```
import torch
import torch.nn as nn
import torch.nn.functional as F
from torchvision import models, datasets, transforms
from torch.utils.data import DataLoader
import matplotlib.pyplot as plt
import numpy as np
import cv2

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print("Device:", device)
```

```
Device: cuda
```

```
model = models.densenet121(weights=models.DenseNet121_Weights.DEFAULT)
num_ftrs = model.classifier.in_features
model.classifier = nn.Sequential(
    nn.Linear(num_ftrs, 256),
    nn.ReLU(),
    nn.Dropout(0.4),
    nn.Linear(256, 2)
)
model = model.to(device)
model.load_state_dict(torch.load("best_pneumonia_densenet121.pt",
map_location=device))
model.eval()
```

```
C:\Users\Ekaansh\AppData\Local\Temp\ipykernel_4564\4104618499.py:10:
FutureWarning: You are using `torch.load` with `weights_only=False`
(the current default value), which uses the default pickle module
implicitly. It is possible to construct malicious pickle data which
will execute arbitrary code during unpickling (See
https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-
models for more details). In a future release, the default value for
`weights_only` will be flipped to `True`. This limits the functions
that could be executed during unpickling. Arbitrary objects will no
longer be allowed to be loaded via this mode unless they are
explicitly allowlisted by the user via
`torch.serialization.add_safe_globals`. We recommend you start setting
`weights_only=True` for any use case where you don't have full control
of the loaded file. Please open an issue on GitHub for any issues
related to this experimental feature.
  model.load_state_dict(torch.load("best_pneumonia_densenet121.pt",
map_location=device))
```

```
DenseNet(
  (features): Sequential(
    (conv0): Conv2d(3, 64, kernel_size=(7, 7), stride=(2, 2),
padding=(3, 3), bias=False)
    (norm0): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True,
```

```
track_running_stats=True)
    (relu0): ReLU(inplace=True)
    (pool0): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1,
ceil_mode=False)
    (denseblock1): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(64, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(96, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(96, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(128, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(160, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(160, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(192, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(192, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(224, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(224, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition1): _Transition(
      (norm): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock2): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(128, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
```

```
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(160, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(160, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(192, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(192, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(224, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(224, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(256, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
```

```
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(288, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(288, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(320, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(320, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(352, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(352, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(384, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(384, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
```

```
        (norm1): BatchNorm2d(416, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(416, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(448, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(448, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(480, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(480, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition2): _Transition(
      (norm): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(512, 256, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock3): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(256, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
```

```
        (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(288, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(288, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(320, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(320, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(352, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(352, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(384, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(384, 128, kernel_size=(1, 1), stride=(1, 1),
```

```
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(416, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(416, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(448, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(448, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(480, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(480, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(512, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(512, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
```

```
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
        (norm1): BatchNorm2d(544, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(544, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(576, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(576, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(608, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(608, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer13): _DenseLayer(
        (norm1): BatchNorm2d(640, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(640, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
```

```
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer14): _DenseLayer(
        (norm1): BatchNorm2d(672, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(672, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer15): _DenseLayer(
        (norm1): BatchNorm2d(704, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(704, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer16): _DenseLayer(
        (norm1): BatchNorm2d(736, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(736, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer17): _DenseLayer(
        (norm1): BatchNorm2d(768, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(768, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer18): _DenseLayer(
        (norm1): BatchNorm2d(800, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(800, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer19): _DenseLayer(
        (norm1): BatchNorm2d(832, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(832, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer20): _DenseLayer(
        (norm1): BatchNorm2d(864, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(864, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer21): _DenseLayer(
        (norm1): BatchNorm2d(896, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
```

```
padding=(1, 1), bias=False)
      )
      (denselayer22): _DenseLayer(
        (norm1): BatchNorm2d(928, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(928, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer23): _DenseLayer(
        (norm1): BatchNorm2d(960, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(960, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer24): _DenseLayer(
        (norm1): BatchNorm2d(992, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(992, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition3): _Transition(
      (norm): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock4): _DenseBlock(
      (denselayer1): _DenseLayer(
```

```
        (norm1): BatchNorm2d(512, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(512, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(544, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(544, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(576, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(576, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(608, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(608, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(640, eps=1e-05, momentum=0.1,
```

```
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(640, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(672, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(672, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(704, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(704, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(736, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(736, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(768, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(768, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
        (norm1): BatchNorm2d(800, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(800, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(832, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(832, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(864, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(864, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer13): _DenseLayer(
        (norm1): BatchNorm2d(896, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
```

```
        (conv1): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer14): _DenseLayer(
        (norm1): BatchNorm2d(928, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(928, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer15): _DenseLayer(
        (norm1): BatchNorm2d(960, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(960, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer16): _DenseLayer(
        (norm1): BatchNorm2d(992, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(992, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (norm5): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  )
  (classifier): Sequential(
```

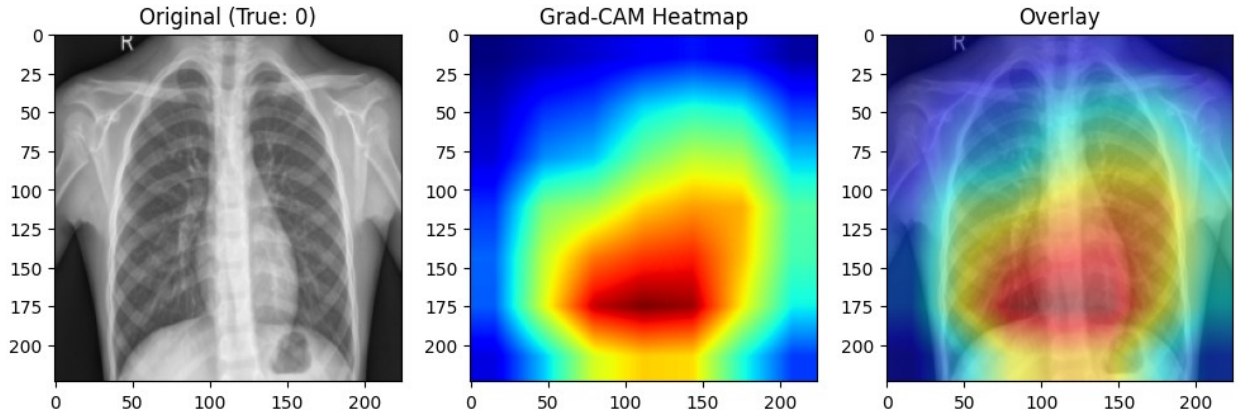
```
    (0): Linear(in_features=1024, out_features=256, bias=True)
    (1): ReLU()
    (2): Dropout(p=0.4, inplace=False)
    (3): Linear(in_features=256, out_features=2, bias=True)
  )
)

val_test_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize([0.485], [0.229])
])

test_data = datasets.ImageFolder(
    r"D:\datasets\chest_xray\chest_xray\test",
    transform=val_test_transform
)

test_loader = DataLoader(test_data, batch_size=32, shuffle=False,
num_workers=2)

class GradCAM:
    def __init__(self, model, target_layer):
        self.model = model
        self.target_layer = target_layer
        self.gradients = None
        self.activations = None

        # hooks
        target_layer.register_forward_hook(self.save_activation)
        target_layer.register_backward_hook(self.save_gradient)

    def save_activation(self, module, input, output):
        self.activations = output.detach()

    def save_gradient(self, module, grad_input, grad_output):
        self.gradients = grad_output[0].detach()

    def generate(self, input_tensor, class_idx=None):
        self.model.eval()
        output = self.model(input_tensor)

        if class_idx is None:
            class_idx = output.argmax(dim=1).item()

        self.model.zero_grad()
        one_hot = torch.zeros_like(output)
        one_hot[0, class_idx] = 1
        output.backward(gradient=one_hot, retain_graph=True)

        weights = self.gradients.mean(dim=(2, 3), keepdim=True)
```

```
        cam = (weights * self.activations).sum(dim=1, keepdim=True)
        cam = F.relu(cam)

        cam = cam.squeeze().cpu().numpy()
        cam = cv2.resize(cam, (224, 224))
        cam = (cam - cam.min()) / (cam.max() - cam.min() + 1e-8)

        return cam

target_layer = model.features.denseblock4
gradcam = GradCAM(model, target_layer)

images, labels = next(iter(test_loader))
images, labels = images.to(device), labels.to(device)

input_tensor = images[0].unsqueeze(0)
true_label = labels[0].item()
cam = gradcam.generate(input_tensor)
```

```
c:\Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\site-
packages\torch\nn\modules\module.py:1827: FutureWarning: Using a non-
full backward hook when the forward contains multiple autograd Nodes
is deprecated and will be removed in future versions. This hook will
be missing some grad_input. Please use register_full_backward_hook to
get the documented behavior.
  self._maybe_warn_non_full_backward_hook(args, result, grad_fn)
c:\Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\site-
packages\torch\autograd\graph.py:825: UserWarning: Attempting to run
cuBLAS, but there was no current CUDA context! Attempting to set the
primary context... (Triggered internally at C:\actions-runner\_work\
pytorch\pytorch\builder\windows\pytorch\aten\src\ATen\cuda\
CublasHandlePool.cpp:135.)
  return Variable._execution_engine.run_backward(  # Calls into the C+
+ engine to run the backward pass
```

```
img = images[0].cpu().numpy().transpose(1, 2, 0)
img = (img * 0.229 + 0.485)
img = np.clip(img, 0, 1)

heatmap = cv2.applyColorMap(np.uint8(255 * cam), cv2.COLORMAP_JET)
heatmap = cv2.cvtColor(heatmap, cv2.COLOR_BGR2RGB)
overlay = cv2.addWeighted((img * 255).astype(np.uint8), 0.5, heatmap,
0.5, 0)

plt.figure(figsize=(12,4))
plt.subplot(1,3,1)
plt.imshow(img.squeeze(), cmap="gray")
plt.title(f"Original (True: {true_label})")

plt.subplot(1,3,2)
plt.imshow(cam, cmap="jet")
```

```
plt.title("Grad-CAM Heatmap")

plt.subplot(1,3,3)
plt.imshow(overlay)
plt.title("Overlay")

plt.show()
```

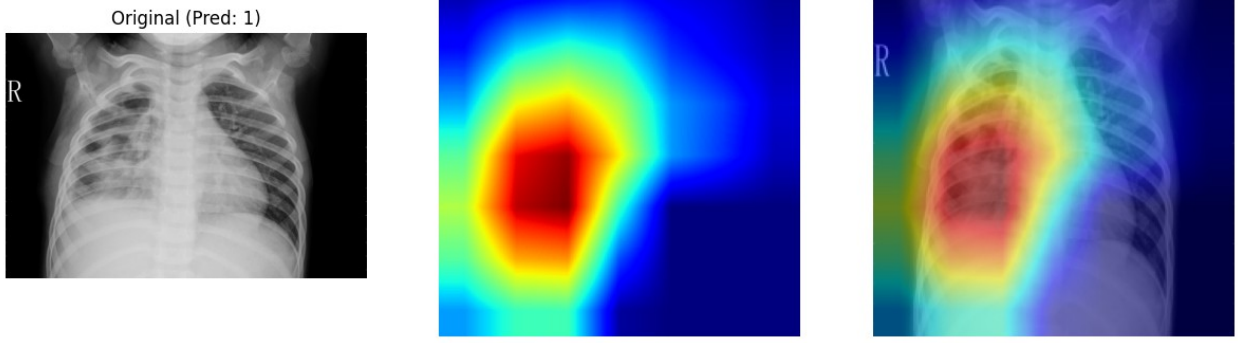



```
import torch
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
from torchvision import transforms
from pytorch_grad_cam import GradCAM
from pytorch_grad_cam.utils.model_targets import
ClassifierOutputTarget
from pytorch_grad_cam.utils.image import show_cam_on_image

cam = GradCAM(model=model, target_layers=[target_layer])

transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize([0.485, 0.456, 0.406],
                         [0.229, 0.224, 0.225])
])

def run_gradcam_on_image(image_path, model, target_layer):
    img = Image.open(image_path).convert('RGB')
    input_tensor = transform(img).unsqueeze(0).to(device)


    model.eval()
    output = model(input_tensor)
    pred = torch.argmax(output, 1).item()
```

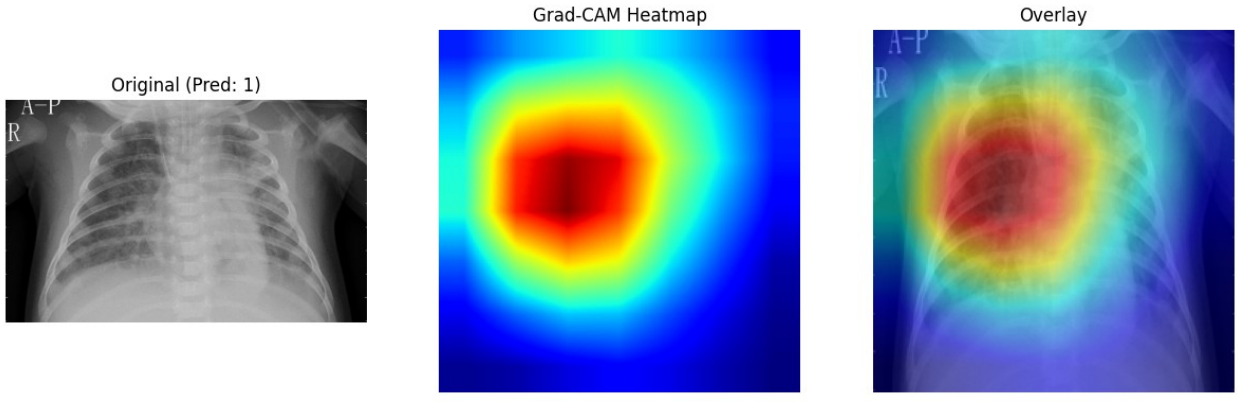
```
    cam = GradCAM(model=model, target_layers=[target_layer])
    grayscale_cam = cam(input_tensor=input_tensor,
targets=[ClassifierOutputTarget(pred)])[0, :]


    rgb_img = np.array(img.resize((224, 224))) / 255.0
    cam_image = show_cam_on_image(rgb_img, grayscale_cam,
use_rgb=True)

    # Plot results
    fig, (ax1, ax2, ax3) = plt.subplots(1, 3, figsize=(15,5))
    ax1.imshow(img, cmap='gray')
    ax1.set_title(f"Original (Pred: {pred})")
    ax1.axis('off')

    ax2.imshow(grayscale_cam, cmap='jet')
    ax2.set_title("Grad-CAM Heatmap")
    ax2.axis('off')

    ax3.imshow(cam_image)
    ax3.set_title("Overlay")
    ax3.axis('off')

    plt.show()

def run_on_multiple_images(image_paths, model, target_layer):
    for img_path in image_paths:
        print(f"\nProcessing: {img_path}")
        run_gradcam_on_image(img_path, model, target_layer)

image_files = [
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1954_bacteria_4886.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4874.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4875.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1947_bacteria_4876.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1949_bacteria_4880.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1950_bacteria_4881.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1951_bacteria_4882.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1952_bacteria_4883.jpeg"
]
```

```
run_on_multiple_images(image_files, model, model.features[-1])
```

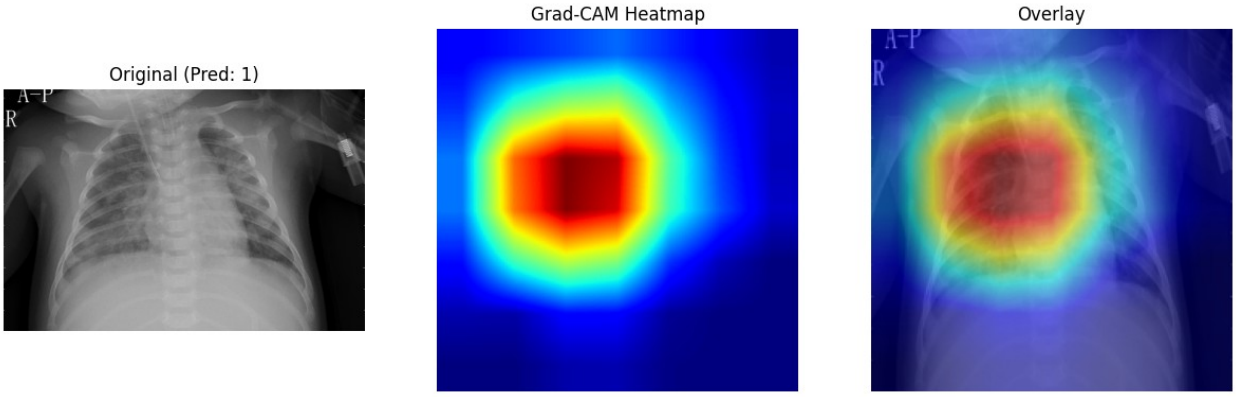
```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1954_bacteria_4886.jpeg

c:\Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\site-
packages\torch\nn\modules\module.py:1827: FutureWarning: Using a non-
full backward hook when the forward contains multiple autograd Nodes
is deprecated and will be removed in future versions. This hook will
be missing some grad_input. Please use register_full_backward_hook to
get the documented behavior.
  self._maybe_warn_non_full_backward_hook(args, result, grad_fn)
```



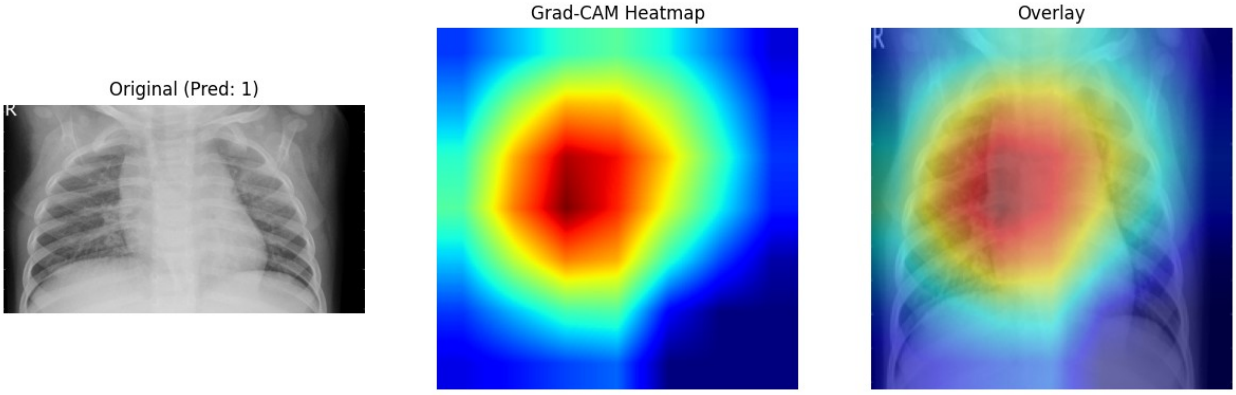

```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4874.jpeg
```



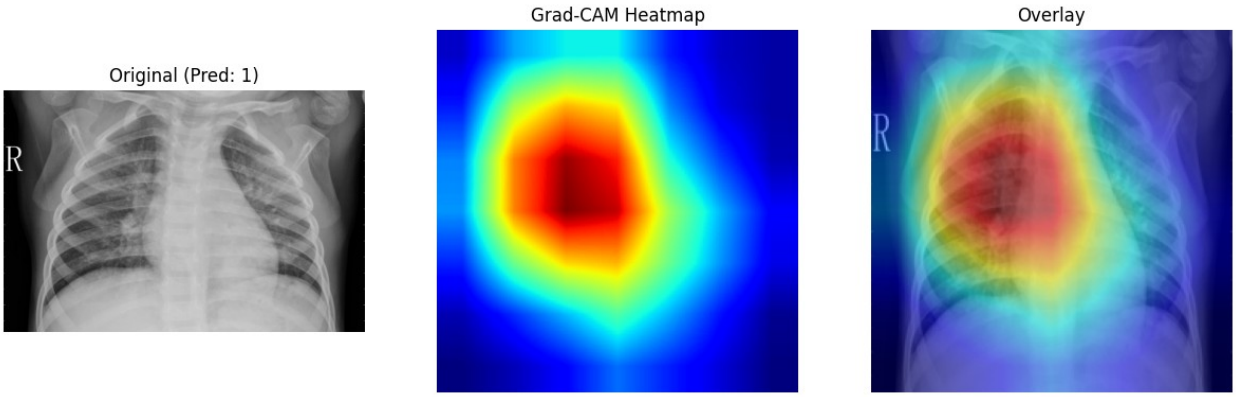

```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4875.jpeg
```



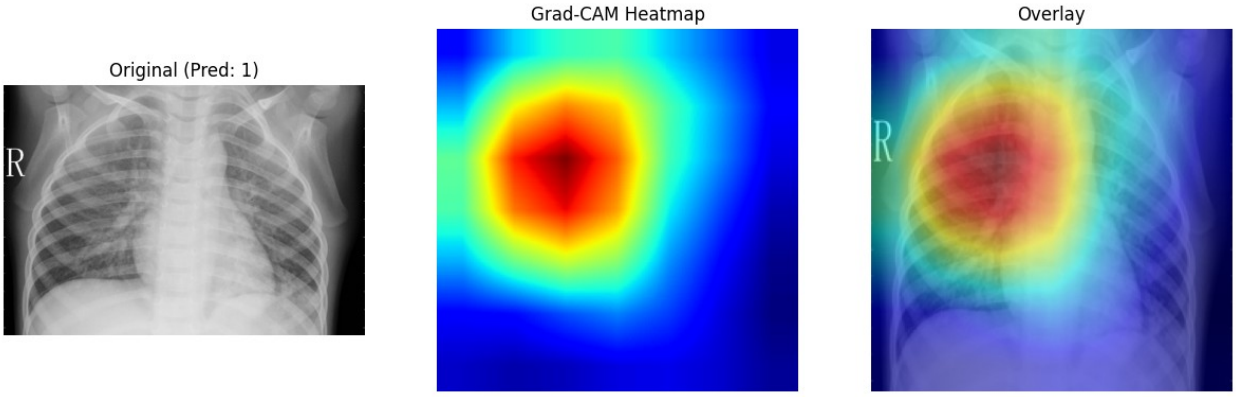

```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1947_bacteria_4876.jpeg
```



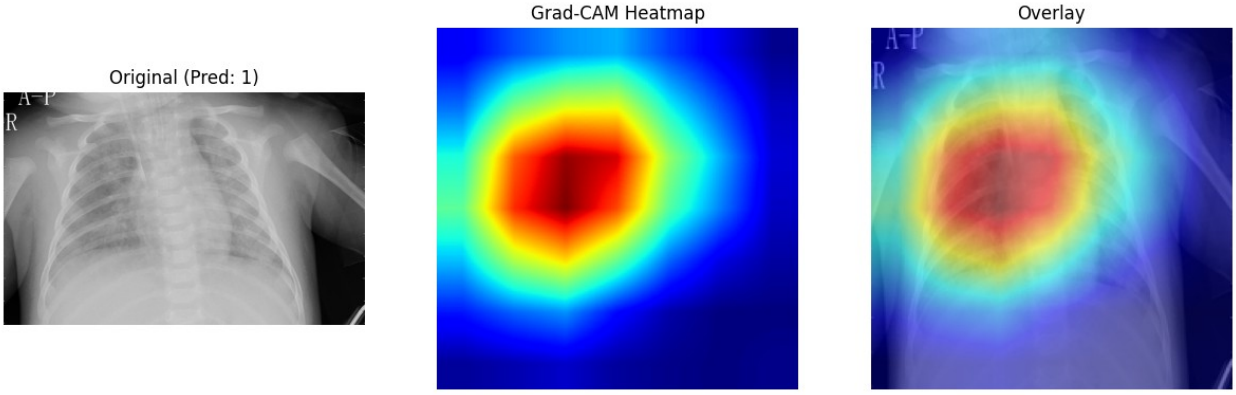

```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1949_bacteria_4880.jpeg
```



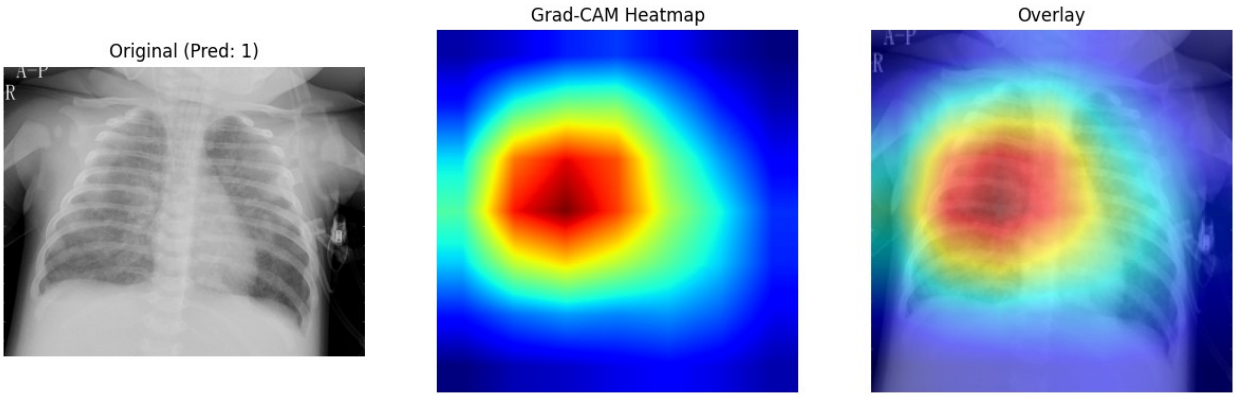

```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1950_bacteria_4881.jpeg
```




```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1951_bacteria_4882.jpeg
```




```
Processing: D:\datasets\chest_xray\val\PNEUMONIA\
person1952_bacteria_4883.jpeg
```



```
import numpy as np
from PIL import Image
import cv2

def load_attrib_image(path, target_size=None, keep_uint8=False):
    # load image (RGB or grayscale), convert to single-channel float
    im = Image.open(path).convert('RGB')
    arr = np.asarray(im)  # H,W,3
    # if image is a colored heatmap, collapse to luminance (approx)
    gray = (0.2989*arr[...,0] + 0.5870*arr[...,1] + 0.1140*arr[...,2])
    if target_size is not None:
        gray = cv2.resize(gray, (target_size[1], target_size[0]),
interpolation=cv2.INTER_LINEAR)
    if keep_uint8:
        return gray
    return gray.astype(np.float32)

def normalize_map(m):
    m = np.abs(m).astype(np.float32)
    if m.max() <= 1e-12:
        return m
    return (m - m.min()) / (m.max() - m.min() + 1e-12)

# Example usage:
p_shap = [
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4876.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4880.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4881.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4882.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4883.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4886.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4874.png",
    r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\final
shap\4875.png"
]
p_grad = [
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1954_bacteria_4886.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4874.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4875.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
```

```
person1947_bacteria_4876.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1949_bacteria_4880.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1950_bacteria_4881.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1951_bacteria_4882.jpeg",
    r"D:\datasets\chest_xray\val\PNEUMONIA\
person1952_bacteria_4883.jpeg"
]


# first load shap to get shape (or choose a desired size)



idx = 2


m1 = load_attrib_image(p_shap[idx], target_size=None)
m2 = load_attrib_image(p_grad[idx], target_size=m1.shape[:2])  #
ensure same HxW

map_shap = normalize_map(m1)
map_gradcam = normalize_map(m2)

# quick sanity prints
print("shap shape:", map_shap.shape, "min/max:", map_shap.min(),
map_shap.max())
print("gradcam shape:", map_gradcam.shape, "min/max:",
map_gradcam.min(), map_gradcam.max())

# save preprocessed arrays if you want
np.save("map_shap.npy", map_shap)
np.save("map_gradcam.npy", map_gradcam)

shap shape: (431, 434) min/max: 0.0 1.0
gradcam shape: (431, 434) min/max: 0.0 1.0

# ------------------------------
# Sparsity + Faithfulness (AOPC)
# ------------------------------
import numpy as np
import pandas as pd
from PIL import Image
import cv2
import os


orig_img_path = r"D:\datasets\chest_xray\val\PNEUMONIA\
person1946_bacteria_4875.jpeg"
```

```
true_class_index = 1
maps = {"SHAP": map_shap, "GradCAM": map_gradcam}

PERTURB_BASELINE = 'mean'

PERT_FRACS = [0.01, 0.03, 0.05, 0.1, 0.2]


OUT_CSV = "explainability_metrics_summary.csv"



def normalize_map(m):
    m = np.array(m, dtype=np.float32)

    m -= m.min()
    if m.max() > 0:
        m /= (m.max() + 1e-12)
    return m

def compute_sparsity_topmass(attrib_map, mass_thresh=0.90):

    nm = np.abs(attrib_map).ravel().astype(np.float32)
    if nm.sum() == 0:
        return 1.0
    order = np.argsort(-nm)
    cum = np.cumsum(nm[order])
    k = np.searchsorted(cum, mass_thresh) + 1
    return float(k / nm.size)


def load_image_rgb(path, target_size=None):
    im = Image.open(path).convert("RGB")
    arr = np.asarray(im).astype(np.float32) / 255.0
    if target_size is not None:
        arr = cv2.resize(arr, (target_size[1], target_size[0]),
interpolation=cv2.INTER_LINEAR)
    return arr


def apply_perturbation(img, indices_to_perturb, baseline='mean'):

    H, W = img.shape[:2]
    flat = img.reshape(-1, 3).copy()
    if baseline == 'mean':
        fill = img.mean(axis=(0,1))
        flat[indices_to_perturb, :] = fill.reshape(1, 3)
    elif baseline == 'black':
        flat[indices_to_perturb, :] = 0.0
    elif baseline == 'blur':
```

```
        # reconstruct mask and apply gaussian blur to that region
        mask = np.zeros((H, W), dtype=np.uint8).ravel()
        mask[indices_to_perturb] = 1
        mask = mask.reshape(H, W)
        blurred = cv2.GaussianBlur((img*255).astype(np.uint8),
(21,21), 0).astype(np.float32)/255.0
        flat_mask = mask.reshape(-1,1).astype(bool)
        flat[flat_mask[:,0], :] = blurred.reshape(-1,3)
[flat_mask[:,0], :]
    else:
        raise ValueError("Unknown baseline")
    return flat.reshape(H, W, 3)


def predict_fn_pytorch(batch_images_np, torch_model, device='cuda'):
    import torch
    from torchvision import transforms

    mean = [0.485,0.456,0.406]
    std = [0.229,0.224,0.225]
    xs = []
    for im in batch_images_np:
        # im is H,W,3 in [0,1]
        t = torch.tensor(im.transpose(2,0,1), dtype=torch.float32)  #
C,H,W
        for c, m, s in zip(t, mean, std):
            c.sub_(m).div_(s)
        xs.append(t)
    xb = torch.stack(xs).to(device)
    torch_model.to(device)
    torch_model.eval()
    with torch.no_grad():
        logits = torch_model(xb)
        probs = torch.softmax(logits, dim=1).cpu().numpy()
    return probs


def predict_fn_tf(batch_images_np, tf_model):
    import tensorflow as tf
    preds = tf_model(batch_images_np, training=False).numpy()
    if preds.ndim == 2:
        exps = np.exp(preds - np.max(preds, axis=1, keepdims=True))
        probs = exps / np.sum(exps, axis=1, keepdims=True)
        return probs
    else:
        return preds



def compute_aopc_for_map(img_rgb, true_class, attrib_map, predict_fn,
```

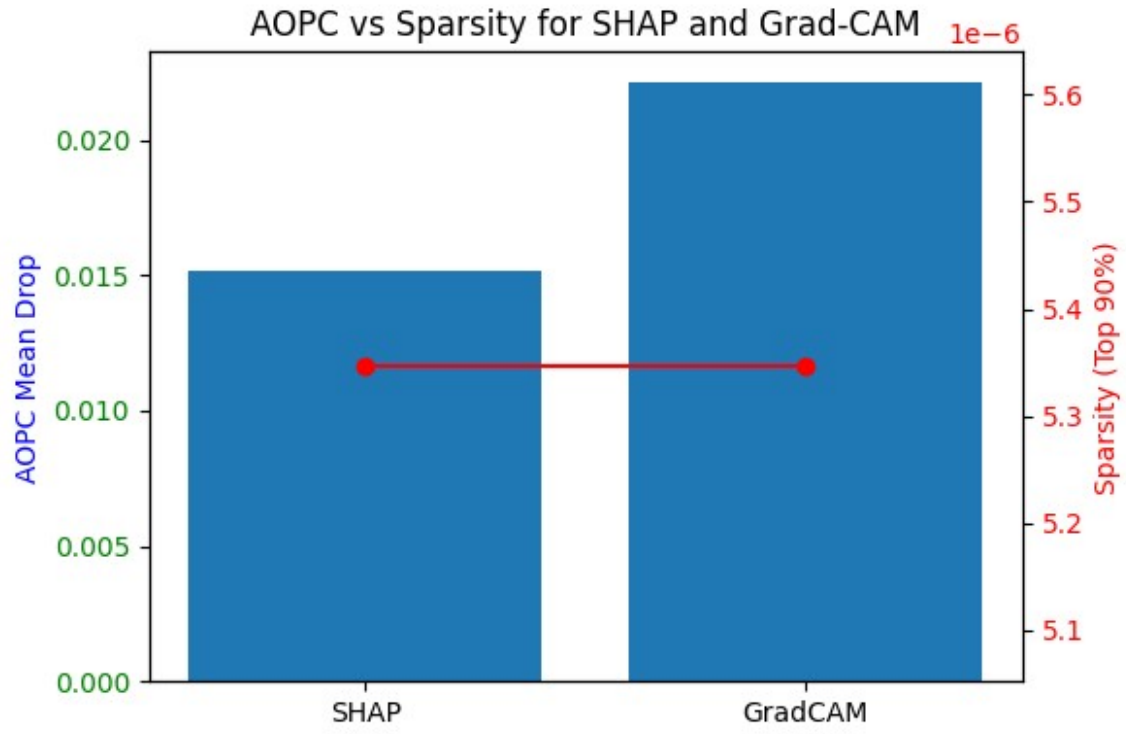
```
                          perturb_fracs=PERT_FRACS,
baseline=PERTURB_BASELINE):

    H, W = attrib_map.shape[:2]
    num_pixels = H * W

    orig_probs = predict_fn(np.expand_dims(img_rgb, 0))[0]  # (C,)
    orig_true_prob = float(orig_probs[true_class])

    flat_map = np.abs(attrib_map).ravel()
    order = np.argsort(-flat_map)
    prob_drops = []
    for frac in perturb_fracs:
        k = max(1, int(num_pixels * frac))
        idxs = order[:k]
        pert_img = apply_perturbation(img_rgb, idxs,
baseline=baseline)
        pert_prob = float(predict_fn(np.expand_dims(pert_img, 0))[0]
[true_class])
        prob_drops.append(orig_true_prob - pert_prob)
    return float(np.mean(prob_drops)), prob_drops, orig_true_prob


def evaluate_one_image(orig_img_path, true_class, maps_dict,
predict_fn_callable):
    img = load_image_rgb(orig_img_path,
target_size=list(maps_dict.values())[0].shape[:2])
    results = []
    for name, attrib in maps_dict.items():
        attrib_n = normalize_map(attrib)
        spars = compute_sparsity_topmass(attrib_n, mass_thresh=0.90)
        aopc_value, drops, orig_prob = compute_aopc_for_map(img,
true_class, attrib_n, predict_fn_callable)
        results.append({
            "method": name,
            "sparsity_topmass_90": spars,
            "aopc_mean_drop": aopc_value,
            "orig_true_prob": orig_prob,
            "prob_drops_by_frac": drops
        })
    return results


def predict_fn_dummy(batch_images_np):

    batch = np.asarray(batch_images_np, dtype=np.float32)
    means = batch.mean(axis=(1, 2, 3))
    probs1 = np.clip(means, 0.0, 1.0)
    probs0 = 1.0 - probs1
    probs = np.stack([probs0, probs1], axis=1)
```

```
    return probs

predict_fn_callable = predict_fn_dummy


res = evaluate_one_image(orig_img_path, true_class_index, maps,
predict_fn_callable)

# Save/print
df_rows = []
for r in res:
    df_rows.append({
        "method": r["method"],
        "sparsity_topmass_90": r["sparsity_topmass_90"],
        "aopc_mean_drop": r["aopc_mean_drop"],
        "orig_true_prob": r["orig_true_prob"],
        "prob_drops_by_frac": ";".join([f"{p:.4f}" for p in
r["prob_drops_by_frac"]])
    })
df = pd.DataFrame(df_rows)
df.to_csv(OUT_CSV, index=False)
print(df.to_string(index=False))
print(f"\nSaved summary CSV -> {os.path.abspath(OUT_CSV)}")
```

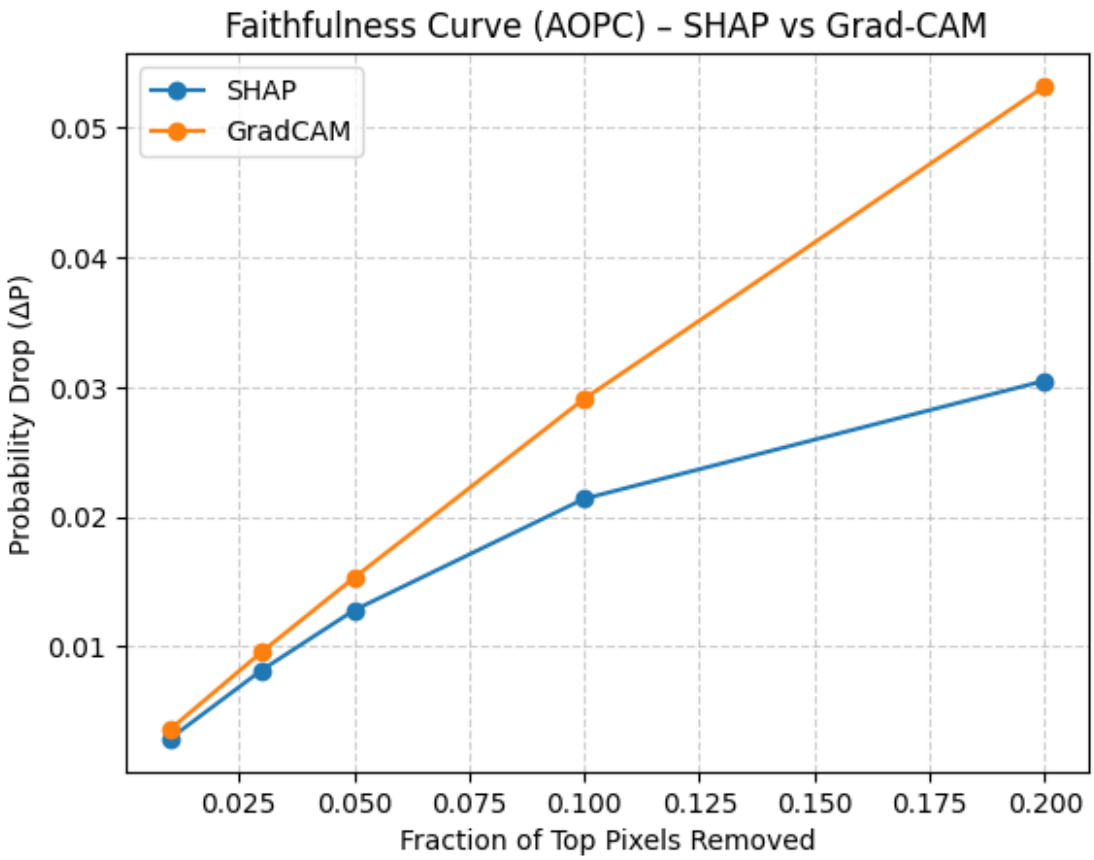
```
 method  sparsity_topmass_90  aopc_mean_drop  orig_true_prob
prob_drops_by_frac
   SHAP             0.000005        0.015150        0.386419
0.0029;0.0082;0.0128;0.0214;0.0305
GradCAM             0.000005        0.022143        0.386419
0.0036;0.0096;0.0153;0.0291;0.0532

Saved summary CSV -> c:\Users\Ekaansh\OneDrive\Desktop\AB\research\
SHAP\explainability_metrics_summary.csv
```

```
import pandas as pd
import matplotlib.pyplot as plt


df = pd.read_csv("explainability_metrics_summary.csv")

# Bar plot comparison
fig, ax1 = plt.subplots(figsize=(6,4))
ax1.bar(df['method'], df['aopc_mean_drop'], label='AOPC Mean Drop')
ax1.set_ylabel('AOPC Mean Drop', color='blue')
ax1.tick_params(axis='y', labelcolor='green')

ax2 = ax1.twinx()
ax2.plot(df['method'], df['sparsity_topmass_90'], color='red',
marker='o', label='Sparsity (Top 90%)')
ax2.set_ylabel('Sparsity (Top 90%)', color='red')
ax2.tick_params(axis='y', labelcolor='red')
```

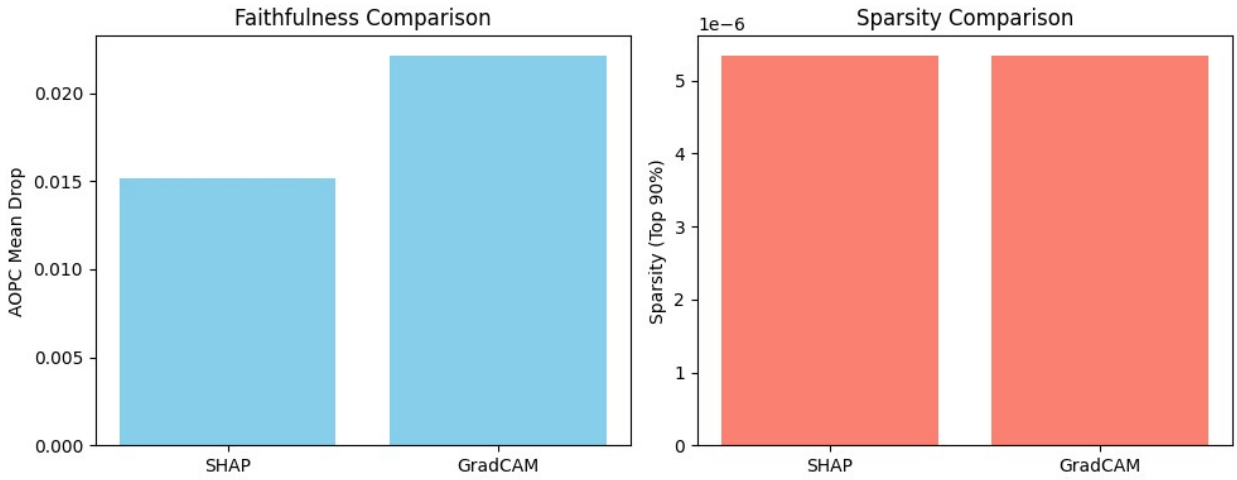
```
plt.title('AOPC vs Sparsity for SHAP and Grad-CAM')
plt.tight_layout()
plt.show()
```



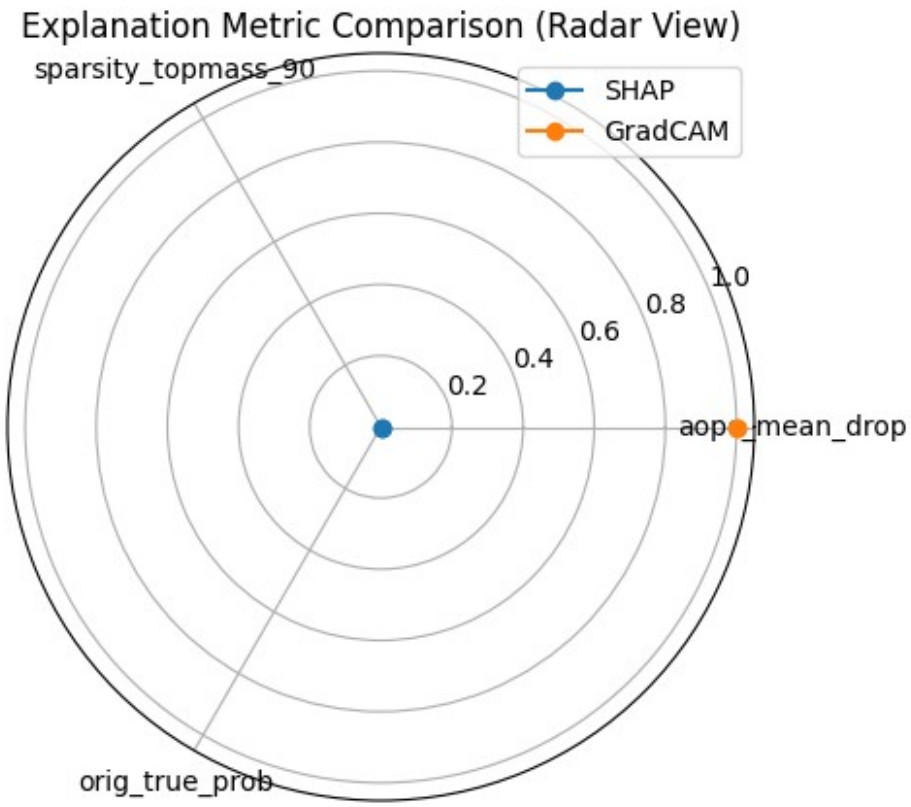

```
import numpy as np
import matplotlib.pyplot as plt

# Define perturbation fractions
pert_fracs = [0.01, 0.03, 0.05, 0.1, 0.2]

for i, row in df.iterrows():
    drops = list(map(float, row['prob_drops_by_frac'].split(';')))
    plt.plot(pert_fracs, drops, marker='o', label=row['method'])

plt.xlabel('Fraction of Top Pixels Removed')
plt.ylabel('Probability Drop (ΔP)')
plt.title('Faithfulness Curve (AOPC) — SHAP vs Grad-CAM')
plt.legend()
plt.grid(True, linestyle='--', alpha=0.6)
plt.show()
```



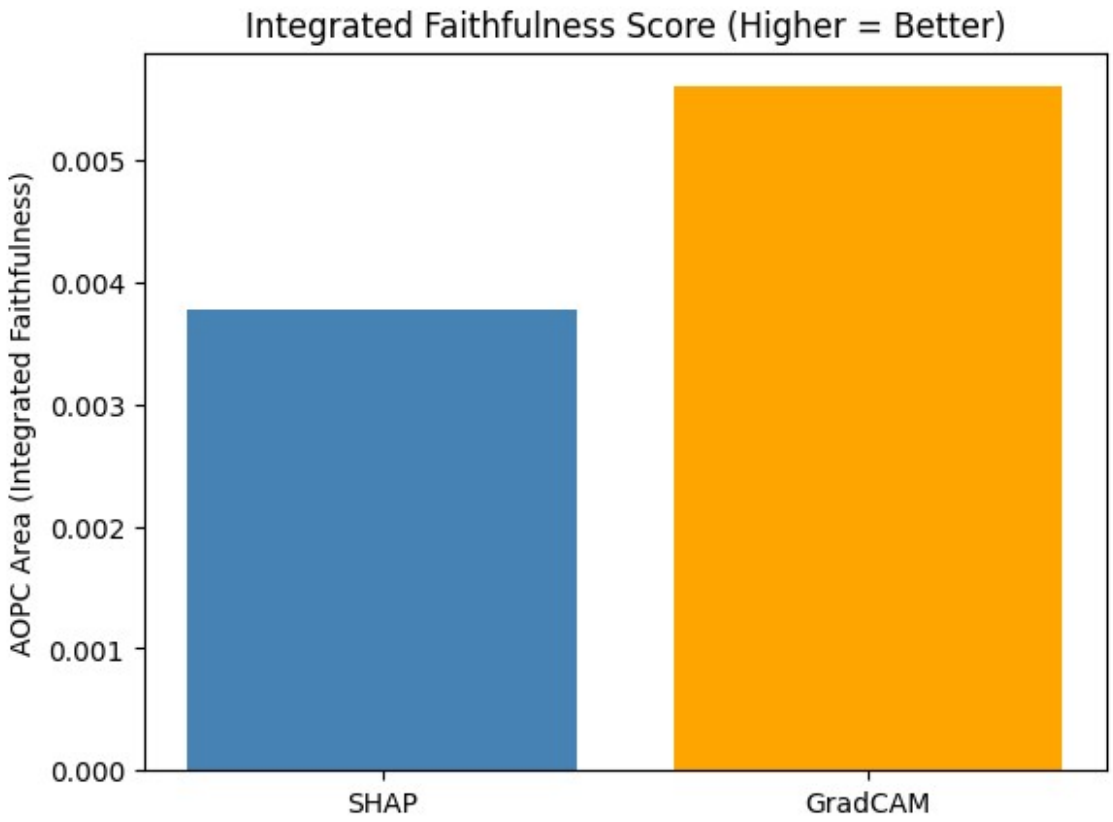

```
fig, axes = plt.subplots(1, 2, figsize=(10,4))

# Left: AOPC & Sparsity
axes[0].bar(df['method'], df['aopc_mean_drop'], color='skyblue',
label='AOPC')
axes[0].set_ylabel('AOPC Mean Drop')
axes[0].set_title('Faithfulness Comparison')

axes[1].bar(df['method'], df['sparsity_topmass_90'], color='salmon',
label='Sparsity')
axes[1].set_ylabel('Sparsity (Top 90%)')
axes[1].set_title('Sparsity Comparison')

plt.tight_layout()
plt.show()
```



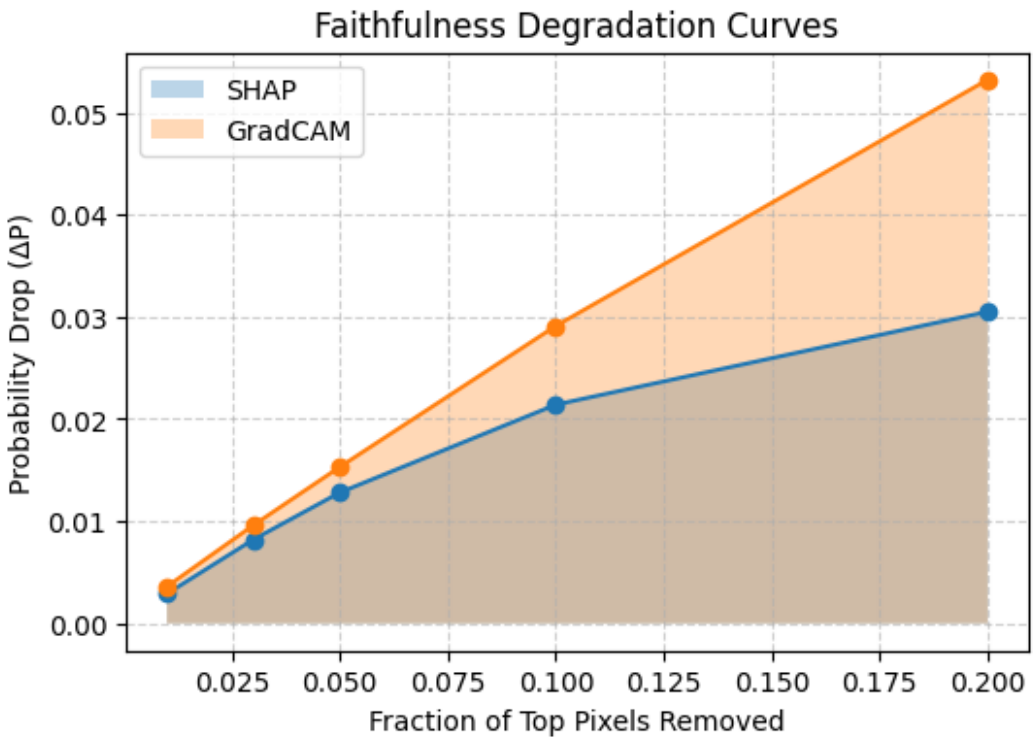

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

df = pd.read_csv("explainability_metrics_summary.csv")

metrics = ['aopc_mean_drop', 'sparsity_topmass_90', 'orig_true_prob']
methods = df['method']
values = [df[m] for m in metrics]

values = [(v - v.min()) / (v.max() - v.min()) for v in values]
values = np.array(values).T

angles = np.linspace(0, 2*np.pi, len(metrics),
endpoint=False).tolist()
angles += angles[:1]

fig, ax = plt.subplots(figsize=(5,5), subplot_kw=dict(polar=True))

for i, method in enumerate(methods):
    vals = values[i].tolist()
    vals += vals[:1]
    ax.plot(angles, vals, marker='o', label=method)
    ax.fill(angles, vals, alpha=0.2)

ax.set_xticks(angles[:-1])
ax.set_xticklabels(metrics)
plt.title("Explanation Metric Comparison (Radar View)")
plt.legend()
plt.show()
```



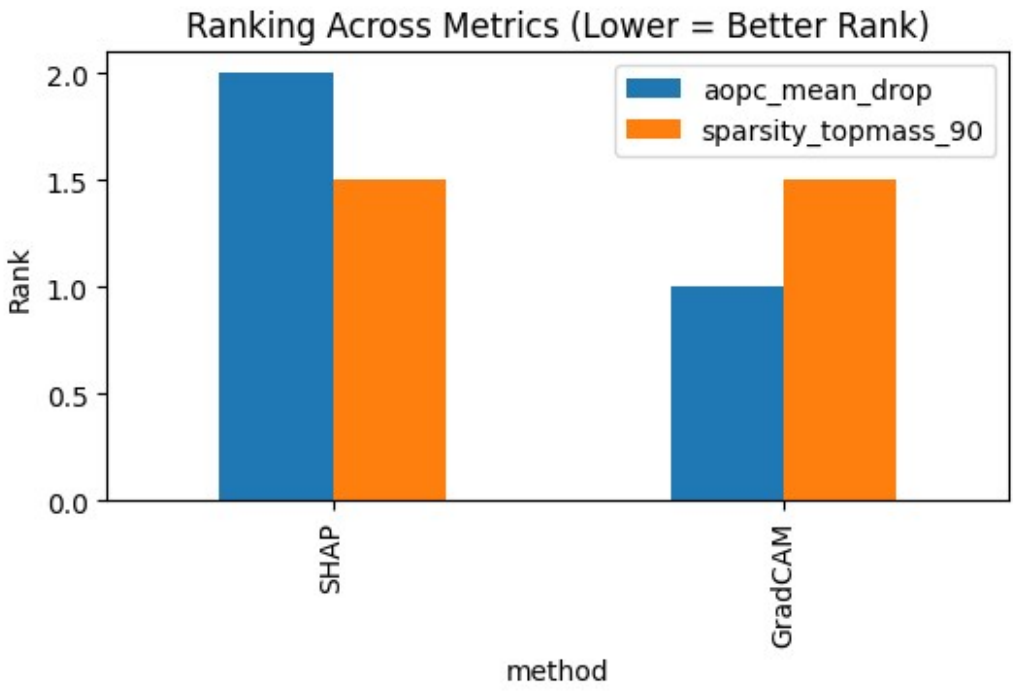

```
import numpy as np
import matplotlib.pyplot as plt

pert_fracs = np.array([0.01, 0.03, 0.05, 0.1, 0.2])
areas = []

for _, row in df.iterrows():
    drops = np.array(list(map(float,
row['prob_drops_by_frac'].split(';'))))
    area = np.trapz(drops, pert_fracs)  # numerical integration
    areas.append(area)

plt.bar(df['method'], areas, color=['steelblue', 'orange'])
plt.ylabel('AOPC Area (Integrated Faithfulness)')
plt.title('Integrated Faithfulness Score (Higher = Better)')
plt.show()
```




```
plt.figure(figsize=(6,4))

for _, row in df.iterrows():
    drops = list(map(float, row['prob_drops_by_frac'].split(';')))
    plt.fill_between(pert_fracs, drops, alpha=0.3,
label=row['method'])
    plt.plot(pert_fracs, drops, marker='o')

plt.xlabel('Fraction of Top Pixels Removed')
plt.ylabel('Probability Drop (ΔP)')
plt.title('Faithfulness Degradation Curves')
plt.legend()
plt.grid(True, linestyle='--', alpha=0.6)
plt.show()
```




```
metrics = ['aopc_mean_drop', 'sparsity_topmass_90']
fig, ax = plt.subplots(figsize=(6,3))
rank_data = df.set_index('method')[metrics].rank(ascending=False)
rank_data.plot(kind='bar', ax=ax)
plt.title('Ranking Across Metrics (Lower = Better Rank)')
plt.ylabel('Rank')
plt.show()
```


Ranking Across Metrics (Lower = Better Rank)
aopc_mean_drop
sparsity_topmass_90
Rank
2.0
1.5
1.0
0.5
0.0
SHAP
GradCAM
method

```python
import os
import torch
import torch.nn as nn
import numpy as np
import shap
from torchvision import models
from PIL import Image
import matplotlib.pyplot as plt


device = torch.device("cpu")
print(f"Using device: {device}")


model_path = r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\
best_pneumonia_densenet121.pt"
dataset_dir = r"D:\datasets\chest_xray\val\PNEUMONIA"  # pneumonia
images
save_dir = r"C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\
finding\shap_pred_1"

os.makedirs(save_dir, exist_ok=True)
class_names = ['NORMAL', 'PNEUMONIA']


model = models.densenet121(weights=None)
num_features = model.classifier.in_features
model.classifier = nn.Sequential(
    nn.Linear(num_features, 256),
    nn.ReLU(),
    nn.Dropout(0.3),
    nn.Linear(256, len(class_names))
)
model.load_state_dict(torch.load(model_path, map_location=device))
model.eval()


mean = np.array([0.485, 0.456, 0.406])
std = np.array([0.229, 0.224, 0.225])

def preprocess_input(images):
    images = images / 255.0
    images = (images - mean) / std
    images = images.transpose(0, 3, 1, 2)
    return torch.tensor(images).float()


def f(x):
    x_tensor = preprocess_input(x)
```

```
    with torch.no_grad():
        out = model(x_tensor)
        probs = torch.softmax(out, dim=1)
    return probs.cpu().numpy()


all_images = sorted([os.path.join(dataset_dir, f) for f in
os.listdir(dataset_dir) if f.lower().endswith(('.jpg', '.jpeg',
'.png'))])
selected_images = all_images[:50]

X = np.stack([np.array(Image.open(p).convert("RGB").resize((224,
224))) for p in selected_images])


masker = shap.maskers.Image("inpaint_telea", X[0].shape)
explainer = shap.Explainer(f, masker, output_names=class_names)

for idx, img_array in enumerate(X):
    shap_values = explainer(img_array[np.newaxis, ...],
max_evals=5000, batch_size=20,
outputs=shap.Explanation.argsort.flip[:2])

    plt.figure(figsize=(5, 5))
    shap.image_plot(shap_values, show=False)

    filename = os.path.basename(selected_images[idx])
    save_path = os.path.join(save_dir, f"shap_{filename}.png")
    plt.savefig(save_path, bbox_inches='tight', pad_inches=0)
    plt.close()
    print(f"Saved: {save_path}")
```

```
Using device: cpu

C:\Users\Ekaansh\AppData\Local\Temp\ipykernel_24304\32912513.py:32:
FutureWarning: You are using `torch.load` with `weights_only=False`
(the current default value), which uses the default pickle module
implicitly. It is possible to construct malicious pickle data which
will execute arbitrary code during unpickling (See
https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-
models for more details). In a future release, the default value for
`weights_only` will be flipped to `True`. This limits the functions
that could be executed during unpickling. Arbitrary objects will no
longer be allowed to be loaded via this mode unless they are
explicitly allowlisted by the user via
`torch.serialization.add_safe_globals`. We recommend you start setting
`weights_only=True` for any use case where you don't have full control
of the loaded file. Please open an issue on GitHub for any issues
related to this experimental feature.
  model.load_state_dict(torch.load(model_path, map_location=device))
```

{"model_id":"a2c66d2c3ab343849203f7f373e61cd8","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [07:03, 423.67s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1946_bacteria_4874.jpeg.png
```

{"model_id":"e874237e52e44f25a16cdd5c62b9d851","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [06:24, 384.69s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1946_bacteria_4875.jpeg.png
```

{"model_id":"e174e46cdf334c0a8f3bb4f77c41a079","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [06:08, 368.94s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1947_bacteria_4876.jpeg.png
```

{"model_id":"7c7e4ee201e14916bff939ba7c6a06c5","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [05:54, 354.41s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1949_bacteria_4880.jpeg.png
```

{"model_id":"8f30020bcc734b87b3a656003f341039","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [06:04, 364.12s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1950_bacteria_4881.jpeg.png
```

{"model_id":"b5b6f434f3794862a2b8512fb4137e94","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [06:14, 374.05s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\shap_pred_1\shap_person1951_bacteria_4882.jpeg.png
```

{"model_id":"d7b615d73d294a1bb4bb4dd96f196f24","version_major":2,"version_minor":0}

```
PartitionExplainer explainer: 2it [07:12, 432.91s/it]
```

```
Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\
shap_pred_1\shap_person1952_bacteria_4883.jpeg.png

{"model_id":"f8e996ae30754942a29c1ac897e9e593","version_major":2,"vers
ion_minor":0}

PartitionExplainer explainer: 2it [08:02, 482.68s/it]

Saved: C:\Users\Ekaansh\OneDrive\Desktop\AB\research\SHAP\finding\
shap_pred_1\shap_person1954_bacteria_4886.jpeg.png

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>

<Figure size 500x500 with 0 Axes>
```

```
%pip install torch torchvision --upgrade
import torch
import torch.nn as nn
import torch.optim as optim
from torch.optim.lr_scheduler import OneCycleLR
from torchvision import models, transforms, datasets
from torch.utils.data import DataLoader
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.utils.class_weight import compute_class_weight
import numpy as np
```

```
Requirement already satisfied: torch in c:\users\ekaansh\appdata\
local\programs\python\python311\lib\site-packages (2.5.1+cu121)
Collecting torch
  Downloading torch-2.9.0-cp311-cp311-win_amd64.whl.metadata (30 kB)
Requirement already satisfied: torchvision in c:\users\ekaansh\
appdata\local\programs\python\python311\lib\site-packages
(0.20.1+cu121)
Collecting torchvision
  Downloading torchvision-0.24.0-cp311-cp311-win_amd64.whl.metadata
(5.9 kB)
Requirement already satisfied: filelock in c:\users\ekaansh\appdata\
local\programs\python\python311\lib\site-packages (from torch)
(3.13.1)
Requirement already satisfied: typing-extensions>=4.10.0 in c:\users\
ekaansh\appdata\local\programs\python\python311\lib\site-packages
(from torch) (4.15.0)
Collecting sympy>=1.13.3 (from torch)
  Using cached sympy-1.14.0-py3-none-any.whl.metadata (12 kB)
Requirement already satisfied: networkx>=2.5.1 in c:\users\ekaansh\
appdata\local\programs\python\python311\lib\site-packages (from torch)
(3.3)
Requirement already satisfied: jinja2 in c:\users\ekaansh\appdata\
local\programs\python\python311\lib\site-packages (from torch) (3.1.4)
Requirement already satisfied: fsspec>=0.8.5 in c:\users\ekaansh\
appdata\local\programs\python\python311\lib\site-packages (from torch)
(2024.6.1)
Requirement already satisfied: numpy in c:\users\ekaansh\appdata\
local\programs\python\python311\lib\site-packages (from torchvision)
(1.26.4)
Requirement already satisfied: pillow!=8.3.*,>=5.3.0 in c:\users\
ekaansh\appdata\local\programs\python\python311\lib\site-packages
(from torchvision) (11.0.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in c:\users\ekaansh\
appdata\local\programs\python\python311\lib\site-packages (from
sympy>=1.13.3->torch) (1.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\ekaansh\
appdata\local\programs\python\python311\lib\site-packages (from
jinja2->torch) (2.1.5)
Downloading torch-2.9.0-cp311-cp311-win_amd64.whl (109.3 MB)
```

```
   ---------------------------------------- 0.0/109.3 MB ? eta -:--:--
   ---------------------------------------- 1.0/109.3 MB 7.1 MB/s eta
0:00:16
   - -------------------------------------- 4.7/109.3 MB 13.6 MB/s eta
0:00:08
   -- ------------------------------------- 6.6/109.3 MB 11.8 MB/s eta
0:00:09
   -- ------------------------------------- 7.3/109.3 MB 11.3 MB/s eta
0:00:10
   ---- ----------------------------------- 11.5/109.3 MB 11.8 MB/s
eta 0:00:09
   ----- ---------------------------------- 15.7/109.3 MB 12.1 MB/s
eta 0:00:08
   ------ --------------------------------- 18.6/109.3 MB 12.2 MB/s
eta 0:00:08
   ------- -------------------------------- 20.7/109.3 MB 12.0 MB/s
eta 0:00:08
   -------- ------------------------------- 23.3/109.3 MB 12.1 MB/s
eta 0:00:08
   -------- ------------------------------- 24.4/109.3 MB 11.7 MB/s
eta 0:00:08
   -------- ------------------------------- 24.4/109.3 MB 11.7 MB/s
eta 0:00:08
   -------- ------------------------------- 24.4/109.3 MB 11.7 MB/s
eta 0:00:08
   --------- ------------------------------ 25.7/109.3 MB 9.2 MB/s eta
0:00:10
   --------- ------------------------------ 27.0/109.3 MB 9.1 MB/s eta
0:00:10
   ---------- ----------------------------- 27.5/109.3 MB 8.6 MB/s eta
0:00:10
   ---------- ----------------------------- 27.5/109.3 MB 8.6 MB/s eta
0:00:10
   ---------- ----------------------------- 27.5/109.3 MB 8.6 MB/s eta
0:00:10
   ---------- ----------------------------- 27.5/109.3 MB 8.6 MB/s eta
0:00:10
   ---------- ----------------------------- 29.9/109.3 MB 7.4 MB/s eta
0:00:11
   ----------- ---------------------------- 30.7/109.3 MB 7.2 MB/s eta
0:00:11
   ----------- ---------------------------- 32.2/109.3 MB 7.2 MB/s eta
0:00:11
   ------------ --------------------------- 34.1/109.3 MB 7.3 MB/s eta
0:00:11
   ------------- -------------------------- 35.9/109.3 MB 7.3 MB/s eta
0:00:11
   ------------- -------------------------- 36.7/109.3 MB 7.2 MB/s eta
0:00:11
```

```
   ------------ -------------------------- 37.5/109.3 MB 7.0 MB/s eta
0:00:11
   ------------ -------------------------- 37.7/109.3 MB 7.0 MB/s eta
0:00:11
   ------------- -------------------------- 38.3/109.3 MB 6.6 MB/s eta
0:00:11
   ------------- -------------------------- 38.5/109.3 MB 6.5 MB/s eta
0:00:11
   ------------- -------------------------- 39.1/109.3 MB 6.3 MB/s eta
0:00:12
   ------------- -------------------------- 40.4/109.3 MB 6.3 MB/s eta
0:00:11
   -------------- ------------------------- 41.9/109.3 MB 6.4 MB/s eta
0:00:11
   ---------------- ----------------------- 44.8/109.3 MB 6.6 MB/s eta
0:00:10
   ---------------- ----------------------- 46.1/109.3 MB 6.6 MB/s eta
0:00:10
   ----------------- ---------------------- 47.4/109.3 MB 6.6 MB/s eta
0:00:10
   ----------------- ---------------------- 48.5/109.3 MB 6.5 MB/s eta
0:00:10
   ------------------ --------------------- 50.6/109.3 MB 6.6 MB/s eta
0:00:09
   ------------------- -------------------- 52.4/109.3 MB 6.7 MB/s eta
0:00:09
   ------------------- -------------------- 54.3/109.3 MB 6.7 MB/s eta
0:00:09
   -------------------- ------------------- 56.1/109.3 MB 6.8 MB/s eta
0:00:08
   --------------------- ------------------ 58.2/109.3 MB 6.8 MB/s eta
0:00:08
   --------------------- ------------------ 60.0/109.3 MB 6.9 MB/s eta
0:00:08
   ---------------------- ----------------- 61.9/109.3 MB 6.9 MB/s eta
0:00:07
   ----------------------- ---------------- 64.0/109.3 MB 7.0 MB/s eta
0:00:07
   ----------------------- ---------------- 65.5/109.3 MB 7.0 MB/s eta
0:00:07
   ------------------------ --------------- 66.6/109.3 MB 7.0 MB/s eta
0:00:07
   ------------------------ --------------- 69.2/109.3 MB 7.1 MB/s eta
0:00:06
   ------------------------- -------------- 71.8/109.3 MB 7.2 MB/s eta
0:00:06
   ------------------------- -------------- 73.4/109.3 MB 7.2 MB/s eta
0:00:05
   -------------------------- ------------- 75.5/109.3 MB 7.3 MB/s eta
```

```
0:00:05
   ---------------------------- ----------- 78.1/109.3 MB 7.4 MB/s eta
0:00:05
   ---------------------------- ----------- 78.4/109.3 MB 7.4 MB/s eta
0:00:05
   ---------------------------- ----------- 78.4/109.3 MB 7.4 MB/s eta
0:00:05
   ---------------------------- ----------- 78.4/109.3 MB 7.4 MB/s eta
0:00:05
   ----------------------------- ---------- 80.7/109.3 MB 7.1 MB/s eta
0:00:05
   ----------------------------- ---------- 81.8/109.3 MB 7.0 MB/s eta
0:00:04
   ------------------------------ --------- 83.4/109.3 MB 7.1 MB/s eta
0:00:04
   ------------------------------ --------- 84.7/109.3 MB 7.0 MB/s eta
0:00:04
   ------------------------------- -------- 87.0/109.3 MB 7.1 MB/s eta
0:00:04
   -------------------------------- ------- 89.7/109.3 MB 7.2 MB/s eta
0:00:03
   -------------------------------- ------ 91.2/109.3 MB 7.2 MB/s eta
0:00:03
   -------------------------------- ------ 91.8/109.3 MB 7.1 MB/s eta
0:00:03
   --------------------------------- ----- 93.1/109.3 MB 7.1 MB/s eta
0:00:03
   --------------------------------- ----- 94.6/109.3 MB 7.1 MB/s eta
0:00:03
   ---------------------------------- ---- 96.2/109.3 MB 7.1 MB/s eta
0:00:02
   ---------------------------------- ---- 98.0/109.3 MB 7.2 MB/s eta
0:00:02
   ----------------------------------- --- 100.1/109.3 MB 7.2 MB/s
eta 0:00:02
   ----------------------------------- -- 102.2/109.3 MB 7.2 MB/s
eta 0:00:01
   ----------------------------------- -- 103.8/109.3 MB 7.2 MB/s
eta 0:00:01
   ------------------------------------ - 104.6/109.3 MB 7.2 MB/s
eta 0:00:01
   ------------------------------------  107.5/109.3 MB 7.3 MB/s
eta 0:00:01
   ------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
```

```
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------  109.1/109.3 MB 7.3 MB/s
eta 0:00:01
   ---------------------------------------- 109.3/109.3 MB 5.4 MB/s
0:00:20
Downloading torchvision-0.24.0-cp311-cp311-win_amd64.whl (4.0 MB)
```

```
   ---------------------------------------- 0.0/4.0 MB ? eta -:--:--
   ---------- ----------------------------- 1.0/4.0 MB 6.3 MB/s eta
0:00:01
   ----------------------- ---------------- 2.4/4.0 MB 6.1 MB/s eta
0:00:01
   --------------------------------- ------ 3.4/4.0 MB 6.1 MB/s eta
0:00:01
   ------------------------------------ --- 3.7/4.0 MB 5.5 MB/s eta
0:00:01
   ---------------------------------------  3.9/4.0 MB 4.4 MB/s eta
0:00:01
   ---------------------------------------- 4.0/4.0 MB 3.5 MB/s
0:00:01
Using cached sympy-1.14.0-py3-none-any.whl (6.3 MB)
Installing collected packages: sympy, torch, torchvision

  Attempting uninstall: sympy

    Found existing installation: sympy 1.13.1

   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
    Uninstalling sympy-1.13.1:
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
      Successfully uninstalled sympy-1.13.1
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
    ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
```

```
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
   ---------------------------------------- 0/3 [sympy]
  Attempting uninstall: torch
   ---------------------------------------- 0/3 [sympy]
    Found existing installation: torch 2.5.1+cu121
   ---------------------------------------- 0/3 [sympy]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
    Uninstalling torch-2.5.1+cu121:
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
      Successfully uninstalled torch-2.5.1+cu121
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
```

```
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
```

```
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
    ------------- --------------------------- 1/3 [torch]
```

```
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
    ------------- -------------------------- 1/3 [torch]
```

```
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
   ------------- --------------------------- 1/3 [torch]
```

```
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
```

```
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
```

```
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
   ------------- -------------------------- 1/3 [torch]
  Attempting uninstall: torchvision
   ------------- -------------------------- 1/3 [torch]
    Found existing installation: torchvision 0.20.1+cu121
   ------------- -------------------------- 1/3 [torch]
   -------------------------- ------------- 2/3 [torchvision]
    Uninstalling torchvision-0.20.1+cu121:
   -------------------------- ------------- 2/3 [torchvision]
      Successfully uninstalled torchvision-0.20.1+cu121
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
   -------------------------- ------------- 2/3 [torchvision]
```

```
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ------------------------- ------------ 2/3 [torchvision]
   ---------------------------------------- 3/3 [torchvision]

Successfully installed sympy-1.14.0 torch-2.9.0 torchvision-0.24.0
Note: you may need to restart the kernel to use updated packages.

WARNING: Ignoring invalid distribution ~ensorflow (c:\Users\Ekaansh\
AppData\Local\Programs\Python\Python311\Lib\site-packages)
WARNING: Ignoring invalid distribution ~hap (c:\Users\Ekaansh\AppData\
Local\Programs\Python\Python311\Lib\site-packages)
WARNING: Ignoring invalid distribution ~ensorflow (c:\Users\Ekaansh\
AppData\Local\Programs\Python\Python311\Lib\site-packages)
WARNING: Ignoring invalid distribution ~hap (c:\Users\Ekaansh\AppData\
Local\Programs\Python\Python311\Lib\site-packages)
  WARNING: Failed to remove contents in a temporary directory 'C:\
Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\site-
packages\~orch'.
  You can safely remove it manually.
  WARNING: Failed to remove contents in a temporary directory 'C:\
Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\site-
packages\~orchvision'.
  You can safely remove it manually.
WARNING: Ignoring invalid distribution ~ensorflow (c:\Users\Ekaansh\
AppData\Local\Programs\Python\Python311\Lib\site-packages)
WARNING: Ignoring invalid distribution ~hap (c:\Users\Ekaansh\AppData\
Local\Programs\Python\Python311\Lib\site-packages)
ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
torchaudio 2.5.1+cu121 requires torch==2.5.1+cu121, but you have torch
2.9.0 which is incompatible.

[notice] A new release of pip is available: 25.2 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip

----------------------------------------------------------------------
-----
```

```
ImportError                               Traceback (most recent call
last)
Cell In[4], line 2
      1 get_ipython().run_line_magic('pip', 'install torch torchvision
--upgrade')
----> 2 import torch
      3 import torch.nn as nn
      4 import torch.optim as optim

File c:\Users\Ekaansh\AppData\Local\Programs\Python\Python311\Lib\
site-packages\torch\__init__.py:50
     42     return False
     45 from torch._utils import (
     46     _functionalize_sync as _sync,
     47     _import_dotted_name,
     48     classproperty,
     49 )
---> 50 from torch._utils_internal import (
     51     get_file_path,
     52     prepare_multiprocessing_environment,
     53     profiler_allow_cudagraph_cupti_lazy_reinit_cuda12,
     54     USE_GLOBAL_DEPS,
     55     USE_RTLD_GLOBAL_WITH_LIBTORCH,
     56 )
     57 from torch.torch_version import __version__ as __version__
     60 if TYPE_CHECKING:

ImportError: cannot import name
'profiler_allow_cudagraph_cupti_lazy_reinit_cuda12' from
'torch._utils_internal' (c:\Users\Ekaansh\AppData\Local\Programs\
Python\Python311\Lib\site-packages\torch\_utils_internal.py)
```

```
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print("Device:", device)
```

```
Device: cuda
```

```
train_transform = transforms.Compose([
    transforms.RandomResizedCrop(224, scale=(0.8, 1.0)),
    transforms.RandomHorizontalFlip(),
    transforms.RandomRotation(15),
    transforms.ColorJitter(brightness=0.2, contrast=0.2),
    transforms.ToTensor(),
    transforms.Normalize([0.485], [0.229])
])

val_test_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
```

```
    transforms.Normalize([0.485], [0.229])
])

train_data = datasets.ImageFolder(r"D:\datasets\chest_xray\chest_xray\
train", transform=train_transform)
val_data   = datasets.ImageFolder(r"D:\datasets\chest_xray\chest_xray\
val", transform=val_test_transform)
test_data  = datasets.ImageFolder(r"D:\datasets\chest_xray\chest_xray\
test", transform=val_test_transform)

train_loader = DataLoader(train_data, batch_size=32, shuffle=True,
num_workers=2)
val_loader   = DataLoader(val_data, batch_size=32, shuffle=False,
num_workers=2)
test_loader  = DataLoader(test_data, batch_size=32, shuffle=False,
num_workers=2)

y_train = [label for _, label in train_data.samples]
class_weights = compute_class_weight(class_weight='balanced',
                                     classes=np.unique(y_train),
                                     y=y_train)
class_weights = torch.tensor(class_weights,
dtype=torch.float).to(device)
print("Class Weights:", class_weights)

Class Weights: tensor([1.9448, 0.6730], device='cuda:0')

model = models.densenet121(weights=models.DenseNet121_Weights.DEFAULT)


num_ftrs = model.classifier.in_features
model.classifier = nn.Sequential(
    nn.Linear(num_ftrs, 256),
    nn.ReLU(),
    nn.Dropout(0.4),
    nn.Linear(256, 2)
)

for name, param in model.named_parameters():
    param.requires_grad = False
    if "denseblock3" in name or "denseblock4" in name or "classifier"
in name:
        param.requires_grad = True

model = model.to(device)

criterion = nn.CrossEntropyLoss(weight=class_weights)

optimizer = optim.AdamW(model.parameters(), lr=1e-4, weight_decay=1e-
4)
```

```
scheduler = OneCycleLR(optimizer, max_lr=1e-3,
steps_per_epoch=len(train_loader), epochs=10)

def train_model(model, train_loader, val_loader, epochs=10):
    best_acc = 0
    for epoch in range(epochs):
        model.train()
        running_loss, correct, total = 0.0, 0, 0

        for inputs, labels in train_loader:
            inputs, labels = inputs.to(device), labels.to(device)

            optimizer.zero_grad()
            outputs = model(inputs)
            loss = criterion(outputs, labels)
            loss.backward()
            optimizer.step()
            scheduler.step()

            running_loss += loss.item()
            _, preds = torch.max(outputs, 1)
            correct += (preds == labels).sum().item()
            total += labels.size(0)

        train_acc = correct / total * 100
        val_acc = evaluate_model(model, val_loader)

        print(f"Epoch {epoch+1}/{epochs}, Loss:
{running_loss/len(train_loader):.4f}, "
              f"Train Acc: {train_acc:.2f}%, Val Acc: {val_acc:.2f}%")


        if val_acc > best_acc:
            best_acc = val_acc
            torch.save(model.state_dict(),
"best_pneumonia_densenet121.pt")

def evaluate_model(model, loader):
    model.eval()
    correct, total = 0, 0
    with torch.no_grad():
        for inputs, labels in loader:
            inputs, labels = inputs.to(device), labels.to(device)
            outputs = model(inputs)
            _, preds = torch.max(outputs, 1)
            correct += (preds == labels).sum().item()
            total += labels.size(0)
    return correct / total * 100


train_model(model, train_loader, val_loader, epochs=10)
```

```
Epoch 1/10, Loss: 0.2653, Train Acc: 91.55%, Val Acc: 87.50%
Epoch 2/10, Loss: 0.1324, Train Acc: 95.28%, Val Acc: 93.75%
Epoch 3/10, Loss: 0.1325, Train Acc: 94.94%, Val Acc: 100.00%
Epoch 4/10, Loss: 0.1026, Train Acc: 96.20%, Val Acc: 93.75%
Epoch 5/10, Loss: 0.0858, Train Acc: 97.24%, Val Acc: 100.00%
Epoch 6/10, Loss: 0.0721, Train Acc: 97.47%, Val Acc: 100.00%
Epoch 7/10, Loss: 0.0663, Train Acc: 97.60%, Val Acc: 81.25%
Epoch 8/10, Loss: 0.0496, Train Acc: 97.85%, Val Acc: 100.00%
Epoch 9/10, Loss: 0.0447, Train Acc: 98.43%, Val Acc: 93.75%
Epoch 10/10, Loss: 0.0399, Train Acc: 98.60%, Val Acc: 87.50%
```

```
model.load_state_dict(torch.load("best_pneumonia_densenet121.pt"))
model.eval()
```

```
C:\Users\Ekaansh\AppData\Local\Temp\ipykernel_18684\1610110734.py:1:
FutureWarning: You are using `torch.load` with `weights_only=False`
(the current default value), which uses the default pickle module
implicitly. It is possible to construct malicious pickle data which
will execute arbitrary code during unpickling (See
https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-
models for more details). In a future release, the default value for
`weights_only` will be flipped to `True`. This limits the functions
that could be executed during unpickling. Arbitrary objects will no
longer be allowed to be loaded via this mode unless they are
explicitly allowlisted by the user via
`torch.serialization.add_safe_globals`. We recommend you start setting
`weights_only=True` for any use case where you don't have full control
of the loaded file. Please open an issue on GitHub for any issues
related to this experimental feature.
  model.load_state_dict(torch.load("best_pneumonia_densenet121.pt"))
```

```
DenseNet(
  (features): Sequential(
    (conv0): Conv2d(3, 64, kernel_size=(7, 7), stride=(2, 2),
padding=(3, 3), bias=False)
    (norm0): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
    (relu0): ReLU(inplace=True)
    (pool0): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1,
ceil_mode=False)
    (denseblock1): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(64, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
```

```
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(96, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(96, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(128, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(160, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(160, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(192, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(192, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
```

```
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(224, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(224, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition1): _Transition(
      (norm): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock2): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(128, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(160, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(160, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
```

```
        (norm1): BatchNorm2d(192, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(192, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(224, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(224, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(256, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(288, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(288, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(320, eps=1e-05, momentum=0.1,
```

```
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(320, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(352, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(352, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(384, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(384, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
        (norm1): BatchNorm2d(416, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(416, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(448, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(448, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(480, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(480, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition2): _Transition(
      (norm): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(512, 256, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock3): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(256, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(288, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(288, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
```

```
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(320, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(320, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(352, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(352, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(384, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(384, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(416, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(416, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
```

```
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(448, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(448, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(480, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(480, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(512, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(512, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
        (norm1): BatchNorm2d(544, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(544, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(576, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(576, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(608, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(608, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer13): _DenseLayer(
        (norm1): BatchNorm2d(640, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(640, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer14): _DenseLayer(
        (norm1): BatchNorm2d(672, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(672, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
```

```
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer15): _DenseLayer(
        (norm1): BatchNorm2d(704, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(704, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer16): _DenseLayer(
        (norm1): BatchNorm2d(736, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(736, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer17): _DenseLayer(
        (norm1): BatchNorm2d(768, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(768, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer18): _DenseLayer(
        (norm1): BatchNorm2d(800, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(800, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
```

```
padding=(1, 1), bias=False)
      )
      (denselayer19): _DenseLayer(
        (norm1): BatchNorm2d(832, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(832, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer20): _DenseLayer(
        (norm1): BatchNorm2d(864, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(864, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer21): _DenseLayer(
        (norm1): BatchNorm2d(896, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer22): _DenseLayer(
        (norm1): BatchNorm2d(928, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(928, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
```

```
      (denselayer23): _DenseLayer(
        (norm1): BatchNorm2d(960, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(960, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer24): _DenseLayer(
        (norm1): BatchNorm2d(992, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(992, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (transition3): _Transition(
      (norm): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU(inplace=True)
      (conv): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1),
bias=False)
      (pool): AvgPool2d(kernel_size=2, stride=2, padding=0)
    )
    (denseblock4): _DenseBlock(
      (denselayer1): _DenseLayer(
        (norm1): BatchNorm2d(512, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(512, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer2): _DenseLayer(
        (norm1): BatchNorm2d(544, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
```

```
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(544, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer3): _DenseLayer(
        (norm1): BatchNorm2d(576, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(576, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer4): _DenseLayer(
        (norm1): BatchNorm2d(608, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(608, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer5): _DenseLayer(
        (norm1): BatchNorm2d(640, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(640, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer6): _DenseLayer(
        (norm1): BatchNorm2d(672, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
```

```
        (conv1): Conv2d(672, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer7): _DenseLayer(
        (norm1): BatchNorm2d(704, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(704, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer8): _DenseLayer(
        (norm1): BatchNorm2d(736, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(736, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer9): _DenseLayer(
        (norm1): BatchNorm2d(768, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(768, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer10): _DenseLayer(
        (norm1): BatchNorm2d(800, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(800, 128, kernel_size=(1, 1), stride=(1, 1),
```

```
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer11): _DenseLayer(
        (norm1): BatchNorm2d(832, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(832, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer12): _DenseLayer(
        (norm1): BatchNorm2d(864, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(864, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer13): _DenseLayer(
        (norm1): BatchNorm2d(896, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer14): _DenseLayer(
        (norm1): BatchNorm2d(928, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(928, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
```

```
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer15): _DenseLayer(
        (norm1): BatchNorm2d(960, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(960, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
      (denselayer16): _DenseLayer(
        (norm1): BatchNorm2d(992, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu1): ReLU(inplace=True)
        (conv1): Conv2d(992, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
        (norm2): BatchNorm2d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
        (relu2): ReLU(inplace=True)
        (conv2): Conv2d(128, 32, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), bias=False)
      )
    )
    (norm5): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  )
  (classifier): Sequential(
    (0): Linear(in_features=1024, out_features=256, bias=True)
    (1): ReLU()
    (2): Dropout(p=0.4, inplace=False)
    (3): Linear(in_features=256, out_features=2, bias=True)
  )
)

y_true, y_pred = [], []
with torch.no_grad():
    for inputs, labels in test_loader:
        inputs, labels = inputs.to(device), labels.to(device)
        outputs = model(inputs)
        _, preds = torch.max(outputs, 1)
        y_true.extend(labels.cpu().numpy())
        y_pred.extend(preds.cpu().numpy())
```

```
print("\nClassification Report:\n", classification_report(y_true,
y_pred, target_names=["PNEUMONIA", "NORMAL"]))
print("\nConfusion Matrix:\n", confusion_matrix(y_true, y_pred))
```

```
Classification Report:
              precision    recall  f1-score   support

   PNEUMONIA       0.98      0.86      0.91       234
      NORMAL       0.92      0.99      0.95       390

    accuracy                           0.94       624
   macro avg       0.95      0.92      0.93       624
weighted avg       0.94      0.94      0.94       624


Confusion Matrix:
 [[201  33]
 [  5 385]]
```