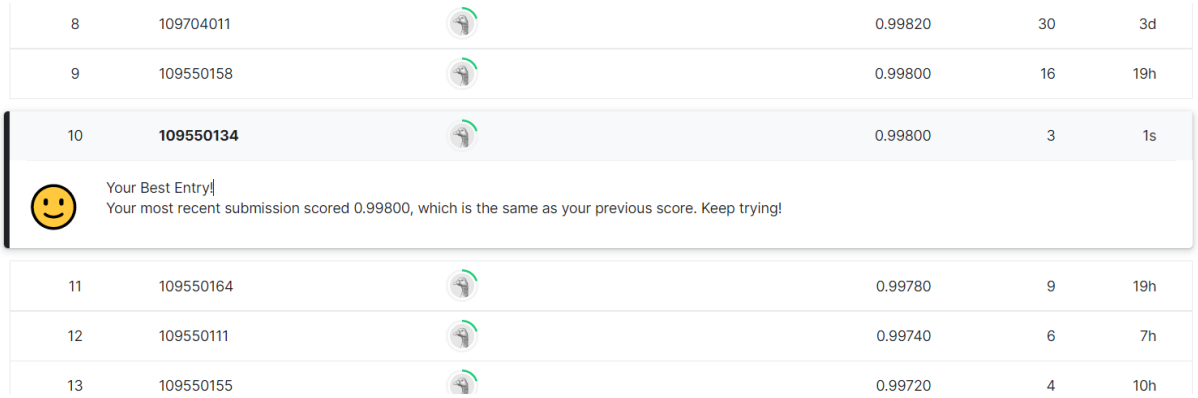
NYCU Introduction to Machine Learning, Homework 5

My result on public leaderboard:



8	109704011		0.99820	30	3d
9	109550158		0.99800	16	19h
10	**109550134**		0.99800	3	1s
	Your Best Entry! Your most recent submission scored 0.99800, which is the same as your previous score. Keep trying!				
11	109550164		0.99780	9	19h
12	109550111		0.99740	6	7h
13	109550155		0.99720	4	10h

Weight link:

https://drive.google.com/drive/folders/1dJCdBKAHEQ3CDslUKRUI6nF-PI4Pu60a?usp=sharing

Environment details:

- Python version: 3.7.12
- Pytorch version: 1.11.0+cpu
- Accelerator: GPU T4 x2
- Trained on kaggle

Implementation details:

- Model architecture:
    - resnet18(pretrain = true)
        - Modify fc layer in resnet18 :
            - Task1 = nn.Linear(512, 10)
            - Task2 = nn.Linear(512, 72)
            - Task3 = nn.Linear(512, 144)
    - optimizer = torch.optim.Adam(model.parameters(), lr=1e-3)
    - loss_fn
        - Task1 = nn.CrossEntropyLoss()
        - Task2 & Task3 = nn.MultiLabelSoftMarginLoss()
- Hyperparameters:
    - Task1:
        - Data preprocessing using torchvision.transforms:
            - Resize(100)

            - Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
        - Train : Val = 7 : 3
        - Batch_size = 200
        - Epoch = 30
            - Save the best weights when epoch >= 15
    - Task2:
        - Data preprocessing using torchvision.transforms:
            - Resize(250)
            - Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
        - Train : Val = 8 : 2
        - Batch_size = 100
        - Epoch = 30
            - Save the best weights when epoch >= 15
    - Task3:
        - Data preprocessing using torchvision.transforms:
            - Resize(275)
            - Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
        - Train : Val = 9 : 1
        - Batch_size = 75
        - Epoch = 30
            - Save the best weights when epoch >= 15
- Used deep learning framework: Pytorch
- Other:
    - Used one-hot encoding to deal with tasks with multiple character inputs (Task2 & Task3)