填空题（常识类）15\*2

基础\*14

mindspore\*1

* 深度学习成功的必要前提包括，足够多的数据、足够强大的计算能力和设定好初始化权值
* 根据学习方式的不同，机器学习方法主要可分为监督式学习、无监督式学习、半监督学习和强化学习四种
* 模型的容量是指模型拟合各种模型的能力
* 常用的两种自动超参数优化算法，分别是网络搜索和随机搜索
* 词嵌入是一种将文本中的词转化为数字向量的方法
* 图像分类算法解决了图像中包含着什么类别的物体的问题
* Cell是mindspore核心编程结构，定义了执行计算的基本模块。
* 打乱(shuffle)操作用来打乱数据集中的数据排序，越大的buffer\_size意味着越高的混洗度，但也意味着会花费更多的时间和计算资源
* 张量mindspore中最基础的数据结构是，他是计算的基础，也是权重和偏置以及特征图的载体
* 深度神经网络的开发流程包括数据处理、模型定义、模型训练、模型推理等过程
* 通过数据集增强、改变输入特征的数量和加入特征对应的参数，可以改变模型的容量
* 深度神经网络的超参数包括：学习率（影响学习速率）、模型深度（影响模型的泛化能力）、Batch size等
* 使机器能够根据外界环境的反馈来进一步完善自身的算法被称为强化学习
* mindspore中常用的算子（operation）可分为Array相关算子arry、数学计算相关算子math、网络类算子nn、控制类算子control
* 深度学习模型训练数据处理pipeline包括加载、打乱、map、分批、重复
* 目标检测是将图像或者视频中目标与其他不感兴趣的部分进行区分，判断是否存在目标，确定目标的位置，识别目标的种类的一种计算机视觉任务
* 图像描述不仅能够识别图像中的物体类别，理解物体之间的关系，并且能够让计算机生成自然语言来流畅、准确地描述图像地主要内容
* 词嵌入（Word Embedding）是一种将文本中的词转换成数字向量的计算方法
* 目前的注意力机制主要分为以下三类：空间注意力、通道注意力和高层语义注意力

简答题6道题（70分）

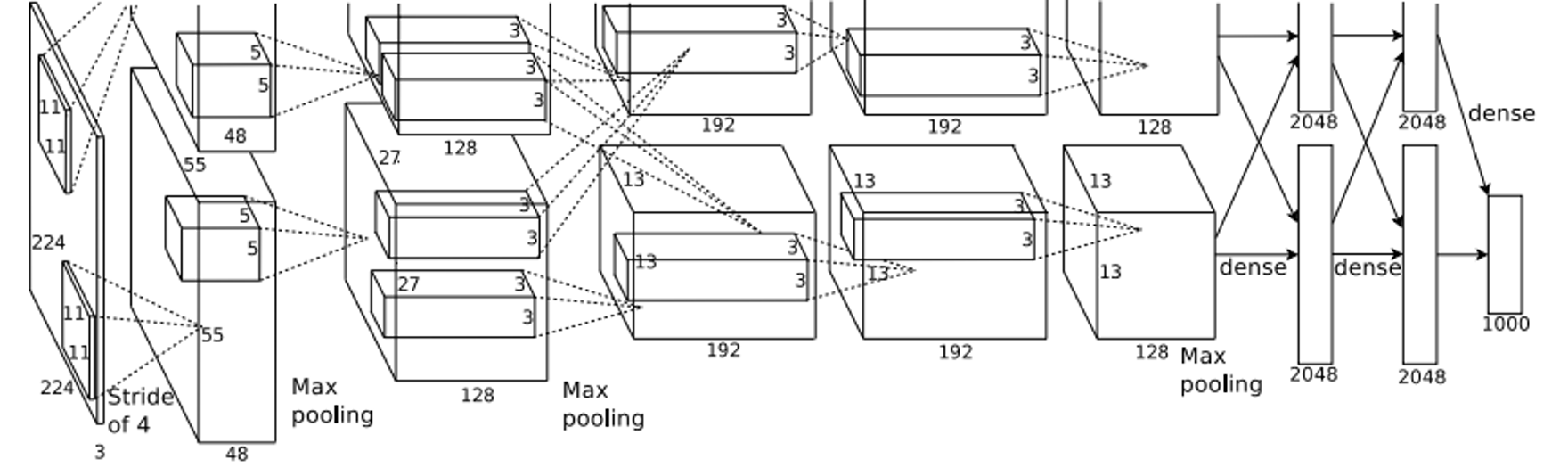
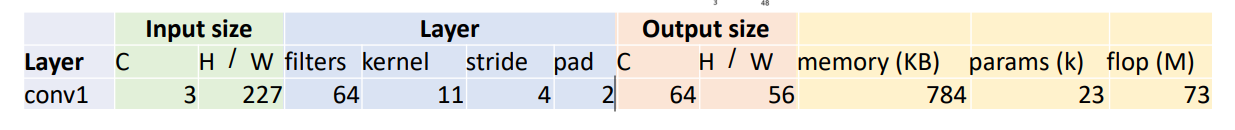
1.cnn参数flops（浮点运算次数）估算

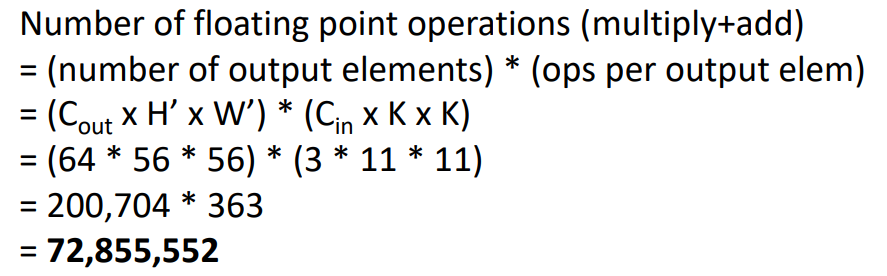
CNN的组成：卷积层（Convolution Layers），池化层（pooling layers）、全连接层（Fully-Connected Layers）、激活函数（Activation function）、归一化

激活函数：把卷积层输出结果做非线性映射。CNN采用的激活函数一般为ReLU(The Rectified Linear Unit/修正线性单元)，它的特点是收敛快，求梯度简单，但较脆弱

池化层：夹在连续的卷积层中间， 用于压缩数据和参数的量，减小过拟合。  
简而言之，如**果输入是图像的话，那么池化层的最主要作用就是压缩图像。**

FLOPs=输出通道数×输出宽×输出高×池化窗口大小=Cin\*K\*K\*Cout\*H’\*W’

以AlexNet为例，计算网络中的FLOPs（Number of floating point operations (multiply+add)）：  
  




！池化层中没有可学习的参数

输出参数（output elems）= Cout \* H’ \* W’

卷积层浮点计算量（Conv FLOPs，bias为1）=Cin\*K\*K\*Cout\*H’\*W’

池化层的浮点运算量（Floating-point ops for pooling layer）=(Cout \* H’ \* W’) \* (K \* K)

扁平化参数量（Flatten Params）=Cin\*Cout+Cout

扁平化输出尺寸（Flatten output size） = Cin \*H \* W

扁平化浮点计算量（Flatten FLOPs）=Cin\*Cout

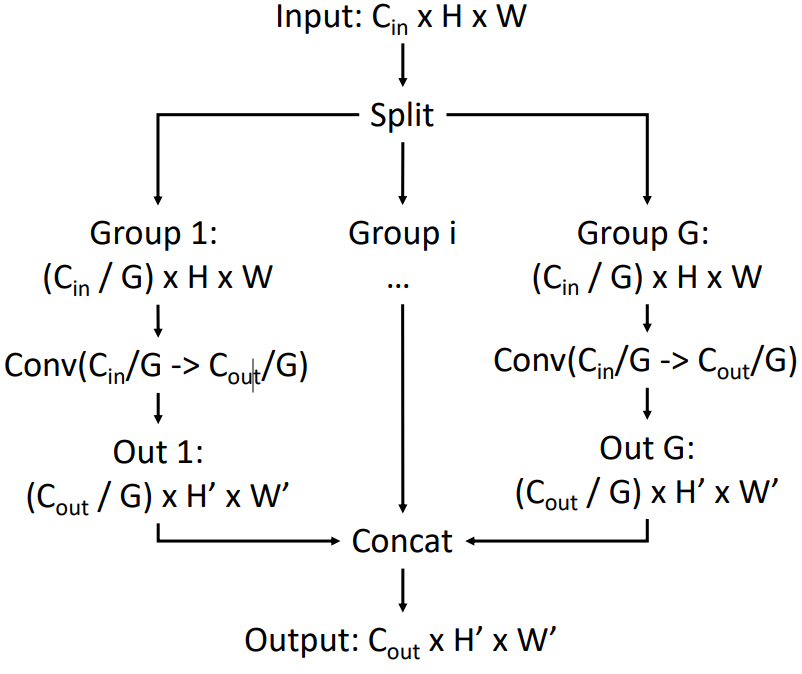
全连接层参数（FC params） = Cin \* Cout + Cout

全连接层浮点计算量（FC FLOPs ）= Cin \* Cout

！大部分内存用在卷起层中，几乎所有参数都在全连接层中，大部分的浮点运算都发生在卷积层中

**可分离卷积（Group Convolution）**

将输入通道分为G组，每组包含Cin/G个通道；将filter分为G组每组look at输入Cin的子集，输出的每个平面取决于一个滤波器和输入通道的子集。



参数计算：  
Input: Cin x H x W

Split to G x [(Cin / G) x H x W]

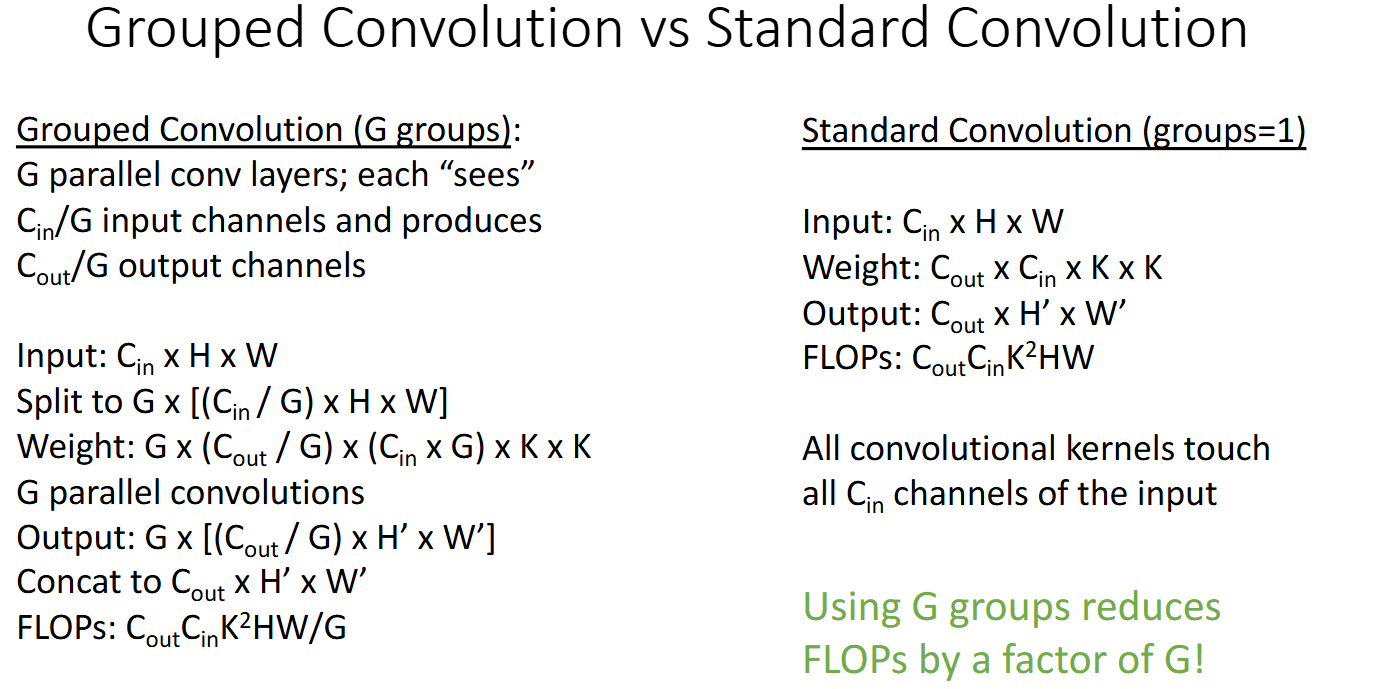
Weight: G x (Cout / G) x (Cin / G) x K x K

G parallel convolutions

Output: G x [(Cout / G) x H’ x W’]

Concat to Cout x H’ x W’

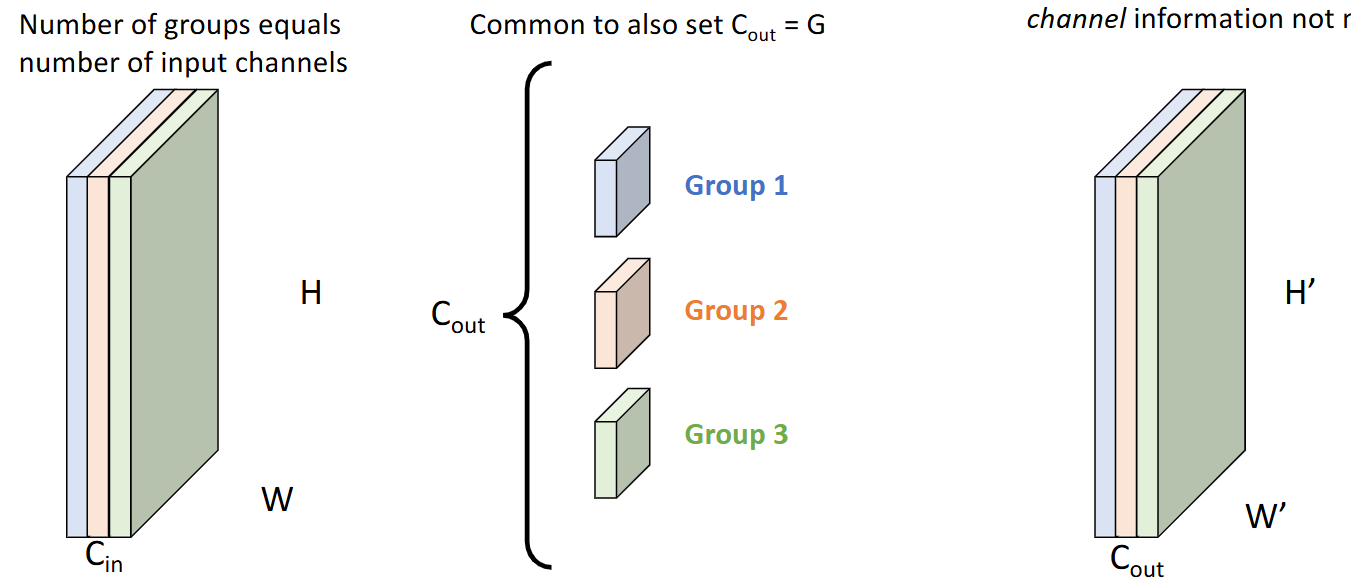
FLOPs: Cout\*Cin\*K\*K\*H\*W/G

对照图如下：  


**可分离卷积的特殊情况——深度卷积（Depthwise Convolution）**

组数等于输入通道数；Cout=G；输出仅混合来自输入的空间信息，频道信息不混合

在以上基础上，每组filter中还可以由多个的filter组成，etc.每组有两个filter，则Cout=2Cin



课件9

循环神经网络RNN

2.mAP的计算

3.自注意力计算，写出公式，算出值

4.bptt梯度流图梯度计算（代码链接：）

H\*，梯度递推式

5.梯度下降优化算法（adam、Adagrad）

6.第一种，给出forward写出backward，

第二种，给python代码，读代码把计算图画出来（输入输出功能），加上文字描述