NavyaBhat_HW5

October 25, 2023

1 Setup environment

[1]:
```
from pathlib import Path
import sys

if 'google.colab' in str(get_ipython()):
    from google.colab import drive
    drive.mount('/content/drive')

    base_folder = Path('/content/drive/MyDrive/')
    data_folder = Path('/content')

    !pip install pytorch-lightning==2.0.9 -qq
    !pip install torchmetrics -U -qq
    !pip install fastdownload -U -qq
    !pip install fastai -U -qq
    !pip install wandb -U -qq

else:
    base_folder = Path('/home/harpreet/Insync/google_drive_shaannoor')
    data_folder = Path('/home/harpreet/data')
```

```
Mounted at /content/drive
                            727.7/727.7
kB 8.9 MB/s eta 0:00:00
                            805.2/805.2
kB 17.4 MB/s eta 0:00:00
                            2.1/2.1 MB
15.6 MB/s eta 0:00:00
                            190.6/190.6
kB 17.9 MB/s eta 0:00:00
                            241.0/241.0
kB 23.7 MB/s eta 0:00:00
  Preparing metadata (setup.py) … done
                            62.7/62.7 kB
```

```
9.5 MB/s eta 0:00:00
  Building wheel for pathtools (setup.py) … done
```

[2]:
```
custom_function_folder = base_folder/'data/custom-functions/fall_2023'
sys.path.append('/content/drive/MyDrive/')
model_folder = base_folder/'data/models/dl_fall_2023/dog_breed/oct-9'
model_folder.mkdir(parents=True, exist_ok=True)
project_folder = base_folder/'data/imagenette2'
kaggle_api_folder = base_folder/'data/.kaggle'
```

[3]:
```
# import Libraries
import yaml

import torch
import torchmetrics
from torchvision import transforms
import pytorch_lightning as pl
from pytorch_lightning import seed_everything
from pytorch_lightning.tuner import Tuner
from pytorch_lightning.callbacks import ModelCheckpoint, EarlyStopping,␣
 ↪LearningRateMonitor
from pytorch_lightning.loggers import CSVLogger, WandbLogger
import wandb
import gc

from data_module_imagenette2 import ImagenetteDataModule
from multiclass_lightning_module_v0 import MultiClassLightningModule
from model_two_layer_bn import TwoLayerMLPBN
from shared_utils import  plot_losses_acc
```

2 Function to load the model

[4]:
```
# Function to load the model
def load_model(config):
    model = TwoLayerMLPBN(**config)
    return model
```

3 Functions for Transformations

[5]:
```
def get_train_transforms(resize_height, resize_width, normalize_mean,␣
 ↪normalize_std):

    return transforms.Compose(
        [
            transforms.Resize((resize_height, resize_width)),
```

```
            transforms.ToTensor(),
            transforms.Normalize(normalize_mean, normalize_std),
        ]
    )

def get_test_transforms(resize_height, resize_width, normalize_mean,␣
 ↪normalize_std):

    return transforms.Compose(
        [
            transforms.Resize((resize_height, resize_width)),
            transforms.ToTensor(),
            transforms.Normalize(normalize_mean, normalize_std),
        ]
    )
```

4 Function to load DataModule

[6]:
```
def load_datamodule(config, data_folder):
    # Fetch the correct transform function based on config and pass the␣
 ↪appropriate arguments
    train_transform = get_train_transforms(**config['train_transform'])
    test_transform = get_test_transforms(**config['test_transform'])
    dm = ImagenetteDataModule(
        data_dir=data_folder,
        train_transform=train_transform,
        test_transform=test_transform,
        **config['data_module']
    )
    return dm
```

5 Function to load LightningModule

[7]:
```
def load_lightning_module(config, model):
    optimizer_cls = eval(config['optimizer_cls'])
    loss_fn = eval(config['loss_fn'])()  # directly instantiate the loss␣
 ↪function
    metric_cls = eval(config['metric_cls'])

    # If scheduler is defined, convert its string to class as well
    if config.get('scheduler_cls'):
        scheduler_cls = eval(config['scheduler_cls'])
        scheduler_options = config['scheduler_options']
        scheduler_params =   config['scheduler_params']
    else:
```

```
        scheduler_cls = None

    lightning_module = MultiClassLightningModule(model=model,
                                                 optimizer_cls=optimizer_cls,
                                                 loss_fn=loss_fn,
                                                 metric_cls=metric_cls,
                                                 scheduler_cls=scheduler_cls,
                                                ␣
 ↪scheduler_options=scheduler_options,
                                                ␣
 ↪scheduler_params=scheduler_params,
                                                 **config['others']
)
    return lightning_module
```

6 Function to load the Trainer

[8]:
```
def load_trainer(model, trainer_config, cl_config, batch_size, model_folder, ␣
 ↪logging=False, checkpointing=True, early_stopping=False):

    lr_monitor = LearningRateMonitor(**cl_config['lr_monitor'])
    callbacks = [lr_monitor]
    if checkpointing:
        model_checkpoint_callback = ModelCheckpoint(dirpath=model_folder/
 ↪cl_config['log_dir'],
                                                    **cl_config['model_checkpoint'])
        callbacks.append(model_checkpoint_callback)

    if early_stopping:
        early_stop_callback = EarlyStopping(**cl_config['early_stopping'] )
        callbacks.append(early_stop_callback)

    if logging:
        # For WandB logger:
        wandb_logger = WandbLogger(project=cl_config['wandb']['project'],␣
 ↪name=cl_config['wandb']['name'], save_dir=model_folder/cl_config['log_dir'])
        wandb_logger.experiment.config.update({'batch_size': batch_size,␣
 ↪'epochs': trainer_config['max_epochs']})
        wandb_logger.watch(model)

        # For CSV logger:
        csv_logger = CSVLogger(save_dir=model_folder/cl_config['log_dir'],␣
 ↪name=cl_config['csv']['name'])
        csv_logger.log_hyperparams(params={'batch_size': batch_size, 'epochs':␣
 ↪trainer_config['max_epochs']})
```

```
        trainer = pl.Trainer(callbacks=callbacks,
                             logger=[csv_logger, wandb_logger],
                             **trainer_config)
    else:
        trainer = pl.Trainer(callbacks=callbacks,
                             **trainer_config
                )
    return trainer
```

7 Function to load components

[9]:
```
def load_components(model_config, data_module_config, lightning_module_config,␣
 ↪data_folder, trainer_config,
cl_config, batch_size,logging=False, checkpointing=True, early_stopping=False):

    # Load the model
    model = load_model(model_config)

    # Load the data module
    dm = load_datamodule(data_module_config, data_folder)

    # Load the lightning module
    lightning_module = load_lightning_module(lightning_module_config, model)

    # Load the trainer
    trainer = load_trainer(model, trainer_config, cl_config, batch_size,␣
 ↪model_folder,  logging=logging,
                           checkpointing=checkpointing,␣
 ↪early_stopping=early_stopping)

    return model, dm, lightning_module, trainer
```

[10]:
```
def load_yaml(filepath):
    with open(filepath, 'r') as file:
        return yaml.safe_load(file)
```

8 Function to Load config files

[11]:
```
# Load configurations from YAML files
def load_all_configs():
    model_config = load_yaml(project_folder/'model_config.yaml')
    data_module_config = load_yaml(project_folder/'data_module_config.yaml')
    lightning_module_config = load_yaml(project_folder/'lightning_module_config.
 ↪yaml')
    cl_config = load_yaml(project_folder/'callbacks_loggers_config.yaml')
```

```
    trainer_config = load_yaml(project_folder/'trainer_config.yaml')

    return model_config, data_module_config, lightning_module_config,␣
 ↪cl_config, trainer_config
```

9 Function to free memory

[12]:
```
def free_memory():
    """
    Attempts to free up memory by deleting variables and running Python's␣
 ↪garbage collector.
    """
    gc.collect()
    for device_id in range(torch.cuda.device_count()):
        torch.cuda.set_device(device_id)
        torch.cuda.empty_cache()
    gc.collect()
```

10 Run One training and validation batch to check bugs

[13]:
```
# Load components
free_memory()
seed_everything(42)
model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()
# override default values
trainer_config['fast_dev_run']=True
model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
                                                       lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                        cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                       logging=False,␣
 ↪checkpointing=False, early_stopping=False)
dm.prepare_data()
trainer.fit(lightning_module, dm)
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
```

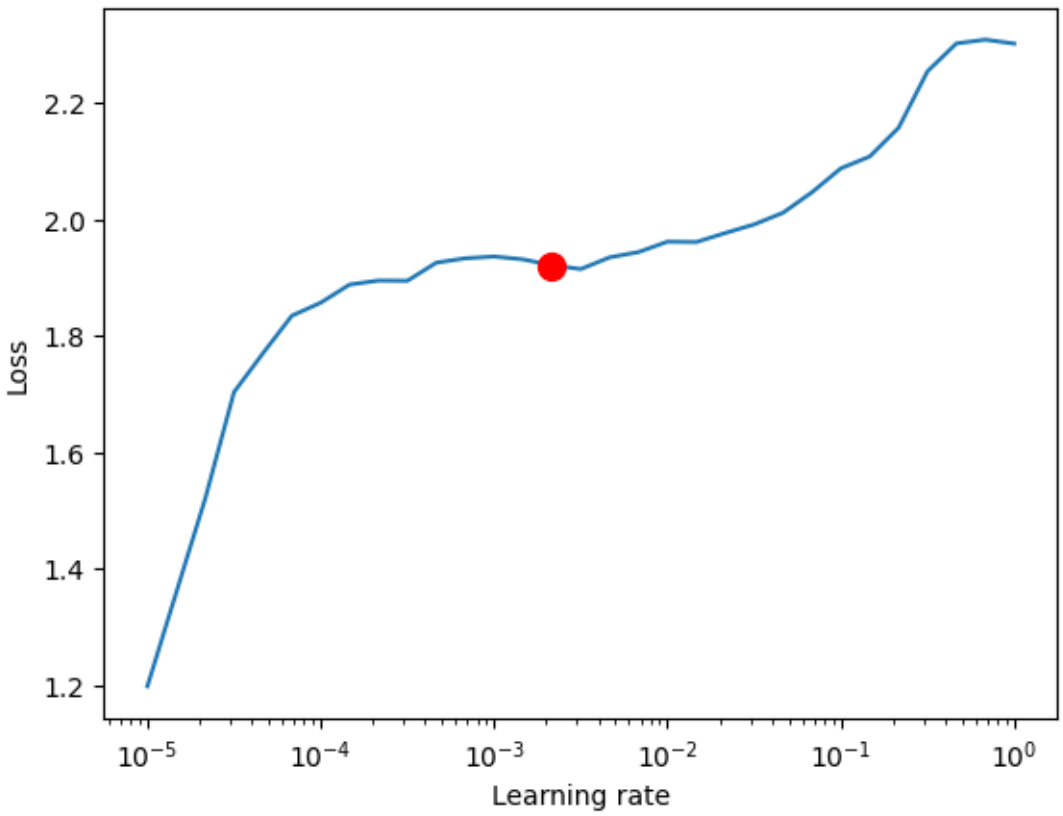
```
INFO:pytorch_lightning.utilities.rank_zero:Running in `fast_dev_run` mode: will
run the requested loop using 1 batch(es). Logging and checkpointing is
suppressed.

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.97, Val_Metric: 0.20 |

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_steps=1`
reached.

Train_Loss: 2.37, Train_Metric: 0.05
```

11 Find Learning Rate

[14]:
```
# Load components
free_memory()
seed_everything(42)
model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()
# override default values
trainer_config['max_epochs']=5
data_module_config['data_module']['batch_size']=64

model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
                                                        lightning_module_config,␣
 ↪data_folder, trainer_config,
```

```
                                                        cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                        logging=False,␣
 ↪checkpointing=False, early_stopping=False)
dm.setup()
tuner = Tuner(trainer)
lr_finder = tuner.lr_find(lightning_module, datamodule=dm, min_lr=1e-5,␣
 ↪max_lr=1, num_training=30, mode='exponential')
fig = lr_finder.plot(suggest=True)
new_lr = lr_finder.suggestion()
print(new_lr)
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_train_batches=1.0)`
was configured so 100% of the batches per epoch will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_val_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
WARNING:pytorch_lightning.loggers.tensorboard:Missing logger folder:
/content/lightning_logs
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]

Epoch 1: Val_Loss: 2.30, Val_Metric: 0.11 |

Finding best initial lr:   0%|          | 0/30 [00:00<?, ?it/s]

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_steps=30`
reached.
INFO:pytorch_lightning.tuner.lr_finder:Learning rate set to 0.002154434690031884
INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/.lr_find_c103e356-aa6b-46c5-afcf-cccc1ab992d1.ckpt

Train_Loss: 2.30, Train_Metric: 0.27

INFO:pytorch_lightning.utilities.rank_zero:Restored all states from the
checkpoint at /content/.lr_find_c103e356-aa6b-46c5-afcf-cccc1ab992d1.ckpt

0.002154434690031884
```



12 Overfit Small Subset

[ ]:
```
# Load components

free_memory()
seed_everything(42)
model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()

# override default values
data_module_config['data_module']['batch_size']=128
trainer_config['overfit_batches']=1
lightning_module_config['others']['learning_rate']=0.007
trainer_config['max_epochs']=3
model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
                                                       lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                        cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
```

```
                                                    logging=False,␣
 ↪checkpointing=False, early_stopping=False)
dm.setup()
trainer.fit(lightning_module, dm)
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(overfit_batches=1)` was
configured so 1 batch will be used.
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Sanity Checking: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.30, Val_Metric: 0.13 |

/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/trainer/connectors/data_connector.py:262:
UserWarning: You requested to overfit but enabled train dataloader shuffling. We
are turning off the train dataloader shuffling for you.
  rank_zero_warn(

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 489.95, Val_Metric: 0.10 | Train_Loss: 2.43, Train_Metric:
0.07

Validation: 0it [00:00, ?it/s]
```

```
Epoch 2: Val_Loss: 19.54, Val_Metric: 0.11 | Train_Loss: 0.63, Train_Metric:
1.00

Validation: 0it [00:00, ?it/s]

Epoch 3: Val_Loss: 24.56, Val_Metric: 0.10 | Train_Loss: 0.56, Train_Metric:
0.91

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=3`
reached.
```

13 Regularization -I

- Early stopping with a patience of 5, total epochs = 50
- Gradient Clipping
- Weight Deacay of 1
- Reduce Learning rate on plateau
- Use 50% of train/val data

[21]:
```
free_memory()
seed_everything(42)
model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()

# override default values
data_module_config['data_module']['batch_size']=128
lightning_module_config['others']['learning_rate']=0.007
trainer_config['max_epochs']=6
trainer_config['gradient_clip_val']=2
trainer_config['log_every_n_steps']=20

lightning_module_config['others']['optimizer_params']['weight_decay']=1
lightning_module_config['others']['learning_rate']=0.007
lightning_module_config['scheduler_cls']='torch.optim.lr_scheduler.
 ↪ReduceLROnPlateau'
lightning_module_config['scheduler_params']= {'mode': 'max', 'patience': 0,␣
 ↪'factor': 0.5, 'verbose': True}
lightning_module_config['scheduler_options']= {'monitor': 'val_metric',␣
 ↪'interval': 'epoch', 'frequency': 1}
cl_config['lr_monitor']['logging_interval']='epoch'
cl_config['wandb']['project']='imagenette2_multiclass'
cl_config['wandb']['name']='two_layer_mlp_bn_v0'

data_module_config['data_module']['small_subset']=True
data_module_config['data_module']['num_samples_small']=0.5

model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
```

```
                                                        lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                         cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                         logging=True,␣
 ↪checkpointing=True, early_stopping=True)
dm.setup()
trainer.fit(lightning_module, dm)
file = f"{trainer.logger.log_dir}/metrics.csv"
plot_losses_acc(file)
ckpt_path = trainer.checkpoint_callback.best_model_path
train_acc = trainer.validate(dataloaders=dm.train_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
valid_acc = trainer.validate(dataloaders=dm.val_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
print(f"Train Accuracy: {train_acc[0]['val_metric']*100:0.2f}")
print(f"Validation Accuracy: {valid_acc[0]['val_metric']*100:0.2f}")
wandb.finish()
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

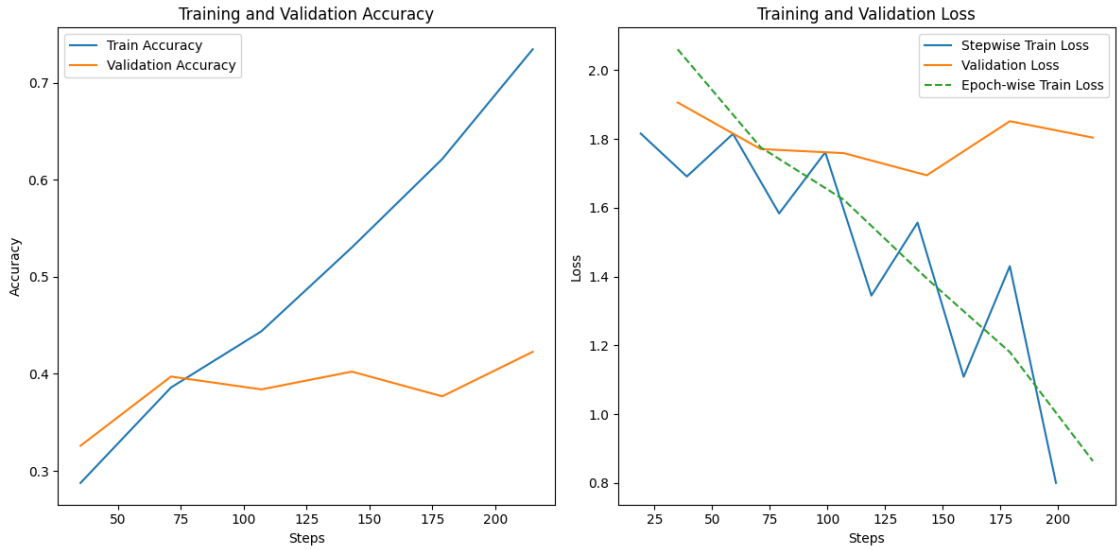
```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
wandb: logging graph, to disable use `wandb.watch(log_graph=False)`
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_train_batches=1.0)`
was configured so 100% of the batches per epoch will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_val_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/callbacks/model_checkpoint.py:617: UserWarning:
Checkpoint directory
/content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-9/logs exists and
is not empty.
  rank_zero_warn(f"Checkpoint directory {dirpath} exists and is not empty.")
```

```
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Sanity Checking: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.31, Val_Metric: 0.10 |

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 1.91, Val_Metric: 0.33 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved. New
best score: 0.326

Train_Loss: 2.06, Train_Metric: 0.29

Validation: 0it [00:00, ?it/s]

Epoch 2: Val_Loss: 1.77, Val_Metric: 0.40 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.071 >= min_delta = 0.0. New best score: 0.397

Train_Loss: 1.78, Train_Metric: 0.39

Validation: 0it [00:00, ?it/s]

Epoch 3: Val_Loss: 1.76, Val_Metric: 0.38 | Train_Loss: 1.62, Train_Metric: 0.44
Epoch 00003: reducing learning rate of group 0 to 3.5000e-03.

Validation: 0it [00:00, ?it/s]

Epoch 4: Val_Loss: 1.69, Val_Metric: 0.40 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.005 >= min_delta = 0.0. New best score: 0.402

Train_Loss: 1.39, Train_Metric: 0.53

Validation: 0it [00:00, ?it/s]
```

```
Epoch 5: Val_Loss: 1.85, Val_Metric: 0.38 | Train_Loss: 1.18, Train_Metric: 0.62
Epoch 00005: reducing learning rate of group 0 to 1.7500e-03.

Validation: 0it [00:00, ?it/s]

Epoch 6: Val_Loss: 1.80, Val_Metric: 0.42 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.020 >= min_delta = 0.0. New best score: 0.423

Train_Loss: 0.86, Train_Metric: 0.73

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=6`
reached.
```




```
INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=216.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=216.ckpt
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/trainer/connectors/data_connector.py:490:
PossibleUserWarning: Your `val_dataloader`'s sampler has shuffling enabled, it
is strongly recommended that you turn shuffling off for val/test dataloaders.
  rank_zero_warn(

Validation: 0it [00:00, ?it/s]

Epoch 7: Val_Loss: 0.66, Val_Metric: 0.81 |
```

```
INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=216.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=216.ckpt

Validation: 0it [00:00, ?it/s]

Epoch 7: Val_Loss: 1.80, Val_Metric: 0.42 | Train Accuracy: 81.12
Validation Accuracy: 42.26

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>
```

14 Regularization -II

- Increase the weight decay to 10

[15]:
```
lightning_module_config
```

[15]:
```
{'optimizer_cls': 'torch.optim.AdamW',
 'loss_fn': 'torch.nn.CrossEntropyLoss',
 'metric_cls': 'torchmetrics.Accuracy',
 'scheduler_cls': 'None',
 'scheduler_options': 'None',
 'scheduler_params': 'None',
 'others': {'optimizer_params': {'weight_decay': 0},
  'num_classes': 10,
  'learning_rate': 0.0001,
  'log_every_n_steps': 1,
  'log_test_metrics': True,
  'display_metrics': True}}
```

[16]:
```
lightning_module_config['others']['optimizer_params']['weight_decay']=10
```

[17]:
```
lightning_module_config
```

[17]:
```
{'optimizer_cls': 'torch.optim.AdamW',
 'loss_fn': 'torch.nn.CrossEntropyLoss',
 'metric_cls': 'torchmetrics.Accuracy',
 'scheduler_cls': 'None',
 'scheduler_options': 'None',
```

```
 'scheduler_params': 'None',
 'others': {'optimizer_params': {'weight_decay': 10},
  'num_classes': 10,
  'learning_rate': 0.0001,
  'log_every_n_steps': 1,
  'log_test_metrics': True,
  'display_metrics': True}}
```

[18]:
```
# Regularization -II
free_memory()
seed_everything(42)

model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
                                                        lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                         cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                        logging=True,␣
 ↪checkpointing=True, early_stopping=True)
dm.setup()
trainer.fit(lightning_module, dm)
file = f"{trainer.logger.log_dir}/metrics.csv"
plot_losses_acc(file)
train_acc = trainer.validate(dataloaders=dm.train_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
valid_acc = trainer.validate(dataloaders=dm.val_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
print(f"Train Accuracy: {train_acc[0]['val_metric']*100:0.2f}")
print(f"Validation Accuracy: {valid_acc[0]['val_metric']*100:0.2f}")
wandb.finish()
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
```

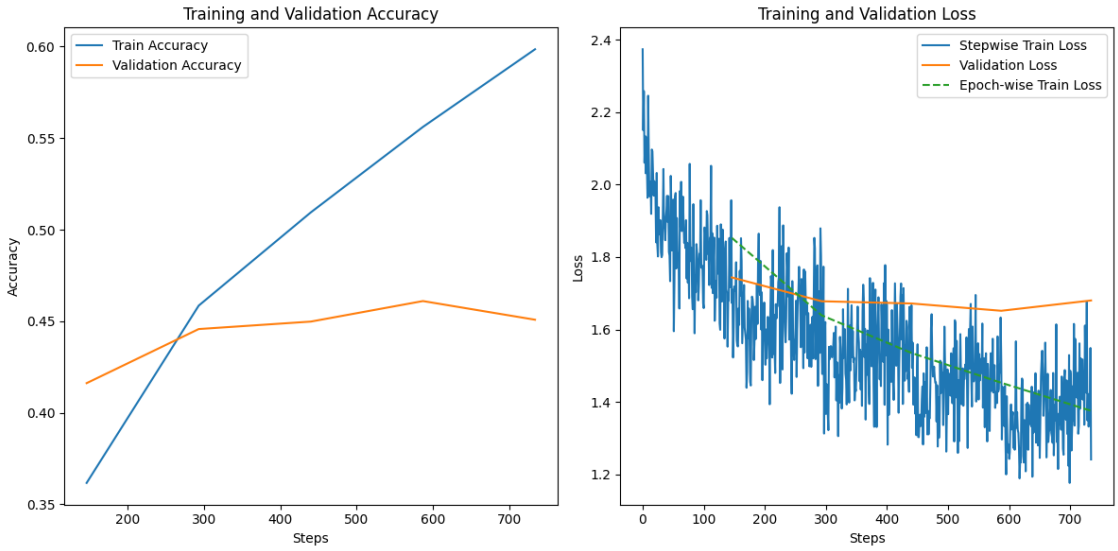
```
<IPython.core.display.Javascript object>
```

```
wandb: Logging into wandb.ai. (Learn how to deploy a W&B server
locally: https://wandb.me/wandb-server)
wandb: You can find your API key in your browser here:
https://wandb.ai/authorize
wandb: Paste an API key from your profile and hit enter, or press ctrl+c to
quit:
 ..........
wandb: Appending key for api.wandb.ai to your netrc file:
/root/.netrc
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

wandb: logging graph, to disable use `wandb.watch(log_graph=False)`
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_train_batches=1.0)`
was configured so 100% of the batches per epoch will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_val_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/callbacks/model_checkpoint.py:617: UserWarning:
Checkpoint directory
/content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-9/logs exists and
is not empty.
  rank_zero_warn(f"Checkpoint directory {dirpath} exists and is not empty.")
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Sanity Checking: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.30, Val_Metric: 0.13 |

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 1.74, Val_Metric: 0.42 |
```

```
INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved. New
best score: 0.416

Train_Loss: 1.85, Train_Metric: 0.36

Validation: 0it [00:00, ?it/s]

Epoch 2: Val_Loss: 1.68, Val_Metric: 0.45 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.030 >= min_delta = 0.0. New best score: 0.446

Train_Loss: 1.64, Train_Metric: 0.46

Validation: 0it [00:00, ?it/s]

Epoch 3: Val_Loss: 1.67, Val_Metric: 0.45 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.004 >= min_delta = 0.0. New best score: 0.450

Train_Loss: 1.54, Train_Metric: 0.51

Validation: 0it [00:00, ?it/s]

Epoch 4: Val_Loss: 1.65, Val_Metric: 0.46 |

INFO:pytorch_lightning.callbacks.early_stopping:Metric val_metric improved by
0.011 >= min_delta = 0.0. New best score: 0.461

Train_Loss: 1.45, Train_Metric: 0.56

Validation: 0it [00:00, ?it/s]

Epoch 5: Val_Loss: 1.68, Val_Metric: 0.45 | Train_Loss: 1.38, Train_Metric: 0.60

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=5`
reached.
```



```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-18-5090562f1c5d> in <cell line: 13>()
     11 file = f"{trainer.logger.log_dir}/metrics.csv"
     12 plot_losses_acc(file)
---> 13 train_acc = trainer.validate(dataloaders=dm.train_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
     14 valid_acc = trainer.validate(dataloaders=dm.val_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
     15 print(f"Train Accuracy: {train_acc[0]['val_metric']*100:0.2f}")

NameError: name 'ckpt_path' is not defined
```

15 HW5 PART A - Complete Regularization -III and IV

16 Regularization -III

- Use one Cycle Learning Rate instead of Reduce Learning Rate on Plateau

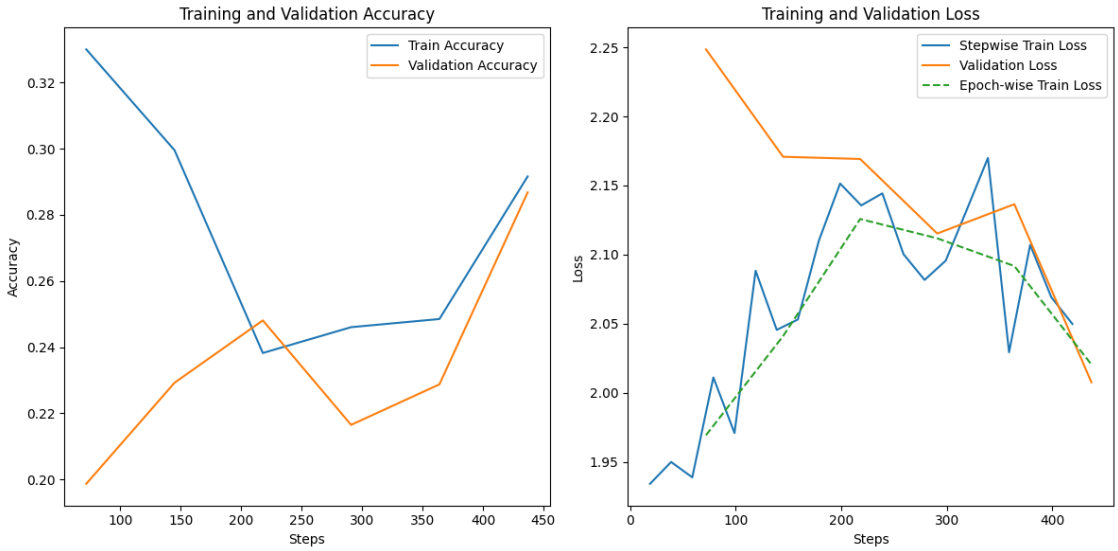
[19]:
```
free_memory()
seed_everything(42)



model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()

# override default values
data_module_config['data_module']['batch_size']=128
lightning_module_config['others']['learning_rate']=0.07
trainer_config['gradient_clip_val']=2
trainer_config['log_every_n_steps']=20
trainer_config['max_epochs']=6

lightning_module_config['others']['optimizer_params']['weight_decay']=10



# Setting the scheduler class
lightning_module_config['scheduler_cls'] = 'torch.optim.lr_scheduler.OneCycleLR'

# Parameters for the OneCycleLR
# Note: 'max_lr' is a required parameter for OneCycleLR; you'll need to specify␣
 ↪it based on your needs
```

```
lightning_module_config['scheduler_params'] = {'max_lr': 0.05,'verbose':
 ↪False,'steps_per_epoch':50,'epochs': 10,'pct_start':0.4,'div_factor':
 ↪25,'final_div_factor':1e4}

# Options related to the monitoring of the scheduler (if needed)
lightning_module_config['scheduler_options'] = {'monitor': 'val_loss','name':␣
 ↪'learning_rate', 'interval':'step','frequency':1}



model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
                                                        lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                         cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                        logging=True,␣
 ↪checkpointing=True, early_stopping=False) # change here
dm.prepare_data()

trainer.fit(lightning_module, dm)
file = f"{trainer.logger.log_dir}/metrics.csv"
plot_losses_acc(file)
ckpt_path = trainer.checkpoint_callback.best_model_path
train_acc = trainer.validate(dataloaders=dm.train_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
valid_acc = trainer.validate(dataloaders=dm.val_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
print(f"Train Accuracy: {train_acc[0]['val_metric']*100:0.2f}")
print(f"Validation Accuracy: {valid_acc[0]['val_metric']*100:0.2f}")
wandb.finish()
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
/usr/local/lib/python3.10/dist-packages/pytorch_lightning/loggers/wandb.py:398:
UserWarning: There is a wandb run already in progress and newly created
instances of `WandbLogger` will reuse this run. If this is not desired, call
`wandb.finish()` before instantiating `WandbLogger`.
  rank_zero_warn(
wandb: logging graph, to disable use `wandb.watch(log_graph=False)`
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_train_batches=1.0)`
was configured so 100% of the batches per epoch will be used..
```

```
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_val_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/callbacks/model_checkpoint.py:617: UserWarning:
Checkpoint directory
/content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-9/logs exists and
is not empty.
  rank_zero_warn(f"Checkpoint directory {dirpath} exists and is not empty.")
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Sanity Checking: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.31, Val_Metric: 0.09 |

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.25, Val_Metric: 0.20 | Train_Loss: 1.97, Train_Metric: 0.33

Validation: 0it [00:00, ?it/s]

Epoch 2: Val_Loss: 2.17, Val_Metric: 0.23 | Train_Loss: 2.04, Train_Metric: 0.30

Validation: 0it [00:00, ?it/s]

Epoch 3: Val_Loss: 2.17, Val_Metric: 0.25 | Train_Loss: 2.13, Train_Metric: 0.24

Validation: 0it [00:00, ?it/s]

Epoch 4: Val_Loss: 2.12, Val_Metric: 0.22 | Train_Loss: 2.11, Train_Metric: 0.25

Validation: 0it [00:00, ?it/s]

Epoch 5: Val_Loss: 2.14, Val_Metric: 0.23 | Train_Loss: 2.09, Train_Metric: 0.25

Validation: 0it [00:00, ?it/s]

Epoch 6: Val_Loss: 2.01, Val_Metric: 0.29 | Train_Loss: 2.02, Train_Metric: 0.29
```

```
INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=6`
reached.
```




```
INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=438.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=438.ckpt
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/trainer/connectors/data_connector.py:490:
PossibleUserWarning: Your `val_dataloader`'s sampler has shuffling enabled, it
is strongly recommended that you turn shuffling off for val/test dataloaders.
  rank_zero_warn(

Validation: 0it [00:00, ?it/s]

Epoch 7: Val_Loss: 1.98, Val_Metric: 0.31 |

INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=438.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=5-step=438.ckpt

Validation: 0it [00:00, ?it/s]
```

```
Epoch 7: Val_Loss: 2.01, Val_Metric: 0.29 | Train Accuracy: 31.00
Validation Accuracy: 28.68
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

17 Regularization-1V

- Use one Step LR instead of One Cycler LR

[20]:
```
free_memory()
seed_everything(42)

model_config, data_module_config, lightning_module_config, cl_config,␣
 ↪trainer_config = load_all_configs()

# override default values
data_module_config['data_module']['batch_size']=128
lightning_module_config['others']['learning_rate']=0.007
trainer_config['gradient_clip_val']=2
trainer_config['log_every_n_steps']=20
trainer_config['max_epochs']=10

lightning_module_config['others']['optimizer_params']['weight_decay']=10


# Setting the scheduler class
lightning_module_config['scheduler_cls'] = 'torch.optim.lr_scheduler.StepLR'

# Parameters for the OneCycleLR
# Note: 'max_lr' is a required parameter for OneCycleLR; you'll need to specify␣
 ↪it based on your needs
lightning_module_config['scheduler_params'] = {'step_size':1,'gamma':0.5}
#'max_lr': 0.05,'verbose': True,

# Options related to the monitoring of the scheduler (if needed)
lightning_module_config['scheduler_options'] = {'monitor': 'val_loss','name':␣
 ↪'learning_rate', 'interval':'step','frequency':1}



model, dm, lightning_module, trainer = load_components(model_config,␣
 ↪data_module_config,
```

```
                                                        lightning_module_config,␣
 ↪data_folder, trainer_config,
                                                         cl_config,␣
 ↪batch_size=data_module_config['data_module']['batch_size'],
                                                        logging=True,␣
 ↪checkpointing=True, early_stopping=False) # change here
dm.prepare_data()

trainer.fit(lightning_module, dm)
file = f"{trainer.logger.log_dir}/metrics.csv"
plot_losses_acc(file)
ckpt_path = trainer.checkpoint_callback.best_model_path
train_acc = trainer.validate(dataloaders=dm.train_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
valid_acc = trainer.validate(dataloaders=dm.val_dataloader(),␣
 ↪ckpt_path=ckpt_path, verbose=False)
print(f"Train Accuracy: {train_acc[0]['val_metric']*100:0.2f}")
print(f"Validation Accuracy: {valid_acc[0]['val_metric']*100:0.2f}")
wandb.finish()
```

```
INFO:lightning_fabric.utilities.seed:Global seed set to 42
wandb: Currently logged in as: navya-190997. Use
`wandb login --relogin` to force relogin
```

```
VBox(children=(Label(value='Waiting for wandb.init()…\r'), FloatProgress(value=0.
 ↪011112525288889527, max=1.0…
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

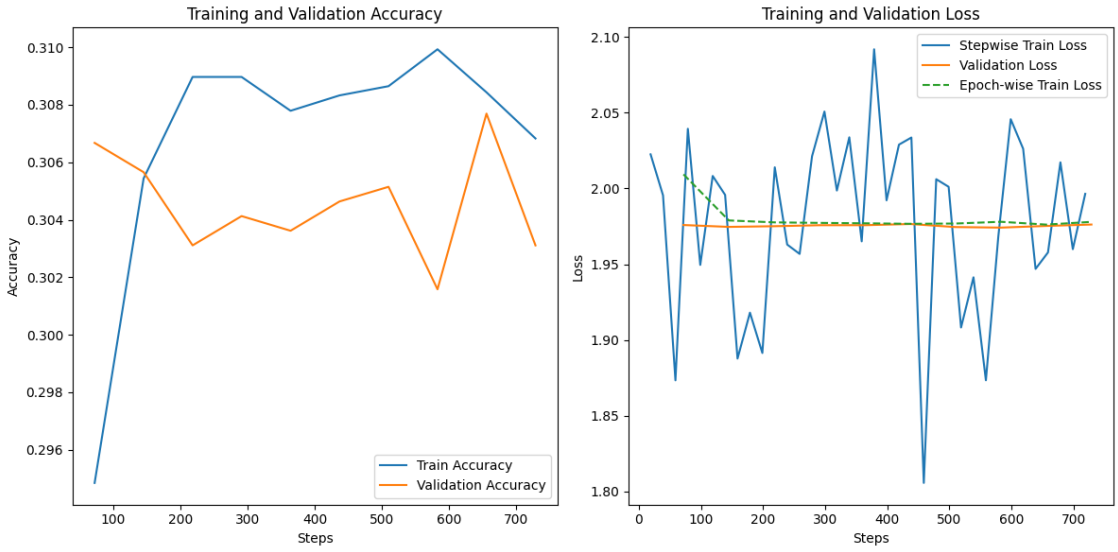
```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
wandb: logging graph, to disable use `wandb.watch(log_graph=False)`
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used:
True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU
cores
INFO:pytorch_lightning.utilities.rank_zero:IPU available: False, using: 0 IPUs
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_train_batches=1.0)`
was configured so 100% of the batches per epoch will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_val_batches=1.0)` was
configured so 100% of the batches will be used..
INFO:pytorch_lightning.utilities.rank_zero:`Trainer(limit_test_batches=1.0)` was
configured so 100% of the batches will be used..
```

```
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/callbacks/model_checkpoint.py:617: UserWarning:
Checkpoint directory
/content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-9/logs exists and
is not empty.
  rank_zero_warn(f"Checkpoint directory {dirpath} exists and is not empty.")
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type               | Params
-------------------------------------------------
0 | model        | TwoLayerMLPBN      | 281 M
1 | loss_fn      | CrossEntropyLoss   | 0
2 | train_metric | MulticlassAccuracy | 0
3 | val_metric   | MulticlassAccuracy | 0
4 | test_metric  | MulticlassAccuracy | 0
-------------------------------------------------
281 M     Trainable params
0         Non-trainable params
281 M     Total params
1,126.032 Total estimated model params size (MB)

Sanity Checking: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 2.31, Val_Metric: 0.09 |

Training: 0it [00:00, ?it/s]

Validation: 0it [00:00, ?it/s]

Epoch 1: Val_Loss: 1.98, Val_Metric: 0.31 | Train_Loss: 2.01, Train_Metric: 0.29

Validation: 0it [00:00, ?it/s]

Epoch 2: Val_Loss: 1.97, Val_Metric: 0.31 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 3: Val_Loss: 1.98, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 4: Val_Loss: 1.98, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 5: Val_Loss: 1.98, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 6: Val_Loss: 1.98, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 7: Val_Loss: 1.97, Val_Metric: 0.31 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]
```

```
Epoch 8: Val_Loss: 1.97, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 9: Val_Loss: 1.98, Val_Metric: 0.31 | Train_Loss: 1.98, Train_Metric: 0.31

Validation: 0it [00:00, ?it/s]

Epoch 10: Val_Loss: 1.98, Val_Metric: 0.30 | Train_Loss: 1.98, Train_Metric:
0.31

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped:
`max_epochs=10` reached.
```




```
INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=8-step=657.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=8-step=657.ckpt
/usr/local/lib/python3.10/dist-
packages/pytorch_lightning/trainer/connectors/data_connector.py:490:
PossibleUserWarning: Your `val_dataloader`'s sampler has shuffling enabled, it
is strongly recommended that you turn shuffling off for val/test dataloaders.
  rank_zero_warn(

Validation: 0it [00:00, ?it/s]

Epoch 11: Val_Loss: 1.97, Val_Metric: 0.31 |

INFO:pytorch_lightning.utilities.rank_zero:Restoring states from the checkpoint
```

```
path at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=8-step=657.ckpt
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES:
[0]
INFO:pytorch_lightning.utilities.rank_zero:Loaded model weights from the
checkpoint at /content/drive/MyDrive/data/models/dl_fall_2023/dog_breed/oct-
9/logs/epoch=8-step=657.ckpt

Validation: 0it [00:00, ?it/s]

Epoch 11: Val_Loss: 1.98, Val_Metric: 0.31 | Train Accuracy: 31.32
Validation Accuracy: 30.77

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>
```

[ ]: