MP4 Report

Shidi Zhao(szhao41)

Part1

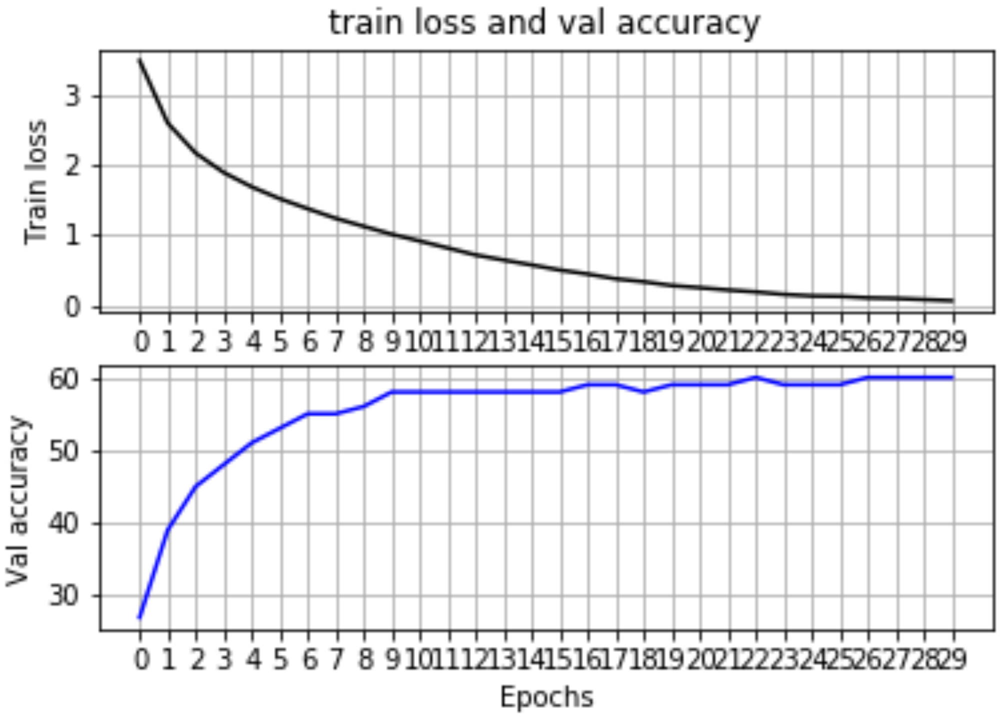
a. **The name under which you submitted on Kaggle.**
   Shidi Zhao
b. **Best accuracy (should match your accuracy on Kaggle).**
   0.62600
c. **Table defining your final architecture similar to the image above.**

Layer No.	Layer Type	Kernel size(for conv layers)	Input | Output dimension	Input | Output Channels(for conv layers)
1	Conv2d	3	32| 30	3|64
2	relu	-	30|30	-
3	Maxpool2d	2	15|15	-
4	Conv2d	3	30| 28	64|128
5	relu	-	28|28	-
6	Maxpool2d	2	14|14	-
7	Conv2d	3	14|12	128|256
8	relu	-	12|12	-
9	Maxpool2d	2	6|6	-
10	Conv2d	3	6|4	256|512
11	relu	-	4|4	-
12	Conv2d	3	4|2	512|512
13	relu	-	-	-
14	Maxpool2d	2	1|1	512|512
15	Linear	-	512|256	-
16	relu	-	256|256	-
17	Linear	-	256|100	-

d. Factors which helped improve your model performance. Explain each factor in 2-3 lines.

Maxpool2d layer helps to improve the performance. And adding Conv2d layers would get more features, that also improve performance. Besides, increase EPOCHS may also works.

e. Final architecture's plot for training loss and validation accuracy. This would have been auto-generated by the notebook.



f. Ablation study to validate the above choices, i.e., a comparison of performance for two variants of a model, one with and one without a certain feature or implementation choice.

I try two models. The first model has 5 conv layers and each of it has a maxpool layer, totally 5 maxpool layers; the second one is a more complex network model which has 13 conv layers, but only has 4 maxpool layers. The accuracy of first model is 0.623, the second one is 0.626. We can see from the models, pooling layer can increase accuracy if the model didn't overfit before adding pooling layers, sometimes more effective than adding conv layers. Because pooling reduces the number of activations and design how your neural network will learn. When adding a max pool, it forces the filters of a convolution layer below to search for a feature, add small translational invariance.

Part2

1. **Report the train and test accuracy achieved by using the ResNet as a fixed feature extractor vs. fine-tuning the whole network.**

   1) fixed feature extractor:
   train accuracy:
   Accuracy 0.0147

   test accuracy:
   Test Accuracy 0.0053

   2) fine-tuning:
   train accuracy:
   Accuracy 0.0080

   test accuracy:
   Test Accuracy 0.0073

2. **Report any hyperparameter settings you used (batch_size, learning_rate, resnet_last_only, num_epochs).**

num_epochs = 20
batch_size = 10
learning_rate = 0.1
resnet_last_only = True (for only last layer)
resnet_last_only = False (fine-tuning the whole network)