Checkpoint 1: Running your own CNN, Show the augmented images

1. Architecture for CNN

In the CNN architecture I am using sequential model and stacking convolution and pooling layer along with the dropout, I am doing this step by three different number of Conv2d filters to get more features of the picture. Then after passing the image to feature extraction I am passing it through to the dense layer. In the dense layer I am taking 512 as number of hidden layers and at last I have taken one more dense layer with the number as that of class that is 10 to classify the image probability.

1. Training Process
   1. Describe the methods used for splitting the training and test data

I used the validation split to split the dataset I have used flow\_from\_dataframe function. So, we have this subset argument which takes either training or validation.

I split the train set to 2 sets one for training and the other for validation just by specifying the argument validation\_split = 0.10 which splits the dataset into to 2 sets where the validation set will have 10% of the total images.

* 1. Approach for choosing hyper parameter

I tried the Grid Search technique . Grid search trains the algorithm for all sscombinations by using the two set of hyper parameters. but I have used only one i.e Number of layers , I have not specified Learning rate

1. Techniques to improve model accuracy like Data Argumentation etc.

I have used dropout as well has done data preprocessing by using preprocessing function of keras to improve model accuracy .

I have also use data Augmentation to improve more accuracy on Data.

1. Performance of Model on a Training and Test Data in terms of Chosen Metric

loss: 0.7910 - accuracy: 0.7298 - val\_loss: 0.7181 - val\_accuracy: 0.7602 on train Data performance

0. 0.7602 Test Data performance

Checkpoint 2: Build a Transfer Learning model that predicts the object present in a given image

1. Which Pretrained model used and Why?

I have used VGG16 model because

Its capability of classifying objects in photographs, is that the model weights are freely available and can be loaded and used in your own models and applications.

Architecture used for Transfer Learning Model

VGG16, In this approach I am loading already a pretrained model VGG16 which is trained on imagenet, In this process I am making a copy of entire feature extraction layer, and freezing the dense layer because it is used to classify the image and replacing VGG16 with my dense layer of unit 10.

1. Training Process
   1. Describe the methods used for splitting the training and test data

I used the validation split to split the dataset I have used flow\_from\_dataframe function. So, we have this subset argument which takes either training or validation.

I split the train set to 2 sets one for training and the other for validation just by specifying the argument validation\_split=0.10 which splits the dataset into to 2 sets where the validation set will have 10% of the total images.

* 1. Approach for choosing hyperparameter

I tried the Grid Search technique . Grid search trains the algorithm for all combinations by using the two set of hyper parameters. but I have used only one i.e Number of layers , I have not specified Learning rate

1. Techniques to improve model accuracy like Data Argumentation etc.
2. I have used dropout as well has done data preprocessing by using preprocessing function of keras to improve model accuracy .

I have also data Augmentation to improve more accuracy on Data

1. Performance of Model on a Training and Test Data in terms of Chosen Metric

Performance on Train Data--loss: 0.7223 - accuracy: 0.7834 - val\_loss: 0.8231 - val\_accuracy: 0.7680

Performance on Test Data-- 0.7680000066757202