DOGS VS. CATS REDUX: KERNELS EDITION

林岳

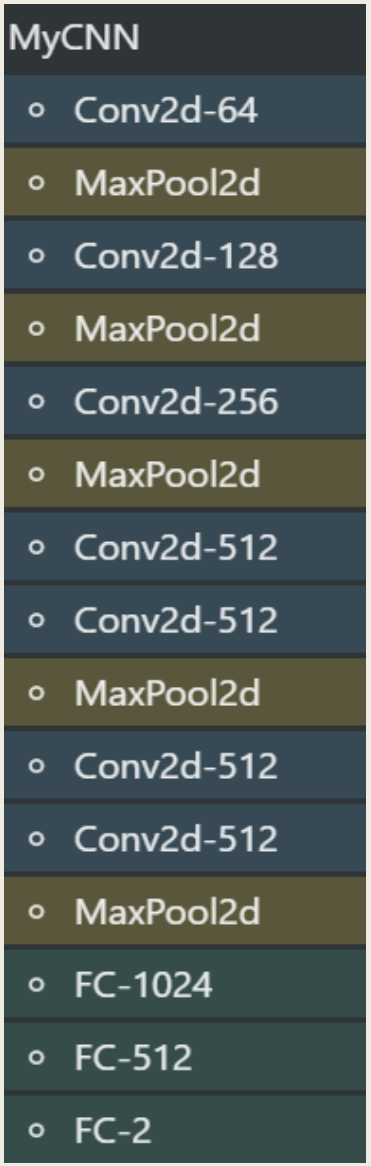
資料處理

- Transforms
  - *transforms.Resize((224,224))*
  - *transforms.ToTensor()*
  - *transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])*

- Train:Valid = 20000 : 5000 (8:2) (by random_split)

Model

- Pre-trained model
  - *VGG-19*
  - *ResNet-50*
  - *Densenet-161*
  - *GoogleNet-1*
  - *Inception-3*
- MyCNN
- Softmax Layer
- 更新最後一層、最後兩層或全部



Training Skill

- nn.BCELoss()

$$\text{LogLoss} = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

- Optimizer : SGD (momentum=0.9) or Adam
- Threshold on output

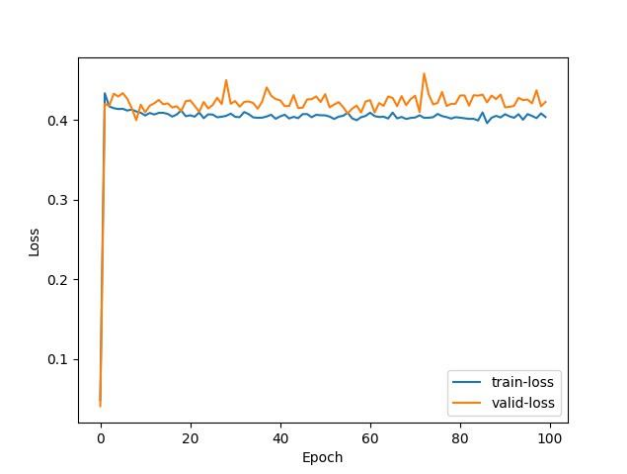
$$Thres(x) = \begin{cases} 0, & x < 1 - T \\ x, & 1 - T \leq x < T \\ 1, & x \geq T \end{cases}$$

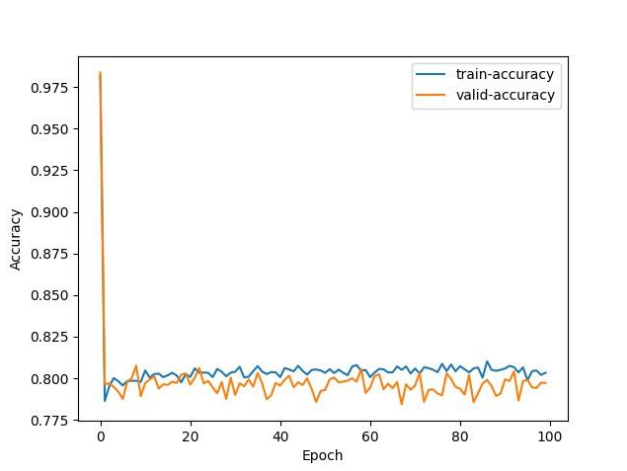
- LR Scheduler by ExponentialLR
- LR Scheduler every epoch or iteration
- Store model at the best valid loss

Experimental Results

- VGG19, lr=1e-3, batch size=32

層數	Optim	Epoch	Score	Loss	Acc.
兩層	SGD	400	0.07558	0.0443 0.8635	0.9848 0.8124
一層	SGD	100	0.06311	0.4037 0.4228	0.8034 0.7972
一層	Adam	100	0.09563	0.4353 0.4464	0.7935 0.7928



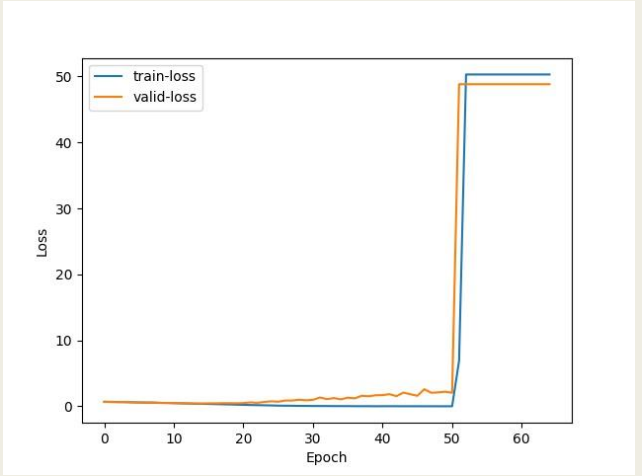


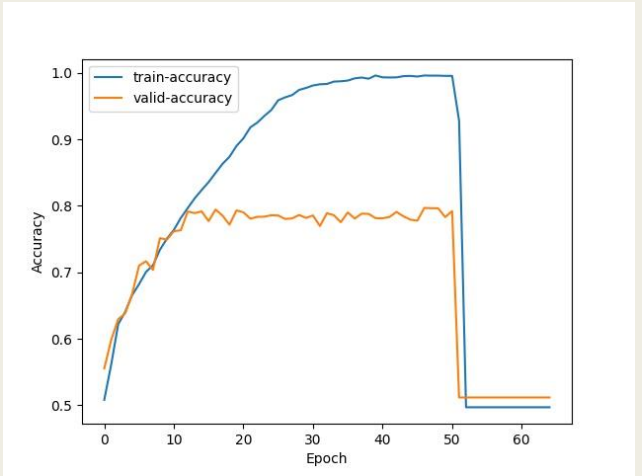
VGG, SGD, epochs=100

Experimental Results

- Update last layer, SGD, lr=1e-3, batch size=32, epochs=1

Model	Score	Loss (T/V)	Acc.
VGG	0.06514	0.0497 0.0429	0.9809 0.9850
DenseNet	0.60971	0.6479 0.6107	0.6210 0.6664
ResNet	0.07436	0.1050 0.0668	0.9641 0.9766
GoogleNet[1]	0.71033	0.6631 0.6459	0.6155 0.6404
Inception[1]	0.70022	0.7014 0.6969	0.5031 0.5110
MyCNN[2]	0.33951	0.0129 2.0610	0.9953 0.7920





1. Epochs=3, lr=1e-5, batch size=1
2. Epochs=65, lr=0.01

Experimental Results

- VGG, SGD, epochs=1, batch size=1

Lr	Update all Score	Update last layer Score
1e-3	x	0.06514*
1e-4	0.09712	0.06791
1e-5	0.07101	0.06121
1e-6	0.06209	0.091
5e-7	0.07324	x
1e-7	0.22005	x

*batch size=32

- VGG_1, SGD, lr=1e-3, epochs=1, batch size=32

T	Score
X	0.06514
0.8	0.26399
0.9	0.23134

$$Thres(x) = \begin{cases} 0, & x < 1 - T \\ x, & 1 - T \leq x < T \\ 1, & x \geq T \end{cases}$$

Experimental Results

- VGG_all, SGD, lr=1e-5, batch size=1, epochs=1

Update	Iter.	Grayscale	GaussianBlur	Score
all	x			0.07101
all	0.9998			0.06244
all	0.99985			0.06038[1] (0.17247[2])
one	0.99985			0.06839
all	0.99985	v	v	0.08187
all	0.99985		v	0.07512
all	0.9999			0.06332

- $10^{-5} * 0.99980^{20000} = 1.831 * 10^{-7}$
- $10^{-5} * 0.99985^{20000} = 4.978 * 10^{-7}$
- $10^{-5} * 0.99990^{20000} = 1.353 * 10^{-6}$

1. 重複執行，score=0.06281
2. threshold=0.99

Conclusion

- 最佳的結果為 0.06038

Model	VGG-19 (pre-trained)	Learning rate	1e-5
更新參數	全部	scheduler	ExponentialLR
Batch size	1	Sched. factor	0.99985
Epochs	1	Sched. time	Every iteration
Optimizer	SGD (m=0.9)	Output threshold	False

- Open Source on Github: https://github.com/lyjeff/dogs-vs-cats

Feature

- 完成可繼續訓練的模型
- 增加模型複雜度
- 擴充資料集，eg. 旋轉、放大縮小、裁減...
- 不要用 softmax，避免過度自信的問題，以得到更準確的機率

Reference

- Pytorch->torchvision.models: https://pytorch.org/vision/stable/models.html
- Pytorch->FINETUNING TORCHVISION MODELS: https://pytorch.org/tutorials/beginner/finetuning_torchvision_models_tutorial.html
- ML 作業三: https://github.com/lyjeff/Machine-Learning-Class/tree/master/hw3-CNN
- 大學畢業專題: https://github.com/Holisha/6axis_revision