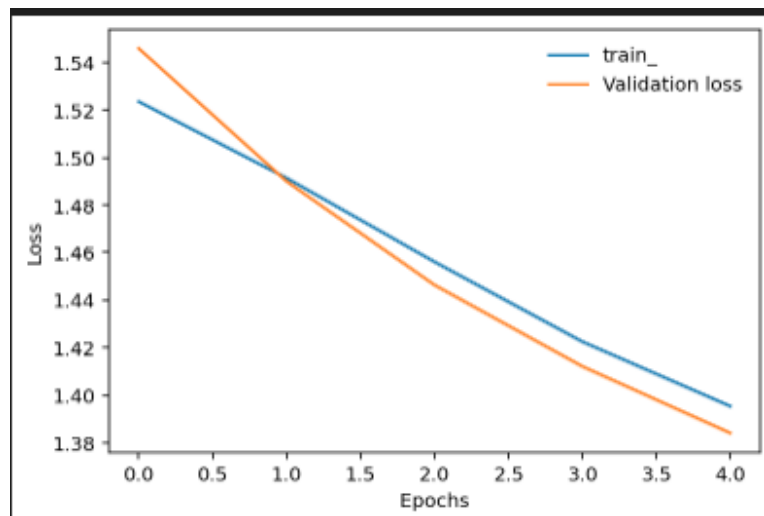


The dataset which was used for this assignment is APOTS Kaggle dataset. This data consists of 5 different classes of diabetes blindness in retina.

For this task, I've used a standard ResNet model and transfer learning to do image classification. The training is done using Pytorch lib and can be run both on CPU or GPU for acceleration.

There is a python notebook file ([Ret classification resnet.ipynb](#)) which run the whole process for data loading, preprocessing, model preparation and training/testing.

Some of the training/test output snapshots can be seen below which content loss and accuracy values:



```
train_losses, valid_losses, acc = train_and_test(10)
```

✓ 28m 2.8s

Model Training started.....

epoch 1 batch 10 completed

validation started for 1

Epoch: 1/5.. Training Loss: 1.524.. Valid Loss: 1.546.. Valid Accuracy: 0.351

Validation loss decreased (inf --> 1.545953). Saving model ...

epoch 2 batch 10 completed

validation started for 2

Epoch: 2/5.. Training Loss: 1.491.. Valid Loss: 1.490.. Valid Accuracy: 0.525

Validation loss decreased (1.545953 --> 1.489982). Saving model ...

epoch 3 batch 10 completed

validation started for 3

Epoch: 3/5.. Training Loss: 1.456.. Valid Loss: 1.446.. Valid Accuracy: 0.568

Validation loss decreased (1.489982 --> 1.446341). Saving model ...

epoch 4 batch 10 completed

validation started for 4

Epoch: 4/5.. Training Loss: 1.422.. Valid Loss: 1.412.. Valid Accuracy: 0.595

Validation loss decreased (1.446341 --> 1.412061). Saving model ...

epoch 5 batch 10 completed

validation started for 5

Epoch: 5/5.. Training Loss: 1.395.. Valid Loss: 1.384.. Valid Accuracy: 0.611

Validation loss decreased (1.412061 --> 1.383991). Saving model ...

Training Completed Successfully !

The ResNet model, specially larger ones can show promising results on this data after even few epochs of training.