Task_IV

March 26, 2021

[1]:

```
import torch
import numpy as np
import dgl

import networkx as nx

import matplotlib.pyplot as plt
```

```
Using backend: pytorch
```

[2]:

```
torch.cuda.device_count()
```

[2]:

```
8
```

[3]:

```
torch.cuda.get_device_name(0)
```

[3]:

```
'A100-SXM4-40GB'
```

[4]:

```
from dgl.nn.pytorch.conv import GraphConv
import torch.nn.functional as F
```

[5]:

```
# Load data
data = np.load('QG_jets.npz')
X = data['X']
y = data['y']
print(X.shape)
```

```
(100000, 139, 4)
```

[6]:

```
# Generate Graphs
def genGraph(event):
    tData = X[event][~np.all(X[event] == 0, axis=1)]
    # Preprocess feature by centering the jets and normalising the pT
    tData[:,1:3] -= np.average(tData[:,1:3], weights=tData[:,0], axis=0)
    tData[:, 0] /= np.sum(X[:, 0])
    tData = tData[tData[:,0].argsort()][::-1].copy()
    numNodes = tData.shape[0]
    tGraph = nx.complete_graph(numNodes)
    tGraph = dgl.from_networkx(tGraph)
    tGraph = dgl.add_self_loop(tGraph)
    tGraph.ndata['features'] = torch.tensor(tData)
    return tGraph.int()
```

[8]:

```
size_limit = 3000
graphs = []
counter = 0
try:
    for i in range(min(len(X), size_limit)):
        graphs.append(genGraph(i))
        counter += 1
        if counter % 1000 == 0:
            print("Processed graph: ", counter)
except KeyboardInterrupt:
    pass
```

```
Processed graph:  1000
Processed graph:  2000
Processed graph:  3000
```

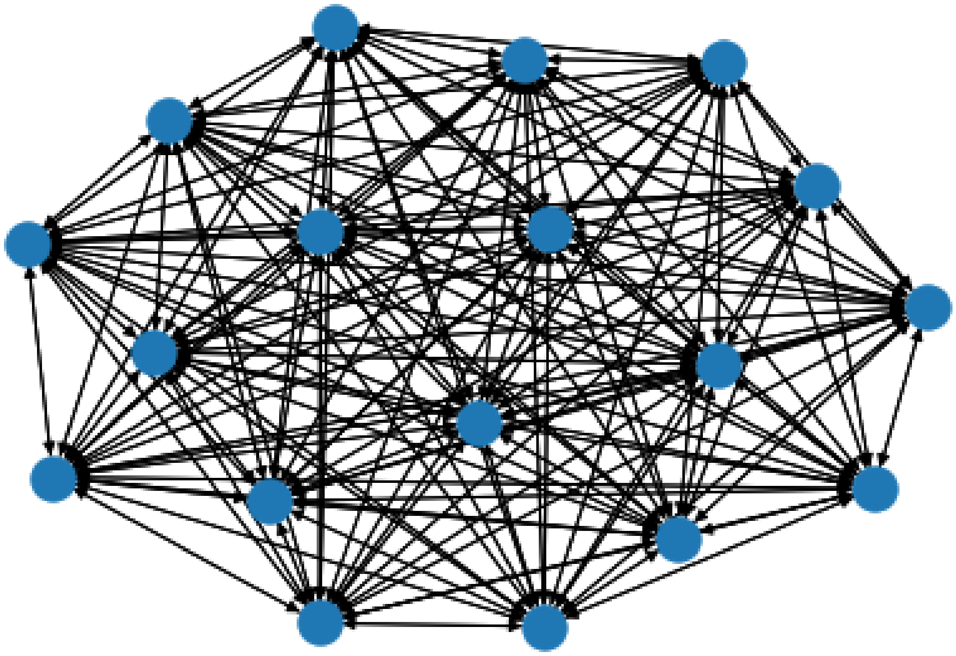
[9]:

```
def display(graph):
    print("Graph with ", graph.number_of_nodes(), " nodes and ", graph.
 ↪number_of_edges(), " edges.")
    nx.draw(graph.to_networkx())
```

[12]:

```
display(graphs[0])
```

```
Graph with  18  nodes and  324  edges.
```



[13]:
```
sizes = (int(0.8* len(graphs)), int(0.1* len(graphs)), len(graphs)-int(0.9*␣
 ↪len(graphs)))
```

[14]:
```
print(sizes)
```

```
(2400, 300, 300)
```

[15]:
```
dsTrain, dsVal, dsTest = torch.utils.data.random_split(list(zip(graphs, y)),␣
 ↪sizes)
print("Train: ", len(dsTrain), "\nValidation: ", len(dsVal), "\nTest: ",␣
 ↪len(dsTest))
```

```
Train:  2400
Validation:  300
Test:  300
```

[16]:
```
trainDataGen = dgl.dataloading.GraphDataLoader(dsTrain, batch_size = sizes[0],␣
 ↪drop_last=False, shuffle=True)
valDataGen = dgl.dataloading.GraphDataLoader(dsVal, batch_size = sizes[1],␣
 ↪drop_last=False, shuffle=False)
testDataGen = dgl.dataloading.GraphDataLoader(dsTest, batch_size = sizes[2],␣
 ↪drop_last=False, shuffle=False)
```

[17]:

```
class GCN(torch.nn.Module):
    def __init__(self, in_dim, hidden_dim, out_feats):
        super(GCN, self).__init__()
        self.conv1 = GraphConv(in_dim, hidden_dim)
        self.conv2 = GraphConv(hidden_dim, hidden_dim)
        self.conv3 = GraphConv(hidden_dim, out_feats)
        self.bn1 = torch.nn.BatchNorm1d(num_features=hidden_dim)
        self.bn2 = torch.nn.BatchNorm1d(num_features=hidden_dim)
        self.bn3 = torch.nn.BatchNorm1d(num_features=out_feats)

    def forward(self, graph, h):
        h = F.relu(self.bn1(self.conv1(graph, h)))
        h = F.relu(self.bn2(self.conv2(graph, h)))
        h = F.relu(self.bn3(self.conv3(graph, h)))
        graph.ndata['tmp_feature'] = h
        h = dgl.mean_nodes(graph, 'tmp_feature')
        h = torch.sigmoid(h)
        h = (h-0.5) * 2
        return h
```

[18]:

```
# Training function.
def train(model, optimizer, epochs=100, loss_func=torch.nn.MSELoss()):
    epoch_losses = {'train': [], 'val': []}
    for epoch in range(epochs):
        train_loss = 0
        for ibatch, (batched_graph, labels) in enumerate(trainDataGen):
            node_features = batched_graph.ndata['features']
            pred = model(batched_graph, node_features.float())
            loss = loss_func(pred, labels.float())
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()

            train_loss += loss.detach().item()

        train_loss /= (ibatch + 1)

        epoch_losses['train'].append(train_loss)

        for ibatch, (batched_graph, labels) in enumerate(valDataGen):
            node_features = batched_graph.ndata['features']
            pred = model(batched_graph, node_features.float())
            val_loss = loss_func(pred, labels.float()).detach().item()
            epoch_losses['val'].append(val_loss)
            assert(ibatch == 0)
        print("Epoch: ", epoch+1, "\t Training Loss: ", train_loss, "\t␣
 ↪Validation Loss:", val_loss)
```

```
    return epoch_losses
```

[21]:
```
model = GCN(graphs[0].ndata[list(graphs[0].ndata.keys())[0]].shape[1],␣
 ↪hidden_dim=9, out_feats=1)
```

[22]:
```
history = train(model, optimizer=torch.optim.Adam(model.parameters(), lr=0.1))
```

```
/lusnlsas/diat/anandw/.linuxbrew/opt/python@3.8/lib/python3.8/site-
packages/torch/nn/modules/loss.py:528: UserWarning: Using a target size
(torch.Size([2400])) that is different to the input size (torch.Size([2400,
1])). This will likely lead to incorrect results due to broadcasting. Please
ensure they have the same size.
  return F.mse_loss(input, target, reduction=self.reduction)
/lusnlsas/diat/anandw/.linuxbrew/opt/python@3.8/lib/python3.8/site-
packages/torch/nn/modules/loss.py:528: UserWarning: Using a target size
(torch.Size([300])) that is different to the input size (torch.Size([300, 1])).
This will likely lead to incorrect results due to broadcasting. Please ensure
they have the same size.
  return F.mse_loss(input, target, reduction=self.reduction)
```

```
Epoch:  1        Training Loss:  0.45926305651664734     Validation Loss:
0.38786134123802185
Epoch:  2        Training Loss:  0.37437817454338074     Validation Loss:
0.3661763072013855
Epoch:  3        Training Loss:  0.3538920283317566      Validation Loss:
0.3484859764575958
Epoch:  4        Training Loss:  0.33937546610832214     Validation Loss:
0.3305249810218811
Epoch:  5        Training Loss:  0.32275885343551636     Validation Loss:
0.315915048122406
Epoch:  6        Training Loss:  0.30986663699150085     Validation Loss:
0.3134375512599945
Epoch:  7        Training Loss:  0.31150245666503906     Validation Loss:
0.3075641393661499
Epoch:  8        Training Loss:  0.30615904927253723     Validation Loss:
0.2960270345211029
Epoch:  9        Training Loss:  0.3004401624202728      Validation Loss:
0.2932964265346527
Epoch:  10       Training Loss:  0.29642945528030396     Validation Loss:
0.2921285927295685
Epoch:  11       Training Loss:  0.29661357402801514     Validation Loss:
0.291047602891922
Epoch:  12       Training Loss:  0.29638248682022095     Validation Loss:
0.28958660364151
Epoch:  13       Training Loss:  0.29495367407798767     Validation Loss:
0.28786689043045044
```

```
Epoch:  14       Training Loss:  0.29287126660346985     Validation Loss:
0.28565555810928345
Epoch:  15       Training Loss:  0.29038798809051514     Validation Loss:
0.28304019570350647
Epoch:  16       Training Loss:  0.28765690326690674     Validation Loss:
0.28013375401496887
Epoch:  17       Training Loss:  0.2846394181251526      Validation Loss:
0.2773338556289673
Epoch:  18       Training Loss:  0.2816648781299591      Validation Loss:
0.27505654096603394
Epoch:  19       Training Loss:  0.27942878007888794     Validation Loss:
0.27367857098579407
Epoch:  20       Training Loss:  0.27789661288261414     Validation Loss:
0.27311477065086365
Epoch:  21       Training Loss:  0.2765974700450897      Validation Loss:
0.2725489139556885
Epoch:  22       Training Loss:  0.2751060426235199      Validation Loss:
0.2720009982585907
Epoch:  23       Training Loss:  0.27364879846572876     Validation Loss:
0.27141255140304565
Epoch:  24       Training Loss:  0.2722169756889343      Validation Loss:
0.2706570625305176
Epoch:  25       Training Loss:  0.2709551751613617      Validation Loss:
0.269696444272995
Epoch:  26       Training Loss:  0.2698689103126526      Validation Loss:
0.2686547338962555
Epoch:  27       Training Loss:  0.2688758373260498      Validation Loss:
0.267645001411438
Epoch:  28       Training Loss:  0.26789233088493347     Validation Loss:
0.2666469216346741
Epoch:  29       Training Loss:  0.26686549186706543     Validation Loss:
0.2655336260795593
Epoch:  30       Training Loss:  0.26578769087791443     Validation Loss:
0.26411172747612
Epoch:  31       Training Loss:  0.26459866762161255     Validation Loss:
0.2626971900463104
Epoch:  32       Training Loss:  0.2633664906024933      Validation Loss:
0.2615220844745636
Epoch:  33       Training Loss:  0.261846125125885       Validation Loss:
0.2605413496494293
Epoch:  34       Training Loss:  0.2601882815361023      Validation Loss:
0.2585625946521759
Epoch:  35       Training Loss:  0.25851017236709595     Validation Loss:
0.2563161253929138
Epoch:  36       Training Loss:  0.25679701566696167     Validation Loss:
0.2542555630207062
Epoch:  37       Training Loss:  0.25517675280570984     Validation Loss:
0.25237539410591125
```

```
Epoch:  38       Training Loss:  0.25343430042266846     Validation Loss:
0.250460147857666
Epoch:  39       Training Loss:  0.2513278126716614      Validation Loss:
0.2495788186788559
Epoch:  40       Training Loss:  0.25061213970184326     Validation Loss:
0.250471293926239
Epoch:  41       Training Loss:  0.25168555974960327     Validation Loss:
0.2513202130794525
Epoch:  42       Training Loss:  0.2527340352535248      Validation Loss:
0.252109169960022
Epoch:  43       Training Loss:  0.2537427246570587      Validation Loss:
0.2526598870754242
Epoch:  44       Training Loss:  0.2544143795967102      Validation Loss:
0.2526434063911438
Epoch:  45       Training Loss:  0.2543122172355652      Validation Loss:
0.25220969319343567
Epoch:  46       Training Loss:  0.2535657286643982      Validation Loss:
0.2517828047275543
Epoch:  47       Training Loss:  0.2527487874031067      Validation Loss:
0.2515217363834381
Epoch:  48       Training Loss:  0.25218692421913147     Validation Loss:
0.2513461709022522
Epoch:  49       Training Loss:  0.2518060803413391      Validation Loss:
0.2511376142501831
Epoch:  50       Training Loss:  0.2514379918575287      Validation Loss:
0.2508256435394287
Epoch:  51       Training Loss:  0.25098103284835815     Validation Loss:
0.2504206895828247
Epoch:  52       Training Loss:  0.25043654441833496     Validation Loss:
0.2501487731933594
Epoch:  53       Training Loss:  0.25004109740257263     Validation Loss:
0.25052592158317566
Epoch:  54       Training Loss:  0.25036630034446716     Validation Loss:
0.25128498673439026
Epoch:  55       Training Loss:  0.2510969042778015      Validation Loss:
0.2514782249927521
Epoch:  56       Training Loss:  0.25126951932907104     Validation Loss:
0.25090107321739197
Epoch:  57       Training Loss:  0.2508133351802826      Validation Loss:
0.25018736720085144
Epoch:  58       Training Loss:  0.2501586079597473      Validation Loss:
0.24993281066417694
Epoch:  59       Training Loss:  0.25000429153442383     Validation Loss:
0.2500235438346863
Epoch:  60       Training Loss:  0.2502082586288452      Validation Loss:
0.2501676082611084
Epoch:  61       Training Loss:  0.25044986605644226     Validation Loss:
0.25023871660232544
```

```
Epoch:  62       Training Loss:  0.2505912780761719      Validation Loss:
0.2502082288265228
Epoch:  63       Training Loss:  0.2505992650985718      Validation Loss:
0.2500857710838318
Epoch:  64       Training Loss:  0.250482976436615       Validation Loss:
0.24990792572498322
Epoch:  65       Training Loss:  0.2502821087837219      Validation Loss:
0.24975088238716125
Epoch:  66       Training Loss:  0.2500801384449005      Validation Loss:
0.24973027408123016
Epoch:  67       Training Loss:  0.2500040829181671      Validation Loss:
0.24990303814411163
Epoch:  68       Training Loss:  0.25012388825416565     Validation Loss:
0.25011393427848816
Epoch:  69       Training Loss:  0.2502892315387726      Validation Loss:
0.250139445066452
Epoch:  70       Training Loss:  0.2502754032611847      Validation Loss:
0.2500055432319641
Epoch:  71       Training Loss:  0.2501116096973419      Validation Loss:
0.24990740418434143
Epoch:  72       Training Loss:  0.24999774992465973     Validation Loss:
0.24992716312408447
Epoch:  73       Training Loss:  0.25001367926597595     Validation Loss:
0.2500055432319641
Epoch:  74       Training Loss:  0.25009340047836304     Validation Loss:
0.25006479024887085
Epoch:  75       Training Loss:  0.2501542568206787      Validation Loss:
0.25006741285324097
Epoch:  76       Training Loss:  0.25015661120414734     Validation Loss:
0.2500167191028595
Epoch:  77       Training Loss:  0.2501039505004883      Validation Loss:
0.24994654953479767
Epoch:  78       Training Loss:  0.250031977891922       Validation Loss:
0.24990437924861908
Epoch:  79       Training Loss:  0.24999114871025085     Validation Loss:
0.24991682171821594
Epoch:  80       Training Loss:  0.2500108778476715      Validation Loss:
0.24995310604572296
Epoch:  81       Training Loss:  0.25006142258644104     Validation Loss:
0.24995039403438568
Epoch:  82       Training Loss:  0.25007858872413635     Validation Loss:
0.24989253282546997
Epoch:  83       Training Loss:  0.25004395842552185     Validation Loss:
0.24982641637325287
Epoch:  84       Training Loss:  0.25000160932540894     Validation Loss:
0.2497965693473816
Epoch:  85       Training Loss:  0.2499922513961792      Validation Loss:
0.249803826212883
```

```
Epoch:  86      Training Loss:  0.25001242756843567     Validation Loss:
0.2498237043619156
Epoch:  87      Training Loss:  0.25003454089164734     Validation Loss:
0.2498362809419632
Epoch:  88      Training Loss:  0.2500379681587219      Validation Loss:
0.2498382180929184
Epoch:  89      Training Loss:  0.2500210702419281      Validation Loss:
0.24984054267406464
Epoch:  90      Training Loss:  0.2499985247850418      Validation Loss:
0.249857097864151
Epoch:  91      Training Loss:  0.2499888837337494      Validation Loss:
0.24988935887813568
Epoch:  92      Training Loss:  0.2499982863664627      Validation Loss:
0.2499203085899353
Epoch:  93      Training Loss:  0.25001317262649536     Validation Loss:
0.2499300092458725
Epoch:  94      Training Loss:  0.2500152587890625      Validation Loss:
0.24991698563098907
Epoch:  95      Training Loss:  0.25000327825546265     Validation Loss:
0.2498970329761505
Epoch:  96      Training Loss:  0.24999147653579712     Validation Loss:
0.2498835027217865
Epoch:  97      Training Loss:  0.24999059736728668     Validation Loss:
0.24987682700157166
Epoch:  98      Training Loss:  0.2499978244304657      Validation Loss:
0.24987028539180756
Epoch:  99      Training Loss:  0.25000354647636414     Validation Loss:
0.24985985457897186
Epoch:  100     Training Loss:  0.25000202655792236     Validation Loss:
0.2498478889465332
```

```
[ ]:
```