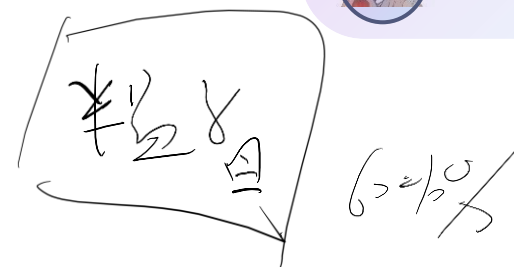
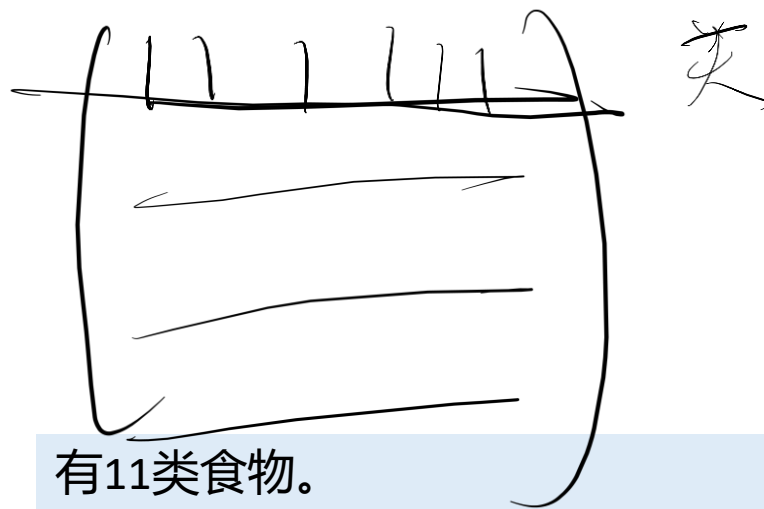


食物分类问题



李哥考研

看看长啥样



有11类食物。

其中带标签的数据: $280 * 11$

不带标签的训练数据: 6786

验证集 $30 * 11$

测试集 3347

8' % - 62%

又批-82% 82%

代码文件: food_classification

数据增广



李哥考研

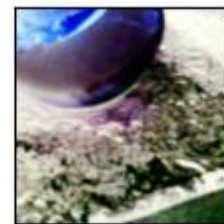
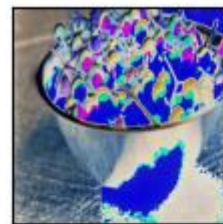
全部作为训练集



图片增广



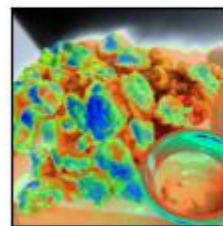
李哥考研



图片增广



李哥考研



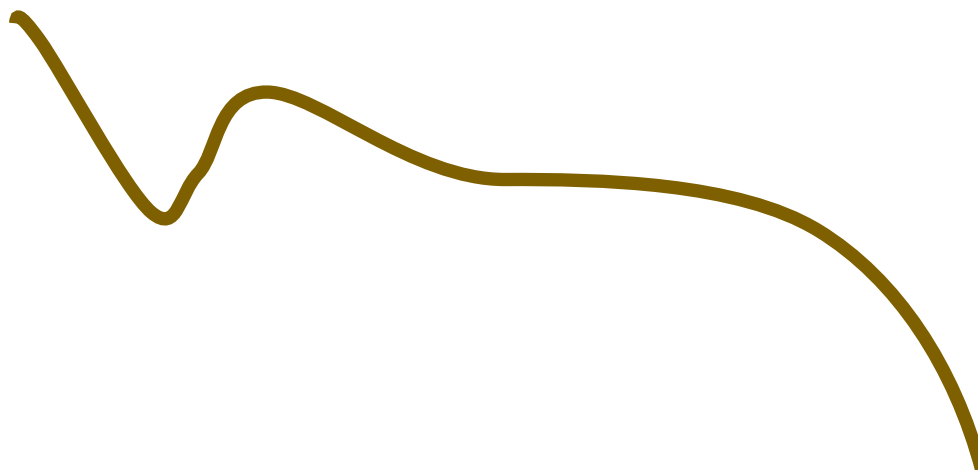
Adam和AdamW



李哥考研

1, 梯度的改变

2, 学习率的改变





```
np.sum(np.argmax(pred.cpu().data.numpy(), axis=1) == data[1].cpu().numpy())
```

```
[  
[0.1, 0.5, 0.4],  
[0.2, 0.3, 0.5],  
[0.2, 0.1, 0.7],  
]
```

```
[  
1  
0  
2  
]
```

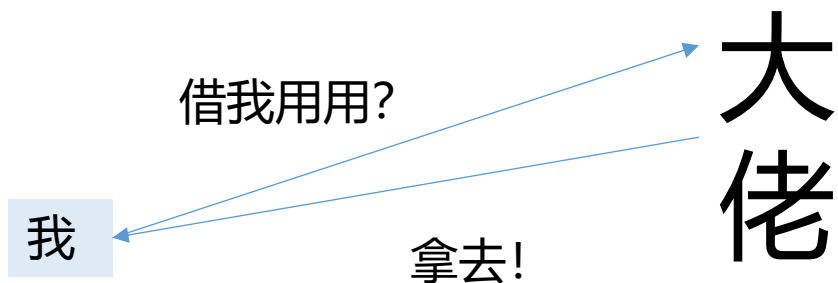

迁移学习：数据量少时的最佳选择



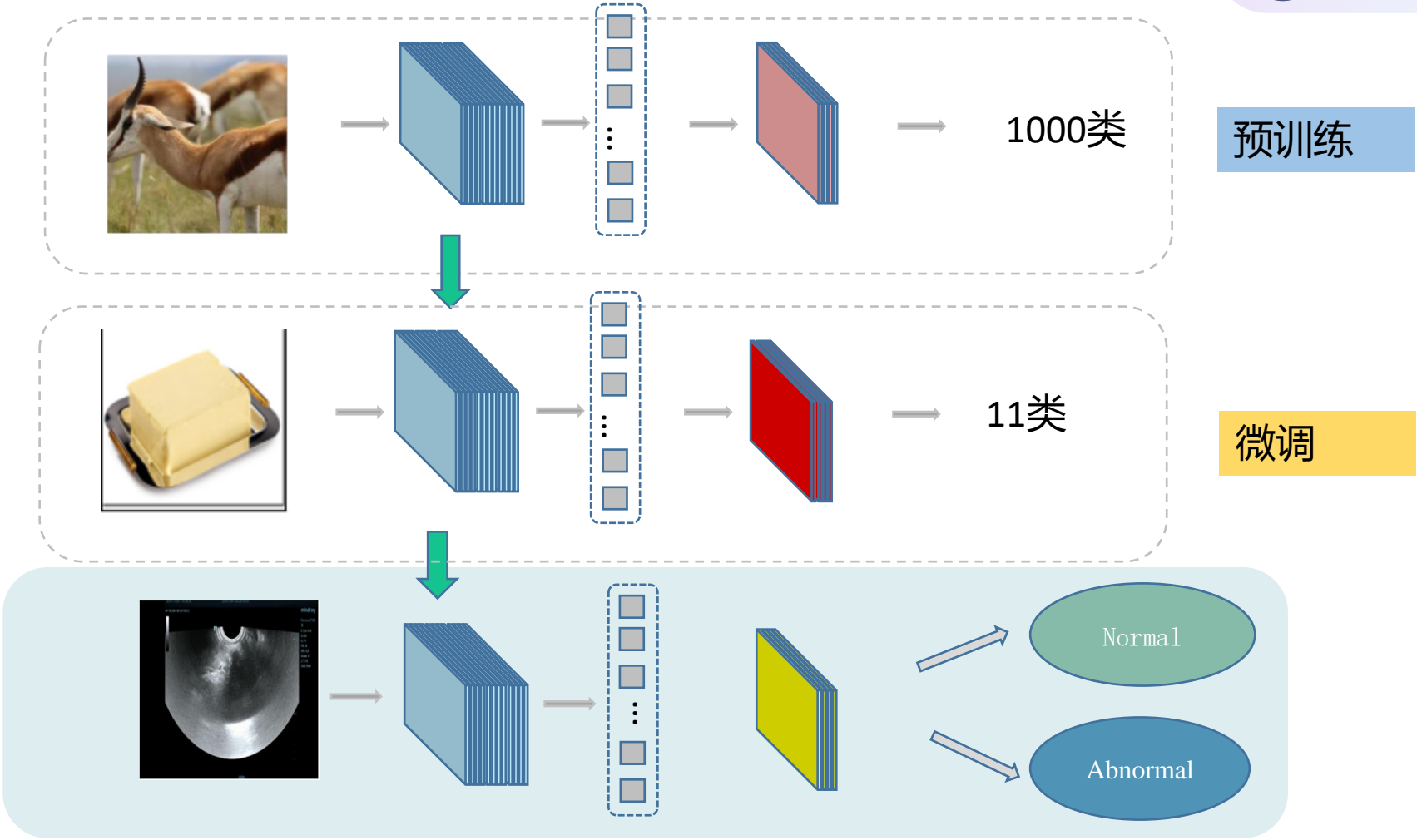
李哥考研

大佬们的模型：花了百万美元，在千万上亿的数据集上训练，提取的特征特别好。

我的模型：5000元电脑，在千张照片上训练，提取的特征：二分类准确率有百分之五十吧



迁移学习



特征提取器 features classifer



李哥考研

读到了660个数据

```
[000/015] 19.62 sec(s) TrainLoss : 2.011990 | valLoss: 0.112902 Trainacc : 0.331818 | valacc: 0.395455
[001/015] 16.78 sec(s) TrainLoss : 1.701985 | valLoss: 0.131512 Trainacc : 0.446104 | valacc: 0.377273
[002/015] 17.09 sec(s) TrainLoss : 1.594606 | valLoss: 0.123432 Trainacc : 0.471753 | valacc: 0.446970
[003/015] 16.92 sec(s) TrainLoss : 1.475840 | valLoss: 0.092428 Trainacc : 0.528896 | valacc: 0.540909
[004/015] 17.06 sec(s) TrainLoss : 1.382760 | valLoss: 0.082835 Trainacc : 0.549351 | valacc: 0.586364
```

100% | 11/11 [00.04<00.00, 2.3810/s]

读到了660个数据

```
[000/015] 19.61 sec(s) TrainLoss : 2.384863 | valLoss: 0.143157 Trainacc : 0.174675 | valacc: 0.209091
[001/015] 16.81 sec(s) TrainLoss : 2.179686 | valLoss: 0.147024 Trainacc : 0.235065 | valacc: 0.225758
[002/015] 16.85 sec(s) TrainLoss : 2.139043 | valLoss: 0.128330 Trainacc : 0.250325 | valacc: 0.304545
[003/015] 16.93 sec(s) TrainLoss : 2.068773 | valLoss: 0.138902 Trainacc : 0.263312 | valacc: 0.222727
[004/015] 16.97 sec(s) TrainLoss : 2.041001 | valLoss: 0.141245 Trainacc : 0.286039 | valacc: 0.256061
[005/015] 17.06 sec(s) TrainLoss : 1.998061 | valLoss: 0.126768 Trainacc : 0.293506 | valacc: 0.290909
[006/015] 17.15 sec(s) TrainLoss : 1.982305 | valLoss: 0.136104 Trainacc : 0.306169 | valacc: 0.263636
[007/015] 17.25 sec(s) TrainLoss : 1.953962 | valLoss: 0.120078 Trainacc : 0.316558 | valacc: 0.343939
```

高

07-60



我们只需要站在巨人的肩膀上



李哥考研

有预训练模型的时候，尽量使用预训练模型

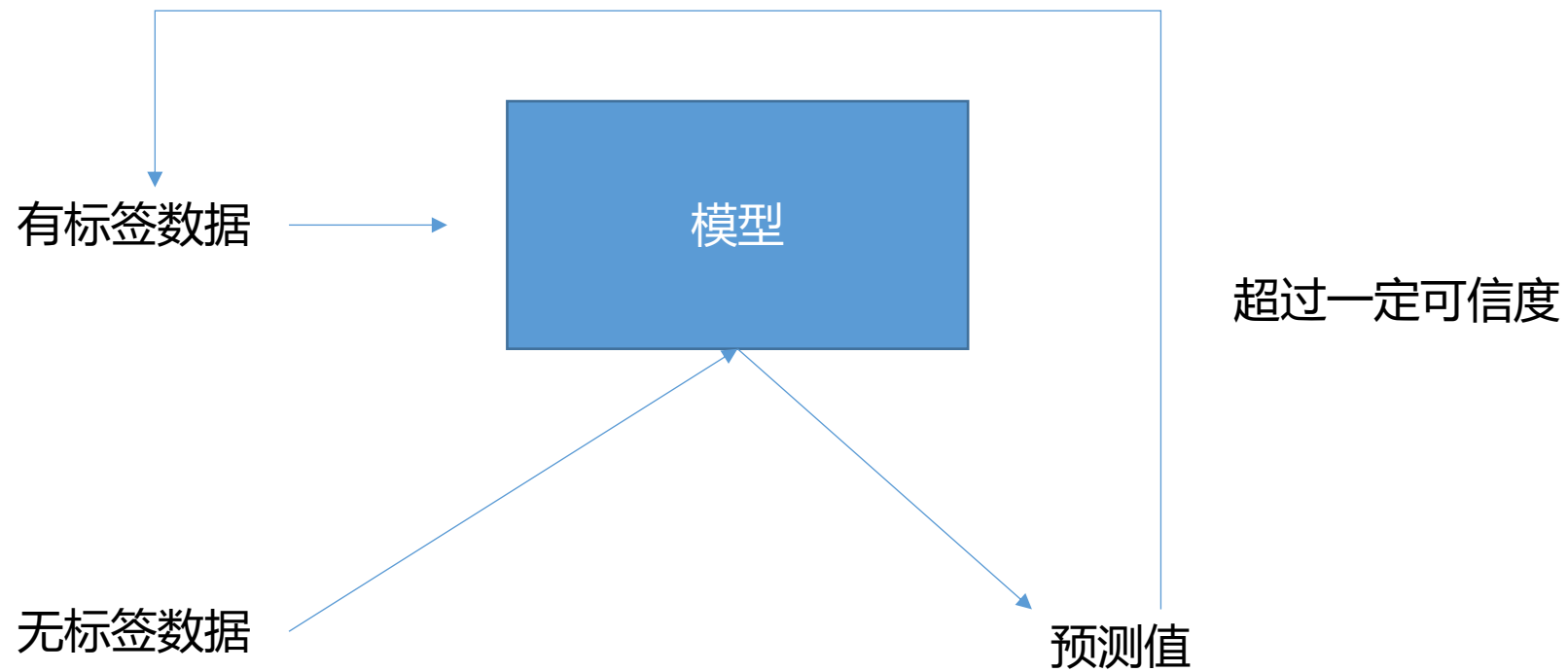
我们使用他人模型架构的原因，最大的原因不是因为他们架构好，而是因为可以迁移。



半监督



李哥考研



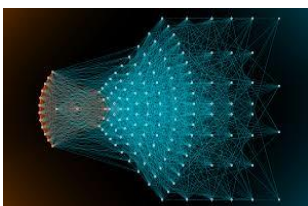
一个最简单的神经网络项目。



李哥考研



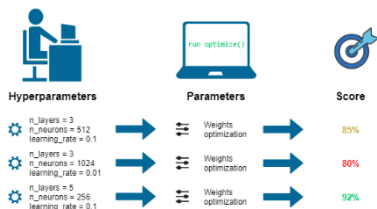
Data: 一般输入是文件地址,
输出是一个存储了数据对X,Y的数据结构。
torch中一般用dataloader来装载。



Model: 定义自己的模型

输入x, 输出预测值

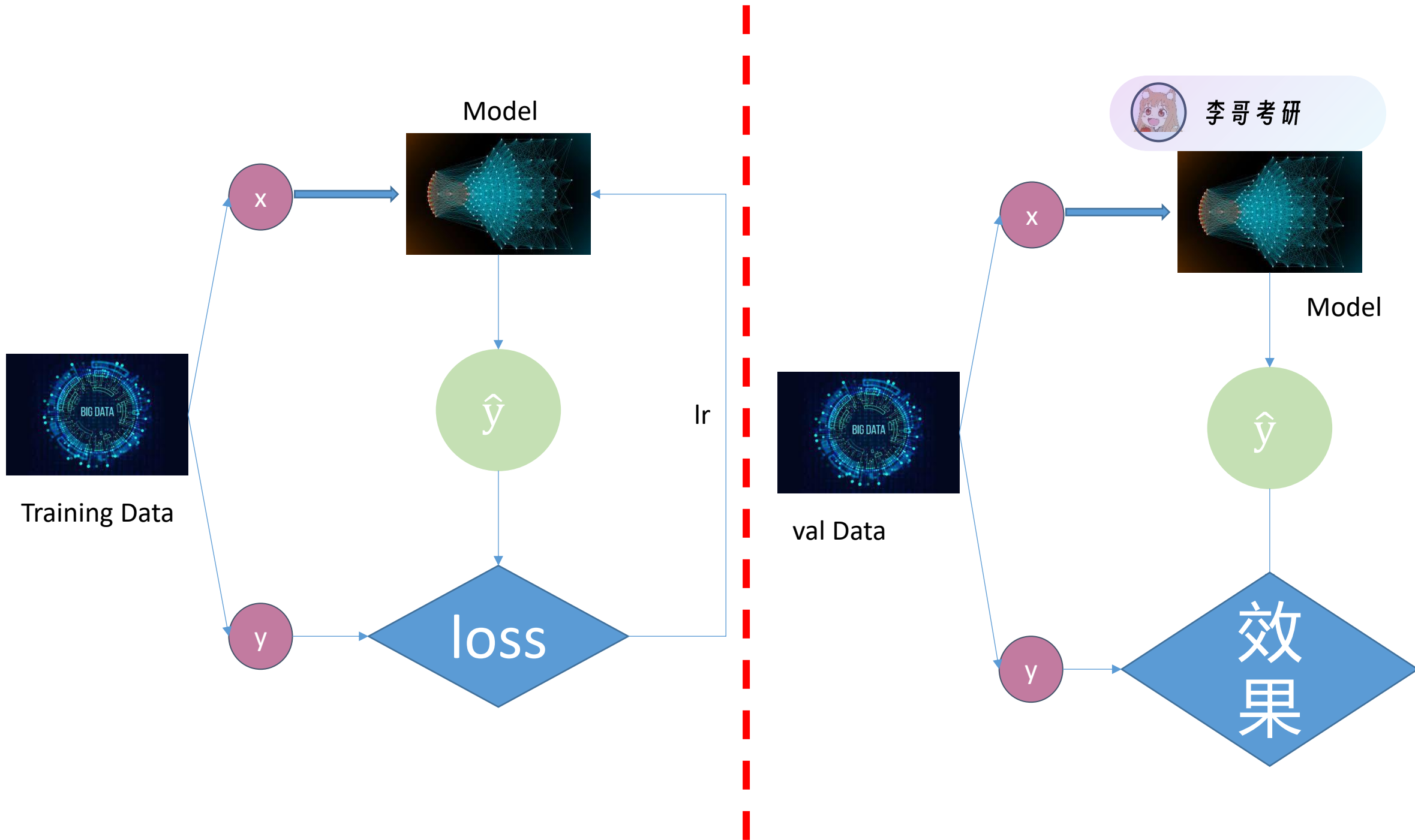
训练流程



hyperPara: 除模型外的超参

一般包含: 学习率, 优化器,
损失函数等

回归损失函数: $LOSS = |\hat{y} - y|$



其他数据集



李哥考研

mnist



注意，当图片很小时，也许你可以尝试直接展平然后使用全连接

然后与卷积进行对比。

Fashion mist



cifar10

