# 第一章 深度学习模型的基本结构

深度学习属于神经网络模型的一部分。神经网络模型属于数据模型，即从数据中学习到信息（数据的模式、数据的样本与标签之间的关系）。

神经网络模型以全连接网络（前馈神经网络）为基本结构，并对不同的具体任务，设计出相应的特殊的结构。

神经网络模型的两个重要特点是：端对端，即输入是我们能通过简单方法获得的样本，输出是我们要得到的结果；模型参数通过学习获得，即我们希望模型表现出某些行为，但这些行为不是预先设定好的，而是模型通过学习而自动获得的（比如CNN学到每个filter的参数，用来提取特征；Attetion机制学到attention score，用来施加注意力）。

第一节 全连接网络（fully-connected）

全连接网络，也称为前馈神经网络，是深度学习模型的最基本的形式。

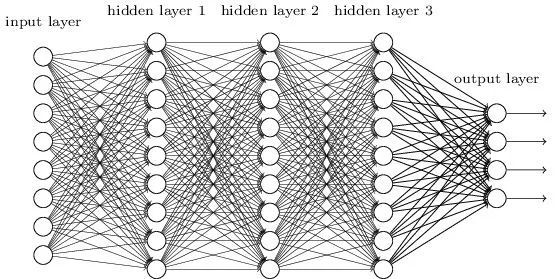


图1 全连接网络的结构

第二节 CNN及其变种

1. CNN的基本结构

（1）CNN的结构：CNN主要用于图片等网格结构的处理，也能用于文本等序列结构的处理。CNN的基本结构（block）是一个卷积层。一个卷积层包括一个卷积操作和一个池化操作，卷积操作后面一般还要接一个ReLU激活函数；池化操作则是用Max或Mean的方式进行下采样。卷积层的作用在于提取图片的特征。在搭建模型时，卷积层可以stack多次，每一个卷积层在上一个卷积层的基础上提取特征，越高层的卷积层提取的特征越抽象。



图2 CNN的结构（李宏毅）

（2）卷积操作：卷积操作就是用一个卷积核（filter，就是一个小矩阵或小tensor）跟输入的图像或者上一层输出的feature map进行滑动求“内积”。卷积操作的超参数是卷积核个数（也就是输出通道个数）、卷积核大小和滑动步长stride。卷积核中的数值都是可学习的参数，它们是通过训练而学习到的。

（3）卷积操作与前馈神经网络的关系：实际上，卷积操作就等价于稀疏连接的前馈神经网络，这种网络的特点是“稀疏连接”和“参数共享”。

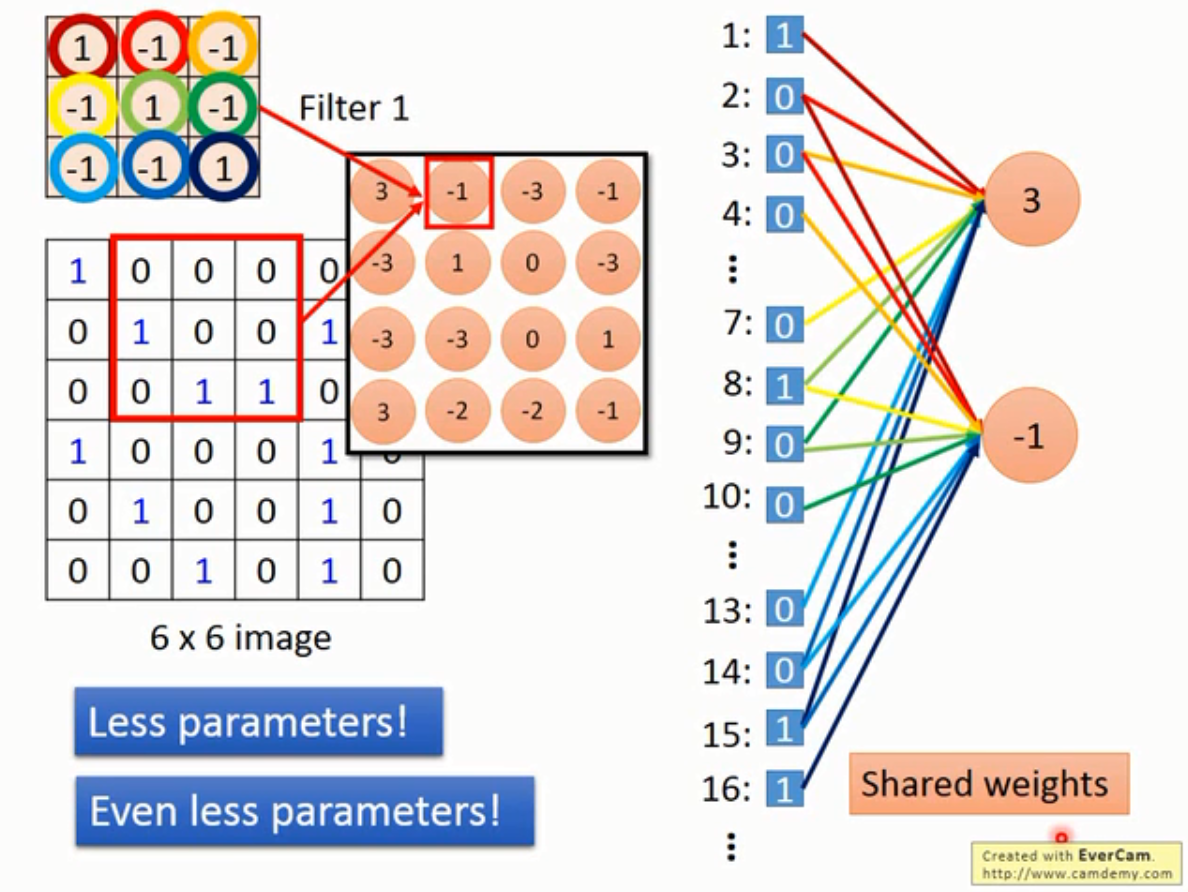


图3 CNN操作等价于稀疏连接的前馈神经网（李宏毅）

（4）通道（channel）：CNN中的一个重要概念是“通道”，它是指网络的输入或者上一层输出到下一层的feature map不一定是一个矩阵，而还可能是一个张量（多个矩阵）。这里矩阵的个数就是通道数。比如输入的图像如果是彩色图像，则有RGB三个通道。

CNN中每一个卷积层可能有多个卷积核，卷积核的个数就是该卷积层的输出通道个数。

从通道的角度来看，卷积层之间可以看成是通道与通道的全连接。如下图（左）所示，为通道角度的全连接示意图，下图（右）为分组卷积，相当于通道角度的稀疏连接。我们后面会介绍分组卷积。

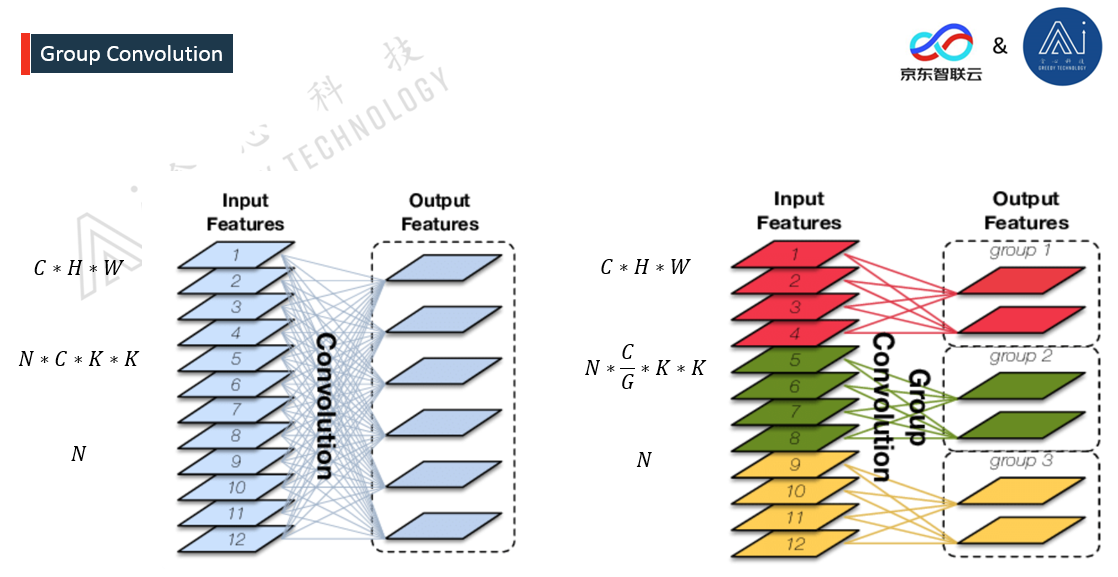


图4 通道全连接与通道稀疏连接

2. 各种形式的卷积

（1）分组卷积：AlexNet中使用了分组卷积，它是从通道的角度将卷积层之间的通道全连接变成通道稀疏连接，减少计算量并使计算能够方便地并行。

为了不让分组影响泛化性能，有时会在每次分组之前做shuffle。

（2）空洞卷积：不增加卷积核的大小，而提高感受野。

（3）小卷积核：VGG中使用多层小卷积核代替大卷积核的思路，增加感受野的同时减少计算量。

（4）1\*1卷积核：AlexNet中使用了1\*1卷积核，其相当于“通道方向的卷积”，其作用已经不是特征提取了，而是在不提取特征的基础上，增减通道数量，以减少整个模型的参数数量。

（5）全图卷积：即卷积核的尺寸跟feature map的尺寸一样。其作用相当于一种较大程度的池化。

（6）多尺寸卷积：同一个卷积层中可以使用不同尺寸的卷积核。 Inception Net、Text CNN中都使用了多尺寸卷积。

3. CNN的变种

（1）ResNet

ResNet并不只是针对CNN的改进。它使用skip-connection结构解决深层神经网络退化的问题。



图5 ResNet的基本结构

（2）DenseNet

DenseNet沿着ResNet的思路，走得更远一些。ResNet只是跳连（跳两层或三层），而DenseNet则是任意两层都连起来。

（3）InceptionNet

使用了1\*1卷积核和多尺寸卷积。

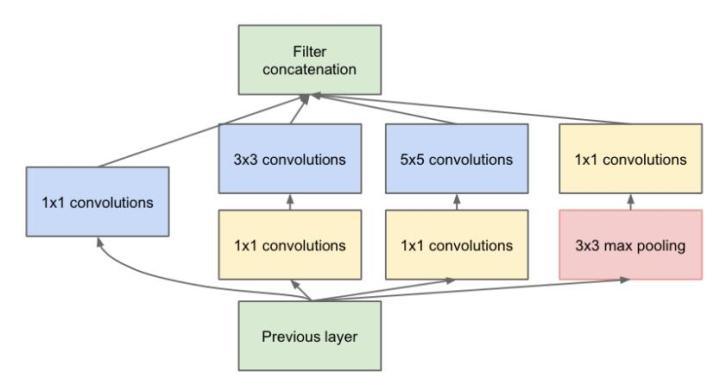


图6 Inception Net的结构

（4）SENet

类似于Attetion机制。通过学习获得上一层的每个channel的权重，然后在下一层的卷积操作时，考虑这些权重，使权重大的channel有更大影响。

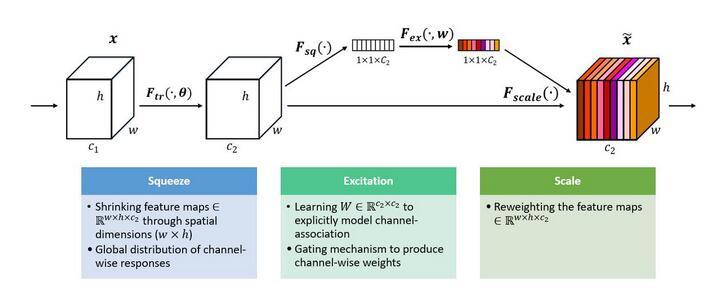


图7 SENet的结构

（5）胶囊网络

原始的CNN结构在处理图片时，对图片中元素的相对位置关系和图片的旋转不变性都不能很好地捕获到。因此提出了胶囊网络。

胶囊网络用“胶囊”来代替传统神经网络中的神经元，每个胶囊都是一个向量。对于分类任务，最终输出时，用向量的模来代表一个类别的概率大小。

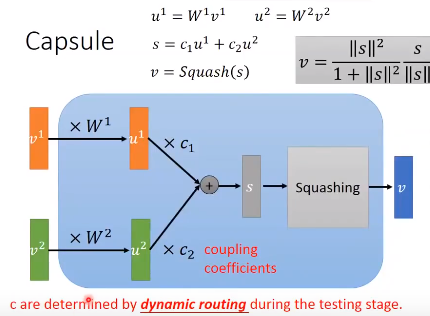


图8 胶囊网络的结构

胶囊网络不是用梯度下降的方法更新参数，而是使用“路由机制”。

（6）Siamese网络

第三节 RNN及其变种

1. RNN

RNN，循环神经网络，是专门为处理序列数据而设计的神经网络结构。RNN可以理解成一种带记忆机制的前馈神经网络，它会记住曾经输入过的数据，也就是说，它的每一时刻的输出，都与前面所有时刻的输入有关。

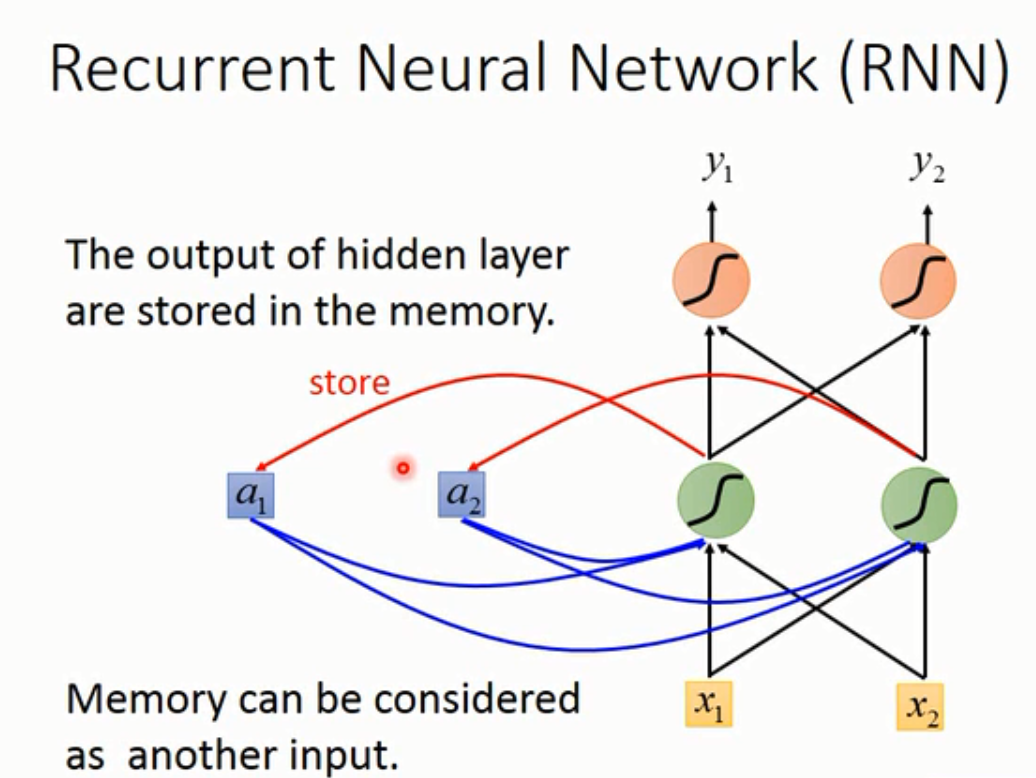


图9 RNN的结构（第一种表示方法，记忆单元）

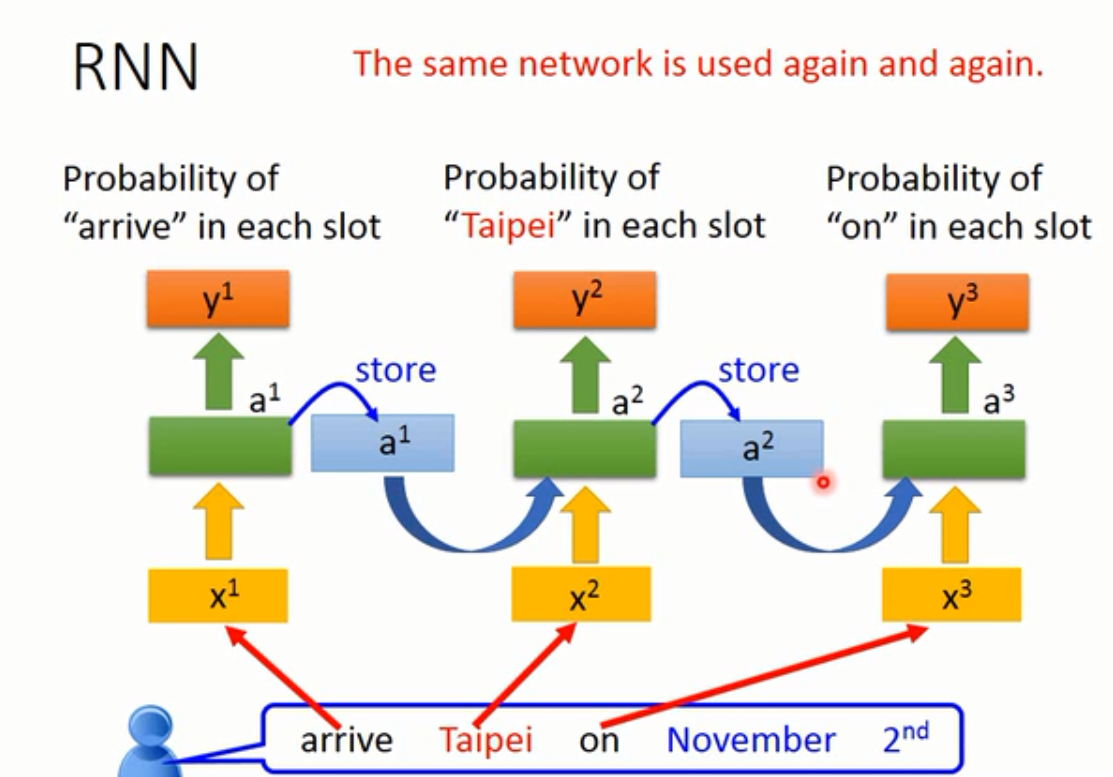


图10 RNN的结构（第二种表示方法，展开表示）

RNN的公式：

1. LSTM

LSTM是对RNN的记忆机制的一种改进。它使用“门机制”对RNN的记忆机制进行改进。

LSTM的公式：

遗忘门：

输入门：

输入（相当于RNN中的隐层）：

记忆单元：

输出门：

隐层输出：

1. GRU

GRU也是利用门机制对RNN的记忆机制进行改进，但GRU只有两个门。

GRU的公式：

重置门：

更新门：

候选隐层输出：

隐层输出：

1. Recursive Net

递归神经网络。RNN相当于Recursive的特例。

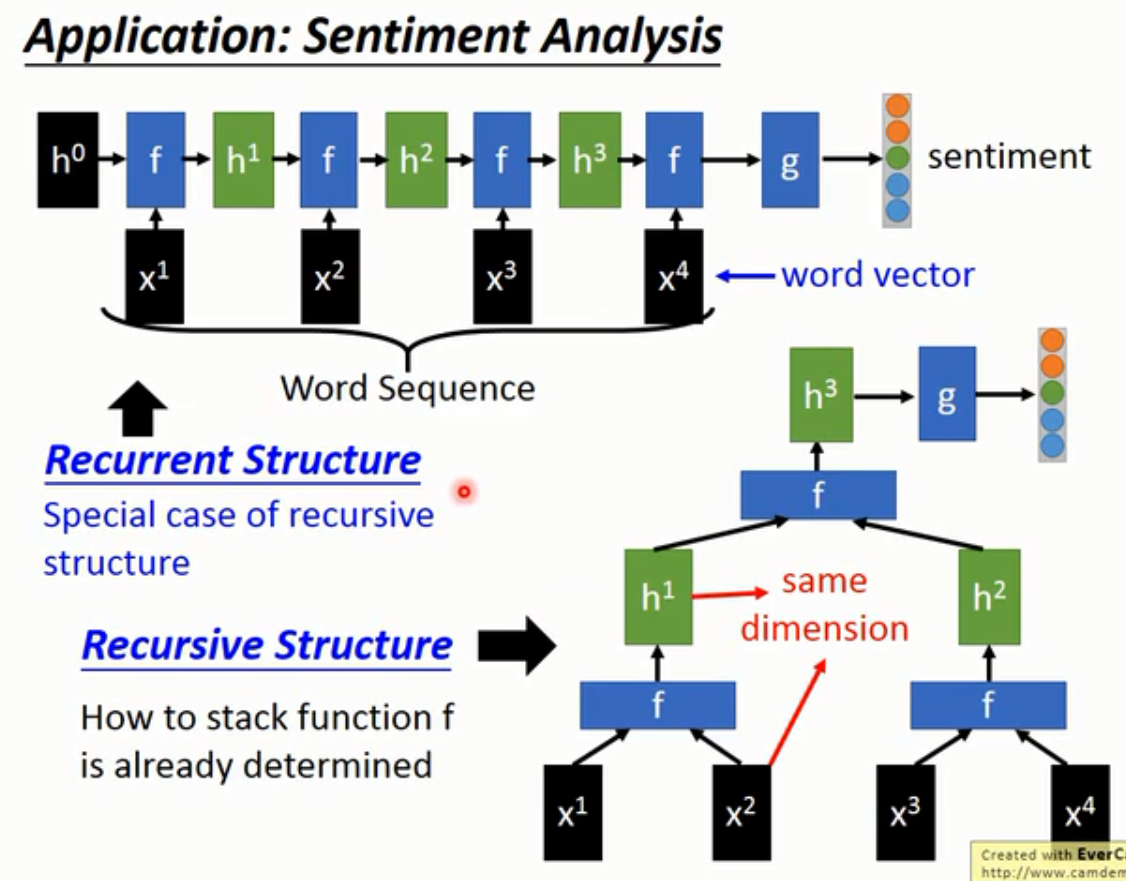


图11 Recursive和RNN结构对比

1. IndRNN

<https://www.zhihu.com/question/269309035>

第四节 AutoEncoder与VAE

1、AutoEncoder

AutoEncoder是一种无监督学习或者说自监督学习的深度学习模型，它的优化目标是最大程度重构数据。它的结构是编码器+解码器。AutoEncoder这样做，其最终目的是要学习得到数据的低维表征，因此它属于一种降维的方法。

（1）AutoEncoder的作用（对于AutoEncoder的理解：（1）AutoEncoder作为一种降维方法，它跟传统的PCA、KPCA、流形学习等方法相比，优点是不是在于它的可扩展性更强，比如可以加入sparse constraint约束等？（2）还有其他的什么优点吗？比如，是否对于特别大的数据，降维的结果会比传统方法更好吗？）

其实AutoEncoder并不是一种具体的网络结构，而是一种框架（或者说一种思想）。它的具体网络结构可以设计为任意的结构，它对于足够大的数据可以很好地拟合其分布（只要模型容量足够大），并得到符合其分布特征的低维嵌入。它具有定制性和可扩展性。并且直接导致了Encoder-Decoder结构的提出。

seq2seq也是一种框架（或者说一种思想）

（2）基本的AutoEcoder可以看成是一种降维或去噪方法，跟PCA、KPCA、流形学习等一样。但是AutoEncoder的强大之处在于，它能够加入一些constraint，即它具有可扩展性。比如它可以增加或减少层数；可以加入人类的先验认识作为约束，得到具有不同特性的code。它还启发产生了encoder-decoder结构。

（3）AutoEncoder相当于用有监督的方式做无监督学习。

（4）AutoEncoder的功能相当于用低维空间中的分布去近似高维空间中的分布。（降维的本质都是这样，从分布的角度，是近似数据在高维空间中的分布；从拓扑的角度，是近似数据在高维空间中的拓扑结构）

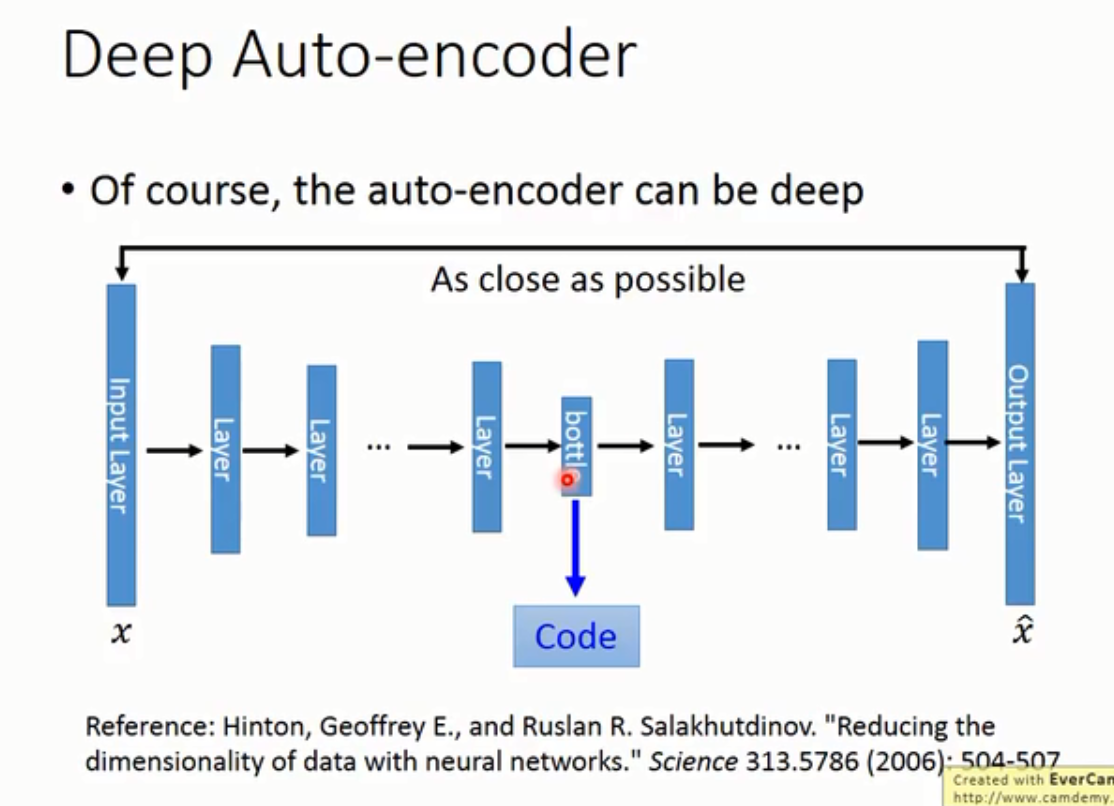


图12 AutoEncoder的结构

2、VAE

VAE属于生成模型。我们知道，生成模型的最终目的是拟合数据集的分布P(X)。VAE就是基于AutoEncoder的思想来拟合数据集的分布。

VAE可以被看作是Gaussian Mixture Model的Distributed Representation版本。

第五节 GAN

GAN是一种生成模型的训练范式。它的结构包括生成器和辨别器两部分。生成器用于生成数据，辨别器用于区分真实数据和生成器生成的数据。我们把生成器和辨别器串在一起进行训练，在训练过程中，它们是“对抗”关系，即生成器尽可能生成出与真实图片一样的数据，以使辨别器无法分辨；分辨器则尽量去区分生成器和真实图片。

GAN与VAE的对比：GAN是隐式地对数据集的分布建模；VAE是显式地对数据集的分布进行建模。

第六节 GNN与GCN

图神经网络，图卷积神经网络。

第七节 Hopfield网络

主流的神经网络结构，神经元之间只有前馈，没有反馈，也就是前馈神经网络。而有一类网络，其神经元间是有反馈的。包括Hopfield网络、玻尔兹曼机等。这些网络根据定义在网络上的“能量”来进行优化。

Hopfield网络的结构是全连接的无向图：

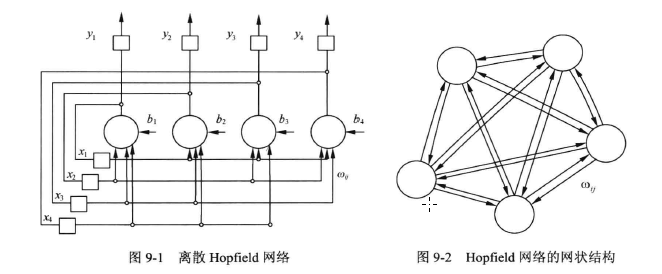


图13 Hopfield网络的结构

Hopfield网络可以用来解旅行商问题（TSP）。

<https://www.cnblogs.com/huty/p/8519271.html>

第八节 RBM与DBN

受限玻尔兹曼机、深度置信网络。

第九节 seq2seq

第十节 强化学习（RL）与深度强化学习（DRL）

强化学习（Reinforcement Learning，RL）是针对这样一类场景：一个Actor与环境不断进行互动。Actor能够观察到环境的state，然后采取action，环境在Actor采取action后会改变state，并给出一个reward。这是一个循环的、序列化的过程。

强化学习的目标是学到一个策略（，就是对每个state，采取一个相应的action），使得最终的回报最大。

强化学习中的环境可能是真实环境，也可能是环境模拟器模拟出来的。

强化学习用于自动驾驶、打电子游戏、下围棋等。

# 第二章 深度学习的意义和特点

深度学习属于神经网络，神经网络的本质其实就是一个拟合函数。深度学习是深层的神经网络，深层的好处是能更高效地表示数据。

深度学习的一个迷人的特点是，它既有强大的表征能力，又具有非常好的泛化性。



<https://www.bilibili.com/video/BV1Sg4y1i7FQ?from=search&seid=15170228889483732389>

# 第三章 深度学习模型的优化

第一节 优化方法

基于梯度的优化算法：（1）梯度下降与随机梯度下降；（2）动量优化；（3）自适应学习率优化；（4）二阶梯度优化。

第二节 优化技巧

1、初始化

2、归一化：BN、LN

# 第四章 深度学习模型的发展方向

当前深度学习在实际中的应用的主要模式，是针对不同的任务和不同的数据，设计相应的网络结构的模型，然后在数据上进行学习。这种简单的学习范式存在诸多局限性：从数据上来说，无法利用多模态数据的信息；从任务上来说，模型通常局限于完成一个任务，而对其他任务则无能为力。为了追求更加智能的模型，人们针对这种学习范式的局限性提出了一些新的学习范式。

这些新的学习范式，听起来非常高大上，但实际上它们的思路都是非常朴素的。

第一节 多模态学习

是指样本数据不是由单一模态组成，而是由多种不同的模态（如文本、图像、视频、音频等）组成，因此多模态学习的主要目标是如何综合利用不同模态数据的信息。

比如多模态文本分类、多模态文本生成。

多模态学习中的几个主要的课题是：

（1）多模态表示表示学习：就是将各种模态的数据都表示成实值向量，又分为联合表示（不同模态的数据都映射到同一空间中）和协同表示（不同模态的数据映射到不同的空间，但映射后的向量满足一定的约束关系）；

（2）模态对齐：从来自同一个实例的两个模态中，寻找子成分之间的关系；

（3）多模态融合：将多个模态的数据都输入到同一个模型中，使模型同时能考虑它们的信息，这就需要模型中对多个模态的数据进行“融合”。多模态融合分为数据级别融合、特征级别融合和决策层的融合。

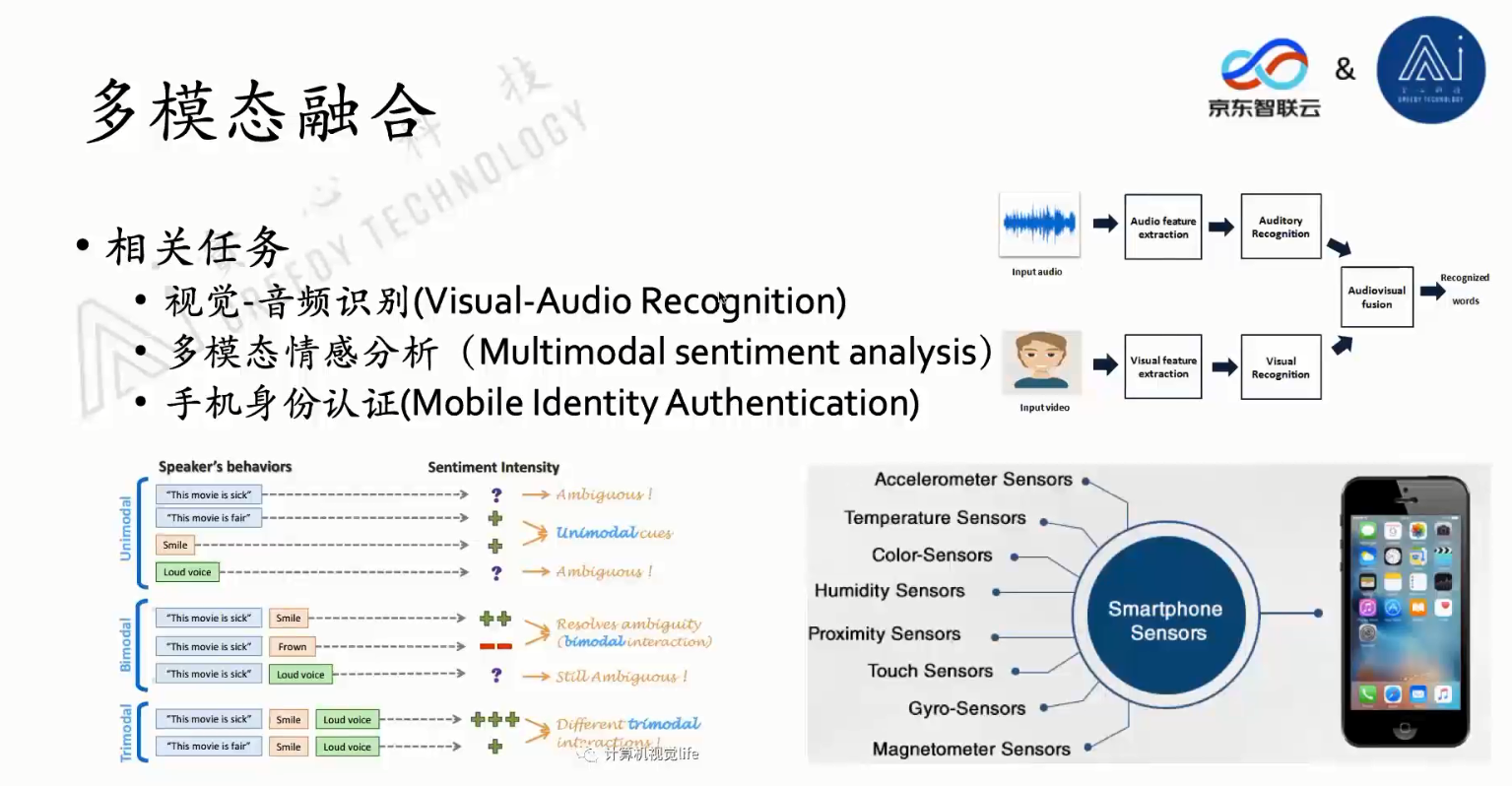


图 多模态融合的几个例子

（4）协同学习：利用数据量较大的模态的知识辅助数据量较小的模态建立模型。

<https://zhuanlan.zhihu.com/p/39878607>

第二节 多任务学习

是指用同一个模型完成多个任务。当然，对于每一个任务都有一个输出层，但是它们会共享输出层以外的结构。多任务学习的好处是，当任务之间有比较紧密的联系时，如果能使模型很好地完成一些任务，那么对它完成另一些任务来说可能有帮助。

<https://zhuanlan.zhihu.com/p/93824566>

第三节 联邦学习

如今神经机器学习的应用存在一个问题：我们的任务需要多种数据，但是由于保密等原因，这些数据的拥有者不能共享数据，这就是“数据孤岛”。联邦学习就是要解决数据孤岛的问题，它让参与方在不共享数据的基础上联合建模。

<https://www.jianshu.com/p/f268baa0f21e>

第四节 迁移学习

是指使用大量数据预训练出来的普适任务的模型，通过fine-tune应用到少量数据的特殊任务上。

迁移学习和协同学习有相似之处，二者都是用大的数据的信息去支持小数据的任务。但二者区别明显：协同学习是不同模态的数据，迁移学习是同一模态的数据；协同学习重点在于数据，迁移学习重点在于模型。

使用迁移学习的思路，就能实现few-shot、one-shot，甚至是zero-shot学习。

第五节 终身学习（增量学习）

是指模型能够先后地在多个任务的数据上进行学习，使模型能完成这些任务，同时保持它在原来任务上的性能。像人类的学习过程那样，不断地完善和提高自身的性能。

终身学习的难点在于，目前的神经网络结构的模型过分依赖数据，导致模型具有“灾难性遗忘”的特点。

第六节 元学习

传统的机器学习模型只能学习如何完成一个具体的任务，而元学习是指让模型学到“学习的能力”，也就是说，希望模型能学到如何针对任务构建一个具体的模型。

传统的机器学习是在一大堆训练数据上进行训练，训练好的模型是输入数据，输出预测值；元学习是在一大堆训练任务上进行训练，训练好的模型是输入一个任务，输出针对该任务的一个模型。

实际中无法取得如此理想的成果，毕竟学会如何构建模型太复杂了。因此我们一般退而求其次，希望让模型学会如何构建优化算法或者构建初始化方法等小目标。

第七节 模型压缩

模型压缩是近年来一个比较火的方向。我们对模型压缩的需求来源于存储和计算的限制。我们希望模型在尽量保持性能的前提下，压缩其参数规模。比如，要将模型部署在小型的终端设备如智能手机、智能手表上，需要模型足够小；又比如BERT，我们将它应用到下游任务时，常常苦于其大和慢。这些情况下就需要对模型进行压缩。

为什么模型压缩是可能的呢？这是因为一个大的模型很有可能其中有相当一部分的神经元或者连接权重在模型运作过程中是影响很小的，去掉它们不会对模型的性能造成太大影响。

那么为什么不直接训练小的模型呢？这是因为大的模型相对小的模型更容易训练。

模型压缩的两个主流的思路：Model Pruning，即当模型训练好以后，对模型进行剪枝，剪掉对模型性能影响较小的神经元或连接权重；Knowledge Distillation，即当模型训练好以后，采用student-teacher模式训练一个较小的模型。