Transfer Learning:

The use of pre-trained models.

torch ← models.densenet121(pretrained = True)

then, we have to freeze model params:

```
for param in model.parameters():
    param.requires_grad = False
```

} we don't want to affect these.

Replace their classifier with our classifier:

```
from Collections import OrderedDict
classifier = nn.Sequential(OrderedDict([
    ('fc1', nn.Linear(1024, 500)),
    ('relu', nn.ReLU()),
    ('fc2', nn.linear(500, 2)),
    ('output', nn.LogSoftmax(dim=1))
]))
```

a new, untrained classifier

```
model.classifier = classifier
```

Using GPU :

```
                                          Params
model.cuda()    # move the model            to GPU
images.cuda()   # move tensor to gpu


model.cpu()   } returning the computations
images.cpu()  }        to cpu.
```