

图像分类原理与应用



前言

- 近年来，深度学习在图像分类研究中取得了突破性进展，本章节首先介绍图像分类问题及其原理，进而介绍CIFAR-10分类实验的整体流程。

目标

- 学完本课程后，您将能够：
 - 了解图像分类任务与评估标准
 - 掌握LeNet5网络的架构
 - 了解图像分类的常用模型

目录

1. 图像分类任务及评估标准
2. 图像分类常用模型
3. CIFAR-10分类任务介绍

图像分类的概念

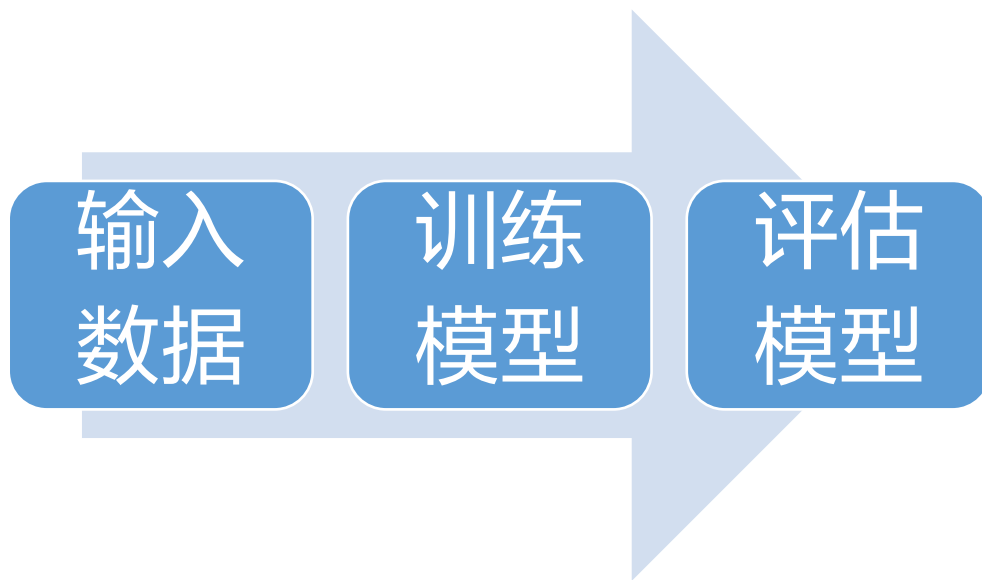
- 图像分类（ Image Classification ）： 需要根据图像处理技术，判断输入的图像属于固定几个类别中的哪一类。图像分类是图像处理在人工智能领域的基本研究主题之一，是计算机视觉领域的核心问题。

图像分类任务面临的困难

- 对于人来说，识别猫、狗非常简单，这是由于我们已经见过了很多猫，也见过了猫的各种动作形态。但是对于计算机来说，同一只猫的不同动作都是两张不同的图片。所以，图像分类任务面临以下困难：
 - 刚体&非刚体的变化：不同类型其变化都不一样；
 - 多视角：收集同一个物体图像，获取的角度是多变的；
 - 尺度：在现实生活中，很多物体的尺度都是千变万化；
 - 遮挡：目标物体可能被挡住。有时候只有物体的一小部分是可见的；
 - 光照条件：在像素层面上，光照的影响非常大；
 - 类内差异：一类物体的个体之间有许多不同的对象，每个都有自己的外形。

图像分类任务流程

- 输入数据：包含N个图像的集合，每个图像的标签是K种分类标签中的一种。这个集合称为训练集。
- 训练模型：使用训练集来训练模型，调整模型中的所有参数，从而学习每个类到底长什么样。
- 评估模型：使用验证集来评估模型表现，根据模型效果来调整模型的超参数。



分类性能度量 (1)

- Ground Truth
 - 图像处理中，数据集中的真实标签被称为Ground Truth（GT）。
- 正确率（accuracy）
 - 判断正确的样本数占有所有样本数的比例。
- 错误率（error rate）
 - 判断错误的样本数占有所有样本数的比例。

分类性能度量 (2)

- 术语:

- P: 正元组, 感兴趣的主要类的元组。
- N: 负元组, 其他元组。
- TP: 真正例, 被分类器正确分类的正元组。
- TN: 真负例, 被分类器正确分类的负元组。
- FP: 假正例, 被错误地标记为正元组的负元组。
- FN: 假负例, 被错误的标记为负元组的正元组。

预测 实际	<i>yes</i>	<i>no</i>	合计
<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
合计	<i>P'</i>	<i>N'</i>	<i>P + N</i>

混淆矩阵

- 混淆矩阵: 是一个至少为 $m \times m$ 的表。前 m 行和 m 列的表目 $CM_{(i,j)}$ 指出类 i 的元组被分类器标记为 j 的个数。
 - 理想地, 对于高正确率的分类器, 大部分元组应该被混淆矩阵从 $CM_{1,1}$ 到 $CM_{(m,m)}$ 的对角线上的表目表示, 而其他表目为 0 或者接近于 0。即 FP 和 FN 接近 0。

分类性能度量 (3)

度量	公式
正确率、识别率	$\frac{TP + TN}{P + N}$
错误率、误分类率	$\frac{FP + FN}{P + N}$
敏感度、真正例率、召回率 (<i>recall</i>)	$\frac{TP}{P}$
特效性、真负例率	$\frac{TN}{N}$
精度 (<i>precision</i>)	$\frac{TP}{TP + FP}$
F 、 F_1 、 F 分数精度和召回率的调和均值	$\frac{2 \times precision \times recall}{precision + recall}$
F_β ，其中 β 是非负实数	$\frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$

分类性能度量举例

- 我们训练了一个机器学习的模型用来识别图片中是不是一只猫，现在用200张图片来验证下模型的性能指标。这200张图片中，170张是猫，30张不是猫。模型的识别结果为160张是猫，40张不是猫。

精度： $P = TP / (TP + FP) = 140 / (140 + 20) = 87.5\%$

召回率： $R = TP / P = 140 / 170 = 82.3\%$

正确率： $ACC = (TP + TN) / (P + N) = (140 + 10) / (170 + 30) = 85\%$

预测 实际	<i>yes</i>	<i>no</i>	合计
<i>yes</i>	140	30	170
<i>no</i>	20	10	30
合计	160	40	200

思考题

1. 在刑侦监控的人脸检测场景中，对犯罪分子黑名单报警宁可误报，不可漏报。这需要系统的哪个指标很高？（ ）
 - A. 精度
 - B. 正确率
 - C. 召回率
 - D. 置信度

思考题

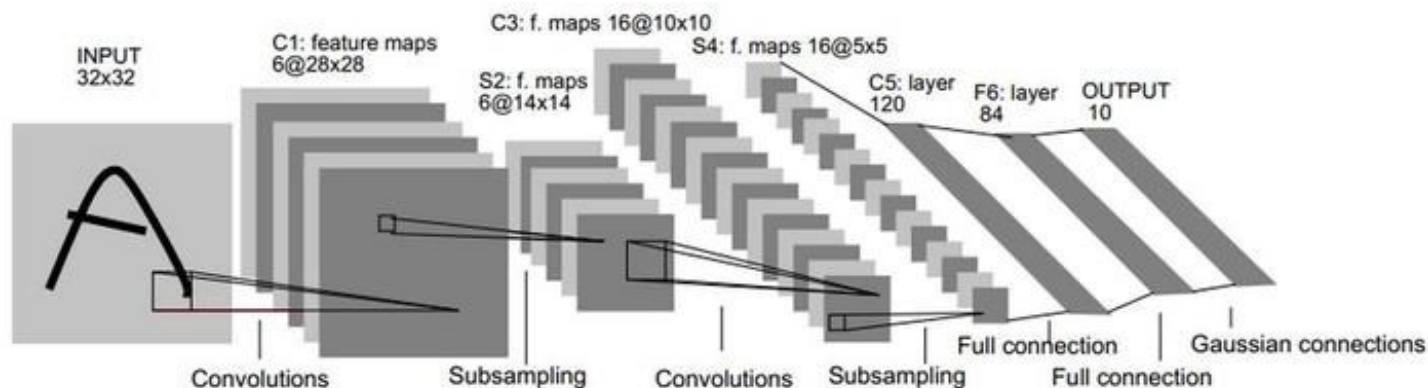
1. 在交通执法的车辆检测场景中，对违章车辆的判罚宁可漏报，不要误报，以免产生错误的罚款。这需要系统的哪个指标很高？（ ）
 - A. 精度
 - B. 正确率
 - C. 召回率
 - D. 置信度

目录

1. 图像分类的任务及评估标准
2. 图像分类常用模型
3. CIFAR-10分类任务介绍

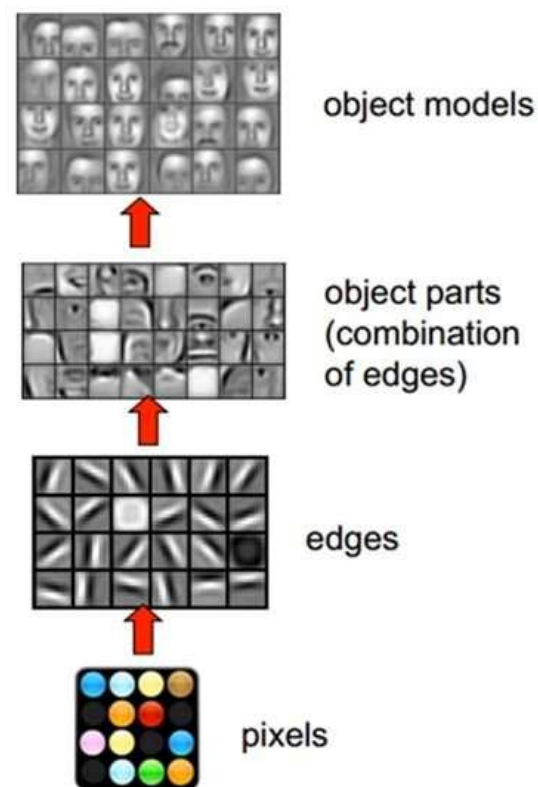
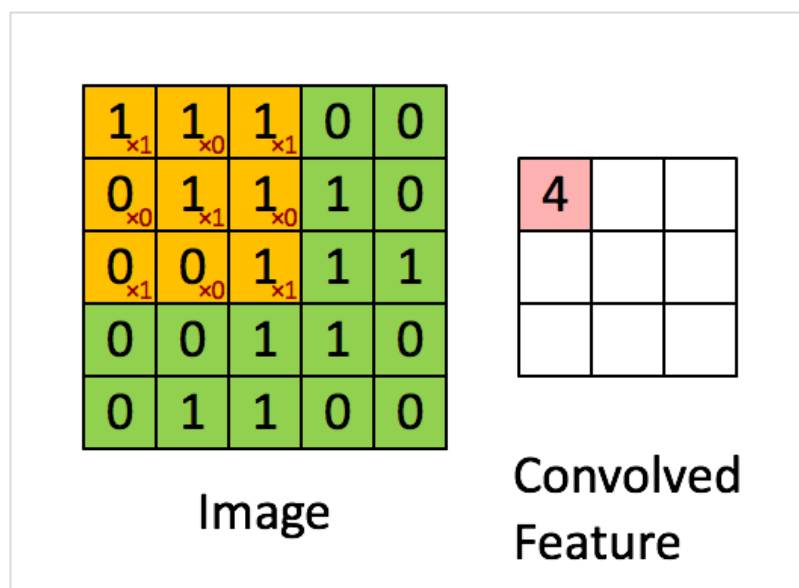
LeNet5

- LeNet-5出自论文Gradient-Based Learning Applied to Document Recognition，是一种用于手写体字符识别的非常高效的卷积神经网络。
- LeNet-5共有7层，不包含输入，共有两层卷积层，两层下采样层和三层全连接层。每层都包含可训练参数；每个层有多个Feature Map，每个Feature Map通过一种卷积滤波器提取输入的一种特征，然后每个Feature Map有多个神经元。



卷积层

- 主要作用：进行卷积操作，提取特征，生成Feature Map。



目标特征



局部特征



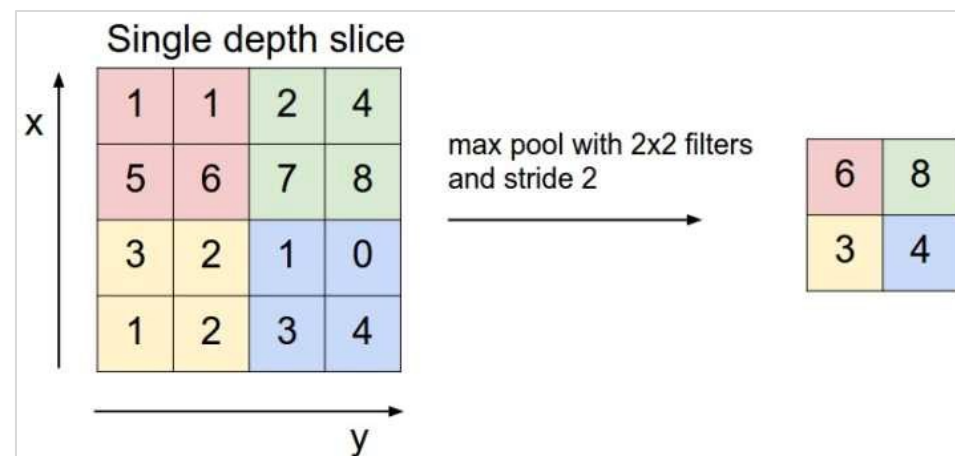
边缘纹理



输入图像像素

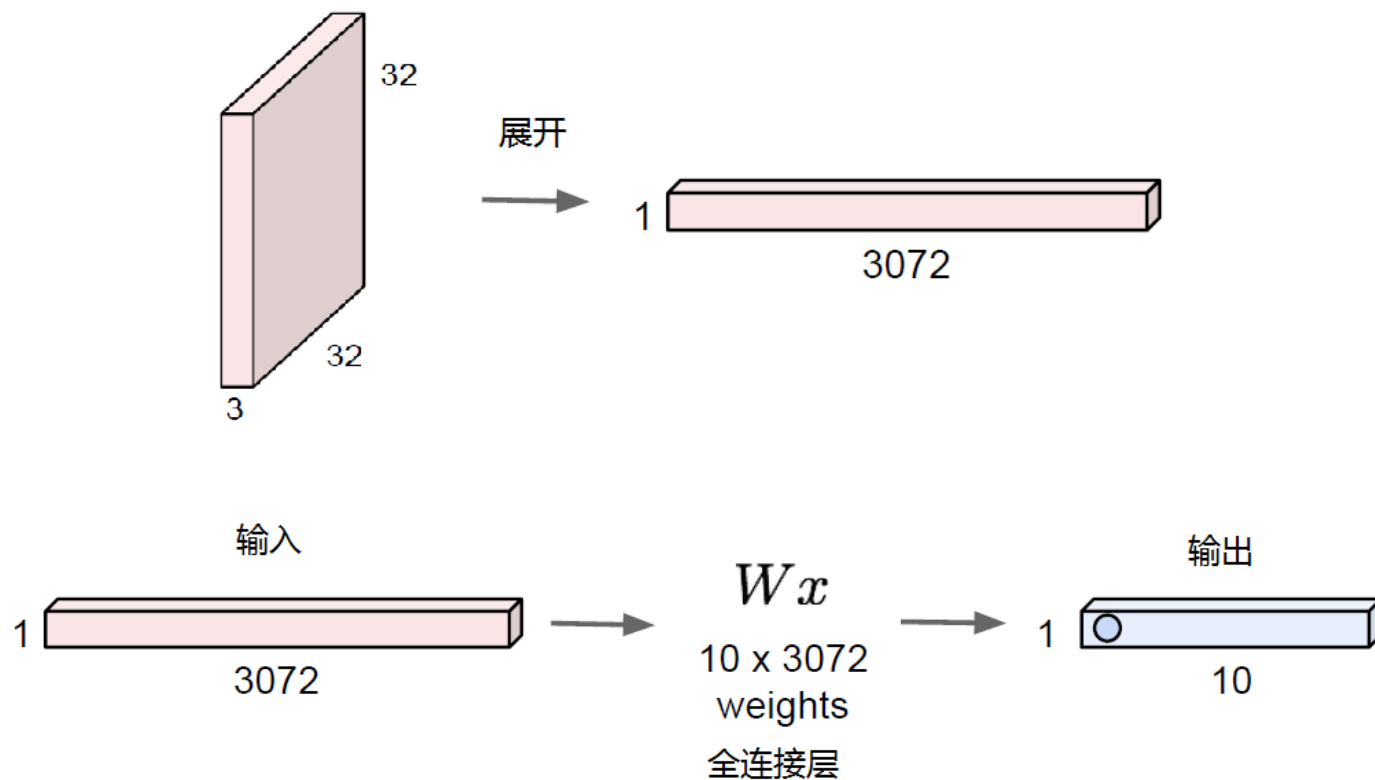
池化层

- 主要作用：减小图像尺寸，减少训练参数的数量。池化层一般分为最大池化（max pooling）和平均池化（average pooling）。



全连接层

- 主要作用：将最后得到的特征映射到线性可分的空间，通常，卷积神经网络会将末端得到的特征图平摊成一个长的列向量，经过全连接层的计算得到最终的输出层。



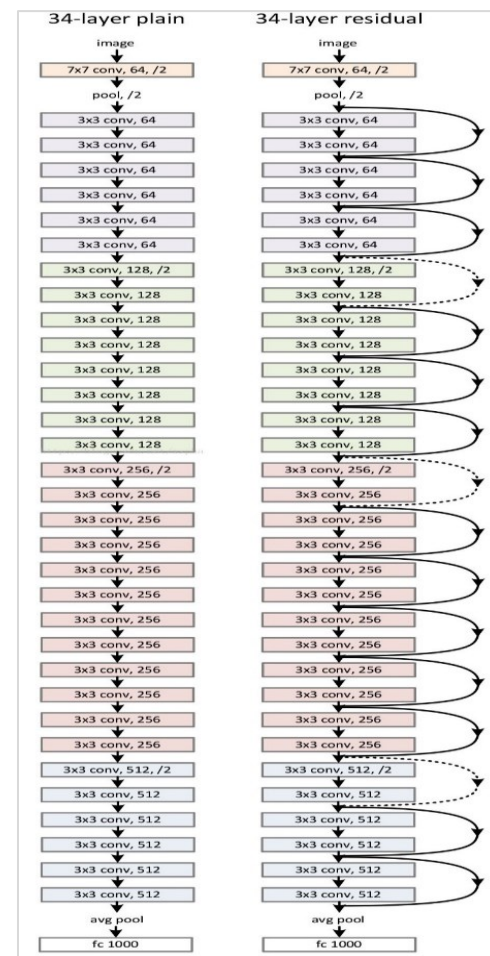
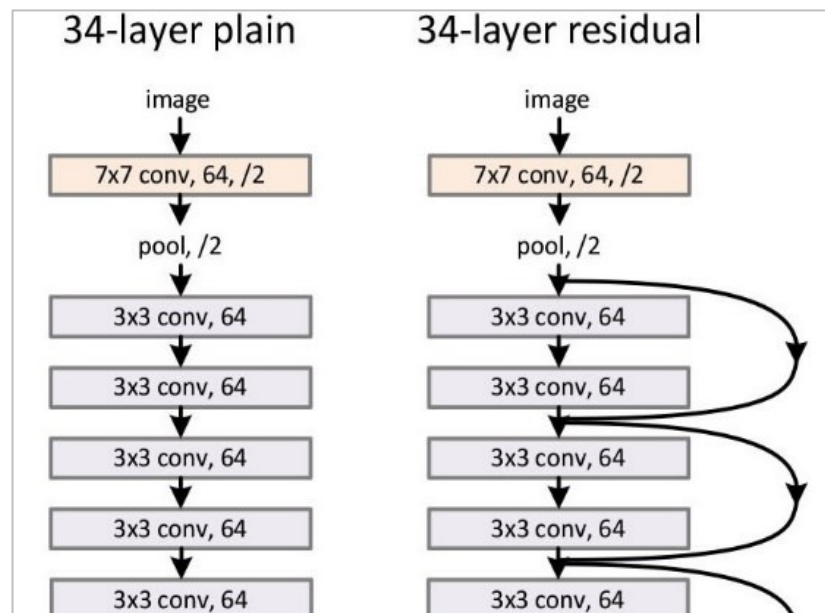
图像分类常用模型：VGG Net

- VGG模型是2014年ILSVRC竞赛的第二名，有很强的特征提取能力。
- 特点：
 - 小卷积核，作者将卷积核全部替换为3x3；
 - 小池化核，VGG全部为2x2的池化核；
 - 层数更深特征图更宽；

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

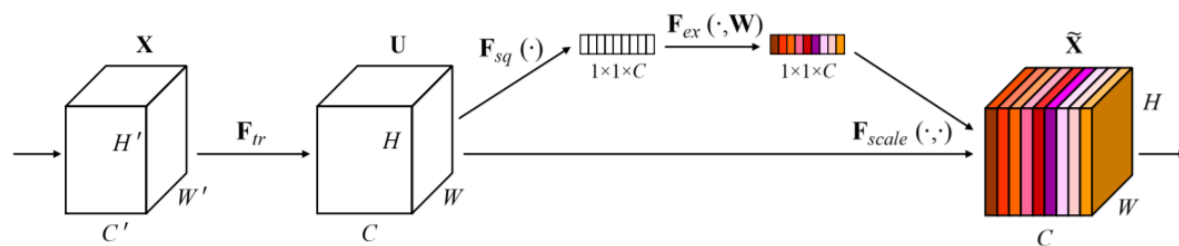
图像分类常用模型：ResNet

- 随着网络深度增加，网络的准确度应该同步增加。但是增加网络深度后，比较靠前的网络层梯度会很小。这意味着这些层基本上学习停滞了，这就产生了梯度消失问题。
- 当网络更深时，网络的参数空间更大，优化问题变得更难，因此简单地增加网络深度反而会导致更高的训练误差。
- 残差网络的提出就是为了解决以上两个问题。



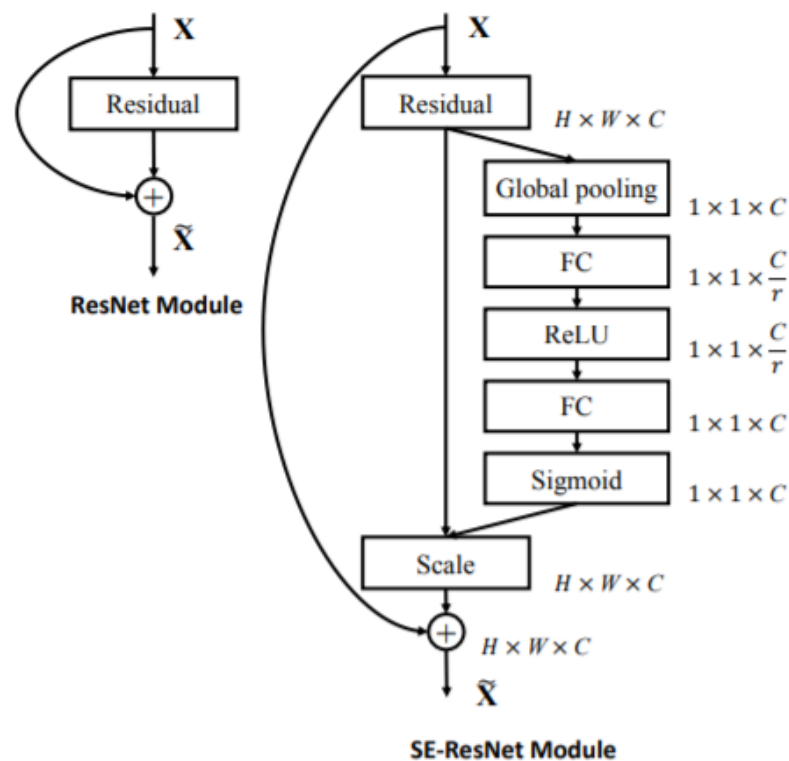
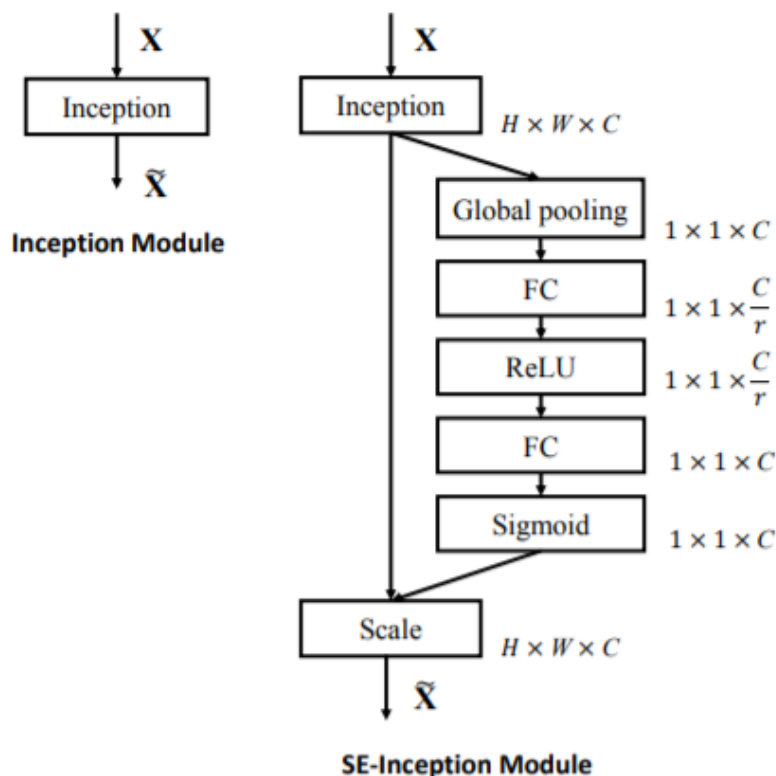
图像分类常用模型：SENet

- 下图是SENet中的核心SE模块，包含了SENet的主要思想。
 - 首先，进行卷积操作，将通道数由 C' 变为 C ；
 - 然后，进行Squeeze操作，顺着空间维度来进行特征压缩，将每个二维的特征通道变成一个实数，这个实数某种程度上具有全局的感受野，并且输出的维度和输入的特征通道数相匹配；
 - 然后，是Excitation操作，它是一个类似于神经网络中门的机制。通过参数来为每个特征通道生成权重，其中参数被学习用来显式地建模特征通道间的相关性；
 - 最后，是Reweight的操作，将Excitation的输出的权重看做是进过特征选择后的每个特征通道的重要性，然后通过乘法逐通道加权到先前的特征上，完成在通道维度上的对原始特征的重标定。



图像分类常用模型：SENet

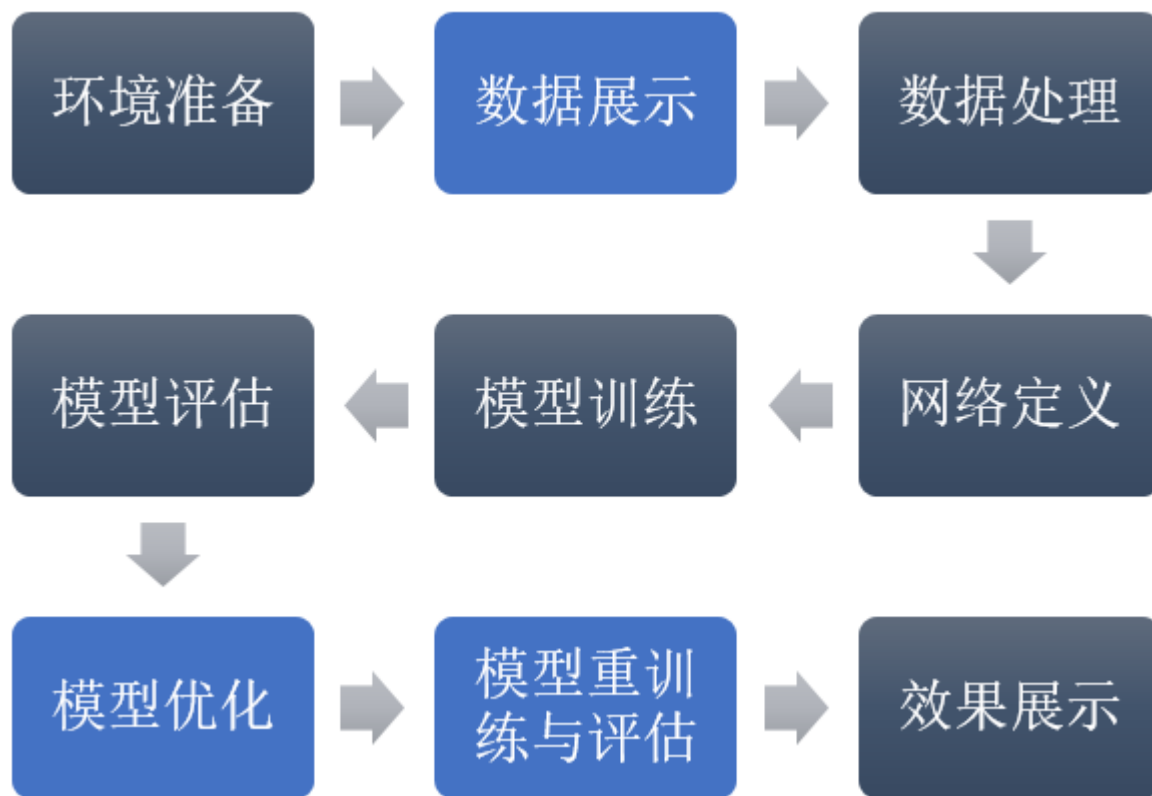
- SE模块可以融入不同的网络来实现其功能，下图分别是将SE模块融入Inception网络和ResNet网络的结构图。



目录

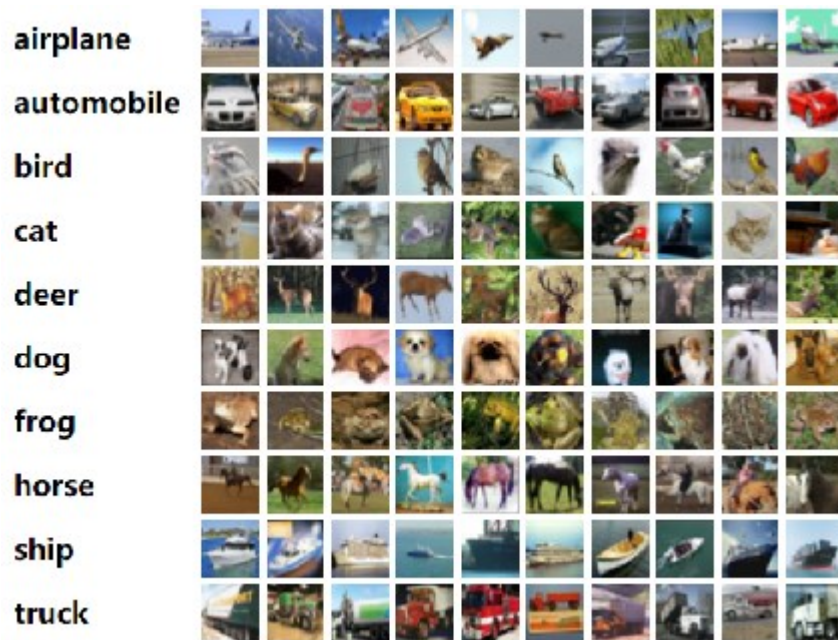
1. 图像分类的任务及评估标准
2. 图像分类常用模型
3. CIFAR-10分类任务介绍

CIFAR-10分类流程



CIFAR-10数据集

- 由10个类的60000个32x32彩色图像组成，每个类有6000个图像。有50000个训练图像和10000个测试图像。数据集分为五个训练批次和一个测试批次，每个批次有10000个图像。测试批次包含来自每个类别的恰好1000个随机选择的图像。训练批次以随机的顺序输入图像，但一些训练批次可能包含来自一个类别的图像比另一个更多。总体来说，五个训练集之和包含来自每个类的正好5000张图像。



数据处理

- 数据读取之后，往往不会直接放入模型进行训练，会先对图像进行一定的处理操作。如：归一化、反转、镜像等。
- 其作用在于经过操作后的图像加入训练集中实现图像增强的效果。同时，提升数据集的样本质量。
- 本实验就采用随机裁剪，随机翻转等处理操作。

```
if usage=='train':
    trans = [
        CV.Normalize(RGB_mean, RGB_std), # 数据归一化

        # 数据增强
        CV.RandomCrop([32, 32], [4, 4, 4, 4]), # 随机裁剪
        CV.RandomHorizontalFlip(), # 随机翻转

        CV.HWC2CHW() # 通道前移 (为适配网络, CHW的格式可最佳发挥昇腾芯片算力)
    ]
else:
    trans = [
        CV.Normalize(RGB_mean, RGB_std), # 数据归一化
        CV.HWC2CHW() # 通道前移 (为适配网络, CHW的格式可最佳发挥昇腾芯片算力)
    ]

typecast_op = C.TypeCast(mstype.int32) # 原始数据的标签是uint, 计算损失需要int
```

网络定义及模型训练

- 根据LeNet5的网络结构，构建网络。
- 同时，为网络设置损失函数、优化器。

```
# 建构网络
def construct(self, x):
    x = self.conv1(x)
    x = self.relu(x)
    x = self.max_pool2d(x)
    x = self.conv2(x)
    x = self.relu(x)
    x = self.max_pool2d(x)
    x = self.flatten(x)
    x = self.fc1(x)
    x = self.relu(x)
    x = self.fc2(x)
    x = self.relu(x)
    x = self.fc3(x)
    return x
```

```
# 网络
network1 = LeNet5(10)

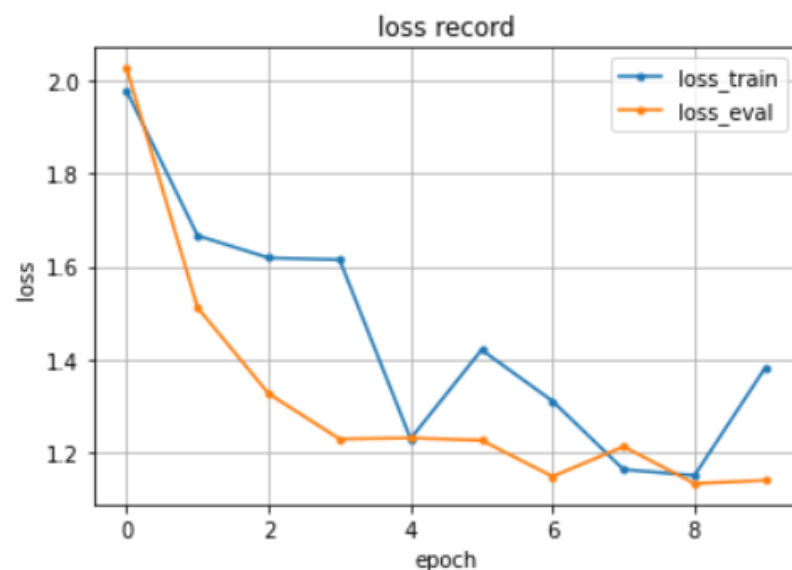
# 损失函数
net_loss = nn.SoftmaxCrossEntropyWithLogits(sparse=True, reduction='mean')

# 优化器
net_opt = nn.Momentum(params=network1.trainable_params(), learning_rate=0.01, momentum=0.9)

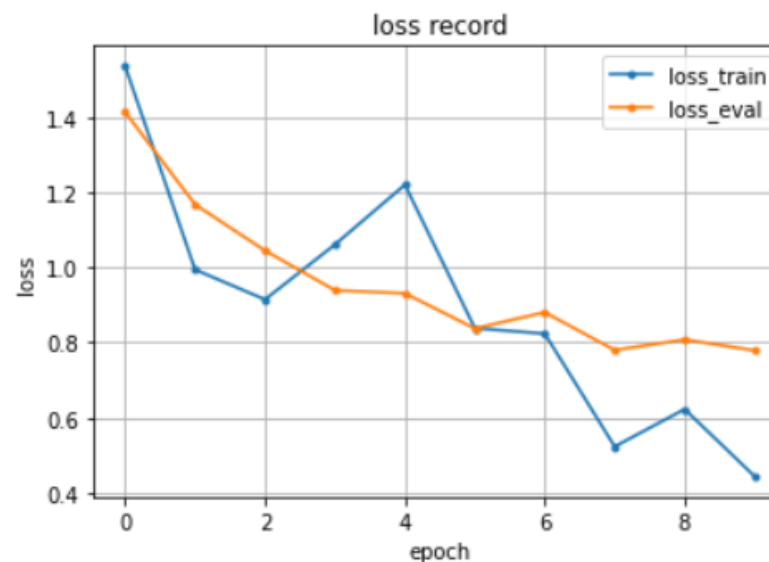
# 模型
model = Model(network = network1, loss_fn=net_loss, optimizer=net_opt, metrics={'accuracy': Accuracy(), 'loss': Loss()})
```

模型评估与优化

- 设定CallBack函数，记录训练中的各项评价指标，可视化后用于了解网络训练情况。
- 调整网络结构后，再次进行训练，观察模型表现。



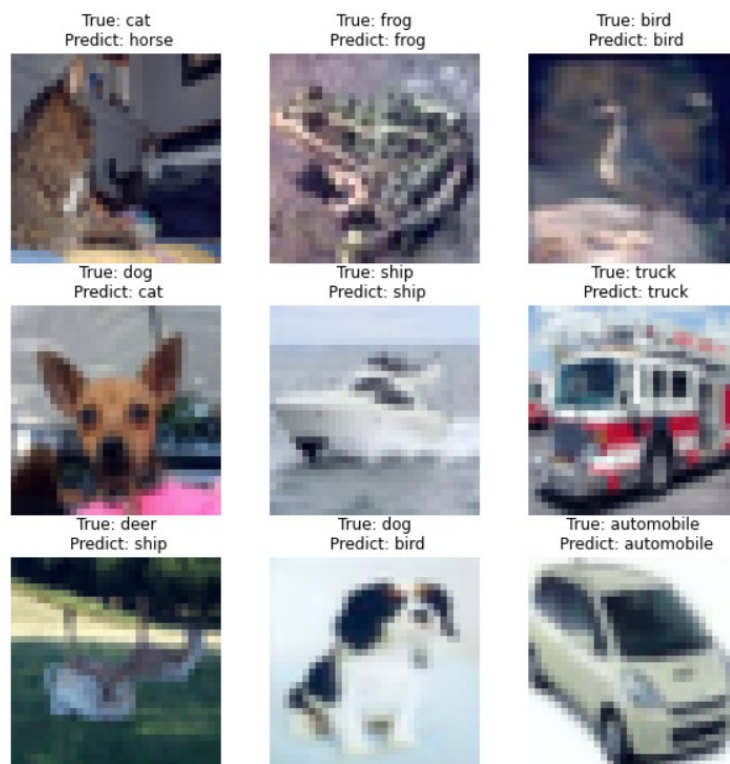
调整前



调整后

效果展示

- 最后，读取测试集图片进行测试，输出预测结果和原始标签。



本章总结

- 本章节介绍图像分类任务及评估标准，进而介绍了图像分类常用模型的架构，并利用该模型完成CIFAR-10分类任务。

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

**Copyright©2021 Huawei Technologies Co., Ltd.
All Rights Reserved.**

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

