Project_resnet

November 26, 2022

[1]:
```
import argparse
import os
import time
import shutil

import torch
import torch.nn as nn
import torch.optim as optim
import torch.nn.functional as F
import torch.backends.cudnn as cudnn


import torchvision
import torchvision.transforms as transforms

from models import *


global best_prec
use_gpu = torch.cuda.is_available()
device = torch.device("cuda")


batch_size = 128
model_name = "Resnet_20_quant_project"
model = resnet20_quant_project()


normalize = transforms.Normalize(mean=[0.491, 0.482, 0.447], std=[0.247, 0.243,␣
↪0.262])


train_dataset = torchvision.datasets.CIFAR10(
    root='./data',
    train=True,
    download=True,
    transform=transforms.Compose([
```

```
        transforms.RandomCrop(32, padding=4),
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        normalize,
    ]))
trainloader = torch.utils.data.DataLoader(train_dataset, batch_size=batch_size,␣
 ↪shuffle=True, num_workers=2)


test_dataset = torchvision.datasets.CIFAR10(
    root='./data',
    train=False,
    download=True,
    transform=transforms.Compose([
        transforms.ToTensor(),
        normalize,
    ]))

testloader = torch.utils.data.DataLoader(test_dataset, batch_size=batch_size,␣
 ↪shuffle=False, num_workers=2)
```

```
Files already downloaded and verified
Files already downloaded and verified
```

[2]:
```
print_freq = 100
def train(trainloader, model, criterion, optimizer, epoch):
    batch_time = AverageMeter()
    data_time = AverageMeter()
    losses = AverageMeter()
    top1 = AverageMeter()

    model.train()

    end = time.time()
    for i, (input, target) in enumerate(trainloader):
        # measure data loading time
        data_time.update(time.time() - end)

        input, target = input.cuda(), target.cuda()

        # compute output
        output = model(input)
        loss = criterion(output, target)

        # measure accuracy and record loss
        prec = accuracy(output, target)[0]
        losses.update(loss.item(), input.size(0))
```

```
        top1.update(prec.item(), input.size(0))

        # compute gradient and do SGD step
        optimizer.zero_grad()
        loss.backward()
        optimizer.step()

        # measure elapsed time
        batch_time.update(time.time() - end)
        end = time.time()


        if i % print_freq == 0:
            print('Epoch: [{0}][{1}/{2}]\t'
                  'Time {batch_time.val:.3f} ({batch_time.avg:.3f})\t'
                  'Data {data_time.val:.3f} ({data_time.avg:.3f})\t'
                  'Loss {loss.val:.4f} ({loss.avg:.4f})\t'
                  'Prec {top1.val:.3f}% ({top1.avg:.3f}%)'.format(
                   epoch, i, len(trainloader), batch_time=batch_time,
                   data_time=data_time, loss=losses, top1=top1))



def validate(val_loader, model, criterion ):
    batch_time = AverageMeter()
    losses = AverageMeter()
    top1 = AverageMeter()
    # switch to evaluate mode
    model.eval()

    end = time.time()
    with torch.no_grad():
        for i, (input, target) in enumerate(val_loader):

            input, target = input.cuda(), target.cuda()

            # compute output
            output = model(input)
            loss = criterion(output, target)

            # measure accuracy and record loss
            prec = accuracy(output, target)[0]
            losses.update(loss.item(), input.size(0))
            top1.update(prec.item(), input.size(0))

            # measure elapsed time
            batch_time.update(time.time() - end)
```

```
            end = time.time()

            if i % print_freq == 0:  # This line shows how frequently print out␣
↪the status. e.g., i%5 => every 5 batch, prints out
                print('Test: [{0}/{1}]\t'
                  'Time {batch_time.val:.3f} ({batch_time.avg:.3f})\t'
                  'Loss {loss.val:.4f} ({loss.avg:.4f})\t'
                  'Prec {top1.val:.3f}% ({top1.avg:.3f}%)'.format(
                   i, len(val_loader), batch_time=batch_time, loss=losses,
                   top1=top1))

    print(' * Prec {top1.avg:.3f}% '.format(top1=top1))
    return top1.avg


def accuracy(output, target, topk=(1,)):
    """Computes the precision@k for the specified values of k"""
    maxk = max(topk)
    batch_size = target.size(0)

    _, pred = output.topk(maxk, 1, True, True)
    pred = pred.t()
    correct = pred.eq(target.view(1, -1).expand_as(pred))

    res = []
    for k in topk:
        correct_k = correct[:k].view(-1).float().sum(0)
        res.append(correct_k.mul_(100.0 / batch_size))
    return res


class AverageMeter(object):
    """Computes and stores the average and current value"""
    def __init__(self):
        self.reset()

    def reset(self):
        self.val = 0
        self.avg = 0
        self.sum = 0
        self.count = 0

    def update(self, val, n=1):
        self.val = val
        self.sum += val * n
        self.count += n
        self.avg = self.sum / self.count
```

```
def save_checkpoint(state, is_best, fdir):
    filepath = os.path.join(fdir, 'checkpoint.pth')
    torch.save(state, filepath)
    if is_best:
        shutil.copyfile(filepath, os.path.join(fdir, 'model_best.pth.tar'))


def adjust_learning_rate(optimizer, epoch,adjust_list):
    """For resnet, the lr starts from 0.1, and is divided by 10 at 80 and 120␣
 ↪epochs"""

    if epoch in adjust_list:
        for param_group in optimizer.param_groups:
            param_group['lr'] = param_group['lr'] * 0.1
```

[3]:
```
lr = 4e-2
weight_decay = 1e-4
epochs = 100
best_prec = 0
model.cuda()
criterion = nn.CrossEntropyLoss().cuda()
optimizer = torch.optim.SGD(model.parameters(), lr=lr, momentum=0.
 ↪9,weight_decay=weight_decay)

if not os.path.exists('result'):
    os.makedirs('result')

fdir = 'result/'+str(model_name)
if not os.path.exists(fdir):
    os.makedirs(fdir)

adjust_list = [80,90]

for epoch in range(0, epochs):
    adjust_learning_rate(optimizer, epoch,adjust_list)

    train(trainloader, model, criterion, optimizer, epoch)

    # evaluate on test set
    print("Validation starts")
    prec = validate(testloader, model, criterion)

    # remember best precision and save checkpoint
    is_best = prec > best_prec
    best_prec = max(prec,best_prec)
```

```
    print('best acc: {:1f}'.format(best_prec))
    save_checkpoint({
        'epoch': epoch + 1,
        'state_dict': model.state_dict(),
        'best_prec': best_prec,
        'optimizer': optimizer.state_dict(),
    }, is_best, fdir)
```

```
Epoch: [0][0/391]       Time 0.697 (0.697)      Data 0.576 (0.576)      Loss
2.6520 (2.6520)    Prec 6.250% (6.250%)
Epoch: [0][100/391]     Time 0.039 (0.052)      Data 0.003 (0.008)      Loss
1.9711 (2.0783)    Prec 25.000% (21.744%)
Epoch: [0][200/391]     Time 0.041 (0.049)      Data 0.002 (0.005)      Loss
1.8291 (1.9321)    Prec 33.594% (27.433%)
Epoch: [0][300/391]     Time 0.046 (0.049)      Data 0.002 (0.004)      Loss
1.6122 (1.8335)    Prec 39.062% (31.364%)
Validation starts
Test: [0/79]    Time 0.585 (0.585)      Loss 1.6914 (1.6914)    Prec 35.938%
(35.938%)
 * Prec 37.970%
best acc: 37.970000
Epoch: [1][0/391]       Time 0.536 (0.536)      Data 0.479 (0.479)      Loss
1.5299 (1.5299)    Prec 44.531% (44.531%)
Epoch: [1][100/391]     Time 0.051 (0.052)      Data 0.002 (0.007)      Loss
1.4429 (1.4517)    Prec 47.656% (47.300%)
Epoch: [1][200/391]     Time 0.048 (0.049)      Data 0.002 (0.005)      Loss
1.3430 (1.4097)    Prec 48.438% (49.114%)
Epoch: [1][300/391]     Time 0.047 (0.048)      Data 0.002 (0.004)      Loss
1.1828 (1.3801)    Prec 54.688% (50.016%)
Validation starts
Test: [0/79]    Time 0.441 (0.441)      Loss 1.7276 (1.7276)    Prec 49.219%
(49.219%)
 * Prec 44.460%
best acc: 44.460000
Epoch: [2][0/391]       Time 0.543 (0.543)      Data 0.481 (0.481)      Loss
1.0911 (1.0911)    Prec 60.156% (60.156%)
Epoch: [2][100/391]     Time 0.044 (0.050)      Data 0.002 (0.007)      Loss
1.2717 (1.2106)    Prec 53.906% (56.235%)
Epoch: [2][200/391]     Time 0.038 (0.048)      Data 0.002 (0.004)      Loss
1.2264 (1.1977)    Prec 50.781% (56.817%)
Epoch: [2][300/391]     Time 0.037 (0.047)      Data 0.002 (0.004)      Loss
1.1645 (1.1675)    Prec 59.375% (58.088%)
Validation starts
Test: [0/79]    Time 0.330 (0.330)      Loss 1.4364 (1.4364)    Prec 54.688%
(54.688%)
 * Prec 56.720%
best acc: 56.720000
```

```
Epoch: [3][0/391]       Time 0.712 (0.712)      Data 0.649 (0.649)      Loss
1.1354 (1.1354)    Prec 55.469% (55.469%)
Epoch: [3][100/391]     Time 0.042 (0.052)      Data 0.002 (0.009)      Loss
1.0503 (1.0600)    Prec 60.938% (62.175%)
Epoch: [3][200/391]     Time 0.044 (0.051)      Data 0.002 (0.005)      Loss
1.0566 (1.0333)    Prec 64.844% (63.122%)
Epoch: [3][300/391]     Time 0.051 (0.049)      Data 0.003 (0.004)      Loss
0.8835 (1.0177)    Prec 69.531% (63.899%)
Validation starts
Test: [0/79]    Time 0.394 (0.394)      Loss 1.2359 (1.2359)    Prec 52.344%
(52.344%)
 * Prec 61.030%
best acc: 61.030000
Epoch: [4][0/391]       Time 0.672 (0.672)      Data 0.612 (0.612)      Loss
0.9447 (0.9447)    Prec 71.875% (71.875%)
Epoch: [4][100/391]     Time 0.041 (0.050)      Data 0.002 (0.008)      Loss
0.8651 (0.9116)    Prec 71.875% (67.528%)
Epoch: [4][200/391]     Time 0.044 (0.048)      Data 0.002 (0.005)      Loss
0.8039 (0.9043)    Prec 75.781% (68.062%)
Epoch: [4][300/391]     Time 0.039 (0.048)      Data 0.002 (0.004)      Loss
0.9637 (0.8946)    Prec 67.188% (68.433%)
Validation starts
Test: [0/79]    Time 0.363 (0.363)      Loss 0.9038 (0.9038)    Prec 67.969%
(67.969%)
 * Prec 67.010%
best acc: 67.010000
Epoch: [5][0/391]       Time 0.545 (0.545)      Data 0.483 (0.483)      Loss
0.8568 (0.8568)    Prec 71.094% (71.094%)
Epoch: [5][100/391]     Time 0.045 (0.054)      Data 0.002 (0.007)      Loss
0.8860 (0.8512)    Prec 68.750% (70.498%)
Epoch: [5][200/391]     Time 0.045 (0.051)      Data 0.002 (0.005)      Loss
0.8183 (0.8449)    Prec 71.094% (70.460%)
Epoch: [5][300/391]     Time 0.040 (0.049)      Data 0.002 (0.004)      Loss
0.8003 (0.8361)    Prec 75.000% (70.720%)
Validation starts
Test: [0/79]    Time 0.327 (0.327)      Loss 0.9208 (0.9208)    Prec 61.719%
(61.719%)
 * Prec 66.110%
best acc: 67.010000
Epoch: [6][0/391]       Time 0.575 (0.575)      Data 0.517 (0.517)      Loss
0.7186 (0.7186)    Prec 79.688% (79.688%)
Epoch: [6][100/391]     Time 0.054 (0.053)      Data 0.003 (0.008)      Loss
0.9389 (0.7764)    Prec 68.750% (72.950%)
Epoch: [6][200/391]     Time 0.046 (0.050)      Data 0.002 (0.005)      Loss
0.6818 (0.7733)    Prec 72.656% (72.862%)
Epoch: [6][300/391]     Time 0.046 (0.049)      Data 0.002 (0.004)      Loss
0.7320 (0.7697)    Prec 71.875% (72.903%)
Validation starts
```

```
Test: [0/79]    Time 0.388 (0.388)      Loss 0.9254 (0.9254)    Prec 66.406%
(66.406%)
 * Prec 70.240%
best acc: 70.240000
Epoch: [7][0/391]       Time 0.841 (0.841)      Data 0.779 (0.779)      Loss
0.6742 (0.6742)    Prec 73.438% (73.438%)
Epoch: [7][100/391]     Time 0.059 (0.064)      Data 0.003 (0.010)      Loss
0.6625 (0.7399)    Prec 71.094% (74.025%)
Epoch: [7][200/391]     Time 0.044 (0.055)      Data 0.002 (0.006)      Loss
0.6622 (0.7461)    Prec 78.906% (73.842%)
Epoch: [7][300/391]     Time 0.043 (0.052)      Data 0.002 (0.005)      Loss
0.7193 (0.7354)    Prec 74.219% (74.255%)
Validation starts
Test: [0/79]    Time 0.279 (0.279)      Loss 0.8234 (0.8234)    Prec 72.656%
(72.656%)
 * Prec 70.390%
best acc: 70.390000
Epoch: [8][0/391]       Time 0.581 (0.581)      Data 0.523 (0.523)      Loss
0.6751 (0.6751)    Prec 75.000% (75.000%)
Epoch: [8][100/391]     Time 0.045 (0.053)      Data 0.002 (0.008)      Loss
0.6582 (0.6747)    Prec 73.438% (76.624%)
Epoch: [8][200/391]     Time 0.037 (0.048)      Data 0.002 (0.005)      Loss
0.8347 (0.6898)    Prec 70.312% (75.847%)
Epoch: [8][300/391]     Time 0.038 (0.044)      Data 0.001 (0.004)      Loss
0.7509 (0.6847)    Prec 75.000% (76.139%)
Validation starts
Test: [0/79]    Time 0.508 (0.508)      Loss 0.5880 (0.5880)    Prec 80.469%
(80.469%)
 * Prec 74.010%
best acc: 74.010000
Epoch: [9][0/391]       Time 0.636 (0.636)      Data 0.578 (0.578)      Loss
0.6594 (0.6594)    Prec 75.781% (75.781%)
Epoch: [9][100/391]     Time 0.049 (0.055)      Data 0.002 (0.008)      Loss
0.8528 (0.6613)    Prec 67.969% (76.756%)
Epoch: [9][200/391]     Time 0.041 (0.050)      Data 0.002 (0.005)      Loss
0.6358 (0.6621)    Prec 79.688% (76.850%)
Epoch: [9][300/391]     Time 0.043 (0.048)      Data 0.002 (0.004)      Loss
0.5533 (0.6616)    Prec 81.250% (76.877%)
Validation starts
Test: [0/79]    Time 0.433 (0.433)      Loss 0.7407 (0.7407)    Prec 74.219%
(74.219%)
 * Prec 75.570%
best acc: 75.570000
Epoch: [10][0/391]      Time 0.525 (0.525)      Data 0.469 (0.469)      Loss
0.7799 (0.7799)    Prec 72.656% (72.656%)
Epoch: [10][100/391]    Time 0.053 (0.053)      Data 0.003 (0.007)      Loss
0.5645 (0.6187)    Prec 80.469% (78.605%)
Epoch: [10][200/391]    Time 0.047 (0.050)      Data 0.002 (0.005)      Loss
```

```
0.6457 (0.6184)    Prec 78.125% (78.607%)
Epoch: [10][300/391]    Time 0.046 (0.049)      Data 0.002 (0.004)      Loss
0.7006 (0.6287)    Prec 71.875% (78.198%)
Validation starts
Test: [0/79]    Time 0.510 (0.510)      Loss 0.7054 (0.7054)    Prec 75.000%
(75.000%)
 * Prec 76.940%
best acc: 76.940000
Epoch: [11][0/391]      Time 0.542 (0.542)      Data 0.488 (0.488)      Loss
0.7296 (0.7296)    Prec 75.781% (75.781%)
Epoch: [11][100/391]    Time 0.045 (0.052)      Data 0.002 (0.007)      Loss
0.4605 (0.6202)    Prec 80.469% (78.055%)
Epoch: [11][200/391]    Time 0.038 (0.050)      Data 0.002 (0.005)      Loss
0.6396 (0.6130)    Prec 79.688% (78.537%)
Epoch: [11][300/391]    Time 0.044 (0.050)      Data 0.002 (0.004)      Loss
0.5929 (0.6107)    Prec 79.688% (78.673%)
Validation starts
Test: [0/79]    Time 0.380 (0.380)      Loss 0.6441 (0.6441)    Prec 78.125%
(78.125%)
 * Prec 76.600%
best acc: 76.940000
Epoch: [12][0/391]      Time 0.558 (0.558)      Data 0.498 (0.498)      Loss
0.4537 (0.4537)    Prec 83.594% (83.594%)
Epoch: [12][100/391]    Time 0.038 (0.049)      Data 0.002 (0.007)      Loss
0.5921 (0.6042)    Prec 78.906% (78.999%)
Epoch: [12][200/391]    Time 0.041 (0.047)      Data 0.002 (0.005)      Loss
0.5482 (0.6017)    Prec 82.812% (79.050%)
Epoch: [12][300/391]    Time 0.037 (0.047)      Data 0.002 (0.004)      Loss
0.5295 (0.6046)    Prec 86.719% (78.935%)
Validation starts
Test: [0/79]    Time 0.515 (0.515)      Loss 0.5531 (0.5531)    Prec 80.469%
(80.469%)
 * Prec 77.480%
best acc: 77.480000
Epoch: [13][0/391]      Time 0.718 (0.718)      Data 0.676 (0.676)      Loss
0.5952 (0.5952)    Prec 78.906% (78.906%)
Epoch: [13][100/391]    Time 0.040 (0.052)      Data 0.002 (0.009)      Loss
0.6260 (0.5625)    Prec 79.688% (80.229%)
Epoch: [13][200/391]    Time 0.041 (0.049)      Data 0.002 (0.006)      Loss
0.4157 (0.5660)    Prec 85.156% (80.263%)
Epoch: [13][300/391]    Time 0.046 (0.048)      Data 0.002 (0.004)      Loss
0.4446 (0.5713)    Prec 86.719% (80.137%)
Validation starts
Test: [0/79]    Time 0.297 (0.297)      Loss 0.6328 (0.6328)    Prec 78.125%
(78.125%)
 * Prec 77.710%
best acc: 77.710000
Epoch: [14][0/391]      Time 0.823 (0.823)      Data 0.778 (0.778)      Loss
```

```
0.4488 (0.4488)    Prec 88.281% (88.281%)
Epoch: [14][100/391]    Time 0.050 (0.055)      Data 0.003 (0.010)      Loss
0.4761 (0.5541)    Prec 81.250% (80.894%)
Epoch: [14][200/391]    Time 0.036 (0.052)      Data 0.002 (0.006)      Loss
0.5584 (0.5569)    Prec 83.594% (80.690%)
Epoch: [14][300/391]    Time 0.037 (0.050)      Data 0.002 (0.005)      Loss
0.5966 (0.5561)    Prec 79.688% (80.741%)
Validation starts
Test: [0/79]    Time 0.424 (0.424)      Loss 0.5909 (0.5909)    Prec 82.031%
(82.031%)
 * Prec 77.640%
best acc: 77.710000
Epoch: [15][0/391]      Time 0.522 (0.522)      Data 0.466 (0.466)      Loss
0.5186 (0.5186)    Prec 82.031% (82.031%)
Epoch: [15][100/391]    Time 0.043 (0.048)      Data 0.002 (0.007)      Loss
0.6145 (0.5471)    Prec 78.906% (81.134%)
Epoch: [15][200/391]    Time 0.052 (0.048)      Data 0.002 (0.005)      Loss
0.5816 (0.5466)    Prec 82.031% (80.889%)
Epoch: [15][300/391]    Time 0.050 (0.048)      Data 0.003 (0.004)      Loss
0.4391 (0.5470)    Prec 83.594% (80.824%)
Validation starts
Test: [0/79]    Time 0.796 (0.796)      Loss 0.5999 (0.5999)    Prec 77.344%
(77.344%)
 * Prec 78.220%
best acc: 78.220000
Epoch: [16][0/391]      Time 0.641 (0.641)      Data 0.595 (0.595)      Loss
0.4703 (0.4703)    Prec 84.375% (84.375%)
Epoch: [16][100/391]    Time 0.056 (0.053)      Data 0.003 (0.008)      Loss
0.5447 (0.5199)    Prec 78.906% (81.877%)
Epoch: [16][200/391]    Time 0.045 (0.051)      Data 0.002 (0.005)      Loss
0.6851 (0.5185)    Prec 75.000% (82.027%)
Epoch: [16][300/391]    Time 0.044 (0.049)      Data 0.002 (0.004)      Loss
0.5540 (0.5217)    Prec 80.469% (81.842%)
Validation starts
Test: [0/79]    Time 0.245 (0.245)      Loss 0.6395 (0.6395)    Prec 74.219%
(74.219%)
 * Prec 75.680%
best acc: 78.220000
Epoch: [17][0/391]      Time 0.591 (0.591)      Data 0.527 (0.527)      Loss
0.5065 (0.5065)    Prec 83.594% (83.594%)
Epoch: [17][100/391]    Time 0.046 (0.053)      Data 0.002 (0.008)      Loss
0.4520 (0.5143)    Prec 82.812% (82.047%)
Epoch: [17][200/391]    Time 0.041 (0.049)      Data 0.002 (0.005)      Loss
0.4913 (0.5076)    Prec 82.812% (82.264%)
Epoch: [17][300/391]    Time 0.037 (0.048)      Data 0.002 (0.004)      Loss
0.5568 (0.5103)    Prec 82.812% (82.192%)
Validation starts
Test: [0/79]    Time 0.393 (0.393)      Loss 0.4998 (0.4998)    Prec 83.594%
```

```
(83.594%)
 * Prec 79.220%
best acc: 79.220000
Epoch: [18][0/391]      Time 0.578 (0.578)      Data 0.519 (0.519)      Loss
0.4343 (0.4343)    Prec 84.375% (84.375%)
Epoch: [18][100/391]    Time 0.044 (0.052)      Data 0.002 (0.007)      Loss
0.4590 (0.5145)    Prec 83.594% (81.745%)
Epoch: [18][200/391]    Time 0.036 (0.049)      Data 0.002 (0.005)      Loss
0.5646 (0.5052)    Prec 81.250% (82.257%)
Epoch: [18][300/391]    Time 0.047 (0.049)      Data 0.002 (0.004)      Loss
0.4629 (0.5072)    Prec 81.250% (82.229%)
Validation starts
Test: [0/79]    Time 0.300 (0.300)      Loss 0.6272 (0.6272)    Prec 77.344%
(77.344%)
 * Prec 75.850%
best acc: 79.220000
Epoch: [19][0/391]      Time 0.556 (0.556)      Data 0.493 (0.493)      Loss
0.4065 (0.4065)    Prec 87.500% (87.500%)
Epoch: [19][100/391]    Time 0.044 (0.051)      Data 0.002 (0.007)      Loss
0.3701 (0.4789)    Prec 89.844% (82.836%)
Epoch: [19][200/391]    Time 0.044 (0.048)      Data 0.002 (0.005)      Loss
0.5877 (0.4934)    Prec 78.906% (82.548%)
Epoch: [19][300/391]    Time 0.051 (0.048)      Data 0.003 (0.004)      Loss
0.5256 (0.4953)    Prec 78.906% (82.581%)
Validation starts
Test: [0/79]    Time 0.303 (0.303)      Loss 0.4849 (0.4849)    Prec 84.375%
(84.375%)
 * Prec 78.130%
best acc: 79.220000
Epoch: [20][0/391]      Time 0.598 (0.598)      Data 0.539 (0.539)      Loss
0.3695 (0.3695)    Prec 87.500% (87.500%)
Epoch: [20][100/391]    Time 0.048 (0.053)      Data 0.003 (0.008)      Loss
0.6348 (0.4707)    Prec 78.125% (83.377%)
Epoch: [20][200/391]    Time 0.043 (0.051)      Data 0.002 (0.005)      Loss
0.4748 (0.4767)    Prec 84.375% (83.326%)
Epoch: [20][300/391]    Time 0.045 (0.049)      Data 0.002 (0.004)      Loss
0.3608 (0.4834)    Prec 87.500% (83.140%)
Validation starts
Test: [0/79]    Time 0.339 (0.339)      Loss 0.4761 (0.4761)    Prec 82.812%
(82.812%)
 * Prec 80.920%
best acc: 80.920000
Epoch: [21][0/391]      Time 0.664 (0.664)      Data 0.600 (0.600)      Loss
0.3877 (0.3877)    Prec 85.938% (85.938%)
Epoch: [21][100/391]    Time 0.045 (0.055)      Data 0.002 (0.008)      Loss
0.3209 (0.4577)    Prec 86.719% (83.973%)
Epoch: [21][200/391]    Time 0.054 (0.051)      Data 0.003 (0.005)      Loss
0.4491 (0.4715)    Prec 83.594% (83.570%)
```

```
Epoch: [21][300/391]    Time 0.048 (0.049)      Data 0.002 (0.004)      Loss
0.6024 (0.4772)    Prec 79.688% (83.498%)
Validation starts
Test: [0/79]    Time 0.461 (0.461)      Loss 0.4941 (0.4941)    Prec 81.250%
(81.250%)
 * Prec 80.150%
best acc: 80.920000
Epoch: [22][0/391]      Time 0.690 (0.690)      Data 0.630 (0.630)      Loss
0.3934 (0.3934)    Prec 87.500% (87.500%)
Epoch: [22][100/391]    Time 0.043 (0.054)      Data 0.002 (0.008)      Loss
0.4671 (0.4538)    Prec 82.812% (84.205%)
Epoch: [22][200/391]    Time 0.051 (0.050)      Data 0.003 (0.005)      Loss
0.4217 (0.4621)    Prec 85.156% (83.874%)
Epoch: [22][300/391]    Time 0.043 (0.049)      Data 0.002 (0.004)      Loss
0.5122 (0.4648)    Prec 78.906% (83.770%)
Validation starts
Test: [0/79]    Time 0.457 (0.457)      Loss 0.5885 (0.5885)    Prec 77.344%
(77.344%)
 * Prec 80.000%
best acc: 80.920000
Epoch: [23][0/391]      Time 0.732 (0.732)      Data 0.677 (0.677)      Loss
0.4861 (0.4861)    Prec 82.812% (82.812%)
Epoch: [23][100/391]    Time 0.042 (0.055)      Data 0.002 (0.009)      Loss
0.4643 (0.4410)    Prec 83.594% (84.499%)
Epoch: [23][200/391]    Time 0.047 (0.049)      Data 0.002 (0.006)      Loss
0.4364 (0.4451)    Prec 84.375% (84.519%)
Epoch: [23][300/391]    Time 0.059 (0.049)      Data 0.003 (0.004)      Loss
0.4405 (0.4481)    Prec 82.031% (84.476%)
Validation starts
Test: [0/79]    Time 0.500 (0.500)      Loss 0.5932 (0.5932)    Prec 78.125%
(78.125%)
 * Prec 79.530%
best acc: 80.920000
Epoch: [24][0/391]      Time 1.098 (1.098)      Data 1.047 (1.047)      Loss
0.5728 (0.5728)    Prec 80.469% (80.469%)
Epoch: [24][100/391]    Time 0.048 (0.057)      Data 0.002 (0.013)      Loss
0.4823 (0.4450)    Prec 79.688% (84.506%)
Epoch: [24][200/391]    Time 0.044 (0.051)      Data 0.002 (0.008)      Loss
0.4105 (0.4504)    Prec 87.500% (84.274%)
Epoch: [24][300/391]    Time 0.045 (0.049)      Data 0.002 (0.006)      Loss
0.5537 (0.4507)    Prec 79.688% (84.359%)
Validation starts
Test: [0/79]    Time 0.686 (0.686)      Loss 0.5138 (0.5138)    Prec 84.375%
(84.375%)
 * Prec 81.550%
best acc: 81.550000
Epoch: [25][0/391]      Time 0.702 (0.702)      Data 0.637 (0.637)      Loss
0.5204 (0.5204)    Prec 83.594% (83.594%)
```

```
Epoch: [25][100/391]    Time 0.040 (0.053)      Data 0.002 (0.009)      Loss
0.4542 (0.4283)    Prec 81.250% (85.110%)
Epoch: [25][200/391]    Time 0.044 (0.050)      Data 0.002 (0.005)      Loss
0.5575 (0.4313)    Prec 84.375% (84.880%)
Epoch: [25][300/391]    Time 0.047 (0.048)      Data 0.002 (0.004)      Loss
0.5234 (0.4353)    Prec 79.688% (84.749%)
Validation starts
Test: [0/79]    Time 0.493 (0.493)      Loss 0.3820 (0.3820)    Prec 88.281%
(88.281%)
 * Prec 81.840%
best acc: 81.840000
Epoch: [26][0/391]      Time 0.521 (0.521)      Data 0.463 (0.463)      Loss
0.5043 (0.5043)    Prec 82.812% (82.812%)
Epoch: [26][100/391]    Time 0.054 (0.053)      Data 0.003 (0.007)      Loss
0.5119 (0.4258)    Prec 84.375% (85.636%)
Epoch: [26][200/391]    Time 0.044 (0.050)      Data 0.002 (0.005)      Loss
0.4541 (0.4238)    Prec 86.719% (85.452%)
Epoch: [26][300/391]    Time 0.047 (0.048)      Data 0.002 (0.004)      Loss
0.3604 (0.4279)    Prec 88.281% (85.182%)
Validation starts
Test: [0/79]    Time 0.352 (0.352)      Loss 0.4834 (0.4834)    Prec 82.031%
(82.031%)
 * Prec 80.690%
best acc: 81.840000
Epoch: [27][0/391]      Time 0.617 (0.617)      Data 0.560 (0.560)      Loss
0.3419 (0.3419)    Prec 87.500% (87.500%)
Epoch: [27][100/391]    Time 0.041 (0.051)      Data 0.002 (0.008)      Loss
0.4454 (0.4071)    Prec 82.812% (85.721%)
Epoch: [27][200/391]    Time 0.054 (0.049)      Data 0.003 (0.005)      Loss
0.3638 (0.4272)    Prec 87.500% (85.110%)
Epoch: [27][300/391]    Time 0.037 (0.048)      Data 0.002 (0.004)      Loss
0.3775 (0.4319)    Prec 86.719% (85.016%)
Validation starts
Test: [0/79]    Time 0.404 (0.404)      Loss 0.4090 (0.4090)    Prec 83.594%
(83.594%)
 * Prec 81.530%
best acc: 81.840000
Epoch: [28][0/391]      Time 0.594 (0.594)      Data 0.518 (0.518)      Loss
0.3874 (0.3874)    Prec 85.938% (85.938%)
Epoch: [28][100/391]    Time 0.047 (0.052)      Data 0.002 (0.007)      Loss
0.4825 (0.4379)    Prec 84.375% (84.847%)
Epoch: [28][200/391]    Time 0.037 (0.048)      Data 0.002 (0.005)      Loss
0.3985 (0.4287)    Prec 87.500% (85.016%)
Epoch: [28][300/391]    Time 0.047 (0.048)      Data 0.003 (0.004)      Loss
0.5825 (0.4278)    Prec 82.031% (85.058%)
Validation starts
Test: [0/79]    Time 0.345 (0.345)      Loss 0.4670 (0.4670)    Prec 85.156%
(85.156%)
```

```
 * Prec 80.550%
best acc: 81.840000
Epoch: [29][0/391]      Time 0.513 (0.513)      Data 0.458 (0.458)      Loss
0.3137 (0.3137)    Prec 87.500% (87.500%)
Epoch: [29][100/391]    Time 0.046 (0.050)      Data 0.003 (0.007)      Loss
0.3582 (0.4082)    Prec 88.281% (85.907%)
Epoch: [29][200/391]    Time 0.043 (0.049)      Data 0.002 (0.005)      Loss
0.4982 (0.4115)    Prec 82.031% (85.821%)
Epoch: [29][300/391]    Time 0.051 (0.049)      Data 0.002 (0.004)      Loss
0.2727 (0.4101)    Prec 91.406% (85.841%)
Validation starts
Test: [0/79]    Time 0.424 (0.424)      Loss 0.4923 (0.4923)    Prec 82.031%
(82.031%)
 * Prec 81.490%
best acc: 81.840000
Epoch: [30][0/391]      Time 0.955 (0.955)      Data 0.900 (0.900)      Loss
0.5181 (0.5181)    Prec 84.375% (84.375%)
Epoch: [30][100/391]    Time 0.044 (0.056)      Data 0.002 (0.011)      Loss
0.4475 (0.4050)    Prec 83.594% (85.930%)
Epoch: [30][200/391]    Time 0.037 (0.050)      Data 0.003 (0.007)      Loss
0.5510 (0.4109)    Prec 81.250% (85.557%)
Epoch: [30][300/391]    Time 0.046 (0.048)      Data 0.002 (0.005)      Loss
0.4876 (0.4106)    Prec 80.469% (85.670%)
Validation starts
Test: [0/79]    Time 0.339 (0.339)      Loss 0.4842 (0.4842)    Prec 82.031%
(82.031%)
 * Prec 82.590%
best acc: 82.590000
Epoch: [31][0/391]      Time 1.075 (1.075)      Data 1.026 (1.026)      Loss
0.3488 (0.3488)    Prec 90.625% (90.625%)
Epoch: [31][100/391]    Time 0.045 (0.059)      Data 0.002 (0.012)      Loss
0.3139 (0.3979)    Prec 88.281% (86.123%)
Epoch: [31][200/391]    Time 0.034 (0.053)      Data 0.002 (0.007)      Loss
0.4960 (0.3981)    Prec 82.031% (86.023%)
Epoch: [31][300/391]    Time 0.037 (0.048)      Data 0.001 (0.005)      Loss
0.3585 (0.4020)    Prec 85.156% (85.922%)
Validation starts
Test: [0/79]    Time 0.384 (0.384)      Loss 0.4215 (0.4215)    Prec 87.500%
(87.500%)
 * Prec 79.750%
best acc: 82.590000
Epoch: [32][0/391]      Time 0.957 (0.957)      Data 0.914 (0.914)      Loss
0.3458 (0.3458)    Prec 85.938% (85.938%)
Epoch: [32][100/391]    Time 0.043 (0.056)      Data 0.002 (0.011)      Loss
0.5772 (0.4068)    Prec 77.344% (85.953%)
Epoch: [32][200/391]    Time 0.045 (0.050)      Data 0.002 (0.007)      Loss
0.4162 (0.4055)    Prec 85.156% (85.875%)
Epoch: [32][300/391]    Time 0.044 (0.049)      Data 0.002 (0.005)      Loss
```

```
0.4506 (0.4080)    Prec 85.156% (85.870%)
Validation starts
Test: [0/79]    Time 0.349 (0.349)      Loss 0.5196 (0.5196)    Prec 82.812%
(82.812%)
 * Prec 80.170%
best acc: 82.590000
Epoch: [33][0/391]      Time 0.904 (0.904)      Data 0.850 (0.850)      Loss
0.3991 (0.3991)    Prec 84.375% (84.375%)
Epoch: [33][100/391]    Time 0.044 (0.054)      Data 0.002 (0.011)      Loss
0.4365 (0.3985)    Prec 85.156% (85.721%)
Epoch: [33][200/391]    Time 0.047 (0.050)      Data 0.002 (0.006)      Loss
0.4893 (0.3984)    Prec 84.375% (85.914%)
Epoch: [33][300/391]    Time 0.046 (0.049)      Data 0.002 (0.005)      Loss
0.4493 (0.3973)    Prec 86.719% (86.010%)
Validation starts
Test: [0/79]    Time 0.642 (0.642)      Loss 0.4456 (0.4456)    Prec 83.594%
(83.594%)
 * Prec 82.210%
best acc: 82.590000
Epoch: [34][0/391]      Time 0.410 (0.410)      Data 0.337 (0.337)      Loss
0.4469 (0.4469)    Prec 82.812% (82.812%)
Epoch: [34][100/391]    Time 0.052 (0.049)      Data 0.003 (0.005)      Loss
0.2801 (0.3969)    Prec 92.969% (86.046%)
Epoch: [34][200/391]    Time 0.044 (0.047)      Data 0.002 (0.004)      Loss
0.3937 (0.3940)    Prec 86.719% (86.217%)
Epoch: [34][300/391]    Time 0.040 (0.047)      Data 0.002 (0.003)      Loss
0.2769 (0.4020)    Prec 90.625% (86.065%)
Validation starts
Test: [0/79]    Time 0.343 (0.343)      Loss 0.4738 (0.4738)    Prec 84.375%
(84.375%)
 * Prec 80.660%
best acc: 82.590000
Epoch: [35][0/391]      Time 0.675 (0.675)      Data 0.618 (0.618)      Loss
0.3364 (0.3364)    Prec 88.281% (88.281%)
Epoch: [35][100/391]    Time 0.052 (0.053)      Data 0.002 (0.008)      Loss
0.3882 (0.3863)    Prec 87.500% (86.603%)
Epoch: [35][200/391]    Time 0.049 (0.050)      Data 0.002 (0.005)      Loss
0.3743 (0.3901)    Prec 85.938% (86.447%)
Epoch: [35][300/391]    Time 0.038 (0.049)      Data 0.002 (0.004)      Loss
0.5369 (0.3928)    Prec 82.031% (86.358%)
Validation starts
Test: [0/79]    Time 0.635 (0.635)      Loss 0.4625 (0.4625)    Prec 84.375%
(84.375%)
 * Prec 81.350%
best acc: 82.590000
Epoch: [36][0/391]      Time 0.711 (0.711)      Data 0.650 (0.650)      Loss
0.4067 (0.4067)    Prec 82.812% (82.812%)
Epoch: [36][100/391]    Time 0.039 (0.054)      Data 0.002 (0.009)      Loss
```

```
0.3652 (0.3816)    Prec 85.156% (86.726%)
Epoch: [36][200/391]    Time 0.039 (0.050)      Data 0.002 (0.006)      Loss
0.3161 (0.3817)    Prec 89.062% (86.614%)
Epoch: [36][300/391]    Time 0.057 (0.049)      Data 0.003 (0.004)      Loss
0.4354 (0.3825)    Prec 83.594% (86.682%)
Validation starts
Test: [0/79]    Time 0.461 (0.461)      Loss 0.5018 (0.5018)    Prec 85.938%
(85.938%)
 * Prec 83.090%
best acc: 83.090000
Epoch: [37][0/391]      Time 0.804 (0.804)      Data 0.742 (0.742)      Loss
0.3576 (0.3576)    Prec 85.156% (85.156%)
Epoch: [37][100/391]    Time 0.047 (0.053)      Data 0.002 (0.010)      Loss
0.5126 (0.3673)    Prec 81.250% (87.090%)
Epoch: [37][200/391]    Time 0.046 (0.051)      Data 0.002 (0.006)      Loss
0.3991 (0.3775)    Prec 86.719% (86.855%)
Epoch: [37][300/391]    Time 0.047 (0.050)      Data 0.003 (0.005)      Loss
0.3494 (0.3806)    Prec 89.844% (86.698%)
Validation starts
Test: [0/79]    Time 0.390 (0.390)      Loss 0.3730 (0.3730)    Prec 88.281%
(88.281%)
 * Prec 81.890%
best acc: 83.090000
Epoch: [38][0/391]      Time 0.690 (0.690)      Data 0.631 (0.631)      Loss
0.3630 (0.3630)    Prec 83.594% (83.594%)
Epoch: [38][100/391]    Time 0.042 (0.052)      Data 0.002 (0.008)      Loss
0.3859 (0.3644)    Prec 86.719% (86.904%)
Epoch: [38][200/391]    Time 0.043 (0.049)      Data 0.002 (0.005)      Loss
0.3925 (0.3653)    Prec 85.156% (87.026%)
Epoch: [38][300/391]    Time 0.048 (0.049)      Data 0.002 (0.004)      Loss
0.4904 (0.3755)    Prec 85.938% (86.789%)
Validation starts
Test: [0/79]    Time 0.353 (0.353)      Loss 0.5509 (0.5509)    Prec 81.250%
(81.250%)
 * Prec 82.010%
best acc: 83.090000
Epoch: [39][0/391]      Time 0.514 (0.514)      Data 0.460 (0.460)      Loss
0.4188 (0.4188)    Prec 84.375% (84.375%)
Epoch: [39][100/391]    Time 0.039 (0.051)      Data 0.002 (0.007)      Loss
0.3090 (0.3664)    Prec 87.500% (87.252%)
Epoch: [39][200/391]    Time 0.054 (0.049)      Data 0.003 (0.005)      Loss
0.3678 (0.3673)    Prec 89.844% (87.255%)
Epoch: [39][300/391]    Time 0.044 (0.049)      Data 0.002 (0.004)      Loss
0.3333 (0.3716)    Prec 89.844% (87.095%)
Validation starts
Test: [0/79]    Time 0.516 (0.516)      Loss 0.4395 (0.4395)    Prec 85.938%
(85.938%)
 * Prec 82.980%
```

```
best acc: 83.090000
Epoch: [40][0/391]      Time 0.537 (0.537)      Data 0.479 (0.479)      Loss
0.2937 (0.2937)    Prec 90.625% (90.625%)
Epoch: [40][100/391]    Time 0.044 (0.050)      Data 0.002 (0.007)      Loss
0.3555 (0.3718)    Prec 86.719% (87.430%)
Epoch: [40][200/391]    Time 0.039 (0.049)      Data 0.002 (0.005)      Loss
0.3408 (0.3743)    Prec 88.281% (87.057%)
Epoch: [40][300/391]    Time 0.047 (0.048)      Data 0.002 (0.004)      Loss
0.3529 (0.3723)    Prec 86.719% (87.142%)
Validation starts
Test: [0/79]    Time 0.619 (0.619)      Loss 0.3183 (0.3183)    Prec 89.844%
(89.844%)
 * Prec 84.470%
best acc: 84.470000
Epoch: [41][0/391]      Time 0.603 (0.603)      Data 0.507 (0.507)      Loss
0.3871 (0.3871)    Prec 86.719% (86.719%)
Epoch: [41][100/391]    Time 0.050 (0.052)      Data 0.003 (0.007)      Loss
0.3077 (0.3632)    Prec 86.719% (87.222%)
Epoch: [41][200/391]    Time 0.037 (0.049)      Data 0.002 (0.005)      Loss
0.4430 (0.3643)    Prec 85.938% (87.158%)
Epoch: [41][300/391]    Time 0.048 (0.048)      Data 0.002 (0.004)      Loss
0.3366 (0.3700)    Prec 88.281% (86.996%)
Validation starts
Test: [0/79]    Time 0.404 (0.404)      Loss 0.5803 (0.5803)    Prec 81.250%
(81.250%)
 * Prec 81.140%
best acc: 84.470000
Epoch: [42][0/391]      Time 0.773 (0.773)      Data 0.716 (0.716)      Loss
0.4345 (0.4345)    Prec 82.812% (82.812%)
Epoch: [42][100/391]    Time 0.049 (0.053)      Data 0.003 (0.009)      Loss
0.3845 (0.3715)    Prec 86.719% (87.098%)
Epoch: [42][200/391]    Time 0.041 (0.048)      Data 0.002 (0.006)      Loss
0.3604 (0.3627)    Prec 88.281% (87.376%)
Epoch: [42][300/391]    Time 0.040 (0.047)      Data 0.002 (0.004)      Loss
0.4031 (0.3616)    Prec 85.156% (87.373%)
Validation starts
Test: [0/79]    Time 0.253 (0.253)      Loss 0.2920 (0.2920)    Prec 89.062%
(89.062%)
 * Prec 83.490%
best acc: 84.470000
Epoch: [43][0/391]      Time 0.532 (0.532)      Data 0.472 (0.472)      Loss
0.4248 (0.4248)    Prec 83.594% (83.594%)
Epoch: [43][100/391]    Time 0.044 (0.049)      Data 0.002 (0.007)      Loss
0.2562 (0.3523)    Prec 91.406% (87.469%)
Epoch: [43][200/391]    Time 0.059 (0.047)      Data 0.003 (0.005)      Loss
0.4075 (0.3552)    Prec 84.375% (87.356%)
Epoch: [43][300/391]    Time 0.050 (0.047)      Data 0.002 (0.004)      Loss
0.3064 (0.3583)    Prec 89.062% (87.324%)
```

```
Validation starts
Test: [0/79]    Time 0.621 (0.621)      Loss 0.4258 (0.4258)    Prec 87.500%
(87.500%)
 * Prec 82.130%
best acc: 84.470000
Epoch: [44][0/391]      Time 0.880 (0.880)      Data 0.823 (0.823)      Loss
0.3316 (0.3316)    Prec 86.719% (86.719%)
Epoch: [44][100/391]    Time 0.043 (0.056)      Data 0.002 (0.010)      Loss
0.4665 (0.3344)    Prec 83.594% (88.359%)
Epoch: [44][200/391]    Time 0.046 (0.050)      Data 0.002 (0.006)      Loss
0.3048 (0.3454)    Prec 90.625% (87.947%)
Epoch: [44][300/391]    Time 0.047 (0.049)      Data 0.002 (0.005)      Loss
0.4379 (0.3484)    Prec 82.031% (87.739%)
Validation starts
Test: [0/79]    Time 0.633 (0.633)      Loss 0.4757 (0.4757)    Prec 85.938%
(85.938%)
 * Prec 83.240%
best acc: 84.470000
Epoch: [45][0/391]      Time 0.637 (0.637)      Data 0.579 (0.579)      Loss
0.1772 (0.1772)    Prec 93.750% (93.750%)
Epoch: [45][100/391]    Time 0.046 (0.052)      Data 0.002 (0.008)      Loss
0.3743 (0.3334)    Prec 85.938% (88.150%)
Epoch: [45][200/391]    Time 0.048 (0.050)      Data 0.002 (0.005)      Loss
0.2952 (0.3431)    Prec 89.844% (87.803%)
Epoch: [45][300/391]    Time 0.046 (0.048)      Data 0.002 (0.004)      Loss
0.4560 (0.3523)    Prec 85.938% (87.503%)
Validation starts
Test: [0/79]    Time 0.528 (0.528)      Loss 0.4761 (0.4761)    Prec 85.156%
(85.156%)
 * Prec 83.370%
best acc: 84.470000
Epoch: [46][0/391]      Time 0.860 (0.860)      Data 0.799 (0.799)      Loss
0.3677 (0.3677)    Prec 87.500% (87.500%)
Epoch: [46][100/391]    Time 0.049 (0.056)      Data 0.002 (0.010)      Loss
0.4591 (0.3518)    Prec 83.594% (87.438%)
Epoch: [46][200/391]    Time 0.046 (0.051)      Data 0.002 (0.006)      Loss
0.4097 (0.3537)    Prec 86.719% (87.352%)
Epoch: [46][300/391]    Time 0.052 (0.049)      Data 0.002 (0.005)      Loss
0.3269 (0.3565)    Prec 89.844% (87.344%)
Validation starts
Test: [0/79]    Time 0.522 (0.522)      Loss 0.4968 (0.4968)    Prec 82.031%
(82.031%)
 * Prec 80.740%
best acc: 84.470000
Epoch: [47][0/391]      Time 0.603 (0.603)      Data 0.521 (0.521)      Loss
0.3529 (0.3529)    Prec 87.500% (87.500%)
Epoch: [47][100/391]    Time 0.052 (0.053)      Data 0.003 (0.007)      Loss
0.3608 (0.3433)    Prec 89.062% (87.871%)
```

```
Epoch: [47][200/391]    Time 0.046 (0.052)      Data 0.003 (0.005)      Loss
0.3699 (0.3537)    Prec 82.812% (87.395%)
Epoch: [47][300/391]    Time 0.060 (0.051)      Data 0.003 (0.004)      Loss
0.2322 (0.3543)    Prec 92.969% (87.440%)
Validation starts
Test: [0/79]    Time 0.592 (0.592)      Loss 0.4180 (0.4180)    Prec 85.938%
(85.938%)
 * Prec 83.870%
best acc: 84.470000
Epoch: [48][0/391]      Time 0.651 (0.651)      Data 0.591 (0.591)      Loss
0.2269 (0.2269)    Prec 92.969% (92.969%)
Epoch: [48][100/391]    Time 0.045 (0.055)      Data 0.002 (0.008)      Loss
0.3518 (0.3252)    Prec 87.500% (88.730%)
Epoch: [48][200/391]    Time 0.053 (0.053)      Data 0.002 (0.005)      Loss
0.3392 (0.3319)    Prec 91.406% (88.483%)
Epoch: [48][300/391]    Time 0.052 (0.053)      Data 0.002 (0.004)      Loss
0.3975 (0.3392)    Prec 86.719% (88.219%)
Validation starts
Test: [0/79]    Time 0.300 (0.300)      Loss 0.4234 (0.4234)    Prec 85.938%
(85.938%)
 * Prec 82.790%
best acc: 84.470000
Epoch: [49][0/391]      Time 0.677 (0.677)      Data 0.613 (0.613)      Loss
0.3905 (0.3905)    Prec 86.719% (86.719%)
Epoch: [49][100/391]    Time 0.054 (0.058)      Data 0.003 (0.008)      Loss
0.3581 (0.3335)    Prec 87.500% (88.475%)
Epoch: [49][200/391]    Time 0.047 (0.055)      Data 0.003 (0.005)      Loss
0.2844 (0.3342)    Prec 89.844% (88.242%)
Epoch: [49][300/391]    Time 0.050 (0.054)      Data 0.002 (0.004)      Loss
0.3051 (0.3379)    Prec 91.406% (88.123%)
Validation starts
Test: [0/79]    Time 0.403 (0.403)      Loss 0.3672 (0.3672)    Prec 89.844%
(89.844%)
 * Prec 82.540%
best acc: 84.470000
Epoch: [50][0/391]      Time 0.582 (0.582)      Data 0.520 (0.520)      Loss
0.3456 (0.3456)    Prec 87.500% (87.500%)
Epoch: [50][100/391]    Time 0.051 (0.056)      Data 0.002 (0.007)      Loss
0.2424 (0.3101)    Prec 92.188% (89.264%)
Epoch: [50][200/391]    Time 0.052 (0.051)      Data 0.003 (0.005)      Loss
0.3192 (0.3234)    Prec 88.281% (88.697%)
Epoch: [50][300/391]    Time 0.045 (0.049)      Data 0.002 (0.004)      Loss
0.2739 (0.3293)    Prec 92.969% (88.520%)
Validation starts
Test: [0/79]    Time 0.476 (0.476)      Loss 0.4115 (0.4115)    Prec 87.500%
(87.500%)
 * Prec 84.020%
best acc: 84.470000
```

```
Epoch: [51][0/391]      Time 0.508 (0.508)      Data 0.434 (0.434)      Loss
0.4221 (0.4221)    Prec 88.281% (88.281%)
Epoch: [51][100/391]    Time 0.050 (0.052)      Data 0.003 (0.006)      Loss
0.3531 (0.3239)    Prec 88.281% (88.629%)
Epoch: [51][200/391]    Time 0.040 (0.049)      Data 0.002 (0.004)      Loss
0.3653 (0.3220)    Prec 85.938% (88.720%)
Epoch: [51][300/391]    Time 0.046 (0.048)      Data 0.002 (0.004)      Loss
0.2573 (0.3311)    Prec 90.625% (88.476%)
Validation starts
Test: [0/79]    Time 0.346 (0.346)      Loss 0.4217 (0.4217)    Prec 87.500%
(87.500%)
 * Prec 83.550%
best acc: 84.470000
Epoch: [52][0/391]      Time 0.880 (0.880)      Data 0.820 (0.820)      Loss
0.2934 (0.2934)    Prec 89.062% (89.062%)
Epoch: [52][100/391]    Time 0.047 (0.056)      Data 0.002 (0.010)      Loss
0.3247 (0.3150)    Prec 88.281% (88.908%)
Epoch: [52][200/391]    Time 0.047 (0.051)      Data 0.002 (0.006)      Loss
0.3337 (0.3255)    Prec 85.156% (88.685%)
Epoch: [52][300/391]    Time 0.046 (0.049)      Data 0.003 (0.005)      Loss
0.3747 (0.3271)    Prec 83.594% (88.582%)
Validation starts
Test: [0/79]    Time 0.500 (0.500)      Loss 0.3796 (0.3796)    Prec 88.281%
(88.281%)
 * Prec 84.140%
best acc: 84.470000
Epoch: [53][0/391]      Time 0.758 (0.758)      Data 0.702 (0.702)      Loss
0.2253 (0.2253)    Prec 91.406% (91.406%)
Epoch: [53][100/391]    Time 0.048 (0.054)      Data 0.002 (0.009)      Loss
0.3006 (0.3150)    Prec 87.500% (89.318%)
Epoch: [53][200/391]    Time 0.045 (0.048)      Data 0.002 (0.006)      Loss
0.2982 (0.3238)    Prec 87.500% (88.934%)
Epoch: [53][300/391]    Time 0.045 (0.047)      Data 0.002 (0.004)      Loss
0.3785 (0.3285)    Prec 83.594% (88.725%)
Validation starts
Test: [0/79]    Time 0.717 (0.717)      Loss 0.4200 (0.4200)    Prec 87.500%
(87.500%)
 * Prec 84.170%
best acc: 84.470000
Epoch: [54][0/391]      Time 0.802 (0.802)      Data 0.743 (0.743)      Loss
0.3566 (0.3566)    Prec 88.281% (88.281%)
Epoch: [54][100/391]    Time 0.051 (0.053)      Data 0.002 (0.010)      Loss
0.2520 (0.3162)    Prec 90.625% (88.946%)
Epoch: [54][200/391]    Time 0.052 (0.053)      Data 0.002 (0.006)      Loss
0.5052 (0.3290)    Prec 79.688% (88.444%)
Epoch: [54][300/391]    Time 0.051 (0.054)      Data 0.003 (0.005)      Loss
0.2723 (0.3280)    Prec 92.969% (88.406%)
Validation starts
```

```
Test: [0/79]    Time 0.357 (0.357)      Loss 0.4325 (0.4325)    Prec 86.719%
(86.719%)
 * Prec 83.540%
best acc: 84.470000
Epoch: [55][0/391]      Time 0.642 (0.642)      Data 0.599 (0.599)      Loss
0.2571 (0.2571)    Prec 90.625% (90.625%)
Epoch: [55][100/391]    Time 0.042 (0.050)      Data 0.002 (0.008)      Loss
0.3850 (0.3103)    Prec 87.500% (89.001%)
Epoch: [55][200/391]    Time 0.048 (0.048)      Data 0.002 (0.005)      Loss
0.3162 (0.3176)    Prec 91.406% (88.755%)
Epoch: [55][300/391]    Time 0.042 (0.048)      Data 0.002 (0.004)      Loss
0.2824 (0.3256)    Prec 91.406% (88.473%)
Validation starts
Test: [0/79]    Time 0.620 (0.620)      Loss 0.4845 (0.4845)    Prec 85.156%
(85.156%)
 * Prec 80.570%
best acc: 84.470000
Epoch: [56][0/391]      Time 0.594 (0.594)      Data 0.532 (0.532)      Loss
0.3053 (0.3053)    Prec 88.281% (88.281%)
Epoch: [56][100/391]    Time 0.047 (0.053)      Data 0.002 (0.008)      Loss
0.1939 (0.3223)    Prec 92.969% (88.653%)
Epoch: [56][200/391]    Time 0.047 (0.049)      Data 0.002 (0.005)      Loss
0.2991 (0.3232)    Prec 89.844% (88.592%)
Epoch: [56][300/391]    Time 0.041 (0.048)      Data 0.002 (0.004)      Loss
0.2055 (0.3248)    Prec 93.750% (88.445%)
Validation starts
Test: [0/79]    Time 0.325 (0.325)      Loss 0.7412 (0.7412)    Prec 79.688%
(79.688%)
 * Prec 78.560%
best acc: 84.470000
Epoch: [57][0/391]      Time 0.532 (0.532)      Data 0.489 (0.489)      Loss
0.2709 (0.2709)    Prec 90.625% (90.625%)
Epoch: [57][100/391]    Time 0.046 (0.051)      Data 0.002 (0.007)      Loss
0.3686 (0.3043)    Prec 85.938% (89.418%)
Epoch: [57][200/391]    Time 0.041 (0.048)      Data 0.002 (0.004)      Loss
0.3585 (0.3146)    Prec 85.938% (88.798%)
Epoch: [57][300/391]    Time 0.051 (0.048)      Data 0.003 (0.004)      Loss
0.2051 (0.3185)    Prec 93.750% (88.746%)
Validation starts
Test: [0/79]    Time 0.435 (0.435)      Loss 0.5469 (0.5469)    Prec 85.156%
(85.156%)
 * Prec 83.590%
best acc: 84.470000
Epoch: [58][0/391]      Time 0.766 (0.766)      Data 0.705 (0.705)      Loss
0.3135 (0.3135)    Prec 89.844% (89.844%)
Epoch: [58][100/391]    Time 0.045 (0.052)      Data 0.002 (0.009)      Loss
0.3136 (0.3062)    Prec 85.938% (88.885%)
Epoch: [58][200/391]    Time 0.046 (0.049)      Data 0.004 (0.006)      Loss
```

```
0.2968 (0.3204)    Prec 89.062% (88.569%)
Epoch: [58][300/391]    Time 0.048 (0.048)      Data 0.003 (0.005)      Loss
0.2160 (0.3216)    Prec 94.531% (88.697%)
Validation starts
Test: [0/79]    Time 0.626 (0.626)      Loss 0.3330 (0.3330)    Prec 87.500%
(87.500%)
 * Prec 83.680%
best acc: 84.470000
Epoch: [59][0/391]      Time 0.499 (0.499)      Data 0.442 (0.442)      Loss
0.3729 (0.3729)    Prec 85.938% (85.938%)
Epoch: [59][100/391]    Time 0.036 (0.049)      Data 0.002 (0.007)      Loss
0.3455 (0.3025)    Prec 88.281% (89.612%)
Epoch: [59][200/391]    Time 0.043 (0.047)      Data 0.002 (0.004)      Loss
0.2668 (0.3069)    Prec 90.625% (89.230%)
Epoch: [59][300/391]    Time 0.036 (0.047)      Data 0.002 (0.004)      Loss
0.2658 (0.3125)    Prec 90.625% (89.083%)
Validation starts
Test: [0/79]    Time 0.372 (0.372)      Loss 0.4099 (0.4099)    Prec 88.281%
(88.281%)
 * Prec 82.550%
best acc: 84.470000
Epoch: [60][0/391]      Time 0.781 (0.781)      Data 0.724 (0.724)      Loss
0.2489 (0.2489)    Prec 93.750% (93.750%)
Epoch: [60][100/391]    Time 0.039 (0.054)      Data 0.002 (0.009)      Loss
0.2362 (0.3024)    Prec 89.844% (89.295%)
Epoch: [60][200/391]    Time 0.048 (0.050)      Data 0.002 (0.006)      Loss
0.3708 (0.3081)    Prec 87.500% (89.144%)
Epoch: [60][300/391]    Time 0.046 (0.050)      Data 0.002 (0.005)      Loss
0.4169 (0.3151)    Prec 85.156% (88.922%)
Validation starts
Test: [0/79]    Time 0.997 (0.997)      Loss 0.3705 (0.3705)    Prec 86.719%
(86.719%)
 * Prec 85.510%
best acc: 85.510000
Epoch: [61][0/391]      Time 0.534 (0.534)      Data 0.477 (0.477)      Loss
0.3589 (0.3589)    Prec 87.500% (87.500%)
Epoch: [61][100/391]    Time 0.052 (0.052)      Data 0.003 (0.007)      Loss
0.3037 (0.3034)    Prec 88.281% (89.217%)
Epoch: [61][200/391]    Time 0.044 (0.048)      Data 0.002 (0.005)      Loss
0.2808 (0.3073)    Prec 89.062% (89.125%)
Epoch: [61][300/391]    Time 0.043 (0.047)      Data 0.002 (0.004)      Loss
0.3935 (0.3110)    Prec 88.281% (89.024%)
Validation starts
Test: [0/79]    Time 0.488 (0.488)      Loss 0.3072 (0.3072)    Prec 89.062%
(89.062%)
 * Prec 85.460%
best acc: 85.510000
Epoch: [62][0/391]      Time 0.604 (0.604)      Data 0.509 (0.509)      Loss
```

```
0.2207 (0.2207)    Prec 93.750% (93.750%)
Epoch: [62][100/391]    Time 0.036 (0.050)      Data 0.002 (0.007)      Loss
0.3309 (0.2777)    Prec 90.625% (90.114%)
Epoch: [62][200/391]    Time 0.050 (0.046)      Data 0.002 (0.005)      Loss
0.3072 (0.2872)    Prec 90.625% (89.871%)
Epoch: [62][300/391]    Time 0.038 (0.047)      Data 0.002 (0.004)      Loss
0.2786 (0.2982)    Prec 92.188% (89.525%)
Validation starts
Test: [0/79]    Time 0.658 (0.658)      Loss 0.5027 (0.5027)    Prec 85.938%
(85.938%)
 * Prec 83.840%
best acc: 85.510000
Epoch: [63][0/391]      Time 0.864 (0.864)      Data 0.809 (0.809)      Loss
0.3828 (0.3828)    Prec 86.719% (86.719%)
Epoch: [63][100/391]    Time 0.044 (0.053)      Data 0.002 (0.010)      Loss
0.2619 (0.2952)    Prec 90.625% (89.720%)
Epoch: [63][200/391]    Time 0.044 (0.050)      Data 0.002 (0.006)      Loss
0.2506 (0.3013)    Prec 90.625% (89.486%)
Epoch: [63][300/391]    Time 0.051 (0.050)      Data 0.003 (0.005)      Loss
0.2911 (0.3055)    Prec 88.281% (89.351%)
Validation starts
Test: [0/79]    Time 1.406 (1.406)      Loss 0.3286 (0.3286)    Prec 87.500%
(87.500%)
 * Prec 84.400%
best acc: 85.510000
Epoch: [64][0/391]      Time 0.791 (0.791)      Data 0.723 (0.723)      Loss
0.3554 (0.3554)    Prec 88.281% (88.281%)
Epoch: [64][100/391]    Time 0.055 (0.057)      Data 0.002 (0.010)      Loss
0.2840 (0.3038)    Prec 89.844% (89.372%)
Epoch: [64][200/391]    Time 0.053 (0.054)      Data 0.002 (0.006)      Loss
0.3025 (0.3067)    Prec 89.844% (89.101%)
Epoch: [64][300/391]    Time 0.049 (0.052)      Data 0.002 (0.005)      Loss
0.3112 (0.3116)    Prec 90.625% (89.000%)
Validation starts
Test: [0/79]    Time 0.601 (0.601)      Loss 0.4902 (0.4902)    Prec 84.375%
(84.375%)
 * Prec 82.980%
best acc: 85.510000
Epoch: [65][0/391]      Time 0.831 (0.831)      Data 0.752 (0.752)      Loss
0.2716 (0.2716)    Prec 91.406% (91.406%)
Epoch: [65][100/391]    Time 0.051 (0.055)      Data 0.002 (0.010)      Loss
0.2901 (0.3099)    Prec 91.406% (89.093%)
Epoch: [65][200/391]    Time 0.049 (0.050)      Data 0.003 (0.006)      Loss
0.2351 (0.3052)    Prec 92.969% (89.164%)
Epoch: [65][300/391]    Time 0.042 (0.049)      Data 0.002 (0.005)      Loss
0.2395 (0.3070)    Prec 93.750% (89.177%)
Validation starts
Test: [0/79]    Time 0.426 (0.426)      Loss 0.4739 (0.4739)    Prec 87.500%
```

```
(87.500%)
 * Prec 83.600%
best acc: 85.510000
Epoch: [66][0/391]      Time 0.549 (0.549)      Data 0.488 (0.488)      Loss
0.3742 (0.3742)    Prec 89.844% (89.844%)
Epoch: [66][100/391]    Time 0.045 (0.049)      Data 0.002 (0.007)      Loss
0.3101 (0.2998)    Prec 87.500% (89.550%)
Epoch: [66][200/391]    Time 0.043 (0.048)      Data 0.002 (0.005)      Loss
0.3916 (0.2991)    Prec 88.281% (89.614%)
Epoch: [66][300/391]    Time 0.043 (0.048)      Data 0.002 (0.004)      Loss
0.2400 (0.3028)    Prec 90.625% (89.452%)
Validation starts
Test: [0/79]    Time 0.306 (0.306)      Loss 0.5583 (0.5583)    Prec 83.594%
(83.594%)
 * Prec 82.010%
best acc: 85.510000
Epoch: [67][0/391]      Time 0.666 (0.666)      Data 0.609 (0.609)      Loss
0.3316 (0.3316)    Prec 88.281% (88.281%)
Epoch: [67][100/391]    Time 0.037 (0.050)      Data 0.002 (0.008)      Loss
0.3624 (0.3028)    Prec 88.281% (89.248%)
Epoch: [67][200/391]    Time 0.040 (0.047)      Data 0.002 (0.005)      Loss
0.4057 (0.3070)    Prec 87.500% (89.051%)
Epoch: [67][300/391]    Time 0.040 (0.046)      Data 0.002 (0.004)      Loss
0.2984 (0.3082)    Prec 88.281% (88.977%)
Validation starts
Test: [0/79]    Time 0.511 (0.511)      Loss 0.3406 (0.3406)    Prec 90.625%
(90.625%)
 * Prec 83.550%
best acc: 85.510000
Epoch: [68][0/391]      Time 0.626 (0.626)      Data 0.570 (0.570)      Loss
0.3710 (0.3710)    Prec 86.719% (86.719%)
Epoch: [68][100/391]    Time 0.043 (0.053)      Data 0.003 (0.008)      Loss
0.2902 (0.2911)    Prec 86.719% (89.558%)
Epoch: [68][200/391]    Time 0.049 (0.050)      Data 0.002 (0.005)      Loss
0.2174 (0.2907)    Prec 90.625% (89.743%)
Epoch: [68][300/391]    Time 0.040 (0.049)      Data 0.002 (0.004)      Loss
0.2812 (0.2997)    Prec 89.062% (89.457%)
Validation starts
Test: [0/79]    Time 0.448 (0.448)      Loss 0.4536 (0.4536)    Prec 82.812%
(82.812%)
 * Prec 80.730%
best acc: 85.510000
Epoch: [69][0/391]      Time 0.539 (0.539)      Data 0.483 (0.483)      Loss
0.3082 (0.3082)    Prec 86.719% (86.719%)
Epoch: [69][100/391]    Time 0.053 (0.054)      Data 0.002 (0.007)      Loss
0.2220 (0.2984)    Prec 89.062% (89.256%)
Epoch: [69][200/391]    Time 0.037 (0.049)      Data 0.001 (0.005)      Loss
0.2510 (0.2925)    Prec 92.969% (89.700%)
```

```
Epoch: [69][300/391]    Time 0.050 (0.048)      Data 0.002 (0.004)      Loss
0.2581 (0.2967)    Prec 89.062% (89.605%)
Validation starts
Test: [0/79]    Time 0.412 (0.412)      Loss 0.4253 (0.4253)    Prec 87.500%
(87.500%)
 * Prec 83.910%
best acc: 85.510000
Epoch: [70][0/391]      Time 0.722 (0.722)      Data 0.667 (0.667)      Loss
0.2560 (0.2560)    Prec 89.844% (89.844%)
Epoch: [70][100/391]    Time 0.047 (0.056)      Data 0.002 (0.009)      Loss
0.2322 (0.2816)    Prec 92.188% (90.370%)
Epoch: [70][200/391]    Time 0.041 (0.051)      Data 0.002 (0.006)      Loss
0.2810 (0.2949)    Prec 89.062% (89.848%)
Epoch: [70][300/391]    Time 0.052 (0.049)      Data 0.002 (0.004)      Loss
0.2752 (0.2963)    Prec 88.281% (89.740%)
Validation starts
Test: [0/79]    Time 0.597 (0.597)      Loss 0.7280 (0.7280)    Prec 77.344%
(77.344%)
 * Prec 79.140%
best acc: 85.510000
Epoch: [71][0/391]      Time 0.715 (0.715)      Data 0.655 (0.655)      Loss
0.3989 (0.3989)    Prec 85.156% (85.156%)
Epoch: [71][100/391]    Time 0.048 (0.053)      Data 0.002 (0.009)      Loss
0.3641 (0.3016)    Prec 88.281% (89.364%)
Epoch: [71][200/391]    Time 0.042 (0.050)      Data 0.002 (0.005)      Loss
0.2851 (0.3011)    Prec 89.062% (89.455%)
Epoch: [71][300/391]    Time 0.044 (0.048)      Data 0.002 (0.004)      Loss
0.2500 (0.3009)    Prec 90.625% (89.400%)
Validation starts
Test: [0/79]    Time 0.542 (0.542)      Loss 0.3181 (0.3181)    Prec 88.281%
(88.281%)
 * Prec 84.820%
best acc: 85.510000
Epoch: [72][0/391]      Time 1.001 (1.001)      Data 0.940 (0.940)      Loss
0.3465 (0.3465)    Prec 87.500% (87.500%)
Epoch: [72][100/391]    Time 0.044 (0.056)      Data 0.002 (0.011)      Loss
0.2614 (0.2869)    Prec 92.188% (89.913%)
Epoch: [72][200/391]    Time 0.046 (0.051)      Data 0.002 (0.007)      Loss
0.2222 (0.2884)    Prec 92.188% (89.894%)
Epoch: [72][300/391]    Time 0.052 (0.049)      Data 0.003 (0.005)      Loss
0.2551 (0.2939)    Prec 91.406% (89.652%)
Validation starts
Test: [0/79]    Time 0.420 (0.420)      Loss 0.4244 (0.4244)    Prec 87.500%
(87.500%)
 * Prec 83.790%
best acc: 85.510000
Epoch: [73][0/391]      Time 0.661 (0.661)      Data 0.603 (0.603)      Loss
0.3386 (0.3386)    Prec 91.406% (91.406%)
```

```
Epoch: [73][100/391]    Time 0.050 (0.051)      Data 0.002 (0.008)      Loss
0.2736 (0.3029)    Prec 89.062% (89.209%)
Epoch: [73][200/391]    Time 0.046 (0.049)      Data 0.002 (0.005)      Loss
0.3404 (0.3001)    Prec 91.406% (89.424%)
Epoch: [73][300/391]    Time 0.053 (0.049)      Data 0.003 (0.004)      Loss
0.2095 (0.2965)    Prec 93.750% (89.535%)
Validation starts
Test: [0/79]    Time 0.415 (0.415)      Loss 0.4080 (0.4080)    Prec 89.844%
(89.844%)
 * Prec 84.750%
best acc: 85.510000
Epoch: [74][0/391]      Time 0.535 (0.535)      Data 0.479 (0.479)      Loss
0.2227 (0.2227)    Prec 94.531% (94.531%)
Epoch: [74][100/391]    Time 0.057 (0.053)      Data 0.003 (0.007)      Loss
0.2739 (0.2931)    Prec 89.844% (89.890%)
Epoch: [74][200/391]    Time 0.045 (0.050)      Data 0.002 (0.005)      Loss
0.2766 (0.2885)    Prec 88.281% (89.758%)
Epoch: [74][300/391]    Time 0.045 (0.049)      Data 0.002 (0.004)      Loss
0.2483 (0.2885)    Prec 91.406% (89.766%)
Validation starts
Test: [0/79]    Time 0.340 (0.340)      Loss 0.4275 (0.4275)    Prec 84.375%
(84.375%)
 * Prec 84.850%
best acc: 85.510000
Epoch: [75][0/391]      Time 0.689 (0.689)      Data 0.629 (0.629)      Loss
0.2881 (0.2881)    Prec 86.719% (86.719%)
Epoch: [75][100/391]    Time 0.049 (0.052)      Data 0.002 (0.008)      Loss
0.3196 (0.2886)    Prec 87.500% (90.114%)
Epoch: [75][200/391]    Time 0.062 (0.049)      Data 0.003 (0.005)      Loss
0.2760 (0.2827)    Prec 90.625% (90.213%)
Epoch: [75][300/391]    Time 0.043 (0.049)      Data 0.002 (0.004)      Loss
0.2639 (0.2834)    Prec 89.844% (90.038%)
Validation starts
Test: [0/79]    Time 0.402 (0.402)      Loss 0.4012 (0.4012)    Prec 86.719%
(86.719%)
 * Prec 84.930%
best acc: 85.510000
Epoch: [76][0/391]      Time 0.536 (0.536)      Data 0.475 (0.475)      Loss
0.2370 (0.2370)    Prec 95.312% (95.312%)
Epoch: [76][100/391]    Time 0.050 (0.052)      Data 0.003 (0.007)      Loss
0.2463 (0.2687)    Prec 89.844% (90.733%)
Epoch: [76][200/391]    Time 0.058 (0.049)      Data 0.003 (0.005)      Loss
0.3314 (0.2784)    Prec 89.062% (90.314%)
Epoch: [76][300/391]    Time 0.043 (0.047)      Data 0.002 (0.004)      Loss
0.2450 (0.2789)    Prec 89.062% (90.207%)
Validation starts
Test: [0/79]    Time 0.509 (0.509)      Loss 0.5306 (0.5306)    Prec 85.938%
(85.938%)
```

```
 * Prec 83.070%
best acc: 85.510000
Epoch: [77][0/391]      Time 0.747 (0.747)      Data 0.682 (0.682)      Loss
0.1254 (0.1254)    Prec 98.438% (98.438%)
Epoch: [77][100/391]    Time 0.041 (0.050)      Data 0.002 (0.009)      Loss
0.2512 (0.2830)    Prec 95.312% (89.859%)
Epoch: [77][200/391]    Time 0.042 (0.047)      Data 0.002 (0.005)      Loss
0.4202 (0.2913)    Prec 85.156% (89.564%)
Epoch: [77][300/391]    Time 0.046 (0.046)      Data 0.002 (0.004)      Loss
0.2912 (0.2918)    Prec 90.625% (89.665%)
Validation starts
Test: [0/79]    Time 0.395 (0.395)      Loss 0.3673 (0.3673)    Prec 86.719%
(86.719%)
 * Prec 84.760%
best acc: 85.510000
Epoch: [78][0/391]      Time 0.783 (0.783)      Data 0.742 (0.742)      Loss
0.3878 (0.3878)    Prec 85.156% (85.156%)
Epoch: [78][100/391]    Time 0.046 (0.052)      Data 0.002 (0.010)      Loss
0.3510 (0.2769)    Prec 86.719% (90.277%)
Epoch: [78][200/391]    Time 0.048 (0.049)      Data 0.002 (0.006)      Loss
0.3533 (0.2824)    Prec 89.062% (90.081%)
Epoch: [78][300/391]    Time 0.044 (0.048)      Data 0.002 (0.005)      Loss
0.3673 (0.2863)    Prec 87.500% (89.974%)
Validation starts
Test: [0/79]    Time 0.305 (0.305)      Loss 0.3759 (0.3759)    Prec 89.062%
(89.062%)
 * Prec 86.240%
best acc: 86.240000
Epoch: [79][0/391]      Time 0.864 (0.864)      Data 0.822 (0.822)      Loss
0.2965 (0.2965)    Prec 89.062% (89.062%)
Epoch: [79][100/391]    Time 0.044 (0.054)      Data 0.002 (0.010)      Loss
0.2031 (0.2700)    Prec 95.312% (90.470%)
Epoch: [79][200/391]    Time 0.053 (0.051)      Data 0.003 (0.006)      Loss
0.3291 (0.2763)    Prec 86.719% (90.159%)
Epoch: [79][300/391]    Time 0.047 (0.049)      Data 0.002 (0.005)      Loss
0.3990 (0.2849)    Prec 87.500% (89.883%)
Validation starts
Test: [0/79]    Time 0.526 (0.526)      Loss 0.3380 (0.3380)    Prec 89.844%
(89.844%)
 * Prec 84.940%
best acc: 86.240000
Epoch: [80][0/391]      Time 0.879 (0.879)      Data 0.817 (0.817)      Loss
0.2146 (0.2146)    Prec 92.969% (92.969%)
Epoch: [80][100/391]    Time 0.048 (0.055)      Data 0.002 (0.010)      Loss
0.1866 (0.2169)    Prec 94.531% (92.481%)
Epoch: [80][200/391]    Time 0.048 (0.051)      Data 0.002 (0.006)      Loss
0.1869 (0.2136)    Prec 92.188% (92.673%)
Epoch: [80][300/391]    Time 0.047 (0.049)      Data 0.002 (0.005)      Loss
```

```
0.2242 (0.2098)    Prec 93.750% (92.839%)
Validation starts
Test: [0/79]    Time 0.306 (0.306)      Loss 0.2892 (0.2892)    Prec 91.406%
(91.406%)
 * Prec 88.600%
best acc: 88.600000
Epoch: [81][0/391]      Time 0.596 (0.596)      Data 0.536 (0.536)      Loss
0.1780 (0.1780)    Prec 95.312% (95.312%)
Epoch: [81][100/391]    Time 0.049 (0.051)      Data 0.002 (0.008)      Loss
0.1882 (0.1876)    Prec 93.750% (93.402%)
Epoch: [81][200/391]    Time 0.048 (0.049)      Data 0.002 (0.005)      Loss
0.2598 (0.1914)    Prec 89.062% (93.303%)
Epoch: [81][300/391]    Time 0.046 (0.049)      Data 0.002 (0.004)      Loss
0.1360 (0.1892)    Prec 94.531% (93.420%)
Validation starts
Test: [0/79]    Time 0.487 (0.487)      Loss 0.2913 (0.2913)    Prec 90.625%
(90.625%)
 * Prec 88.710%
best acc: 88.710000
Epoch: [82][0/391]      Time 0.681 (0.681)      Data 0.621 (0.621)      Loss
0.1243 (0.1243)    Prec 96.875% (96.875%)
Epoch: [82][100/391]    Time 0.061 (0.054)      Data 0.003 (0.008)      Loss
0.2488 (0.1748)    Prec 89.844% (94.052%)
Epoch: [82][200/391]    Time 0.050 (0.050)      Data 0.003 (0.005)      Loss
0.1624 (0.1737)    Prec 96.094% (94.038%)
Epoch: [82][300/391]    Time 0.053 (0.050)      Data 0.003 (0.004)      Loss
0.1328 (0.1720)    Prec 96.094% (94.157%)
Validation starts
Test: [0/79]    Time 0.575 (0.575)      Loss 0.3005 (0.3005)    Prec 89.844%
(89.844%)
 * Prec 88.600%
best acc: 88.710000
Epoch: [83][0/391]      Time 0.665 (0.665)      Data 0.603 (0.603)      Loss
0.1850 (0.1850)    Prec 93.750% (93.750%)
Epoch: [83][100/391]    Time 0.043 (0.053)      Data 0.002 (0.008)      Loss
0.1257 (0.1765)    Prec 97.656% (94.137%)
Epoch: [83][200/391]    Time 0.051 (0.050)      Data 0.002 (0.005)      Loss
0.1478 (0.1751)    Prec 94.531% (94.080%)
Epoch: [83][300/391]    Time 0.051 (0.050)      Data 0.002 (0.004)      Loss
0.1725 (0.1718)    Prec 94.531% (94.163%)
Validation starts
Test: [0/79]    Time 0.462 (0.462)      Loss 0.2844 (0.2844)    Prec 92.969%
(92.969%)
 * Prec 88.590%
best acc: 88.710000
Epoch: [84][0/391]      Time 0.532 (0.532)      Data 0.485 (0.485)      Loss
0.1505 (0.1505)    Prec 96.094% (96.094%)
Epoch: [84][100/391]    Time 0.048 (0.052)      Data 0.002 (0.007)      Loss
```

```
0.2356 (0.1626)    Prec 93.750% (94.361%)
Epoch: [84][200/391]    Time 0.045 (0.049)      Data 0.002 (0.005)      Loss
0.2004 (0.1695)    Prec 92.188% (94.127%)
Epoch: [84][300/391]    Time 0.051 (0.049)      Data 0.002 (0.004)      Loss
0.1382 (0.1674)    Prec 95.312% (94.222%)
Validation starts
Test: [0/79]    Time 0.358 (0.358)      Loss 0.3004 (0.3004)    Prec 91.406%
(91.406%)
 * Prec 88.900%
best acc: 88.900000
Epoch: [85][0/391]      Time 0.706 (0.706)      Data 0.650 (0.650)      Loss
0.1440 (0.1440)    Prec 95.312% (95.312%)
Epoch: [85][100/391]    Time 0.043 (0.051)      Data 0.002 (0.009)      Loss
0.1377 (0.1563)    Prec 95.312% (94.748%)
Epoch: [85][200/391]    Time 0.041 (0.049)      Data 0.002 (0.005)      Loss
0.1734 (0.1604)    Prec 93.750% (94.520%)
Epoch: [85][300/391]    Time 0.051 (0.048)      Data 0.002 (0.004)      Loss
0.1838 (0.1608)    Prec 92.188% (94.464%)
Validation starts
Test: [0/79]    Time 0.401 (0.401)      Loss 0.2704 (0.2704)    Prec 93.750%
(93.750%)
 * Prec 88.860%
best acc: 88.900000
Epoch: [86][0/391]      Time 0.717 (0.717)      Data 0.658 (0.658)      Loss
0.1245 (0.1245)    Prec 95.312% (95.312%)
Epoch: [86][100/391]    Time 0.038 (0.051)      Data 0.002 (0.009)      Loss
0.0952 (0.1507)    Prec 97.656% (94.933%)
Epoch: [86][200/391]    Time 0.055 (0.050)      Data 0.002 (0.005)      Loss
0.2116 (0.1546)    Prec 94.531% (94.831%)
Epoch: [86][300/391]    Time 0.049 (0.049)      Data 0.002 (0.004)      Loss
0.1290 (0.1587)    Prec 95.312% (94.643%)
Validation starts
Test: [0/79]    Time 0.752 (0.752)      Loss 0.2442 (0.2442)    Prec 94.531%
(94.531%)
 * Prec 88.800%
best acc: 88.900000
Epoch: [87][0/391]      Time 0.955 (0.955)      Data 0.894 (0.894)      Loss
0.1738 (0.1738)    Prec 93.750% (93.750%)
Epoch: [87][100/391]    Time 0.049 (0.056)      Data 0.003 (0.011)      Loss
0.1765 (0.1513)    Prec 92.969% (94.670%)
Epoch: [87][200/391]    Time 0.047 (0.051)      Data 0.002 (0.007)      Loss
0.1107 (0.1537)    Prec 95.312% (94.652%)
Epoch: [87][300/391]    Time 0.046 (0.051)      Data 0.002 (0.005)      Loss
0.1382 (0.1524)    Prec 95.312% (94.726%)
Validation starts
Test: [0/79]    Time 0.571 (0.571)      Loss 0.2910 (0.2910)    Prec 92.188%
(92.188%)
 * Prec 88.860%
```

```
best acc: 88.900000
Epoch: [88][0/391]      Time 0.601 (0.601)      Data 0.509 (0.509)      Loss
0.1017 (0.1017)    Prec 96.875% (96.875%)
Epoch: [88][100/391]    Time 0.047 (0.052)      Data 0.002 (0.007)      Loss
0.1350 (0.1526)    Prec 96.094% (94.787%)
Epoch: [88][200/391]    Time 0.048 (0.050)      Data 0.002 (0.005)      Loss
0.1382 (0.1500)    Prec 92.969% (94.932%)
Epoch: [88][300/391]    Time 0.044 (0.049)      Data 0.002 (0.004)      Loss
0.1435 (0.1493)    Prec 94.531% (94.923%)
Validation starts
Test: [0/79]    Time 0.448 (0.448)      Loss 0.2677 (0.2677)    Prec 92.188%
(92.188%)
 * Prec 88.870%
best acc: 88.900000
Epoch: [89][0/391]      Time 0.823 (0.823)      Data 0.762 (0.762)      Loss
0.0892 (0.0892)    Prec 96.875% (96.875%)
Epoch: [89][100/391]    Time 0.050 (0.053)      Data 0.002 (0.010)      Loss
0.1206 (0.1413)    Prec 96.094% (95.150%)
Epoch: [89][200/391]    Time 0.048 (0.052)      Data 0.002 (0.006)      Loss
0.1033 (0.1459)    Prec 95.312% (95.009%)
Epoch: [89][300/391]    Time 0.043 (0.051)      Data 0.002 (0.005)      Loss
0.2221 (0.1474)    Prec 88.281% (94.908%)
Validation starts
Test: [0/79]    Time 0.648 (0.648)      Loss 0.2293 (0.2293)    Prec 92.188%
(92.188%)
 * Prec 88.730%
best acc: 88.900000
Epoch: [90][0/391]      Time 0.585 (0.585)      Data 0.532 (0.532)      Loss
0.1618 (0.1618)    Prec 92.969% (92.969%)
Epoch: [90][100/391]    Time 0.047 (0.052)      Data 0.002 (0.008)      Loss
0.1110 (0.1455)    Prec 95.312% (94.856%)
Epoch: [90][200/391]    Time 0.054 (0.051)      Data 0.002 (0.005)      Loss
0.1637 (0.1429)    Prec 94.531% (95.099%)
Epoch: [90][300/391]    Time 0.037 (0.048)      Data 0.002 (0.004)      Loss
0.1527 (0.1408)    Prec 93.750% (95.159%)
Validation starts
Test: [0/79]    Time 0.355 (0.355)      Loss 0.2489 (0.2489)    Prec 92.969%
(92.969%)
 * Prec 88.880%
best acc: 88.900000
Epoch: [91][0/391]      Time 0.588 (0.588)      Data 0.527 (0.527)      Loss
0.1207 (0.1207)    Prec 97.656% (97.656%)
Epoch: [91][100/391]    Time 0.053 (0.052)      Data 0.003 (0.007)      Loss
0.1468 (0.1363)    Prec 96.094% (95.467%)
Epoch: [91][200/391]    Time 0.046 (0.049)      Data 0.002 (0.005)      Loss
0.0951 (0.1391)    Prec 95.312% (95.293%)
Epoch: [91][300/391]    Time 0.039 (0.045)      Data 0.002 (0.004)      Loss
0.1328 (0.1375)    Prec 96.875% (95.297%)
```

```
Validation starts
Test: [0/79]    Time 0.545 (0.545)      Loss 0.2551 (0.2551)    Prec 92.969%
(92.969%)
 * Prec 88.870%
best acc: 88.900000
Epoch: [92][0/391]      Time 0.523 (0.523)      Data 0.474 (0.474)      Loss
0.0858 (0.0858)    Prec 97.656% (97.656%)
Epoch: [92][100/391]    Time 0.061 (0.052)      Data 0.002 (0.007)      Loss
0.1292 (0.1348)    Prec 95.312% (95.514%)
Epoch: [92][200/391]    Time 0.041 (0.050)      Data 0.002 (0.005)      Loss
0.1656 (0.1361)    Prec 93.750% (95.460%)
Epoch: [92][300/391]    Time 0.044 (0.049)      Data 0.002 (0.004)      Loss
0.1911 (0.1356)    Prec 95.312% (95.476%)
Validation starts
Test: [0/79]    Time 0.372 (0.372)      Loss 0.2372 (0.2372)    Prec 93.750%
(93.750%)
 * Prec 89.000%
best acc: 89.000000
Epoch: [93][0/391]      Time 1.364 (1.364)      Data 1.301 (1.301)      Loss
0.1841 (0.1841)    Prec 93.750% (93.750%)
Epoch: [93][100/391]    Time 0.048 (0.059)      Data 0.002 (0.015)      Loss
0.0889 (0.1403)    Prec 96.875% (95.320%)
Epoch: [93][200/391]    Time 0.044 (0.052)      Data 0.002 (0.009)      Loss
0.1125 (0.1357)    Prec 95.312% (95.398%)
Epoch: [93][300/391]    Time 0.045 (0.051)      Data 0.002 (0.007)      Loss
0.1131 (0.1350)    Prec 96.875% (95.427%)
Validation starts
Test: [0/79]    Time 0.758 (0.758)      Loss 0.2183 (0.2183)    Prec 93.750%
(93.750%)
 * Prec 88.960%
best acc: 89.000000
Epoch: [94][0/391]      Time 0.645 (0.645)      Data 0.589 (0.589)      Loss
0.1344 (0.1344)    Prec 93.750% (93.750%)
Epoch: [94][100/391]    Time 0.049 (0.052)      Data 0.002 (0.008)      Loss
0.1698 (0.1325)    Prec 94.531% (95.382%)
Epoch: [94][200/391]    Time 0.052 (0.049)      Data 0.003 (0.005)      Loss
0.1077 (0.1352)    Prec 96.094% (95.336%)
Epoch: [94][300/391]    Time 0.046 (0.047)      Data 0.002 (0.004)      Loss
0.2228 (0.1372)    Prec 91.406% (95.255%)
Validation starts
Test: [0/79]    Time 0.449 (0.449)      Loss 0.2117 (0.2117)    Prec 94.531%
(94.531%)
 * Prec 89.200%
best acc: 89.200000
Epoch: [95][0/391]      Time 0.618 (0.618)      Data 0.556 (0.556)      Loss
0.2114 (0.2114)    Prec 91.406% (91.406%)
Epoch: [95][100/391]    Time 0.051 (0.053)      Data 0.002 (0.008)      Loss
0.0914 (0.1385)    Prec 96.875% (95.382%)
```

```
Epoch: [95][200/391]    Time 0.038 (0.049)      Data 0.002 (0.005)      Loss
0.1407 (0.1330)    Prec 94.531% (95.503%)
Epoch: [95][300/391]    Time 0.046 (0.048)      Data 0.002 (0.004)      Loss
0.1021 (0.1341)    Prec 94.531% (95.383%)
Validation starts
Test: [0/79]    Time 0.558 (0.558)      Loss 0.2683 (0.2683)    Prec 93.750%
(93.750%)
 * Prec 88.960%
best acc: 89.200000
Epoch: [96][0/391]      Time 0.529 (0.529)      Data 0.480 (0.480)      Loss
0.1394 (0.1394)    Prec 96.875% (96.875%)
Epoch: [96][100/391]    Time 0.041 (0.050)      Data 0.002 (0.007)      Loss
0.1416 (0.1274)    Prec 95.312% (95.707%)
Epoch: [96][200/391]    Time 0.049 (0.048)      Data 0.002 (0.005)      Loss
0.1888 (0.1294)    Prec 93.750% (95.588%)
Epoch: [96][300/391]    Time 0.046 (0.048)      Data 0.002 (0.004)      Loss
0.2807 (0.1316)    Prec 87.500% (95.497%)
Validation starts
Test: [0/79]    Time 0.376 (0.376)      Loss 0.2515 (0.2515)    Prec 90.625%
(90.625%)
 * Prec 89.170%
best acc: 89.200000
Epoch: [97][0/391]      Time 0.696 (0.696)      Data 0.636 (0.636)      Loss
0.0994 (0.0994)    Prec 96.094% (96.094%)
Epoch: [97][100/391]    Time 0.045 (0.053)      Data 0.002 (0.008)      Loss
0.1235 (0.1278)    Prec 94.531% (95.738%)
Epoch: [97][200/391]    Time 0.041 (0.050)      Data 0.002 (0.005)      Loss
0.1084 (0.1287)    Prec 96.875% (95.643%)
Epoch: [97][300/391]    Time 0.040 (0.049)      Data 0.002 (0.004)      Loss
0.1064 (0.1315)    Prec 96.875% (95.572%)
Validation starts
Test: [0/79]    Time 0.473 (0.473)      Loss 0.2536 (0.2536)    Prec 93.750%
(93.750%)
 * Prec 89.210%
best acc: 89.210000
Epoch: [98][0/391]      Time 0.573 (0.573)      Data 0.517 (0.517)      Loss
0.1087 (0.1087)    Prec 98.438% (98.438%)
Epoch: [98][100/391]    Time 0.055 (0.053)      Data 0.003 (0.007)      Loss
0.2222 (0.1360)    Prec 91.406% (95.591%)
Epoch: [98][200/391]    Time 0.051 (0.052)      Data 0.002 (0.005)      Loss
0.0949 (0.1332)    Prec 96.875% (95.546%)
Epoch: [98][300/391]    Time 0.040 (0.050)      Data 0.002 (0.004)      Loss
0.0684 (0.1318)    Prec 99.219% (95.564%)
Validation starts
Test: [0/79]    Time 0.622 (0.622)      Loss 0.2355 (0.2355)    Prec 94.531%
(94.531%)
 * Prec 89.070%
best acc: 89.210000
```

```
Epoch: [99][0/391]      Time 0.521 (0.521)      Data 0.464 (0.464)      Loss
0.1042 (0.1042)    Prec 96.875% (96.875%)
Epoch: [99][100/391]    Time 0.044 (0.052)      Data 0.002 (0.007)      Loss
0.1587 (0.1318)    Prec 94.531% (95.521%)
Epoch: [99][200/391]    Time 0.040 (0.049)      Data 0.002 (0.004)      Loss
0.0695 (0.1324)    Prec 98.438% (95.480%)
Epoch: [99][300/391]    Time 0.047 (0.048)      Data 0.002 (0.004)      Loss
0.1050 (0.1344)    Prec 96.094% (95.414%)
Validation starts
Test: [0/79]    Time 1.324 (1.324)      Loss 0.2383 (0.2383)    Prec 92.969%
(92.969%)
 * Prec 89.180%
best acc: 89.210000
```

[3]:
```python
PATH = "result/Resnet_20_quant_project/model_best.pth.tar"
checkpoint = torch.load(PATH)
model.load_state_dict(checkpoint['state_dict'])
device = torch.device("cuda")

model.cuda()
model.eval()

test_loss = 0
correct = 0

with torch.no_grad():
    for data, target in testloader:
        data, target = data.to(device), target.to(device) # loading to GPU
        output = model(data)
        pred = output.argmax(dim=1, keepdim=True)
        correct += pred.eq(target.view_as(pred)).sum().item()

test_loss /= len(testloader.dataset)

print('\nTest set: Accuracy: {}/{} ({:.0f}%)\n'.format(
        correct, len(testloader.dataset),
        100. * correct / len(testloader.dataset)))
```

```
Test set: Accuracy: 8922/10000 (89%)
```

[4]:
```python
class SaveOutput:
    def __init__(self):
        self.outputs = []
    def __call__(self, module, module_in):
        self.outputs.append(module_in)
```

```
    def clear(self):
        self.outputs = []

######### Save inputs from selected layer ##########
save_output = SaveOutput()
i = 0

for layer in model.modules():
    i = i+1
    if isinstance(layer, QuantConv2d):
        print(i,"-th layer prehooked")
        layer.register_forward_pre_hook(save_output)
####################################################

dataiter = iter(testloader)
images, labels = dataiter.next()
images = images.to(device)
out = model(images)
```

```
7 -th layer prehooked
9 -th layer prehooked
13 -th layer prehooked
15 -th layer prehooked
21 -th layer prehooked
25 -th layer prehooked
27 -th layer prehooked
34 -th layer prehooked
36 -th layer prehooked
42 -th layer prehooked
46 -th layer prehooked
48 -th layer prehooked
54 -th layer prehooked
56 -th layer prehooked
63 -th layer prehooked
65 -th layer prehooked
71 -th layer prehooked
75 -th layer prehooked
77 -th layer prehooked
83 -th layer prehooked
85 -th layer prehooked
```

[5]:
```
## Layer 9
## Layer 13
## save_outputs.output[1][0]
## save_outputs.output[2][0]
model.layer1[0].conv2.weight_quant.wgt_alpha
```

[5]:
```
Parameter containing:
tensor(2.6745, device='cuda:0', requires_grad=True)
```

[6]:
```
w_bit = 4
weight_q = model.layer1[0].conv2.weight_q
w_alpha = model.layer1[0].conv2.weight_quant.wgt_alpha
w_delta = w_alpha/(2**(w_bit-1)-1)
weight_int = weight_q/w_delta

x_bit = 4
x = save_output.outputs[1][0]
x_alpha = model.layer1[0].conv2.act_alpha
x_delta = x_alpha/(2**x_bit-1)
act_quant = act_quantization(x_bit)
x_q = act_quant(x,x_alpha)
x_int = x_q/x_delta

conv_int = nn.Conv2d(8,8,kernel_size=3, padding=1, bias=False)
conv_int.weight = torch.nn.parameter.Parameter(weight_int)
output_int = conv_int(x_int)
psum_recovered = output_int*w_delta*x_delta
relu = nn.ReLU(inplace=True)
psum_after_relu = relu(psum_recovered)

difference = (save_output.outputs[2][0] - psum_after_relu).mean()

print("The difference between psum original and psum recovered = {}".
 ↪format(difference))
```

```
The difference between psum original and psum recovered = -5.591659828496631e-07
```

[8]:
```
act_int = x_int[0,:,:,:]  # pick only one input out of batch
# a_int.size() = [64, 32, 32]

# conv_int.weight.size() = torch.Size([64, 64, 3, 3])  <- output_ch, input_ch,␣
 ↪ki, kj
w_int = torch.reshape(weight_int, (weight_int.size(0), weight_int.size(1), -1))␣
 ↪ # merge ki, kj index to kij

padding = 1
stride = 1
array_size = 8
x_size = x_int.size()
nig = range(x_size[2])
njg = range(x_size[3])

kijg = range(w_int.size(2))
```

```
ki_dim = int(math.sqrt(w_int.size(2)))  ## Kernel's 1 dim size

icg = range(int(w_int.size(1)))  ## input channel
ocg = range(int(w_int.size(0)))  ## output channel

a_pad = torch.zeros((x_size[1],x_size[2]+2*padding,x_size[3]+2*padding))

a_pad[ :, padding:padding+len(nig), padding:padding+len(njg)] = act_int.cuda()
a_pad = torch.reshape(a_pad, (a_pad.size(0), -1))

ic_tile = range(int(int(act_int.size(0))/array_size))
oc_tile = range(int(int(w_int.size(0))/array_size))

a_tile = torch.
 ↪zeros(len(ic_tile),array_size,len(nig)+padding*2,len(njg)+padding*2).cuda()
a_tile = torch.reshape(a_tile,(a_tile.size(0),a_tile.size(1),-1))

for ict in ic_tile:
    a_tile[ict,:,:] = a_pad[(ict*array_size):((ict+1)*array_size),:]

w_tile = torch.zeros(len(ic_tile),len(oc_tile),array_size,array_size,len(kijg)).
 ↪cuda()

for oct in oc_tile:
    for ict in ic_tile:
        w_tile[ict,oct,:,:,:] = w_int[(oct*array_size):
 ↪((oct+1)*array_size),(ict*array_size):((ict+1)*array_size),:]

p_nijg = range(a_pad.size(1)) ## paded activation's nij group

psum = torch.zeros(len(ic_tile),len(oc_tile),array_size,len(p_nijg),len(kijg)).
 ↪cuda()

for kij in kijg:
    for ict in ic_tile:
        for oct in oc_tile:
            for nij in p_nijg:     # time domain, sequentially given input
                m = nn.Linear(array_size, array_size, bias=False)
                m.weight = torch.nn.Parameter(w_tile[ict,oct,:,:,kij])
                psum[ict,oct,:, nij, kij] = m(a_tile[ict,:,nij]).cuda()
```

[9]:
```
import math

a_pad_ni_dim = int(math.sqrt(a_pad.size(1))) # 32 + 2*pad = 34

o_ni_dim = int((a_pad_ni_dim - (ki_dim- 1) - 1)/stride + 1) #34 - 2 - 1 + 1 = 32
o_nijg = range(o_ni_dim**2)
```

```
out = torch.zeros(len(ocg), len(o_nijg)).cuda()


### SFP accumulation ###
for o_nij in o_nijg:
    for kij in kijg:
        for ict in ic_tile:
            for oct in oc_tile:
                out[oct*array_size:(oct+1)*array_size,o_nij] =␣
↪out[oct*array_size:(oct+1)*array_size,o_nij] + \
                psum[ict,oct,:, int(o_nij/o_ni_dim)*a_pad_ni_dim +␣
↪o_nij%o_ni_dim + int(kij/ki_dim)*a_pad_ni_dim + kij%ki_dim, kij]
                ## 2nd index = (int(o_nij/30)*32 + o_nij%30) + (int(kij/3)*32 +␣
↪kij%3)
```

[143]:
```
## dump all the files

## Helper functions to dump files
def dec_to_bin(arr,bit):
    bin_arr = []
    for a in arr:
        a = int(a)
        if a < 0:
            a+=bit
        b = '{0:04b}'.format(int(a))
        bin_arr.append(b)
    return bin_arr

def dec_to_bin_psum(arr,bit):
    bin_arr = []
    for a in arr:
        a = int(a)
        sign = 0
        if a < 0:
            sign = 1
            a+=bit
        b = '{0:16b}'.format(int(a))
        if sign == 1:
            b = b.replace(" ", "1")
        else:
            b = b.replace(" ","0")
        bin_arr.append(b)
    return bin_arr


def convert_to_list(arr):
```

```
    return arr.tolist()
```

[144]:
```
## Activation dump

fp_act = open('Renet_activation_project.txt','w')
fp_act.write("#####\n")
fp_act.write("#####\n")
fp_act.write("#####\n")
fp_act_dec = open('Resnet_activation_dec.txt','w')
for i in range(a_pad.size()[1]):
    act_line = a_pad[:,i]
    act_arr = [int(j+0.001) for j in convert_to_list(act_line)]
    for a in act_arr[::-1]:
        #print(int(a))
        fp_act_dec.write(str(int(a)))
        fp_act_dec.write(" ")
    fp_act_dec.write("\n")
    bin_act = dec_to_bin(act_arr,16)
    for b in bin_act[::-1]:
        #print(b)
        fp_act.write(b)
    fp_act.write('\n')


fp_act.close()
fp_act_dec.close()
```

[145]:
```
## Weight dump
fp_wgt = open('Resnet_weight_project.txt','w')
fp_wgt.write("#####\n")
fp_wgt.write("#####\n")
fp_wgt.write("#####\n")
fp_wgt_dec = open('Resnet_weight_dec.txt','w')

for kij in range(9):
    for w in range(8):
        w_line = w_int[w,:,kij]
        w_arr = []
        for i in convert_to_list(w_line):
            if i < 0:
                w_arr.append(int(i-0.001))
            else:
                w_arr.append(int(i+0.001))

        for ww in w_arr[::-1]:
            #print(int(ww))
            fp_wgt_dec.write(str(int(ww)))
```

```
            fp_wgt_dec.write(" ")
        fp_wgt_dec.write("\n")
        bin_wgt = dec_to_bin(w_arr,16)
        for b in bin_wgt[::-1]:
            #print("Count = {} , b = {}".format(count,b))
            fp_wgt.write(b)
        fp_wgt.write('\n')

fp_wgt.close()
fp_wgt_dec.close()
```

[149]:
```
## psum dump and output.txt dump
fp_psum = open('Resnet_psum_project.txt','w')
fp_psum.write("#####\n")
fp_psum.write("#####\n")
fp_psum.write("#####\n")

fp_psum_dec = open('Resnet_psum_dec.txt','w')
fp_psum_relu = open('Resnet_output_project.txt','w')
fp_psum_relu.write("#\n")
fp_psum_relu.write("#\n")
fp_psum_relu.write("#\n")
for kij in range(o_ni_dim*o_ni_dim):
    psum_line = out[:,kij]
    psum_arr = []
    out_arr = []
    for i in convert_to_list(psum_line):
        if i < 0:
            psum_arr.append(int(i-0.001))
            out_arr.append(0)
        else:
            psum_arr.append(int(i+0.001))
            out_arr.append(int(i+0.001))
    for p in psum_arr:
        #print(int(a))
        fp_psum_dec.write(str(int(p)))
        fp_psum_dec.write(" ")
    fp_psum_dec.write("\n")
    bin_act = dec_to_bin_psum(psum_arr,65536)
    out_bin_act = dec_to_bin_psum(out_arr,65536)
    for b in bin_act:
        #print(b)
        fp_psum.write(b)
    fp_psum.write('\n')

    for b in out_bin_act:
        fp_psum_relu.write(b)
```

```
    fp_psum_relu.write("\n")
fp_psum.close()
fp_psum_dec.close()
fp_psum_relu.close()
```

[10]:
```
## For input activations tiling. Handled in testbench. Written here just for␣
 ↪reference logic
hardware_ni_dim = 6
hor_step = hardware_ni_dim-kernel_dim+1 ## 4
ver_step = (hardware_ni_dim-kernel_dim+1)*a_pad_ni_dim ## 136
stop_point = (a_pad_ni_dim-hardware_ni_dim)*a_pad_ni_dim+1 ## 953
group_count = 0
act_arr = []
## Vertical movement loop
for v in range(0,stop_point,ver_step):
    ## Now move horizontally
    for h in range(v,v+34,hor_step):
        if h+hardware_ni_dim>v+a_pad_ni_dim:
            break
        group = []
        group_count+=1
        for hh in range(h,h+(hardware_ni_dim)*a_pad_ni_dim,a_pad_ni_dim):

            for hhh in range(hardware_ni_dim):
                group.append(hh+hhh)
        act_group.append(group)
```

```
64
```

[15]:
```
## For output tiling. Handled in testbench. Written here just for reference␣
 ↪logic
our_out_hw = 4
out_ni_dim = int(math.sqrt(out.size(1)))
out_stop_point = (out_ni_dim-our_out_hw)*out_ni_dim+1
out_ver_step = our_out_hw*out_ni_dim
group_count = 0
out_group = []
for v in range(0,out_stop_point,out_ver_step):
    ## Now move horizontally
    for h in range(v,v+32,our_out_hw):
        group = []
        group_count+=1
        for hh in range(h,h+our_out_hw*out_ni_dim,out_ni_dim):
            for hhh in range(our_out_hw):
                group.append(hh+hhh)
        out_group.append(group)
```

[ ]: