Tutorial 4
Introduction to PyTorch

Kai CHEN

Overview

PyTorch

Get Started | Features | Ecosystem | Blog | Tutorials | Docs | Resources | Github

FROM RESEARCH TO PRODUCTION

An open source deep learning platform that provides a seamless path from research prototyping to production deployment.

Get Started >

KEY FEATURES & CAPABILITIES

See all Features >

Hybrid Front-End

A new hybrid front-end seamlessly

Distributed Training

Scalable distributed training and

Python-First

Deep integration into Python allows

Tools & Libraries

A rich ecosystem of tools and

Overview

Developers

facebook

NVIDIA

salesforce

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIE
PARIS INSTITUTE OF TECHNOLOGY

Carnegie Mellon University

UNIVERSITE PIERRE & MARIE CURIE
LA SCIENCE A PARIS

Digital Reasoning

Stanford University

UNIVERSITY OF OXFORD

NYU

Inria

ENS
ÉCOLE NORMALE SUPÉRIEURE

Overview

Main components

- Model
- DataLoader
- Optimizer

Example

Model

Define layers

Define computational graph (network structure)

```python
import torch
import torch.nn as nn
import torch.nn.functional as F


class Net(nn.Module):

    def __init__(self):
        super(Net, self).__init__()
        self.conv1 = nn.Conv2d(1, 10, kernel_size=5)
        self.conv2 = nn.Conv2d(10, 20, kernel_size=5)
        self.fc = nn.Linear(320, 10)
        self.max_pool = nn.MaxPool2d(2)
        self.relu = nn.ReLU(inplace=True)

    def forward(self, x):
        x = self.conv1(x)
        x = self.max_pool(x)
        x = self.relu(x)
        x = self.conv2(x)
        x = self.max_pool(x)
        x = self.relu(x)
        x = x.view(-1, 320)
        x = self.fc(x)
        return x
```

Example

DataLoader

```
from torchvision import datasets, transforms

dataset = datasets.MNIST(
    '../data',
    train=True,
    download=True,
    transform=transforms.Compose(
        [transforms.ToTensor(),
         transforms.Normalize((0.1307, ), (0.3081, ))]))
train_loader = torch.utils.data.DataLoader(dataset)
```

Example

Optimizer

```
import torch.optim as optim

model = Net()

optimizer = optim.SGD(model.parameters(), lr=0.01, momentum=0.9)
```

Example

Training

```
if torch.cuda.is_available():
    device = torch.device('cuda')
else:
    device = torch.device('cpu')

model.train()
for i, (data, target) in enumerate(train_loader):
    data, target = data.to(device), target.to(device)
    optimizer.zero_grad()
    output = model(data)
    loss = F.cross_entropy(output, target)
    loss.backward()
    optimizer.step()
    print('Iteration {}\tLoss: {:.6f}'.format(i, loss.item()))
```

Overview

Full Example

https://github.com/pytorch/examples/blob/master/mnist/main.py

Thank you!