Deep Learning(AI2100,AI5100)

Kaggle Challenge

Instructor: Dr. Konda Reddy Mopuri

CS21BTECH11014 (Rutheesh Reddy)

CS21BTECH11017 (Harsha Vardhan Reddy)

Goal: Image Classification

- Dataset: ImageNet like dataset with trainset of 65,000 images and testset of 38,000 images
- Dataset is classified into 50 classes

- Loss Criterion : We used Cross-Entropy Loss
- Learning Rate = 0.001
- Optimizer = Adam
- Batch Size = 64

Attempt 1- Run some SOTA Models

We picked some SOTA models for ImageNet dataset and started running for our data. We observed the following results

Model	Epochs	Public Score
ResNet50	10	0.42
ResNet50 with Augmentation	10	0.39
ResNet50 with Augmentation	15	0.36
EfficientNet	15	0.36
EfficientNet	10	0.40
convnext	10	0.54
Convnext with augmentation	10	0.54
Resnet34 with augmentation	15	0.35
ViT	20	--

Augmentations We Used

```python
transformation = transforms.Compose([
    #resize to a specific size
    transforms.Resize((224,224)),
    # Randomly augment the image data
        # Random horizontal flip
    transforms.RandomHorizontalFlip(0.5),
#        # Random vertical flip
    transforms.RandomVerticalFlip(0.3),
    # transform to tensors
    transforms.ToTensor(),
    # Normalize the pixel values (in R, G, and B channels)
    transforms.Normalize(mean=[0.5, 0.5, 0.5], std=[0.5, 0.5, 0.5])
])
```

Attempt 2 – Create a Validation Set

- Then we thought of creating a validation set(10% of train set) from train set and evaluate models on this set, we believed that this validation set is similar to test data and hence acts as a good measure for the private score.

Model	Epochs	Validation accuracy	Public Score
ResNet50	15	0.72	0.37
ResNet34	15	0.69	0.35
Convnext	20	0.68	0.53
densenet121	15	0.74	0.40

Attempt 3 – Go Ensemble

- Then thought of using ensemble models. Because ensemble models are more robust and have better generalization performance. We trained a gradientboost ensemble with 2 base resnet learners, even then we got a validation accuracy of 74% and 46% public score.

- We have saved model weights of different models we tried earlier, we could have used voting algorithm.

Some Doubts 🤔

- Densenet model had validation accuracy of 74% yet its private score was 10% which is same as random baseline predictor?

- ConvNext has performed relatively better than other models, it has lesser validation accuracy, why?