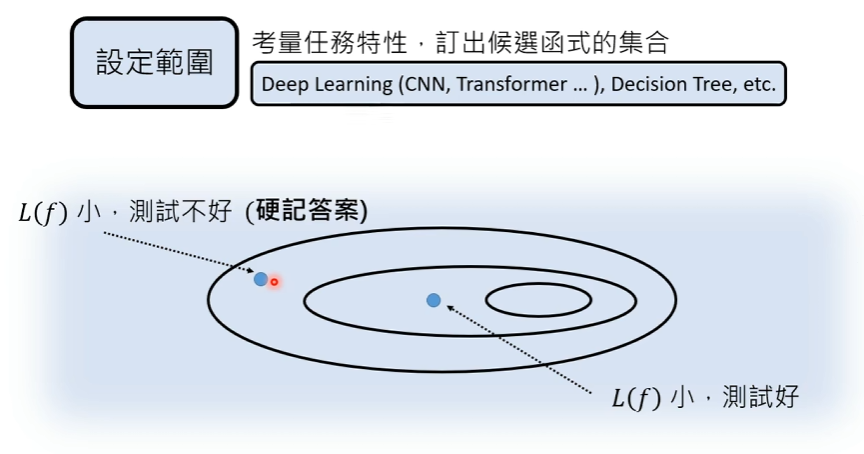
1. **机器学习**

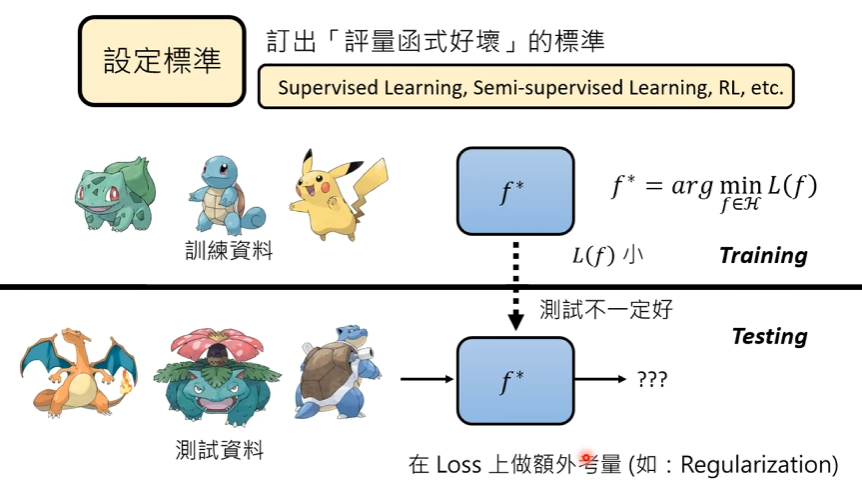
* 回归：函式的输出是一个数值
* 分类：函式的输出是一个类别（选择题）
* structured learning 又叫生成式学习：生成有结构的物件（例如：影响、文句），ChatGPT实际是分类，使用者感受到的是生成式学习，精确来讲就是把生成式学习拆解成多个分类问题.

1. **机器学习找出函式的三个步骤：（倒着来3-2-1）**
2. 设定范围（H）：定出候选函式的集合（model ），范围适中，训练资料少划的越大，深度学习中类神经网络的结构指的就是不同的候选函式集合

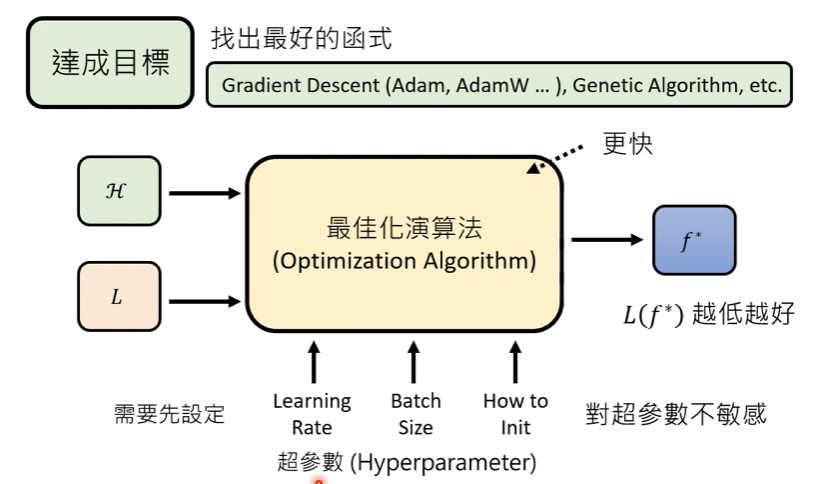
Deep Learning（CNN，Transformer...）,Decision Tree,etc.

1. 设定标准（L）：定出评量函式好坏的标准（loss）  
   L（f）=根据训练资料（监督学习）的差距来评判+相似物的差距

Supervised Learning,Semi-supervised Learning,RL,etc.

1. 连成目标：找出最好的函式——最佳化：

Gradient Descent(Adam,AdamW...),Genetic Algorithm,etc.



1. **生成式学习的两种策略：**

* 各个击破（AR）：每一个字（像素）都需要等前面的字（像素）生成、生成品质好，例如：文字
* 一次到位（NAR）：只要有足够的的平行运算能力，就可以很快、生成品质差，例如：影像

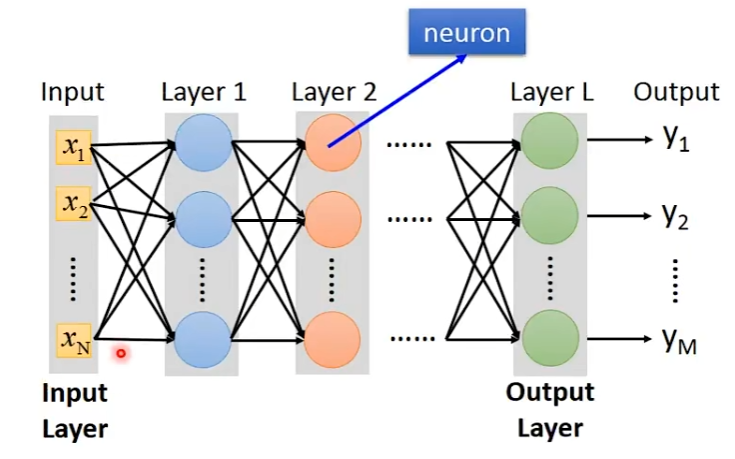
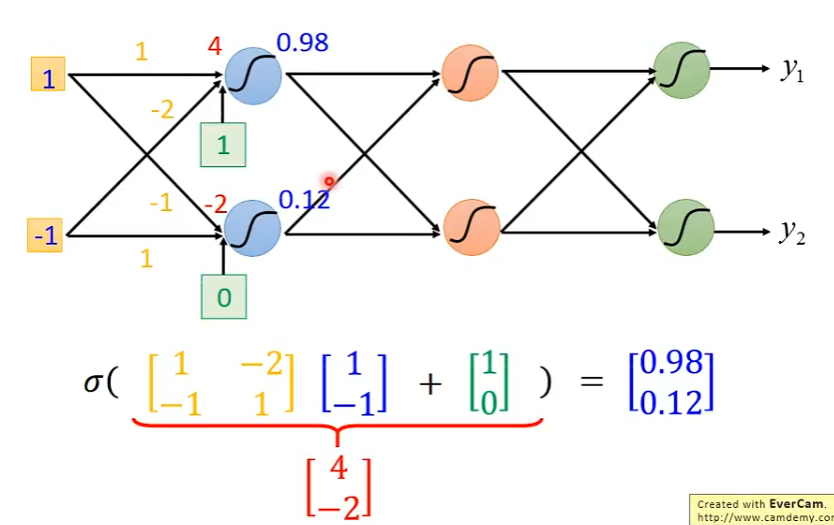
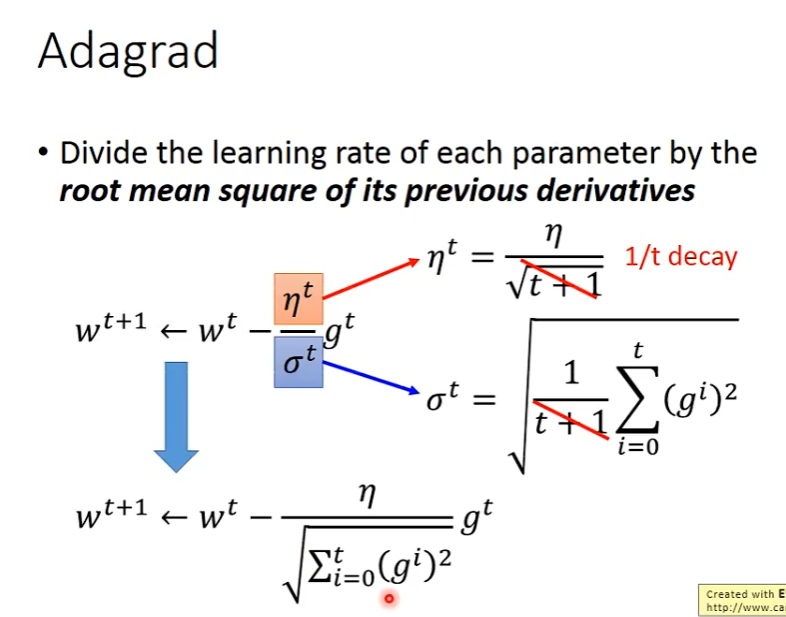
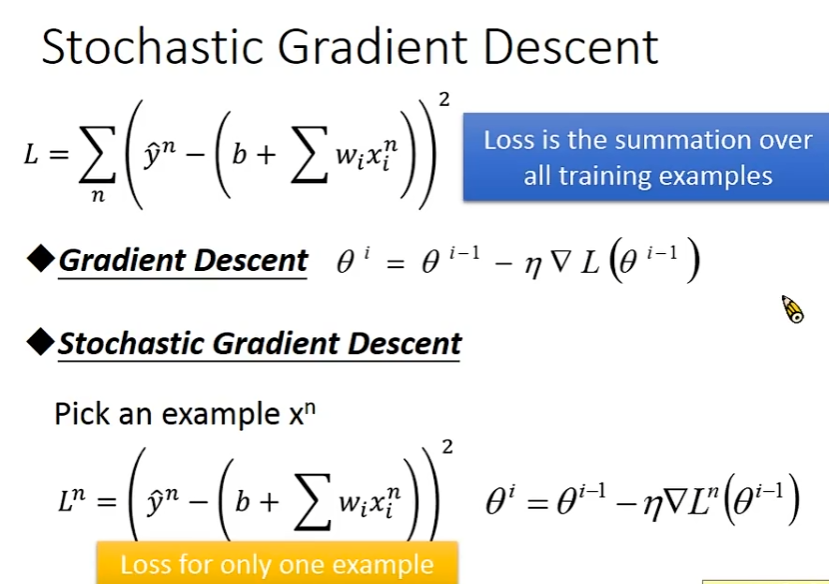
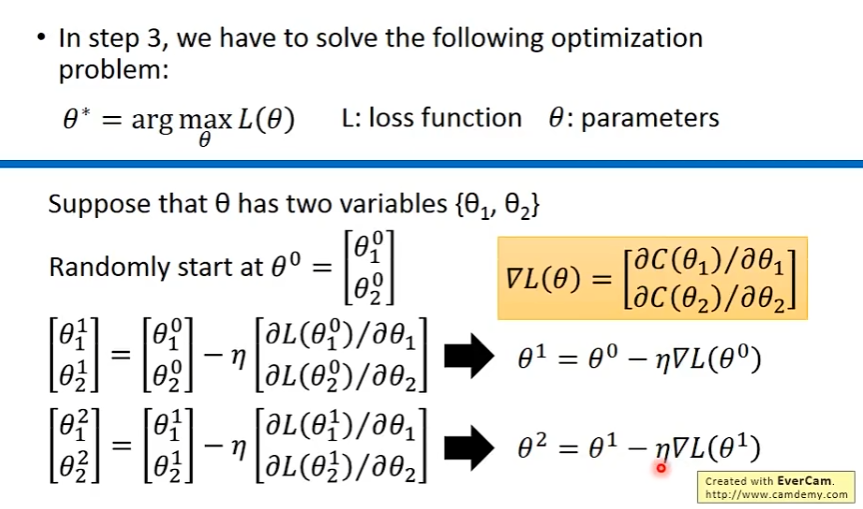
注：各个击破+一次到位：例如语音，先各个击破再一次到位  
n次到位：影像

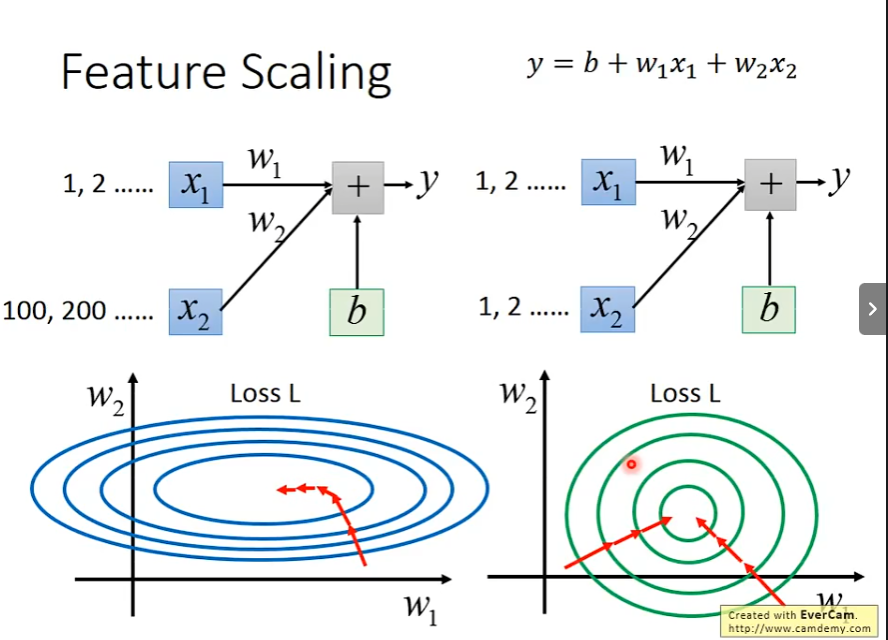
1. **能够使用工具的AI：**

New Bing：会进行网络搜寻（由自己决定）

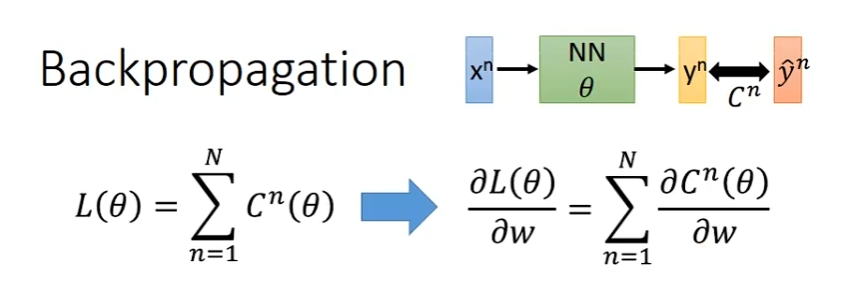
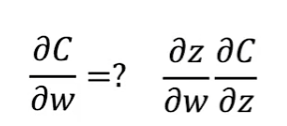
WebGPT：使用搜寻引擎的GPT （使用搜寻引擎也是文字接龙）

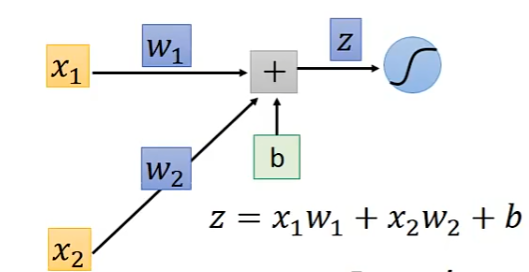
Toolformer：使用多重不同的工具（使用工具就是文字接龙）

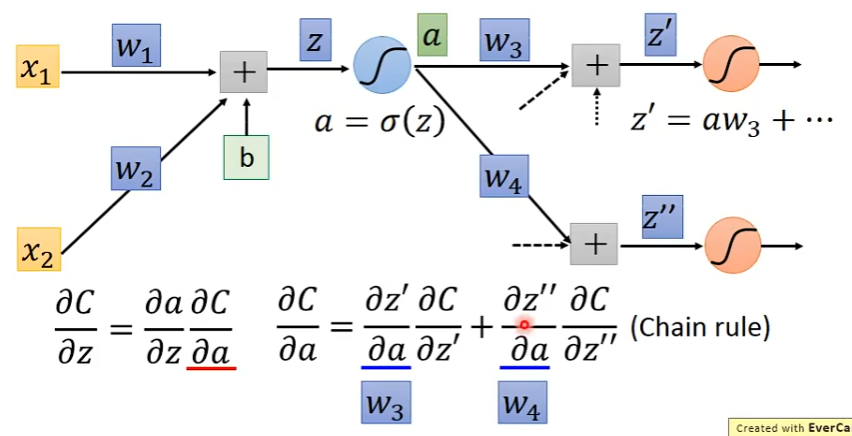
1. **深度学习：**
2. ****定义一个函式（Neural Network），Neural Network的各个Neural需要连接，常见的：Fully Connect Feedforward  
    deep=many hidden layers
3. 定义一个函式的好坏：差值和
4. 找出最好的函式：Gradient Descent
5. **Gradient Descent：**
6. IMG_258IMG_257  
     
   需要设置一个learning rate （合适），使loss收敛在最低，  
   eg：Adagrad使用了这个
7. Stochastic Gradient Descent（更快）：（每一个参数，L（f）一次）
8. Feature Scaling：

  
将数据标准化

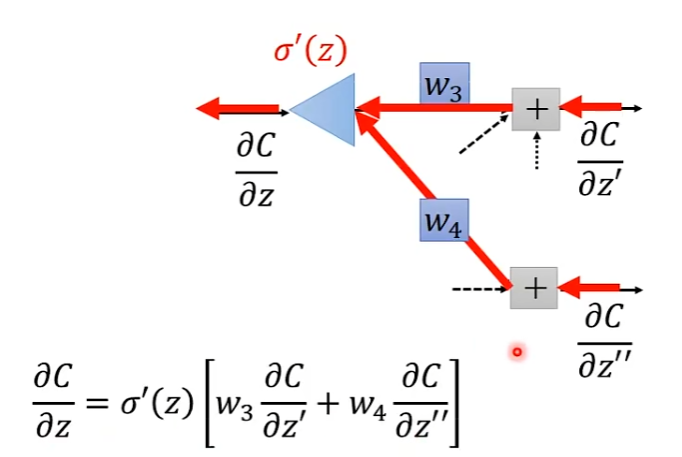
1. **Backpropagation：链式法则**

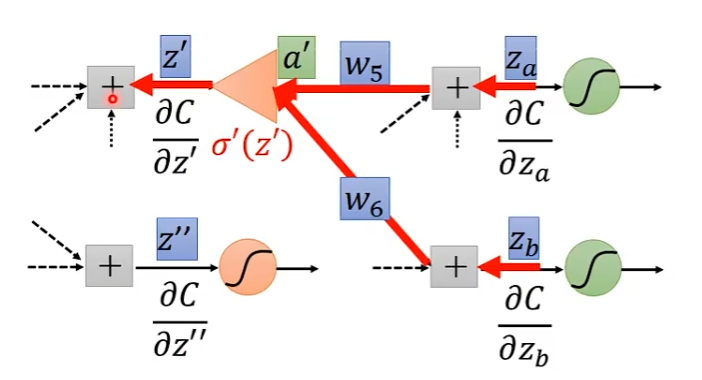
步骤：

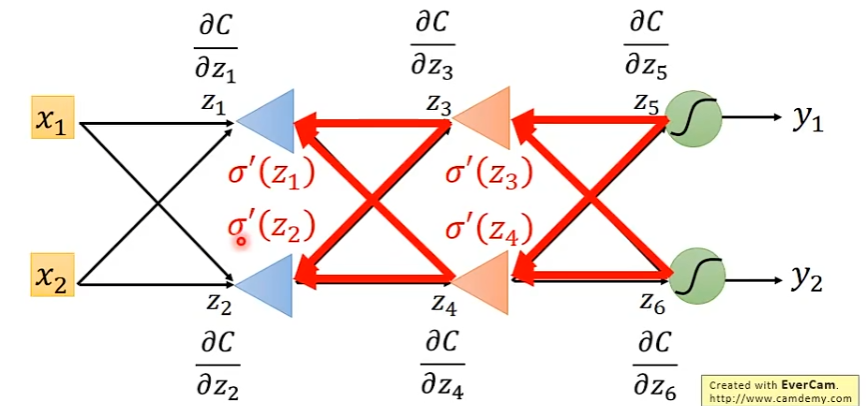
1. 



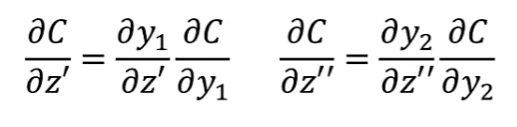
等同于

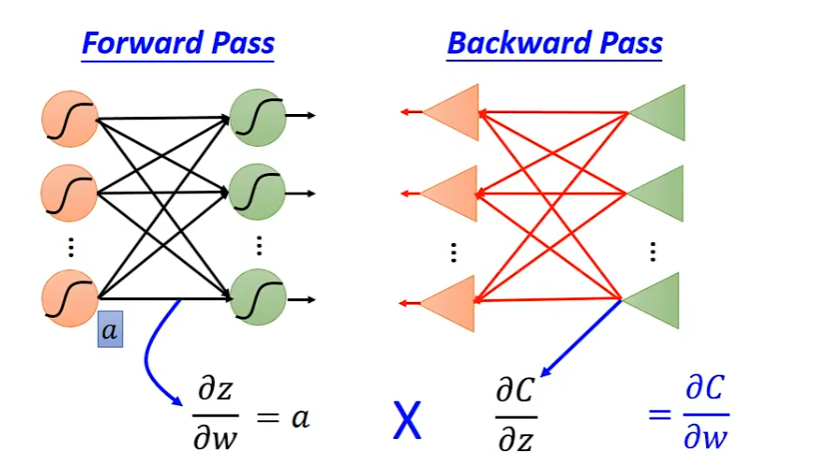






（倒推）……以此类推，最后输出时。



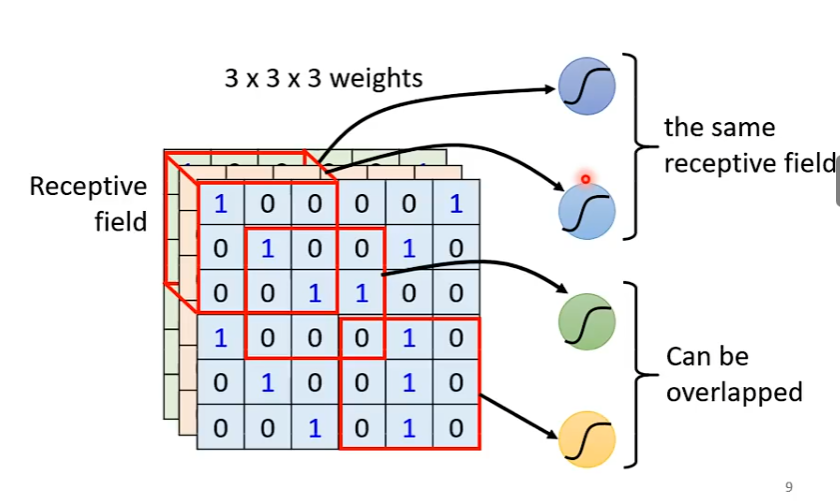
总结：

1. **CNN：**
2. 第一个简化：

影像RGB3色的数据组成向量，但是影像只输入部分图片，会设置一个Receptive Field（侦测范围）作为一个参数，一个Receptive Field有多个neuron。

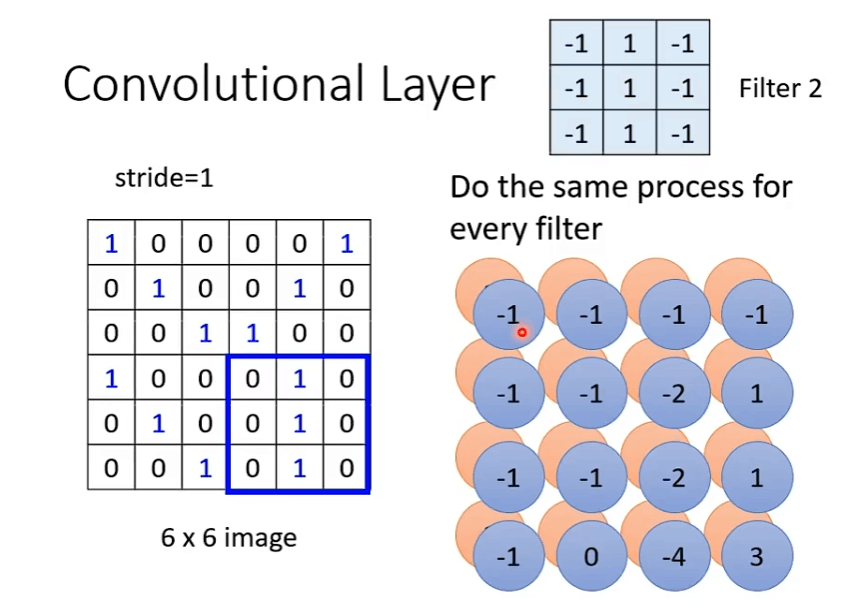
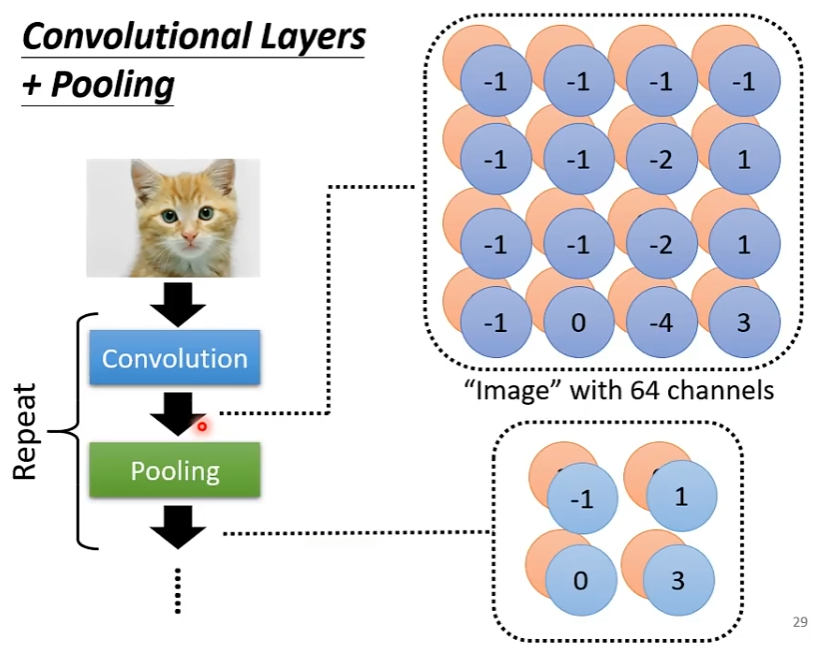
Receptive Field的经典安排方式：看全部的channels（深度），kernel size（高和宽，eg：3\*3）。

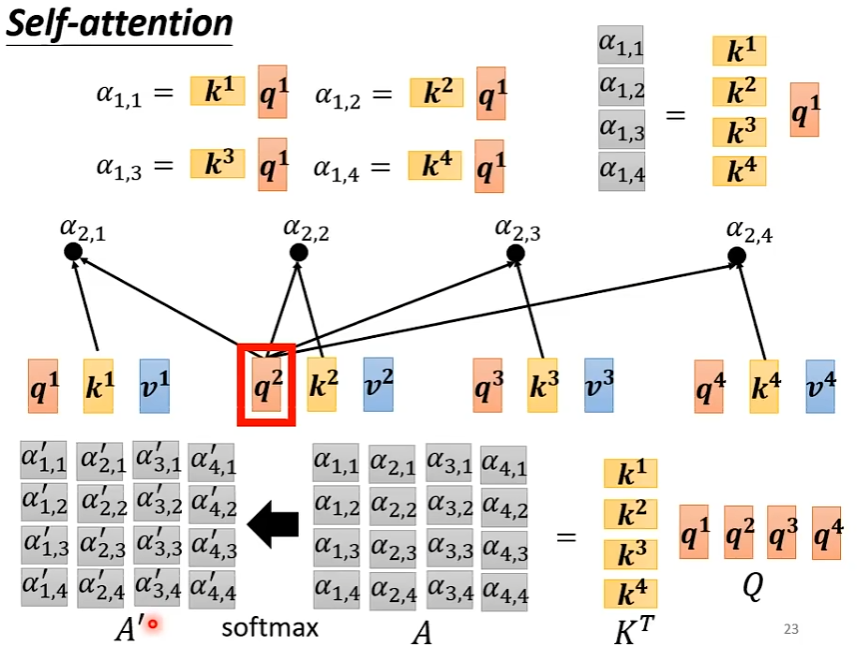
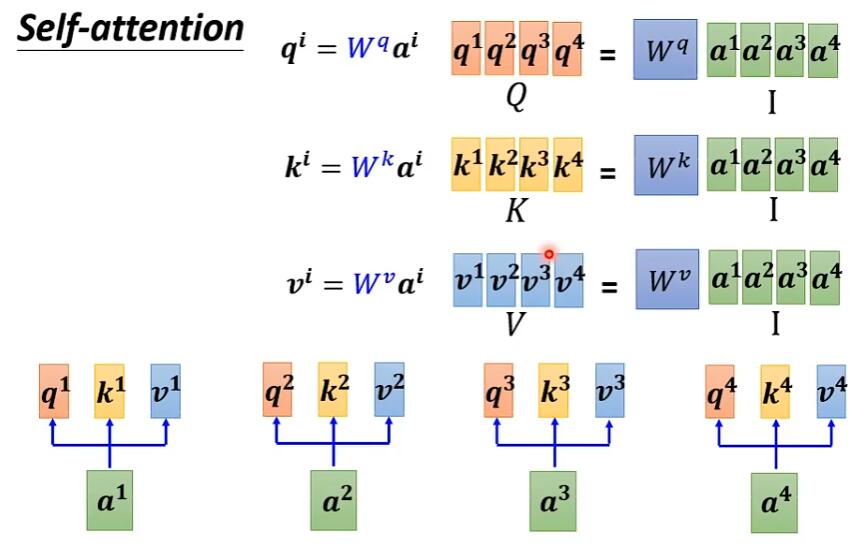
其他的Receptive Field的产生：移动（stride=2移动2个），如果超出边补0

1. 第二个简化：

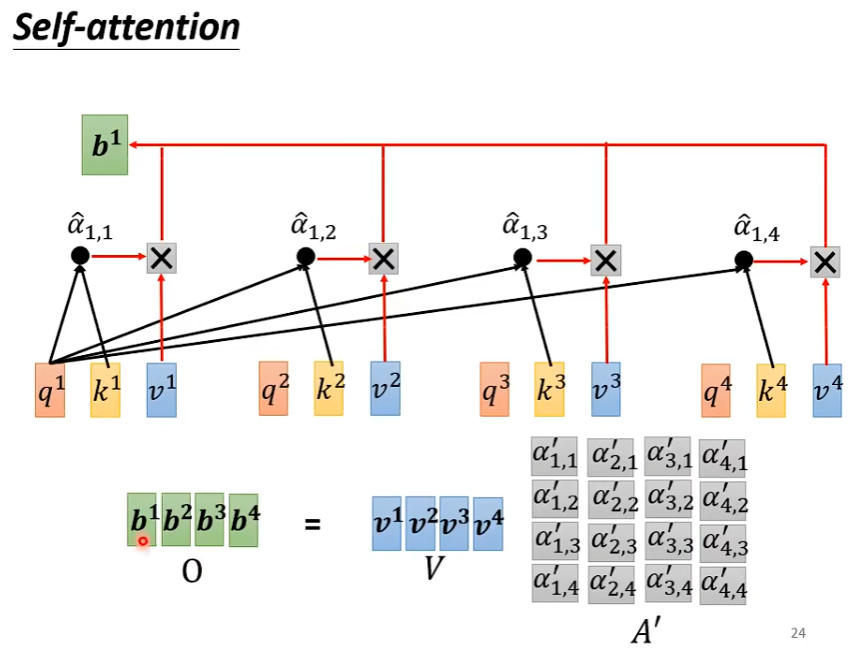
不同的Receptive Field共享参数（w相同），但是输入不一样  
  
做了2个简化就是CNN

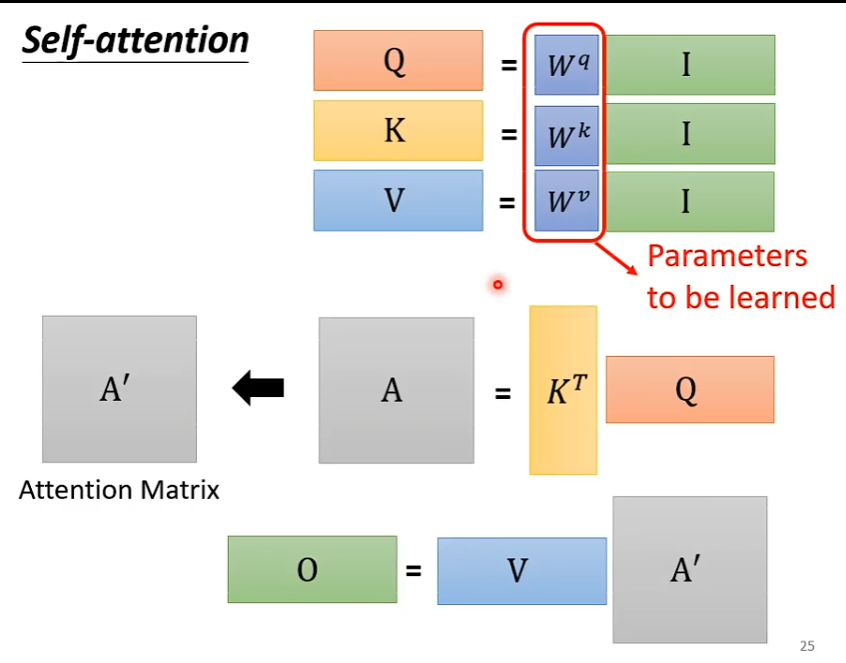
**另一种理解方式：**

通过Filter（给出了）产生channels  
  
如果有64个fileter就产生64个channels，图片就变成了64\*n\*n，然后pooling（将上一步产生的图片channels不变，分组(eg:2\*2一组)来减少运算量）  
  
，然后拉直形成参数。

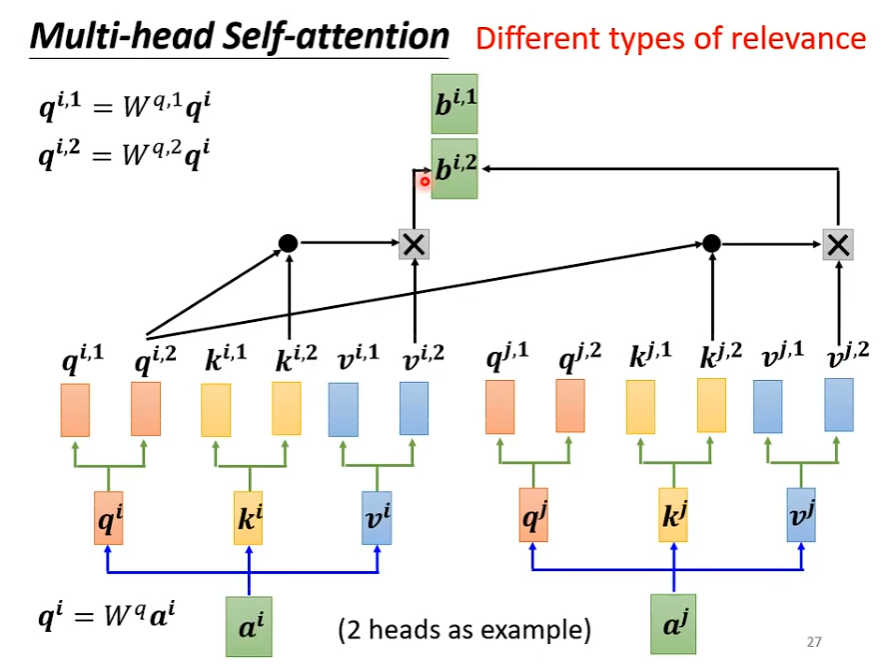
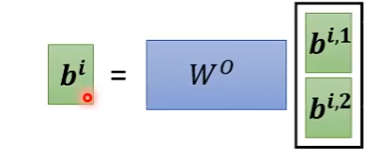
1. **Self-attention**

（W为矩阵）

****总结：

****

1. **Multi-head Self-attention**

****（两个head）然后  
每必要看所有的参数，看部分

CNN 是简化的 self-attention

cnn：适合训练资料少的  
self-attention:适合训练资料多