

# 连接主义模型总结

---

## 神经网络

---

- 最早的神经元模型--MP模型
- 感知器模型
  - 无法解决线性不可分；
  - 即使少量样本也可能导致不收敛；
  - 无法利用单一感知器解决XOR问题（可用三层感知器）
- (BP)前馈神经网络
  - 非线性激活单元；
  - 输入层，隐含层，输出层；
  - 全连接结构；
  - 连接是单向的，传播是双向的（信息前向传输，误差反向传输）；
  - 链式法则；
  - 是一个任意函数的**通用近似器**
- Hopfield
  - 单层网络，全连接，固定参数
  - 两种工作方式，异步（串行）和同步（并行）
  - 每个神经元即是输入也是输出，每个神经元的输出结合其他神经元的输出和自身的输入
  - 引入能量函数概念，在系统运行过程中，能量不断减少，最终处于最小值
  - 离散型适用于联想记忆
  - 连续型适用于优化
  - 连续型较离散型的激活函数，神经元状态，工作方式（连续是同步工作）有不同
- 循环神经网络

- 包括输入层，隐含层，输出层
- 有折叠和展开两种表示方式
- 适用于序列化特征而设计的网络，具有**短期记忆能力**
- 可处理任意长度的序列
- 参数共享（每次迭代过程中，循环节点使用相同的权重系数处理所有时间步）
- 通用计算能力：按精度逼近任意一个**非线性动力系统**
- 单向性局限
- 长序列的记忆丢失

## 深度学习

---

### 深度学习的生物学依据

- 局部感受野
- 方向选择性细胞
- 视觉系统的信息处理是分级的

### 深度学习的目标

学习层次化的特征分布和表达

### 深度卷积神经网络的构成

- 卷积
- 池化层
- 非线性激活单元
- 归一化层
- 全连接层

卷积层

- 局部算子

- 平移同变性
- 特征增强
- 降噪

$$O = \left\lceil \frac{I-k+2p}{s} \right\rceil + 1$$

### 池化层一般不训练学习参数

- 局部变化的不变性
- 增大感受野
- 降维
- 防止过拟合

## 卷积网络的优势

---

- 局部连接-参数减少
- 参数共享-参数再减少

## CNN例题

假设输入是一张 $128 \times 128$  的RGB 彩色图像，如果网络第一个卷积层的卷积核为 $3 \times 3$  的大小，步长为1，共有64 个卷积核后接一个步长为2 的 $2 \times 2$  平均池化层，随后再接两层核大小为 $5 \times 5$ 、步长为1、卷积核个数为128 的卷积层以及一层步长为2 的 $2 \times 2$  的平均池化层，最后接含有128 个神经元的全连接层和一个含有10 个神经元的输出层。试回答：

(1) 通过平均池化层后特征图的大小( $H \times W \times C$ ) 为多少？

(2) 计算该网络的参数量。

(3) 网络的参数量主要由哪部分贡献？贡献了百分之多少？（ $3 \times 3$  卷积的padding 为1， $5 \times 5$  卷积核的padding 为2）。

- 考虑偏置

1. 通过平均池化层后特征图的大小 (H\*W\*C) 为多少?

解:  $64*64*64$

2. 计算该网络的参数量

解: conv-1 参数:  $3^2*3*64+64=1792$

conv-2 参数:  $5^2*64*128+128=204,928$

conv-3 参数:  $5^2*128*128+128=409,728$

FC-1 参数:  $32^2*128*128+128=16,777,344$

FC-2 参数:  $128*10+10=1,290$

总参数:  $1792+204928+409728+16777344+12190=17,395,082$

3. 网络的参数量主要有由哪部分贡献? 贡献了百分之几?

解: 主要由全连接层贡献, 贡献了:

$$\frac{16,777,344}{17,395,082} * 100\% = 96.45\%$$

- 不考虑偏置

- 经过第一个平均池化层后，特征图大小为  $64 \times 64 \times 64$ ;  
经过第二个平均池化层后，特征图大小为  $32 \times 32 \times 128$ 。
- 该网络结构如下表所示。在不考虑 bias 的情况下，对于第一个卷积层，参数量为

$$3 \times 3 \times 3 \times 64 = 1,728 \quad (25)$$

同理，之后层参数量为

$$5 \times 5 \times 64 \times 128 = 204,800 \quad (26)$$

$$5 \times 5 \times 128 \times 128 = 409,600 \quad (27)$$

$$32 \times 32 \times 128 \times 128 = 16,777,216 \quad (28)$$

$$128 \times 10 = 1,280 \quad (29)$$

则参数总个数为 17,394,624。

| Type/stride | Filter Shape                         | Input Size                 |
|-------------|--------------------------------------|----------------------------|
| Conv/s1     | $3 \times 3 \times 3 \times 64$      | $128 \times 128 \times 3$  |
| Avg Pool/s2 | Pool $2 \times 2$                    | $128 \times 128 \times 64$ |
| Conv/s1     | $5 \times 5 \times 64 \times 128$    | $64 \times 64 \times 64$   |
| Conv/s1     | $5 \times 5 \times 128 \times 128$   | $64 \times 64 \times 128$  |
| Avg Pool/s2 | Pool $2 \times 2$                    | $64 \times 64 \times 128$  |
| FC/s1       | $32 \times 32 \times 128 \times 128$ | $32 \times 32 \times 128$  |
| FC/s1       |                                      | 128                        |
| sigmoid/s1  | Classifier                           | $1 \times 10$              |

- 参数量主要由全连接层贡献，贡献计算如下：

$$(16,777,216 + 1280) \div 17,394,624 = 0.9446 \quad (30)$$

## 典型CNN模型

- LeNet
  - 平均池化
  - sigmoid or tanh 非线性激活单元
  - 全连接层用于分类
- AlexNet
  - 8层模型
  - ReLu激活函数
  - 正则化

- GPU实现
- VGGNet
  - 更深的模型（16 or 19层）
  - 连续3\*3的卷积
- GoogLeNet
  - Inception模块（1×1, 3×3, 5×5的不同感受野分支用于捕获）
  - 在卷积计算前使用1×1的卷积降低特征通道维数
- ResNet
  - 152层
  - 跳连解决深度网络变深后难以优化的问题

## 基础GAN存在的问题

---

- 生成模型的分布无显示表达
- 比较难训练：生成器和判别器间需要很好的同步

## 典型GAN模型

---

- Deep Convolutional GAN
  - 将CNN与GAN结合
  - 转置卷积进行上采样，判别器中使用步长卷积代替池化层
  - 生成器和判别器都采用了批归一化操作
  - 去掉了全连接层，使用全局池化层代替-网络成为全卷积网络
  - 生成器使用ReLU作为激活函数，输出层使用Tanh激活函数
  - 判别器使用LeakyReLU激活函数
- WGAN
  - 经典距离度量方式，解释了GAN训练不稳定的原因

- 去掉判别器最后一层的sigmoid
  - 损失函数不取log
  - 更新判别器的参数后，截断在某个范围内
  - 不使用基于动量的优化算法
  - CGAN
    - 在判别器和生成器中同时加入条件约束引导数据生成过程
  - Pix2Pix
    - 使用成对的数据进行训练
  - CycleGAN
    - 使用不成对的数据即可进行训练
  - PGGAN
    - 渐进分辨率，渐进训练
-