PyTorch

数据增强

主讲人：龙良曲

Big Data

- The key to prevent Overfitting

Sample more data?

Limited Data

- Small network capacity
- Regularization
- Data argumentation

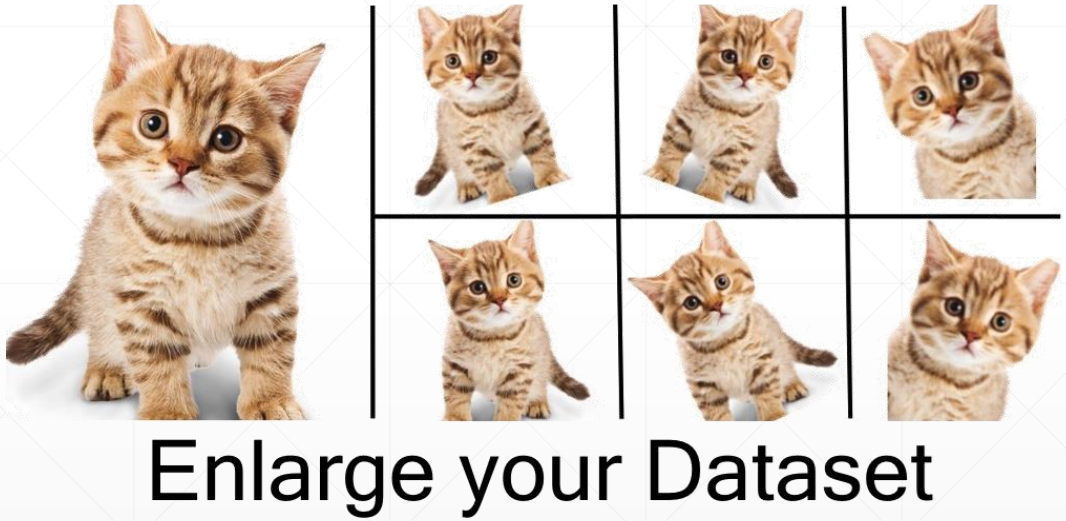
Recap

```
cifar_train = datasets.CIFAR10('cifar', True, transform=transforms.Compose([
    transforms.Resize((32, 32)),
    transforms.ToTensor()
]), download=True)
cifar_train = DataLoader(cifar_train, batch_size=batchsz, shuffle=True)

cifar_test = datasets.CIFAR10('cifar', False, transform=transforms.Compose([
    transforms.Resize((32, 32)),
    transforms.ToTensor()
]), download=True)
```

Data argumentation

- Flip
- Rotate
- Random Move & Crop
- GAN

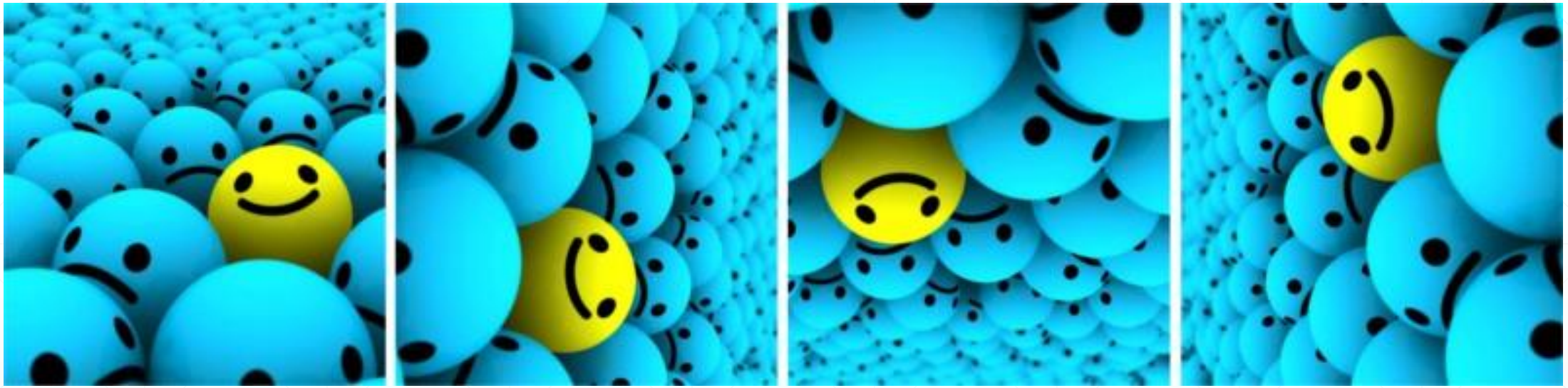


https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced

Flip

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

Rotate



Rotate

```
train_loader = torch.utils.data.DataLoader(
    datasets.MNIST('../data', train=True, download=True,
                   transform=transforms.Compose([
                       transforms.RandomHorizontalFlip(),
                       transforms.RandomVerticalFlip(),
                       transforms.RandomRotation(15),
                       transforms.RandomRotation([90, 180, 270]),
                       transforms.ToTensor(),
                       # transforms.Normalize((0.1307,), (0.3081,))
                   ])),
    batch_size=batch_size, shuffle=True)
```

Scale

```
train_loader = torch.utils.data.DataLoader(
    datasets.MNIST('../data', train=True, download=True,
                   transform=transforms.Compose([
                       transforms.RandomHorizontalFlip(),
                       transforms.RandomVerticalFlip(),
                       transforms.RandomRotation(15),
                       transforms.RandomRotation([90, 180, 270]),
                       transforms.Resize([32, 32]),
                       transforms.ToTensor(),
                       # transforms.Normalize((0.1307,), (0.3081,))
                   ])),
    batch_size=batch_size, shuffle=True)
```

Crop Part

```
train_loader = torch.utils.data.DataLoader(
    datasets.MNIST('../data', train=True, download=True,
                   transform=transforms.Compose([
                       transforms.RandomHorizontalFlip(),
                       transforms.RandomVerticalFlip(),
                       transforms.RandomRotation(15),
                       transforms.RandomRotation([90, 180, 270]),
                       transforms.Resize([32, 32]),
                       transforms.RandomCrop([28, 28]),
                       transforms.ToTensor(),
                       # transforms.Normalize((0.1307,), (0.3081,))
                   ])),
    batch_size=batch_size, shuffle=True)
```

Noise

- Data argumentation will help
- But not too much

下一课时

艺术风格迁移

Thank You.