

simpleconvnet2.py

```
# coding: utf-8
import sys, os
sys.path.append(os.pardir) # 부모 디렉터리의 파일을 가져올 수 있도록 설정
import pickle
import numpy as np
from collections import OrderedDict
from common.layers import *
from common.gradient import numerical_gradient

class SimpleConvNet:
    def __init__(self, input_dim=(1, 28, 28),
                  conv_param={'filter_num':30, 'filter_size':5, 'pad':0,
                              'stride':1},
                  hidden_size=100, output_size=10, weight_init_std=0.01): #
        변수명 수정
        filter_num = conv_param['filter_num']
        filter_size = conv_param['filter_size']
        filter_pad = conv_param['pad']
        filter_stride = conv_param['stride']
        input_size = input_dim[1]
        conv_output_size = (input_size - filter_size + 2*filter_pad) //
        filter_stride + 1
        pool_output_size = int(filter_num * (conv_output_size/2) *
                                (conv_output_size/2))

        # 가중치 초기화
        self.params = {}
        self.params['W1'] = weight_init_std * \
            np.random.randn(filter_num, input_dim[0],
                              filter_size, filter_size)
        self.params['b1'] = np.zeros(filter_num)
        self.params['W2'] = weight_init_std * \
            np.random.randn(pool_output_size, hidden_size)
        self.params['b2'] = np.zeros(hidden_size)
        self.params['W3'] = weight_init_std * \
            np.random.randn(hidden_size, output_size)
        self.params['b3'] = np.zeros(output_size)

        # 계층 생성
        self.layers = OrderedDict()
        self.layers['Conv1'] = Convolution(self.params['W1'],
        self.params['b1'],
                                         conv_param['stride'],
        conv_param['pad'])
        self.layers['BatchNorm1'] = BatchNormalization(filter_num)
        self.layers['Relu1'] = Relu()
```

```

self.layers['Pool1'] = Pooling(pool_h=2, pool_w=2, stride=2)
self.layers['Affine1'] = Affine(self.params['W2'], self.params['b2'])
self.layers['BatchNorm2'] = BatchNormalization(hidden_size)
self.layers['Relu2'] = Relu()
self.layers['Dropout1'] = Dropout(dropout_ratio=0.5)
self.layers['Affine2'] = Affine(self.params['W3'], self.params['b3'])

self.last_layer = SoftmaxWithLoss()

# Batch Normalization 파라미터 초기화
self.gamma = {}
self.beta = {}
self.gamma['gamma1'] = np.ones(filter_num)
self.beta['beta1'] = np.zeros(filter_num)
self.gamma['gamma2'] = np.ones(hidden_size)
self.beta['beta2'] = np.zeros(hidden_size)

# 각 Batch Normalization 레이어에 파라미터 설정
self.layers['BatchNorm1'].gamma = self.gamma['gamma1']
self.layers['BatchNorm1'].beta = self.beta['beta1']
self.layers['BatchNorm2'].gamma = self.gamma['gamma2']
self.layers['BatchNorm2'].beta = self.beta['beta2']

def predict(self, x):
    for layer in self.layers.values():
        x = layer.forward(x)

    return x

def loss(self, x, t):
    y = self.predict(x)
    return self.last_layer.forward(y, t)

def accuracy(self, x, t, batch_size=100):
    if t.ndim != 1 : t = np.argmax(t, axis=1)

    acc = 0.0

    for i in range(int(x.shape[0] / batch_size)):
        tx = x[i*batch_size:(i+1)*batch_size]
        tt = t[i*batch_size:(i+1)*batch_size]
        y = self.predict(tx)
        y = np.argmax(y, axis=1)
        acc += np.sum(y == tt)

    return acc / x.shape[0]

def numerical_gradient(self, x, t):

```

```

        loss_w = lambda w: self.loss(x, t)

        grads = {}
        for idx in (1, 2, 3):
            grads['W' + str(idx)] = numerical_gradient(loss_w, self.params['W'
+ str(idx)])
            grads['b' + str(idx)] = numerical_gradient(loss_w, self.params['b'
+ str(idx)])

        return grads

def gradient(self, x, t):
    # forward
    self.loss(x, t)

    # backward
    dout = 1
    dout = self.last_layer.backward(dout)

    layers = list(self.layers.values())
    layers.reverse()
    for layer in layers:
        dout = layer.backward(dout)

    # 결과 저장
    grads = {}
    grads['W1'], grads['b1'] = self.layers['Conv1'].dW,
self.layers['Conv1'].db
    grads['W2'], grads['b2'] = self.layers['Affine1'].dW,
self.layers['Affine1'].db
    grads['W3'], grads['b3'] = self.layers['Affine2'].dW,
self.layers['Affine2'].db

    # Batch Normalization 에 대한 gradient 추가
    grads['gamma1'] = self.layers['BatchNorm1'].dgamma
    grads['beta1'] = self.layers['BatchNorm1'].dbeta
    grads['gamma2'] = self.layers['BatchNorm2'].dgamma
    grads['beta2'] = self.layers['BatchNorm2'].dbeta

    return grads

def save_params(self, file_name="params.pkl"):
    params = {}
    for key, val in self.params.items():
        params[key] = val
    with open(file_name, 'wb') as f:
        pickle.dump(params, f)

```

```

def load_params(self, file_name="params.pkl"):
    with open(file_name, 'rb') as f:
        params = pickle.load(f)
    for key, val in params.items():
        self.params[key] = val

    for i, key in enumerate(['Conv1', 'Affine1', 'Affine2']):
        self.layers[key].W = self.params['W' + str(i+1)]
        self.layers[key].b = self.params['b' + str(i+1)]

    # Batch Normalization 파라미터 설정
    self.layers['BatchNorm1'].gamma = self.gamma['gamma1']
    self.layers['BatchNorm1'].beta = self.beta['beta1']
    self.layers['BatchNorm2'].gamma = self.gamma['gamma2']
    self.layers['BatchNorm2'].beta = self.beta['beta2']
    def __init__(self, input_dim=(1, 28, 28),
                  conv_param={'filter_num':30, 'filter_size':5, 'pad':0,
                              'stride':1},
                  hidden_size=100, output_size=10, weight_init_std=0.01): #
        변수명 수정
        filter_num = conv_param['filter_num']
        filter_size = conv_param['filter_size']
        filter_pad = conv_param['pad']
        filter_stride = conv_param['stride']
        input_size = input_dim[1]
        conv_output_size = (input_size - filter_size + 2*filter_pad) /
        filter_stride + 1
        pool_output_size = int(filter_num * (conv_output_size/2) *
        (conv_output_size/2))

        # 가중치 초기화
        self.params = {}
        self.params['W1'] = weight_init_std * \
            np.random.randn(filter_num, input_dim[0],
            filter_size, filter_size)
        self.params['b1'] = np.zeros(filter_num)
        self.params['W2'] = weight_init_std * \
            np.random.randn(pool_output_size, hidden_size)
        self.params['b2'] = np.zeros(hidden_size)
        self.params['W3'] = weight_init_std * \
            np.random.randn(hidden_size, output_size)
        self.params['b3'] = np.zeros(output_size)

        # 계층 생성
        self.layers = OrderedDict()
        self.layers['Conv1'] = Convolution(self.params['W1'],
        self.params['b1'],

```

```

conv_param['stride'],
conv_param['pad'])
    self.layers['Relu1'] = Relu()
    self.layers['Pool1'] = Pooling(pool_h=2, pool_w=2, stride=2)
    self.layers['Affine1'] = Affine(self.params['W2'], self.params['b2'])
    self.layers['Relu2'] = Relu()
    self.layers['Dropout1'] = Dropout(dropout_ratio=0.5)
    self.layers['Affine2'] = Affine(self.params['W3'], self.params['b3'])

    self.last_layer = SoftmaxWithLoss()

def predict(self, x):
    for layer in self.layers.values():
        x = layer.forward(x)

    return x

def loss(self, x, t):
    y = self.predict(x)
    return self.last_layer.forward(y, t)

def accuracy(self, x, t, batch_size=100):
    if t.ndim != 1 : t = np.argmax(t, axis=1)

    acc = 0.0

    for i in range(int(x.shape[0] / batch_size)):
        tx = x[i*batch_size:(i+1)*batch_size]
        tt = t[i*batch_size:(i+1)*batch_size]
        y = self.predict(tx)
        y = np.argmax(y, axis=1)
        acc += np.sum(y == tt)

    return acc / x.shape[0]

def numerical_gradient(self, x, t):
    loss_w = lambda w: self.loss(x, t)

    grads = {}
    for idx in (1, 2, 3):
        grads['W' + str(idx)] = numerical_gradient(loss_w, self.params['W'
+ str(idx)])
        grads['b' + str(idx)] = numerical_gradient(loss_w, self.params['b'
+ str(idx)])

    return grads

def gradient(self, x, t):

```

```

# forward
self.loss(x, t)

# backward
dout = 1
dout = self.last_layer.backward(dout)

layers = list(self.layers.values())
layers.reverse()
for layer in layers:
    dout = layer.backward(dout)

# 결과 저장
grads = {}
grads['W1'], grads['b1'] = self.layers['Conv1'].dW,
self.layers['Conv1'].db
grads['W2'], grads['b2'] = self.layers['Affine1'].dW,
self.layers['Affine1'].db
grads['W3'], grads['b3'] = self.layers['Affine2'].dW,
self.layers['Affine2'].db

return grads

def save_params(self, file_name="params.pkl"):
    params = {}
    for key, val in self.params.items():
        params[key] = val
    with open(file_name, 'wb') as f:
        pickle.dump(params, f)

def load_params(self, file_name="params.pkl"):
    with open(file_name, 'rb') as f:
        params = pickle.load(f)
    for key, val in params.items():
        self.params[key] = val

for i, key in enumerate(['Conv1', 'Affine1', 'Affine2']):
    self.layers[key].W = self.params['W' + str(i+1)]
    self.layers[key].b = self.params['b' + str(i+1)]

```

train_cnn.py

```

# coding: utf-8
import sys, os
sys.path.append(os.pardir) # 부모 디렉터리의 파일을 가져올 수 있도록 설정

```

```

import numpy as np
import matplotlib.pyplot as plt
from dataset.fasion_mnist import load_mnist
from simpleconvnet2 import SimpleConvNet
from common.trainer import Trainer

# 데이터 읽기
(x_train, t_train), (x_test, t_test) = load_mnist(flatten=False)

# 시간이 오래 걸릴 경우 데이터를 줄인다.
x_train, t_train = x_train[:5000], t_train[:5000]
x_test, t_test = x_test[:4000], t_test[:4000]

max_epochs = 20

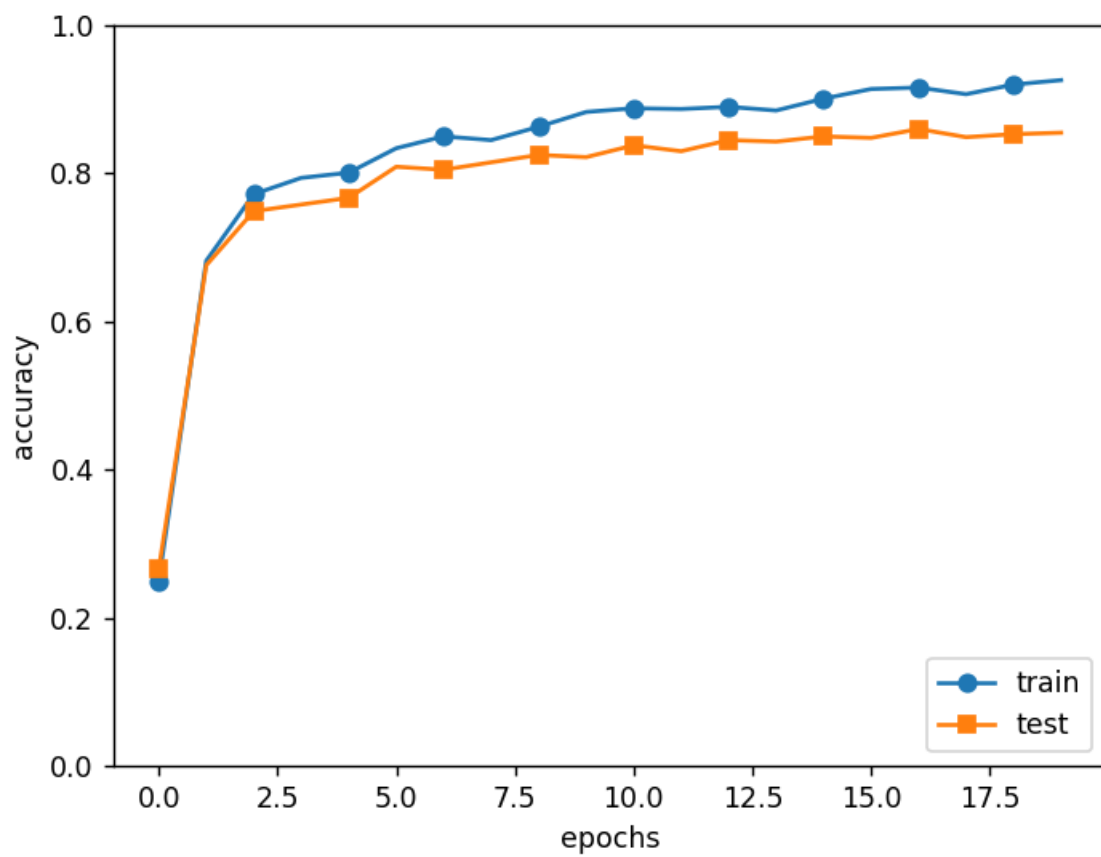
network = SimpleConvNet(input_dim=(1,28,28),
                        conv_param = {'filter_num': 64, 'filter_size': 5,
                                      'pad': 0, 'stride': 1},
                        hidden_size=512, output_size=10, weight_init_std=0.01)

trainer = Trainer(network, x_train, t_train, x_test, t_test,
                  epochs=max_epochs, mini_batch_size=100,
                  optimizer='Adam', optimizer_param={'lr': 0.001},
                  evaluate_sample_num_per_epoch=1000)
trainer.train()

# 매개변수 보존
network.save_params("params.pkl")
print("Saved Network Parameters!")

# 그래프 그리기
markers = {'train': 'o', 'test': 's'}
x = np.arange(max_epochs)
plt.plot(x, trainer.train_acc_list, marker='o', label='train', markevery=2)
plt.plot(x, trainer.test_acc_list, marker='s', label='test', markevery=2)
plt.xlabel("epochs")
plt.ylabel("accuracy")
plt.ylim(0, 1.0)
plt.legend(loc='lower right')
plt.show()

```



터미널 로그

```
train loss:1.2718930640625283
train loss:1.1875702442192253
train loss:1.1365009990727968
train loss:1.2289973465913187
train loss:1.0719864614365988
train loss:1.0806772157581086
train loss:1.024906970744914
train loss:1.2247836803854384
train loss:0.8887986864787567
train loss:1.1248551752032356
train loss:1.035486905465466
train loss:1.015380462504475
train loss:0.9982715208233075
train loss:1.1175710021038459
train loss:1.1299610888266516
train loss:1.0947699127023147
```



```
train loss:1.0220368268997537
train loss:1.0206888738534599
train loss:0.8427836130859054
train loss:0.8530494269908292
train loss:0.8551473802130674
train loss:1.1039885990955798
train loss:0.8687080378585241
train loss:0.9074938222716437
train loss:0.8813102316864005
train loss:0.8174693019147025
train loss:0.7880255559107391
train loss:0.7223449246468405
train loss:0.7949957770843256
train loss:0.7377144034698075
train loss:0.8785433900162676
train loss:0.7837562057573731
train loss:0.9988452090343991
train loss:0.961675064393538
train loss:0.7856735094733641
train loss:0.8697454024157283
train loss:0.7217173740432511
=== epoch:2, train acc:0.682, test acc:0.676 ===
train loss:0.7977343751081432
train loss:0.8257671113255003
train loss:0.8588416245974158
train loss:0.710716344378059
train loss:0.8392741741165914
train loss:0.8693922268459638
train loss:0.76908618654187
train loss:0.8955986115972359
train loss:0.8716404963042459
train loss:0.8360445926368237
train loss:0.7730326492216427
train loss:0.6112837749289273
train loss:0.7631446319756217
train loss:0.605987684366484
train loss:0.8225403341178553
train loss:0.7146144344742456
train loss:0.7837173480941396
train loss:0.6695376210246577
```

```
train loss:0.5746452253550574
train loss:0.7732534667399432
train loss:0.49486866848387934
train loss:0.7354670528748777
train loss:0.7504320337957455
train loss:0.6853465787813242
train loss:0.6922888182550989
train loss:0.7331375498935729
train loss:0.7220614477525978
train loss:0.528422730549196
train loss:0.6642714221014034
train loss:0.5216091598843015
train loss:0.6217306204248904
train loss:0.6226838407161975
train loss:0.62344677793575
train loss:0.6830774286740156
train loss:0.683132667357374
train loss:0.699149552003837
train loss:0.5311072803355809
train loss:0.6950144327473684
train loss:0.6519652295885824
train loss:0.7102847155714609
train loss:0.6758840073828121
train loss:0.7248192135768484
train loss:0.6048694537373879
train loss:0.46930952084568767
train loss:0.5260071910469172
train loss:0.5448293055704769
train loss:0.6490711376260515
train loss:0.6860649194354173
train loss:0.6418003209070924
train loss:0.7367695354252478
=== epoch:3, train acc:0.772, test acc:0.749 ===
train loss:0.9312575288596492
train loss:0.6927216596231304
train loss:0.564997519364376
train loss:0.7546081769208712
train loss:0.6631632268186884
train loss:0.7716589760319313
train loss:0.6303625577370989
```

train loss:0.6886876662240258
train loss:0.6556408413696367
train loss:0.6187240659396495
train loss:0.6855085448282137
train loss:0.658804869399917
train loss:0.5639453577822571
train loss:0.7237939060291518
train loss:0.615286736025469
train loss:0.5100222350528835
train loss:0.6023220808778398
train loss:0.6419135370643251
train loss:0.8138827243519934
train loss:0.5154739305418642
train loss:0.6044906447849372
train loss:0.6527217604862146
train loss:0.6414734808461905
train loss:0.560818690712586
train loss:0.6133825173375008
train loss:0.6607198483267935
train loss:0.7270412945396171
train loss:0.6216620012201614
train loss:0.5240801808350161
train loss:0.432814556871344
train loss:0.49146577510479866
train loss:0.4751863945905167
train loss:0.622554300349231
train loss:0.652952647449354
train loss:0.5235207109396689
train loss:0.44997294715241837
train loss:0.4809863436131186
train loss:0.5900976225050265
train loss:0.6165516574518768
train loss:0.6489814218183387
train loss:0.7447232137936736
train loss:0.5887746546440682
train loss:0.7637490748030104
train loss:0.4974892222408211
train loss:0.3770476816267464
train loss:0.5738933771253607
train loss:0.446599769142545

train loss:0.5928178188898413
train loss:0.6312250126256678
train loss:0.6271928593134458
=== epoch:4, train acc:0.794, test acc:0.758 ===
train loss:0.539290764854633
train loss:0.6226647049703129
train loss:0.6586817906719823
train loss:0.5296344379194167
train loss:0.5317160778254317
train loss:0.5013169807337998
train loss:0.4106283412974684
train loss:0.4446314932222548
train loss:0.5252122282010078
train loss:0.5316643849229298
train loss:0.6052446283679633
train loss:0.6094615774862226
train loss:0.4913332009407555
train loss:0.41307106477312105
train loss:0.33214536241385967
train loss:0.6831119536653225
train loss:0.5990733022969873
train loss:0.6803023074466027
train loss:0.744982182249527
train loss:0.5383889627110696
train loss:0.507237333987616
train loss:0.5049855596075301
train loss:0.4896749007573298
train loss:0.6159564000566307
train loss:0.5422532602185953
train loss:0.5202153576325607
train loss:0.46980442898620384
train loss:0.5764397996440931
train loss:0.5993838802678693
train loss:0.6986071055613955
train loss:0.6506906066971334
train loss:0.43592171502785476
train loss:0.7567744214900699
train loss:0.5703491664655425
train loss:0.5322034283329218
train loss:0.6536588691733034

train loss:0.44511929139879763
train loss:0.5328178201114304
train loss:0.5178995279436767
train loss:0.457853138675346
train loss:0.5571308100232408
train loss:0.4969032951098392
train loss:0.4178437591841149
train loss:0.7957711571453264
train loss:0.5780419572682789
train loss:0.5738034724570804
train loss:0.5762986546469373
train loss:0.4952646438998066
train loss:0.5243977787020528
train loss:0.5856212243196235
=== epoch:5, train acc:0.801, test acc:0.767 ===
train loss:0.5653383123712719
train loss:0.7891313876662661
train loss:0.5228941201324033
train loss:0.3978071279122595
train loss:0.5533393311250417
train loss:0.4932996742244113
train loss:0.676566793763844
train loss:0.6615172165733171
train loss:0.5528202140478888
train loss:0.47983104760434114
train loss:0.45914008378129123
train loss:0.46023345310540875
train loss:0.4894844148798522
train loss:0.42938878745241693
train loss:0.4872525040721291
train loss:0.4277539333789226
train loss:0.5027184762231688
train loss:0.4538574716502787
train loss:0.5302196040995265
train loss:0.3805169267969892
train loss:0.41899039634514673
train loss:0.48125810862228036
train loss:0.4857887365040206
train loss:0.5062045597276095
train loss:0.37276248845938015

train loss:0.7468960157558534
train loss:0.5300593003993643
train loss:0.49991907798150104
train loss:0.5643914292339572
train loss:0.5382084631419964
train loss:0.4543043020075919
train loss:0.5731127345322332
train loss:0.44329479066748834
train loss:0.4975274715671687
train loss:0.4499025375784772
train loss:0.38263404532217843
train loss:0.4948427574020181
train loss:0.36790267873696075
train loss:0.4686490237106101
train loss:0.4161563008372394
train loss:0.44593207728316536
train loss:0.39322876792094463
train loss:0.4572032321430736
train loss:0.5218817117698036
train loss:0.4829923115043176
train loss:0.5605616989528817
train loss:0.6439913745914833
train loss:0.4402296437947465
train loss:0.35981406305892355
train loss:0.4620545782261874
=== epoch:6, train acc:0.834, test acc:0.809 ===
train loss:0.48709518813152464
train loss:0.41629041463669053
train loss:0.5370795794952645
train loss:0.6449843192641588
train loss:0.5837474974737547
train loss:0.38823446243492993
train loss:0.42136809748863646
train loss:0.4013088709687581
train loss:0.5259605691050776
train loss:0.5390501857590775
train loss:0.46783565661642995
train loss:0.581732027398587
train loss:0.505479792016616
train loss:0.4252848807012535

train loss:0.45624071661062254
train loss:0.40995861046876597
train loss:0.4587961363763165
train loss:0.46958782732160675
train loss:0.40191719597189857
train loss:0.426317568208588
train loss:0.42687691615297824
train loss:0.45528360129902956
train loss:0.36368200169376635
train loss:0.33242965932475455
train loss:0.386586695465101
train loss:0.2849451489090326
train loss:0.3683003361903777
train loss:0.3684052416831002
train loss:0.4558435753085533
train loss:0.4568587990980975
train loss:0.5022612847728536
train loss:0.45123710185855975
train loss:0.6167805869348743
train loss:0.4067232673080921
train loss:0.7011134751018693
train loss:0.6013711407279113
train loss:0.543569404307072
train loss:0.4632232727788312
train loss:0.474211767824816
train loss:0.4717370366244022
train loss:0.4005629515105554
train loss:0.5518001463348092
train loss:0.45787800744273743
train loss:0.47100205712666754
train loss:0.45081654633065865
train loss:0.445895066720215
train loss:0.4722391340228599
train loss:0.36374468353283695
train loss:0.3842563266032997
train loss:0.4233355930041393
=== epoch:7, train acc:0.85, test acc:0.805 ===
train loss:0.46946411886826506
train loss:0.3531133496533749
train loss:0.5084219139502761

train loss:0.3815690452219764
train loss:0.42080931473856786
train loss:0.3240143324918662
train loss:0.35670553297454793
train loss:0.6164897929281443
train loss:0.5083151823395271
train loss:0.36721794988993095
train loss:0.5782371756934067
train loss:0.4478026068723559
train loss:0.35524354480807396
train loss:0.44426551894721106
train loss:0.41496949819773854
train loss:0.470935843244146
train loss:0.31233767547743996
train loss:0.3803692525573987
train loss:0.3481033508495659
train loss:0.38098168623998474
train loss:0.3698670347046879
train loss:0.4104678373234046
train loss:0.39169879210578473
train loss:0.37846400469360986
train loss:0.3647605906756979
train loss:0.5464630787378097
train loss:0.3711338627047674
train loss:0.3544055951826067
train loss:0.3531793201470299
train loss:0.3948443659525845
train loss:0.36760560297231915
train loss:0.40172502731906046
train loss:0.4414749718414094
train loss:0.33807357202309324
train loss:0.4790959156786158
train loss:0.40220709098192214
train loss:0.4779138306527367
train loss:0.5318361805603622
train loss:0.429752679830314
train loss:0.5401899396569749
train loss:0.41195476626255484
train loss:0.490210458549491
train loss:0.31373464859149014

train loss:0.3138349299890681
train loss:0.4874090380885145
train loss:0.32849892161206
train loss:0.2818426856007755
train loss:0.36242020978238115
train loss:0.41101869446435224
train loss:0.6534873447029139
=== epoch:8, train acc:0.845, test acc:0.815 ===
train loss:0.44906759192630624
train loss:0.36001630272008306
train loss:0.3781848610732053
train loss:0.3713744922802957
train loss:0.40191779356597407
train loss:0.35655430250606623
train loss:0.33986239864603474
train loss:0.314938241343043
train loss:0.5064427253362922
train loss:0.5345208827981842
train loss:0.4496935499335236
train loss:0.3925855789837305
train loss:0.3287549976034025
train loss:0.3217705102756828
train loss:0.35480664071318313
train loss:0.3604213488902268
train loss:0.4077761530624514
train loss:0.5307960652456467
train loss:0.4370732022379882
train loss:0.43886199585017716
train loss:0.5262421169845899
train loss:0.49634500226837525
train loss:0.21772705237026335
train loss:0.35614986667835824
train loss:0.4332656966895529
train loss:0.43413212142967184
train loss:0.5512710976227345
train loss:0.47997521738989485
train loss:0.27903545758331944
train loss:0.2511814793130556
train loss:0.3406468962199822
train loss:0.4661123028046726

train loss:0.2632772228757463
train loss:0.4327461809757376
train loss:0.5031675847595026
train loss:0.21126253203872727
train loss:0.36050344597525624
train loss:0.41557667427295314
train loss:0.3527892198089556
train loss:0.44780639743774836
train loss:0.3854936859113861
train loss:0.2987089906885823
train loss:0.31814743041617466
train loss:0.31233635060892284
train loss:0.3615458384428596
train loss:0.3797136609436088
train loss:0.4201853741004849
train loss:0.41028156871681687
train loss:0.38716209678393637
train loss:0.35319893283561143
=== epoch:9, train acc:0.863, test acc:0.825 ===
train loss:0.3765161641240604
train loss:0.25155402591505516
train loss:0.3893545742934487
train loss:0.31104588406189876
train loss:0.4936564760098993
train loss:0.4237276859342694
train loss:0.3849588131996427
train loss:0.39054891895541616
train loss:0.3821419219648697
train loss:0.3017831593596677
train loss:0.3730937443696438
train loss:0.4145312916036171
train loss:0.30495701840299
train loss:0.2916253336220949
train loss:0.3629686231315747
train loss:0.35822874181090525
train loss:0.5428920793559363
train loss:0.2779644783183594
train loss:0.46744330059167827
train loss:0.48567688252229496
train loss:0.4623106465002113

```
train loss:0.5113632166425927
train loss:0.39990894540952726
train loss:0.4212803667506534
train loss:0.369315191201121
train loss:0.4230869292990871
train loss:0.4151434859901569
train loss:0.2578839259380232
train loss:0.4549595245897957
train loss:0.3844109834532251
train loss:0.30812494010854646
train loss:0.37188405606309827
train loss:0.4233283678711626
train loss:0.37570271079163736
train loss:0.28622611207057824
train loss:0.5522815670181135
train loss:0.3137285060174688
train loss:0.3307214781383784
train loss:0.3578343528673292
train loss:0.4280174376545765
train loss:0.34901791461819515
train loss:0.36424465757683655
train loss:0.3424454647413504
train loss:0.4021539955996778
train loss:0.3040894470185929
train loss:0.3343071735058328
train loss:0.5253380587516481
train loss:0.27653509209840993
train loss:0.4411530937225707
train loss:0.4129594501892898
=== epoch:10, train acc:0.883, test acc:0.822 ===
train loss:0.37903942022550424
train loss:0.29744256006369085
train loss:0.35674338113180776
train loss:0.33459926094075554
train loss:0.4077005778970089
train loss:0.3145492986093387
train loss:0.14641449838720244
train loss:0.41583138437158723
train loss:0.3573030740148447
train loss:0.31855020535692363
```

train loss:0.4594849587832605
train loss:0.2152871056260085
train loss:0.39297296000986476
train loss:0.2035982673374262
train loss:0.2246383701544964
train loss:0.324171420650749
train loss:0.37785217434406276
train loss:0.4935584719517201
train loss:0.3666137838062764
train loss:0.4302490300739245
train loss:0.40741651577377785
train loss:0.38636371616149356
train loss:0.36460295698098294
train loss:0.29047015015998945
train loss:0.342649967292066
train loss:0.2636237743322397
train loss:0.30249605908256266
train loss:0.3263692821636653
train loss:0.39807810870201416
train loss:0.32150301284642174
train loss:0.4341485519685408
train loss:0.3981293159565005
train loss:0.3121504727240283
train loss:0.37524798056338293
train loss:0.35431922087717743
train loss:0.4327467615276502
train loss:0.37584167210353997
train loss:0.3180004821112092
train loss:0.24526642988897948
train loss:0.43725552619621993
train loss:0.26347485500684514
train loss:0.251645118095112
train loss:0.28518473721329923
train loss:0.36119121152161493
train loss:0.4459440808983856
train loss:0.27632910700921354
train loss:0.42379418714080025
train loss:0.3090438507578403
train loss:0.3687133178285695
train loss:0.3391429801377259

=== epoch:11, train acc:0.888, test acc:0.838 ===

train loss:0.2859764428143606

train loss:0.49725203794239725

train loss:0.38618499785115296

train loss:0.2951497066459837

train loss:0.39487253188697424

train loss:0.3216540339784122

train loss:0.26583942519418385

train loss:0.2728795629713478

train loss:0.30050454450444897

train loss:0.32734482328674885

train loss:0.4168612042393874

train loss:0.39639208454525243

train loss:0.32725212694557676

train loss:0.23374664213184576

train loss:0.22095282859421528

train loss:0.36394845754690797

train loss:0.30736688464015793

train loss:0.32964640596273354

train loss:0.33170204851235036

train loss:0.42451854700160796

train loss:0.3353440468423941

train loss:0.3600052115880575

train loss:0.29905229439285086

train loss:0.3132938886984439

train loss:0.27441214616627113

train loss:0.193207756663449

train loss:0.46331983973776586

train loss:0.26349887574881037

train loss:0.5079256201330008

train loss:0.23551207737073782

train loss:0.31477426500084776

train loss:0.3657560999537091

train loss:0.4274393545368567

train loss:0.25025364753301266

train loss:0.35746779516514704

train loss:0.3321362630053015

train loss:0.22859604199271658

train loss:0.31478223030233243

train loss:0.38836490041367855

train loss:0.4562084727586418
train loss:0.265896419205735
train loss:0.34967633605605714
train loss:0.24587995270143836
train loss:0.45433502706587275
train loss:0.3413431178776539
train loss:0.2914542380707209
train loss:0.25915020975091246
train loss:0.4084565952750196
train loss:0.39616467441994396
train loss:0.35735909154949275
=== epoch:12, train acc:0.887, test acc:0.83 ===
train loss:0.32149009898123276
train loss:0.3795175760915467
train loss:0.35477348434834943
train loss:0.2247313956801089
train loss:0.3025466512258975
train loss:0.4000646025821039
train loss:0.18898825859087118
train loss:0.30811166124255274
train loss:0.3515410017700061
train loss:0.3929241818811288
train loss:0.31005178003094125
train loss:0.305663239089256
train loss:0.2933066266061579
train loss:0.29715684380695256
train loss:0.30508938376195466
train loss:0.3337438457010763
train loss:0.3376112773625246
train loss:0.2882500597687466
train loss:0.3943436427799443
train loss:0.2653471013432858
train loss:0.24000488490735894
train loss:0.4009245886153511
train loss:0.33491185468375567
train loss:0.2139139889464615
train loss:0.24939994169114185
train loss:0.2597576913737317
train loss:0.39376683171454047
train loss:0.24032670855045354

train loss:0.31001366621556875
train loss:0.32851745177961605
train loss:0.23132645416937003
train loss:0.19049949848738607
train loss:0.25961844669511114
train loss:0.26511008759638044
train loss:0.2836614197971181
train loss:0.30489470518604633
train loss:0.4428968983860734
train loss:0.3740078742725574
train loss:0.3529071814069982
train loss:0.216128939700678
train loss:0.2824732768690516
train loss:0.4050072016785328
train loss:0.37610193258444746
train loss:0.2186213771187509
train loss:0.28953513084521454
train loss:0.37848118068813
train loss:0.30964295838863853
train loss:0.3480530673534597
train loss:0.36025252793829154
train loss:0.25508836366700144
=== epoch:13, train acc:0.89, test acc:0.845 ===
train loss:0.2144579723494447
train loss:0.3714046547385837
train loss:0.34827097911030597
train loss:0.21984235532754076
train loss:0.27393719410822637
train loss:0.2847618406137403
train loss:0.3624967418703814
train loss:0.17705172685142567
train loss:0.43957624088454056
train loss:0.37999366172099125
train loss:0.32924670810672274
train loss:0.3447278787267981
train loss:0.23368371643972607
train loss:0.2164317810973962
train loss:0.296703102494575
train loss:0.33124267506200694
train loss:0.3304100850000744

train loss:0.23081287439612164
train loss:0.27819370295382084
train loss:0.33041504909014074
train loss:0.20435633308933032
train loss:0.3449539751982844
train loss:0.29031184995503095
train loss:0.26326780788736265
train loss:0.23787709257059228
train loss:0.33333510932116733
train loss:0.44021391696994866
train loss:0.3310294792988489
train loss:0.28242818486320614
train loss:0.3795347323288985
train loss:0.2918382383858946
train loss:0.15878439679312387
train loss:0.27237098948303273
train loss:0.33875787843674826
train loss:0.3003704243373144
train loss:0.26983054474978285
train loss:0.2179638842154673
train loss:0.210862352822213
train loss:0.30729434985164267
train loss:0.2772713522719136
train loss:0.22696417001826139
train loss:0.2862656606339745
train loss:0.295555567437843
train loss:0.2720318792072418
train loss:0.3294099597171097
train loss:0.24001592508180175
train loss:0.30020169224870435
train loss:0.27954350267410166
train loss:0.30916846302592565
train loss:0.2797081202812717
=== epoch:14, train acc:0.885, test acc:0.843 ===
train loss:0.23515737374260096
train loss:0.3113057865121211
train loss:0.3199494539195401
train loss:0.2842317239739998
train loss:0.3045442665411771
train loss:0.25997475041651186

train loss:0.33907840908075393
train loss:0.22002924779252722
train loss:0.41509734623314537
train loss:0.2558400986066942
train loss:0.27701968431347074
train loss:0.38597466429109345
train loss:0.3622538427542629
train loss:0.17872240725576763
train loss:0.2524354325547618
train loss:0.2205730920799819
train loss:0.3100441985914994
train loss:0.20862012611726105
train loss:0.23505007658796973
train loss:0.24258421560889004
train loss:0.25245800096658727
train loss:0.2331505317785377
train loss:0.42092642470051544
train loss:0.3731123815667601
train loss:0.3723532266442212
train loss:0.3673434714312236
train loss:0.32607966485549006
train loss:0.1518037150422643
train loss:0.2710108992594637
train loss:0.2488735798117503
train loss:0.3135993622302561
train loss:0.3177079442678911
train loss:0.3796419286349655
train loss:0.25789458530182874
train loss:0.2164064496919144
train loss:0.27296982192526337
train loss:0.3313917852021524
train loss:0.3909161593077128
train loss:0.31839905509412403
train loss:0.3038636636057585
train loss:0.1952188722414513
train loss:0.3464093554082855
train loss:0.2298990598755737
train loss:0.3875960514587924
train loss:0.34371123787836466
train loss:0.32830190480647725

train loss:0.28398850052551866
train loss:0.1956066345431859
train loss:0.26039981831679826
train loss:0.21753638055419086
=== epoch:15, train acc:0.901, test acc:0.85 ===
train loss:0.20941945424566522
train loss:0.23881783750720717
train loss:0.19411538245154714
train loss:0.28366187260759745
train loss:0.21439120957487912
train loss:0.23232334105424965
train loss:0.20770395384981277
train loss:0.3097987344328504
train loss:0.24249304698678947
train loss:0.21868752478162143
train loss:0.27597864891538804
train loss:0.274857898441669
train loss:0.24070154147886627
train loss:0.32503776931200035
train loss:0.21000974491373417
train loss:0.22831390973469173
train loss:0.2653637021692638
train loss:0.1659253323673836
train loss:0.2778857903219903
train loss:0.2660649137224234
train loss:0.21859581510052312
train loss:0.21786307998693874
train loss:0.21657600926628645
train loss:0.2929321874774182
train loss:0.2305481661960171
train loss:0.2702794000050663
train loss:0.33657644906802736
train loss:0.2630499050983304
train loss:0.4169101858718293
train loss:0.28775364220200145
train loss:0.2875405740493948
train loss:0.43539768035984866
train loss:0.18234845729133056
train loss:0.34616648279140555
train loss:0.25813687667400964

```
train loss:0.2935557330277801
train loss:0.2825641936488471
train loss:0.24085475454070476
train loss:0.18614712237647127
train loss:0.13305010649825968
train loss:0.3551765673781459
train loss:0.2103378196793836
train loss:0.28556254826280536
train loss:0.25620570859470937
train loss:0.37536235284838865
train loss:0.205829555642254
train loss:0.2688938957194524
train loss:0.3719793617328068
train loss:0.22624041018108862
train loss:0.19027399466475342
=== epoch:16, train acc:0.914, test acc:0.848 ===
train loss:0.29118907922171466
train loss:0.34550694994045933
train loss:0.4195331306577064
train loss:0.3131910949383354
train loss:0.2872468150151294
train loss:0.3501824498394923
train loss:0.3345639287818251
train loss:0.21962230055179105
train loss:0.34884782257236013
train loss:0.31070816081863956
train loss:0.17521737622053876
train loss:0.2517274565865655
train loss:0.3714978245402266
train loss:0.20483225830986862
train loss:0.2524862212162985
train loss:0.27551487381987483
train loss:0.22428988150296525
train loss:0.26575898392091873
train loss:0.23332057931597128
train loss:0.2617051740612112
train loss:0.30574359291878345
train loss:0.22298487495339814
train loss:0.1997577902493195
train loss:0.28395427485479435
```

```
train loss:0.2450107034255125
train loss:0.2863492789400201
train loss:0.28346812609098254
train loss:0.22045797076022594
train loss:0.21710006502303486
train loss:0.20564424259387754
train loss:0.18805348937242392
train loss:0.24534627346857873
train loss:0.23808820364331823
train loss:0.24548764530195843
train loss:0.27589953930967304
train loss:0.3475275126557733
train loss:0.2447854764927527
train loss:0.22724900441347784
train loss:0.21412456324786994
train loss:0.2975278388477962
train loss:0.1940411152837857
train loss:0.2606435068617787
train loss:0.1841489140097156
train loss:0.2570339672001493
train loss:0.16645851311686177
train loss:0.25719454374958384
train loss:0.2224412816064754
train loss:0.30612761676121353
train loss:0.21728682291257545
train loss:0.2562433697042213
=== epoch:17, train acc:0.916, test acc:0.86 ===
train loss:0.25672642719476135
train loss:0.23339615683525092
train loss:0.2696623803543604
train loss:0.1714665305329771
train loss:0.2055642457223265
train loss:0.18114604630993678
train loss:0.25511848528851927
train loss:0.30763107368715886
train loss:0.22817896157531659
train loss:0.22290034001909306
train loss:0.33176506334939515
train loss:0.20810791994212033
train loss:0.18128737876020523
```

train loss:0.3664953772317298
train loss:0.20373634846762126
train loss:0.22261663007793117
train loss:0.1574086317334477
train loss:0.36499736165743407
train loss:0.2262641028782647
train loss:0.2574876932874116
train loss:0.15052592487995364
train loss:0.27833008322541763
train loss:0.2317586504416721
train loss:0.3045850799710278
train loss:0.33139038508502033
train loss:0.2718718330841102
train loss:0.13821049139860497
train loss:0.252678079385745
train loss:0.25206241581835864
train loss:0.2303334138963151
train loss:0.210496054705796
train loss:0.3363738405068804
train loss:0.30996280891124567
train loss:0.30206645178851604
train loss:0.1941918324028883
train loss:0.23491038238993817
train loss:0.28917290423329256
train loss:0.17348493788343475
train loss:0.24693256545439174
train loss:0.2648329623514425
train loss:0.1710461746731865
train loss:0.2693842621991176
train loss:0.1971999430863471
train loss:0.313450251959953
train loss:0.32269934802852995
train loss:0.18702233186467043
train loss:0.26266147245479315
train loss:0.2142510755249085
train loss:0.2993331871362717
train loss:0.27515715104600696
=== epoch:18, train acc:0.907, test acc:0.849 ===
train loss:0.23571849183019963
train loss:0.14629274479710433

train loss:0.21794622615329515
train loss:0.2823593957692461
train loss:0.16386007467078884
train loss:0.21618544286998706
train loss:0.3002887464778099
train loss:0.12297900447297769
train loss:0.2721651339453925
train loss:0.23651784233870013
train loss:0.21698654562755695
train loss:0.18998410215299816
train loss:0.2571286657538874
train loss:0.1975709666122953
train loss:0.27262343176180875
train loss:0.2105836170806824
train loss:0.19381586452196103
train loss:0.2649592492287251
train loss:0.15522626902405665
train loss:0.2107517269879478
train loss:0.230114515258047
train loss:0.15497258065200278
train loss:0.270758369735404
train loss:0.13723683503005726
train loss:0.19015226808127572
train loss:0.19179484261953697
train loss:0.1738532056550241
train loss:0.345410954720792
train loss:0.25837826683324844
train loss:0.26128963749323536
train loss:0.20677024706120725
train loss:0.26629641133397913
train loss:0.28486522676678294
train loss:0.25816576236244343
train loss:0.2252751966633362
train loss:0.2778063581759793
train loss:0.2981868819482935
train loss:0.3293915522329927
train loss:0.22215432864732826
train loss:0.29111823776880785
train loss:0.2526635246379572
train loss:0.23694269753438532

train loss:0.24557547718430345
train loss:0.2523075814000119
train loss:0.22509019161635
train loss:0.2303042113078578
train loss:0.3054954479805245
train loss:0.16796762529255777
train loss:0.287632971790604
train loss:0.21771918953687705
=== epoch:19, train acc:0.92, test acc:0.853 ===
train loss:0.2543721808875864
train loss:0.23272648517158617
train loss:0.2621005534963261
train loss:0.23131692837319304
train loss:0.17537416135362704
train loss:0.2017135857323301
train loss:0.2775378216038335
train loss:0.1387432864337108
train loss:0.20820821704475745
train loss:0.2264556505059681
train loss:0.28391261329087486
train loss:0.19342770864195458
train loss:0.17086170639647633
train loss:0.27695714992237863
train loss:0.22681323178991605
train loss:0.25869328774781963
train loss:0.23893982595733917
train loss:0.2392643666241621
train loss:0.30392422205535075
train loss:0.16357283133468695
train loss:0.17108229493958227
train loss:0.23712884100140827
train loss:0.193809676564742
train loss:0.24237302482396467
train loss:0.18922111564161345
train loss:0.20990088075879484
train loss:0.229844588269361
train loss:0.2390364902106555
train loss:0.15119997789289147
train loss:0.1775575754068043
train loss:0.2662265784358862

train loss:0.26580252832425805
train loss:0.20815376753296305
train loss:0.17743196193410074
train loss:0.2132827730838528
train loss:0.25740476060064466
train loss:0.2569687263061877
train loss:0.19990613343918612
train loss:0.26606095285769055
train loss:0.1881389795246917
train loss:0.24423615200188845
train loss:0.16372616412029958
train loss:0.21418986917235075
train loss:0.2628131293954208
train loss:0.26712255022971604
train loss:0.3061837626094236
train loss:0.10705984745946596
train loss:0.2673617971059104
train loss:0.29429110065509506
train loss:0.14699802211292431
=== epoch:20, train acc:0.926, test acc:0.855 ===
train loss:0.2256898896461125
train loss:0.09558318671041573
train loss:0.18528153833829908
train loss:0.15653381231392413
train loss:0.2092718164001972
train loss:0.15226254732612157
train loss:0.23828747655831722
train loss:0.16351736455267404
train loss:0.18858360081435266
train loss:0.23722699722685284
train loss:0.19763070286306167
train loss:0.22661429051340087
train loss:0.31901485164854343
train loss:0.277975884240098
train loss:0.1759019577229841
train loss:0.12912962780340725
train loss:0.2376689379560845
train loss:0.27403181814493627
train loss:0.16504678817430418
train loss:0.1871334757866861


```
train loss:0.24341872807373455
train loss:0.20142897633994722
train loss:0.192688689882475
train loss:0.34486970730687183
train loss:0.17349694266356558
train loss:0.2627034795343041
train loss:0.29412369633908797
train loss:0.2776343924415566
train loss:0.2870021694069745
train loss:0.22985567291196474
train loss:0.29467468522456014
train loss:0.19407142944562777
train loss:0.19106692672405592
train loss:0.23080751403848532
train loss:0.17830516361298618
train loss:0.22077461614252317
train loss:0.1797140087079008
train loss:0.18007760525684563
train loss:0.19577157424449967
train loss:0.13687111409346342
train loss:0.17925939490824142
train loss:0.3179180764211106
train loss:0.19782040918937366
train loss:0.196959492108584
train loss:0.3270782549533721
train loss:0.2733455774432886
train loss:0.23837083396916894
train loss:0.2826961517608833
train loss:0.21441027736985074
===== Final Test Accuracy =====
test acc:0.85575
Saved Network Parameters!
```