Cifar-10 with ResNet50 Modeling

Tianyi Liang

Pure ResNet50 (Modify output layer to SoftMax 10) - 5 Epochs:

```
1407/1407 [==============================] - 323s 229ms/step - loss: 2.3738 - accuracy: 0.2708 - val_loss: 1.8962 - val_accuracy: 0.3276
Epoch 2/5
1407/1407 [==============================] - 322s 229ms/step - loss: 2.2576 - accuracy: 0.2846 - val_loss: 2.6578 - val_accuracy: 0.2084
Epoch 3/5
1407/1407 [==============================] - 321s 228ms/step - loss: 2.3133 - accuracy: 0.2504 - val_loss: 12.5751 - val_accuracy: 0.2676
Epoch 4/5
1407/1407 [==============================] - 320s 228ms/step - loss: 1.8977 - accuracy: 0.3641 - val_loss: 3.8347 - val_accuracy: 0.1540
Epoch 5/5
1407/1407 [==============================] - 319s 227ms/step - loss: 1.9654 - accuracy: 0.3689 - val_loss: 3.3166 - val_accuracy: 0.3242
```

*Train accuracy: **36.89%** - Validation accuracy: **32.42%***

Classifier (With ResNet50 Embedded) – 15 Epochs:

First, there is an Up Sampling layer to turn (32, 32, 3) to (224, 224, 3) by multiplying 7. The output is sent to the pre-trained ResNet50.

I added several Dense layers with Glorot Normal as my kernel initializer. After Pooling-Flatten-BatchNorm the result from ResNet50, I got my output L1. I added two 2048, one 1024, and one 512 layers named D1, … , D4 proceeding with a 10 Classes Dense layer as the output layer. I also send the L1 directly to D4 by concatenate the output from layer D3 with L1. Batch Norm is added after D1, D2, and D3. There are also three Dropout layers with 0.3 drop rate after D1, D2, D3. I chose SGD as the optimzier

```
Epoch 10/15
704/704 [==============================] - 467s 663ms/step - loss: 0.0081 - accuracy: 0.9979 - val_loss: 0.2217 - val_accuracy: 0.9476
Epoch 11/15
704/704 [==============================] - 467s 663ms/step - loss: 0.0042 - accuracy: 0.9989 - val_loss: 0.2336 - val_accuracy: 0.9452
Epoch 12/15
704/704 [==============================] - 467s 663ms/step - loss: 0.0027 - accuracy: 0.9995 - val_loss: 0.2300 - val_accuracy: 0.9472
Epoch 13/15
704/704 [==============================] - 467s 664ms/step - loss: 0.0018 - accuracy: 0.9998 - val_loss: 0.2136 - val_accuracy: 0.9470
Epoch 14/15
704/704 [==============================] - 467s 663ms/step - loss: 0.0017 - accuracy: 0.9996 - val_loss: 0.2425 - val_accuracy: 0.9430
Epoch 15/15
704/704 [==============================] - 467s 663ms/step - loss: 0.0012 - accuracy: 0.9999 - val_loss: 0.2337 - val_accuracy: 0.9446
```

*Training accuracy: **99.99%** - Validation accuracy **94.46%***

The following figure is my model's structure.