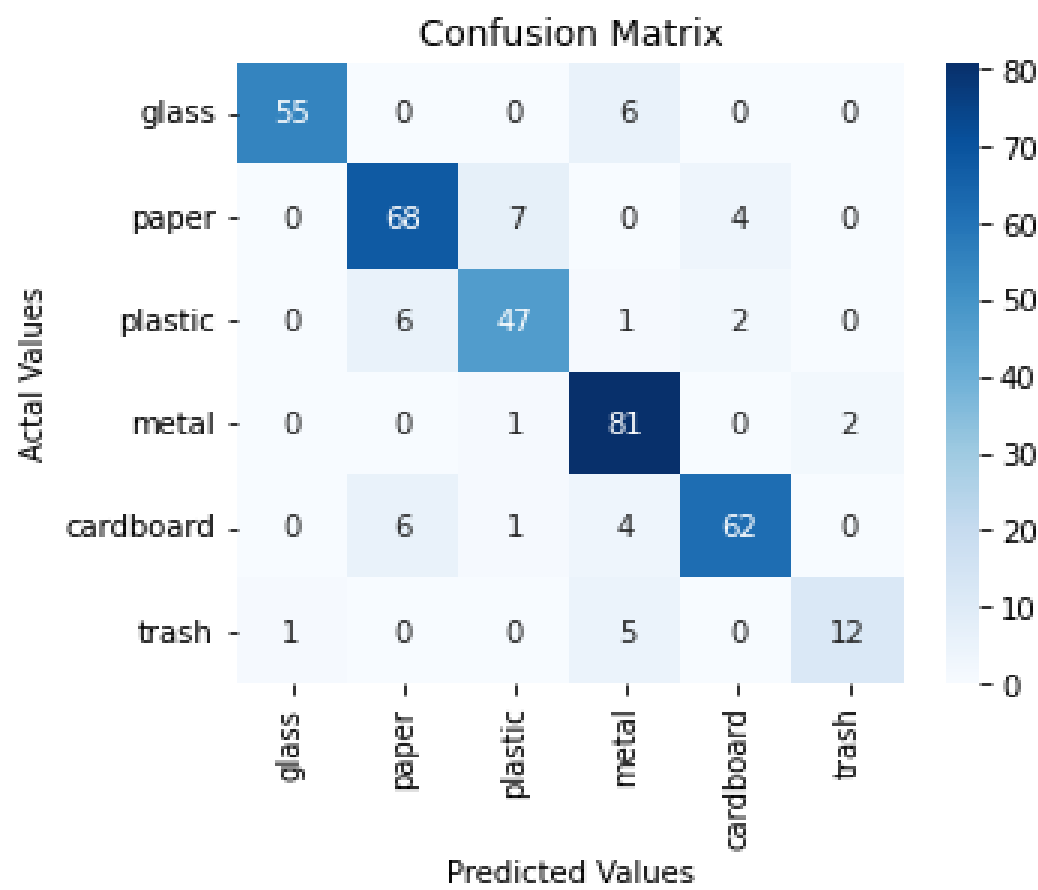


The ResNet Architecture:



V. Results and Findings

After running our model we were able to achieve an accuracy of **89%**. The below graph represents the confusion matrix of all the 6 classes.



The Recall and Precision are calculated as From the confusion matrix we were able to calculate recall and precision as below:

Class	Precision	Recall
Glass	90.2%	98.2%
Paper	86.1%	85%
Plastic	83.9%	83.9%

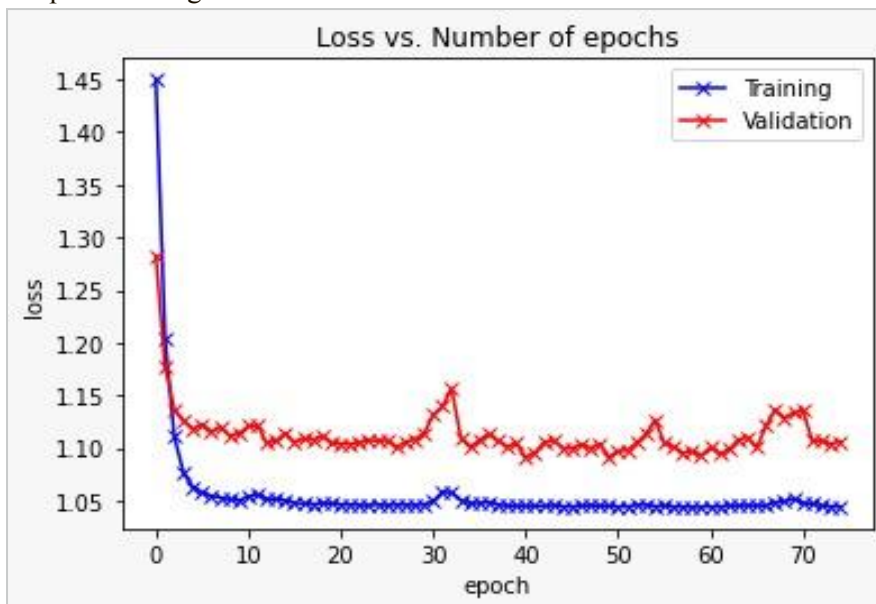
Metal	96.4%	83.5%
Cardboard	84.9%	91.2%
Trash	66.6%	84.7%

- From the table we observe that the highest precision results were found for Metal, which shows reliability of metal images.
- The lowest precision value is found for Trash as we observe that comparatively the number of images of trash are also low.
- Glass has the highest recall value of 98.2% and the least recall id for Metal of 83.5%.

Important Graphs:

1. Loss vs Number of Epochs

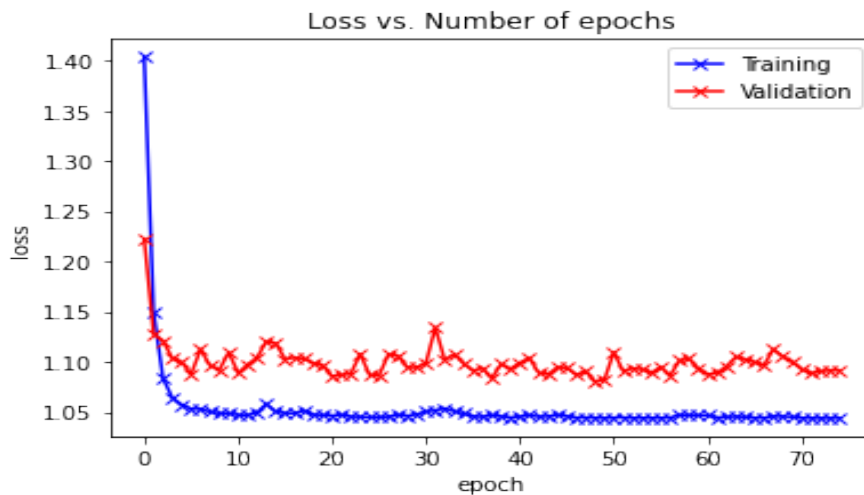
Graph according to ResNet34 Model:



Points to note here:

1. As the number of epochs increased, both training and validation loss values decreased.
2. After the 6th epoch, the training loss takes the value of 1.05 and the trend continues till the 66th epoch. After that, it starts decreasing again.
3. After the 5th epoch, the validation loss takes the value of 1.11 and the trend continues till around the 30th epoch. After that, it faces few fluctuations and finally starts decreasing further after the 70th epoch.

Graph according to ResNet50 Model:

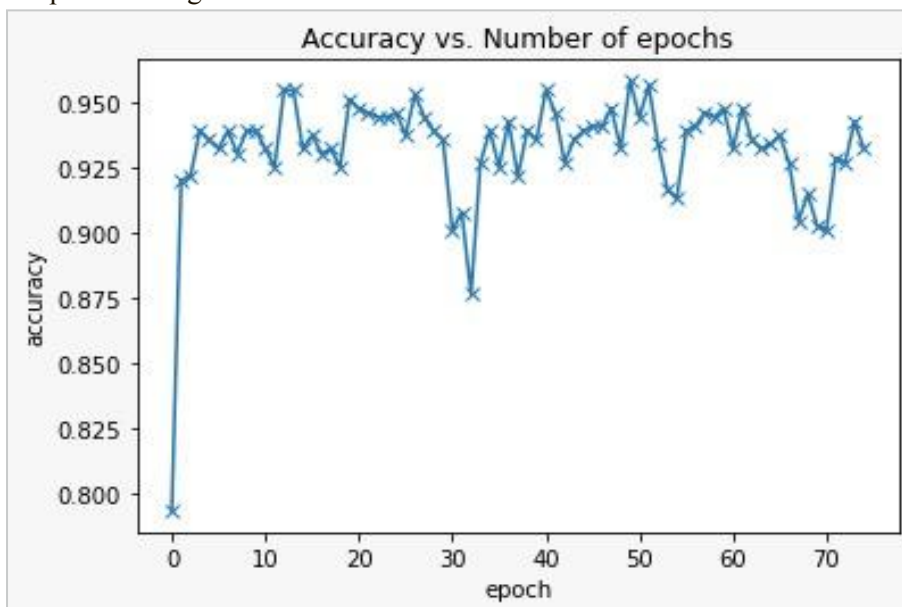


Comparison between ResNet34 and ResNet50:

1. Starting loss values for both training and validation are lower in case of ResNet50.
2. After the 6th epoch, training loss values for both ResNet34 and ResNet50 seem to behave in a similar manner.
3. After the 5th epoch, validation loss values for both ResNet34 and ResNet50 seem to behave in a similar manner.

2. Accuracy vs Number of Epochs (Validation data)

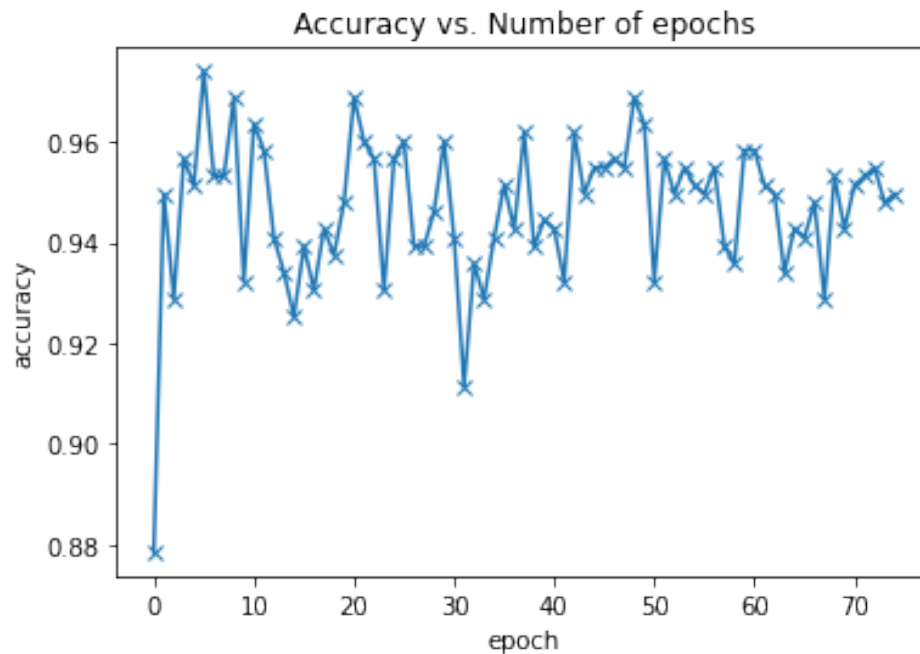
Graph according to ResNet34 Model:



Points to note here:

1. The starting accuracy at 0th epoch is less than 80%.
2. After the 2nd epoch, accuracy increased drastically with a difference of more than 10%.
3. There is a sudden drop in the accuracy level around the 30th and 32nd epoch.
4. The model is run for a total of 75 epochs, so the final accuracy at the 75th epoch is 93.2%.

Graph according to ResNet50 Model:

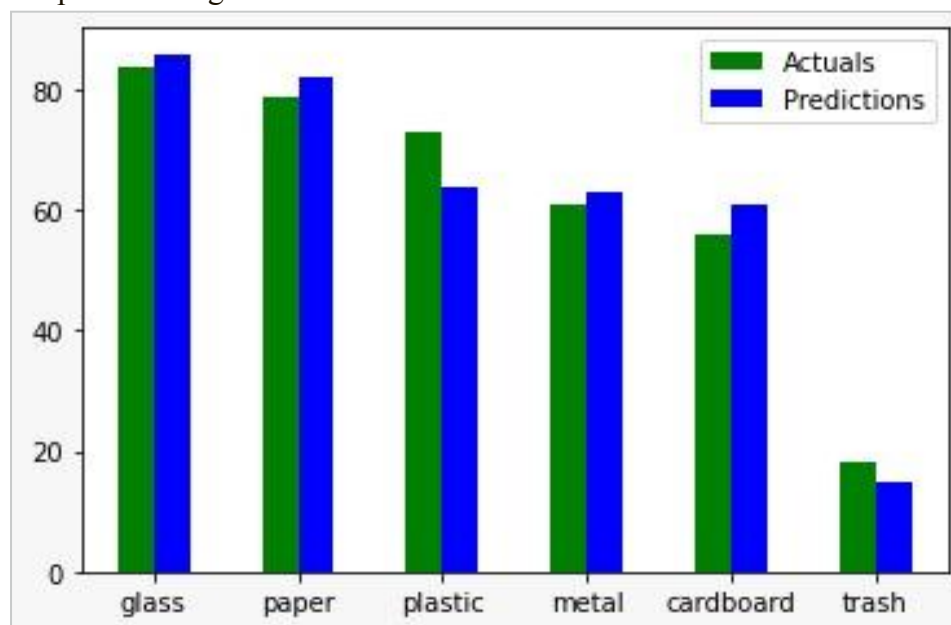


Comparison between ResNet34 and ResNet50:

1. In general, ResNet50 gives better accuracies than ResNet34.
2. In case of ResNet50, the final accuracy at the 75th epoch is around 95% which is greater than the final accuracy in case of ResNet34.
3. Both ResNet34 and ResNet50 see a sudden drop in accuracy around the 30th and 32nd epoch. However, ResNet50 has more variation in accuracy throughout the different epochs as compared to the ResNet34 model.

3. Actuals vs Predictions (Validation data)

Graph according to ResNet34 Model:



Points to note here:

1. In the case of Plastic and Trash classes, the actual value is greater than the prediction value.
2. Overall throughout all the 6 classes, there is a slight difference between the actual value and prediction value, except for Plastic.
3. In the case of Plastic, there is a difference of about 10 images between actual and prediction values.

VI. Conclusion

We were able to find an effective method which can classify the images of waste into 6 categories with 89% accuracy. This project can help reduce the pollution caused by the mixture of different categories of waste. The model chosen was Resnet34 and the coding environment was PyTorch in Python where Resnet34 performed efficiently. The method in this project was chosen considering both the training time and accuracy.

Future Scope:

1. Model accuracy can be improved by using more image transformation techniques.
2. An intelligent waste management system using a web application or a mobile application can be created wherein the applications can smartly classify garbage images and thus help protect the environment.
3. Train the model using balanced data, i.e, use an equal number of images for each class of garbage.

VII. Appendix

Reference Links:

<https://neurohive.io/en/popular-networks/resnet>

<https://stats.stackexchange.com/>