

深度学习综述

戚加欣 2016.11.18

机器学习

机器学习是人工智能中发展最快的分支之一。从二十世纪九十年代以来，机器学习成为人工智能的主流方向。

机器学习研究的是计算机怎样模拟人类的学习行为，以获取新的知识或技能，并重新组织已有的知识结构使之不断改善自身。

[train-经验] - -> [model-学习算法] - -> [test-预测]

人工神经网络

历史

- 二十世纪五十年代中后期，基于神经网络的“连接主义 (Connectionism)”学习开始出现。
- 1986年，D.E.Rumelhart等人重新发明了著名的BP算法，产生了深远影响，神经网络开始兴起。
- 因为理论分析的难度，加上训练方法需要很多经验和技巧，以及巨大的计算量和优化求解难度，神经网络慢慢淡出了科研领域的主流方向。
- 在2006年以前，尝试训练一个深层的、监督的前馈神经网络往往比浅层的（1~2个隐层）网络产生更糟糕的结果（无论是训练误差，还是测试误差）（梯度下降局部最优）
- 在2006年，以Hinton为首的研究人员在深度置信网络（Deep Belief Networks, DBNs）方面的划时代性的工作，解决了上述问题。从此深度学习的方法一路所向披靡

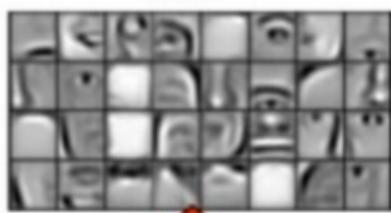
代表性的论文

- Hinton, G. E., Osindero, S. and Teh, Y., A fast learning algorithm for deep belief nets. Neural Computation. 18:1527-1554, 2006

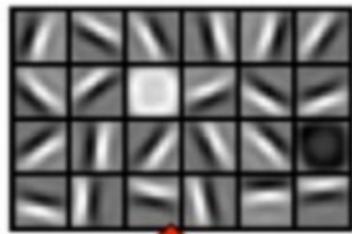
人脑视觉原理



object models



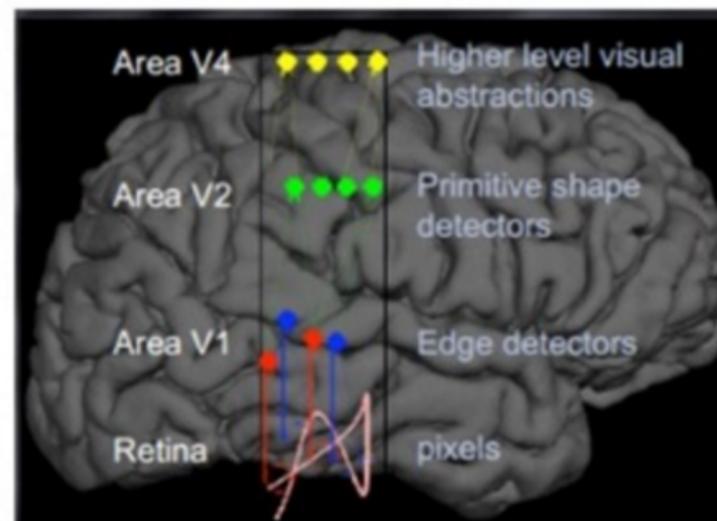
object parts
(combination
of edges)



edges



pixels



人脑视觉原理

总的来说，人的视觉系统的信息处理是分级的，从低级的V1区提取边缘特征，再到V2区的形状或者目标的部分等，再到更高层，整个目标、目标的行为等。也就是说高层的特征是低层特征的组合，从低层到高层的特征表示越来越抽象。

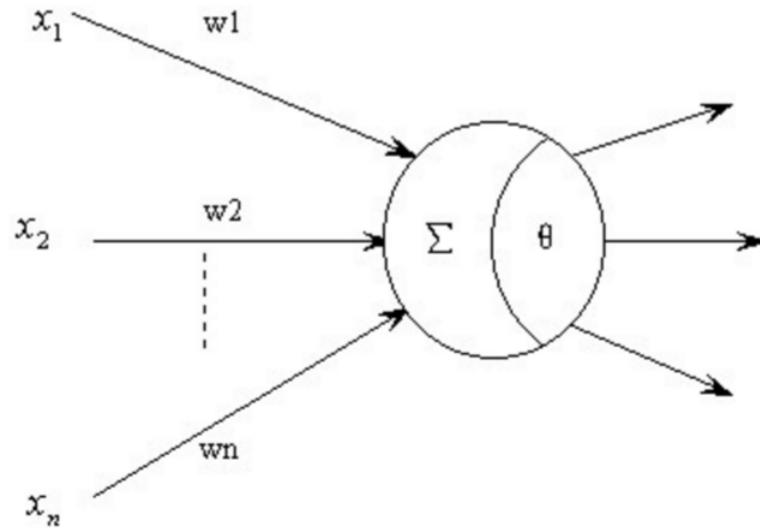
可以看出，大脑是一个深度架构，认知过程也是深度的。受大脑结构分层次启发，神经网络研究人员一直致力于多层次神经网络的研究。

定义

神经网络是由具有适应性的简单单元组成的广泛并行互连的网络，它的组织能够模拟生物神经系统对真实世界物体所做出的交互反应。[Kohonen, 1988]

M-P 神经元模型

1943年，[McCulloch and Pitts, 1943] 将生物神经元模型抽象为一个简单的模型。在这个模型中，神经元接收到来自 n 个其他神经元传递过来的输入信号，这些输入信号通过带权(Weight)的连接进行传递，神经元接受到的总输入值将与神经元的阈值进行对比，然后通过“激活函数”处理以产生神经元的输出。把许多个这样的神经元按一定的层次结构连接起来，就得到了神经网络。



$$y = f\left(\sum_{i=1}^n \omega_i x_i - \theta\right)$$

x_i 来自第*i*个神经元的输入

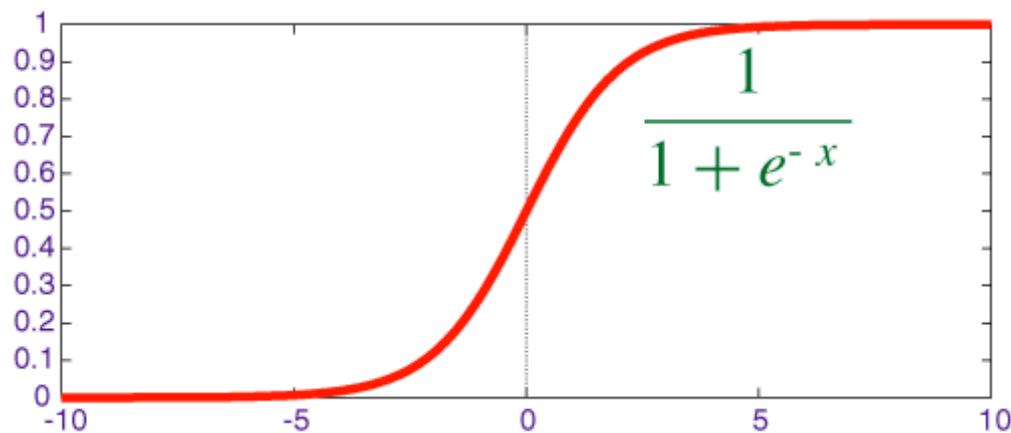
ω_i 第*i*个神经元的连接权重

θ 阈值

激活函数通常有如下一些性质：

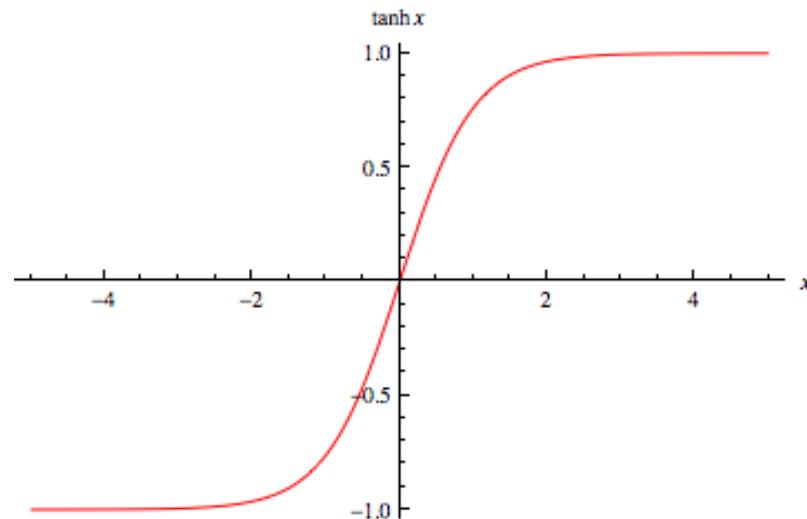
- 非线性：当激活函数是线性的时候，一个两层的神经网络就可以逼近基本上所有的函数了。但是，如果激活函数是恒等激活函数的时候（即 $f(x) \approx x$ ），就不满足这个性质了，而且如果MLP使用的是恒等激活函数，那么其实整个网络跟单层神经网络是等价的。
- 可微性：当优化方法是基于梯度的时候，这个性质是必须的。
- 单调性：当激活函数是单调的时候，单层网络能够保证是凸函数。
- $f(x) \approx x$ ：当激活函数满足这个性质的时候，如果参数的初始化是random的很小的值，那么神经网络的训练将会很高效；如果不满足这个性质，那么就需要很用心的去设置初始值。
- 输出值的范围：当激活函数输出值是有限的时候，基于梯度的优化方法会更加稳定，因为特征的表示受有限权值的影响更显著；当激活函数的输出是无限的时候，模型的训练会更加高效，不过在这种情况下，一般需要更小的learning rate.

Sigmoid 激活函数



$$f(x) = \frac{1}{1 + e^{-x}}$$

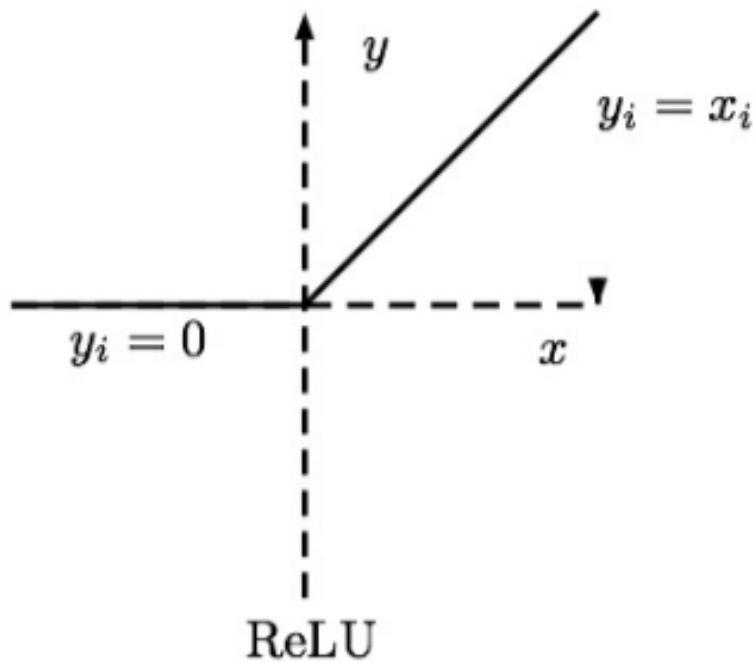
tanh 激活函数



$$\tanh(x) = 2\text{sigmoid}(2x) - 1$$

\tanh 是 sigmoid 的变形

ReLU 激活函数



$$f(x) = \max(0, x)$$

感知机

感知机(Perception)由两层神经元组成，输入层接收外界输入信号后传递给输出层，输出层是M-P神经元。感知机的训练包括多训练样本的输入及计算每个样本的输出。在每一次计算以后，权重 ω 都要调整以最小化输出误差，这个误差由输入样本的标记值与实际计算得出值的差得出。

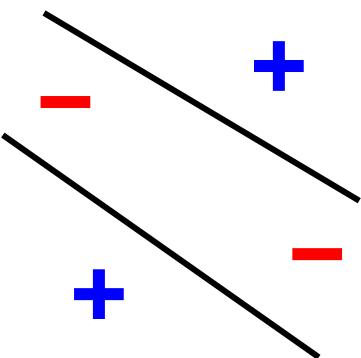
感知机

$$\Delta\omega_i = \eta(y - \hat{y})x_i$$

$$\omega_i \leftarrow \omega_i + \Delta\omega_i$$

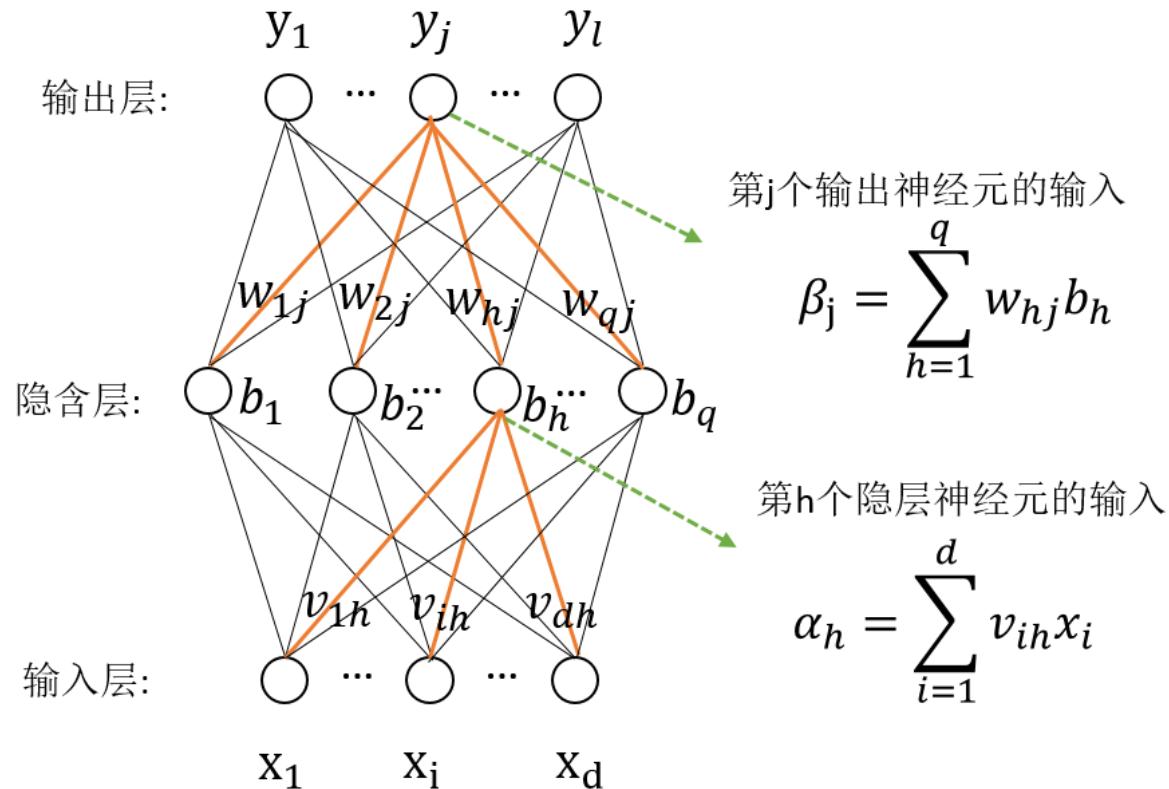
感知机输出为 \hat{y} ，感知机的权重 ω 将按上式调整。其中 η 称为学习率。

这种简单的感知机有一个明显缺陷：只能学习线性可分函数，无法解决异或问题。



多层前馈神经网络

要解决非线性可分问题，需考虑使用多层功能性神经元，在输出层与输入层之间额外插入一层或多层神经元，称为隐层或隐含层。



多层前馈神经网络

- 一个输入层，一个输出层，一个或多个隐含层。
- 每一个神经元都是一个上文提到的感知机。
- 输入层的神经元作为隐含层的输入，同时隐含层的神经元也是输出层神经元的输入。
- 每条建立在神经元之间的连接都有一个权重 ω （与感知机中提到的权重类似）。
- 在 t 层的每个神经元通常与前一层 ($t - 1$ 层) 中的每个神经元都有连接（但你可以通过将这条连接的权重设为0来断开这条连接）。

多层前馈神经网络

- 为了处理输入数据，将输入向量赋到输入层中。这些值将被传播到隐含层，通过加权传递函数传给每一个隐含层神经元（这就是前向传播），隐含层神经元再计算输出（激活函数）。
- 输出层和隐含层一样进行计算，输出层的计算结果就是整个神经网络的输出。
- 神经网络的学习过程，就是根据训练数据来调整神经元之间的“连接权”以及每个功能神经元之间的阈值。即，神经网络学到的知识，蕴含在连接权与阈值中。

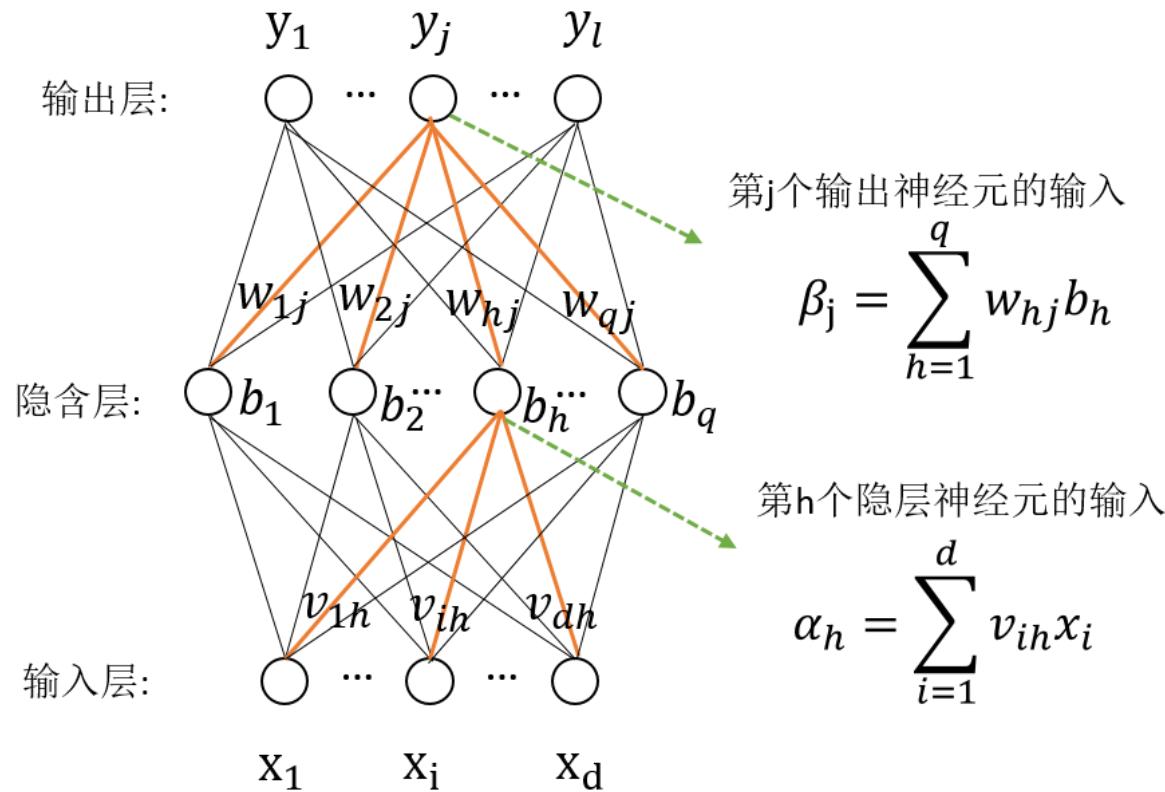
训练ANN

误差逆传播(Back Propagation)算法

在神经网络中，输入参数是一个训练误差的曲线。每个权重的最佳值应该是误差曲线中的全局最小值

反向传播（英语：Backpropagation，缩写为BP）是“误差反向传播”的简称，是一种与最优化方法（如梯度下降法）结合使用的，用来训练人工神经网络的常见方法。该方法计算对网络中所有权重计算损失函数的梯度。这个梯度会反馈给最优化方法，用来更新权值以最小化损失函数。

误差逆传播(Back Propagation)算法



记隐层第 h 个神经元接收到的输入为 α_h ，输出层第 j 个神经元接收到的输入为 β_j

$$\hat{y}_j^k = f(\beta_j - \theta_j)$$

误差逆传播(Back Propagation)算法

更新公式

$$\Delta\omega_{hj} = \eta g_j b_n$$

$$\Delta\gamma_h = -\eta e_h$$

误差逆传播(Back Propagation)算法

输入：训练集 $D = \{(x_k, y_k)\}_{k=1}^m$ 学习率 η

过程：

在(0,1)范围内随机初始化网络中所有连接权和阈值

repeat

 for all $(x_k, y_k) \in D$ do

 根据当前参数合式计算当前样本的输出 \hat{y}_k

 根据梯度下降算法计算输出层神经元的梯度项 g_j

 根据梯度下降算法计算隐层神经元的梯度项 e_h

 根据更新公式更新连接权 ω_{hj}, v_{ih} 与阈值 θ_j, γ_h

 end for

until 达到停止条件

输出：连接权与阈值确定的多层前馈神经网络

误差逆传播(Back Propagation)算法

示例图

http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

深度网络

浅层结构函数表示能力的局限性

浅层模型的一个共性是仅含单个将原始输入信号转换到特定问题空间特征的简单结构。典型的浅层学习结构包括传统隐马尔可夫模型(HMM)、条件随机场(CRFs)、最大熵模型(MaxEnt)、支持向量机(SVM)、核回归及仅含单隐层的多层感知器(MLP)等。

浅层结构函数表示能力的局限性

其局限性在于有限样本和计算单元情况下对复杂函数的表示能力有限,针对复杂分类问题其泛化能力受到一定制约。

在很多情况下，深度为2就已足以在给定精度范围内表示任何函数了，例如逻辑门、正常 神经元、 sigmoid-神经元、 SVM中的 RBF(Radial Basis Function)等，但对于某一类函数，需要的参数的个数与输入的大小是成指数关系的（逻辑门、正常神经元、 RBF单 元）

利用深度为K的多项式级的逻辑门电路实现的函数,对于K-1层电路需要指数倍的计算节点。

浅层结构函数表示能力的局限性

深度学习可通过学习一种深层非线性网络结构,实现复杂函数逼近,表征输入数据分布式表示,并展现了强大的从少数样本集中学习数据集本质特征的能力

很多可以用深层结构有效表示的却无法用浅层的来有效表示

深度网络

深度学习首先利用无监督学习对每一层进行逐层预训练（Layerwise Pre-Training）去学习特征；每次单独训练一层，并将训练结果作为更高一层的输入；然后到最上层改用监督学习从上到下进行微调（Fine-Tune）去学习模型。

深度网络

深度置信网络

受限波尔兹曼机

受限波尔兹曼机（Restricted Boltzmann machines RBM），是一种可以在输入数据集上学习概率分布的生成随机神经网络。

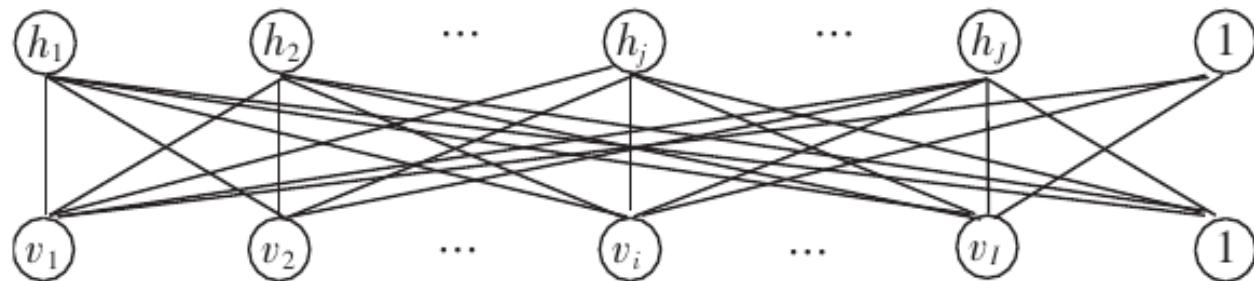


图 2 RBM 模块

受限波尔兹曼机

RBM由隐含层、可见层、偏置层组成。和前馈神经网络不同，可见层和隐含层之间的连接是无方向性（值可以从可见层->隐含层或隐含层->可见层任意传输）且全连接的。每一个当前层的神经元与下一层的每个神经元都有连接——如果允许任意层的任意神经元连接到任意层去，我们就得到了一个波尔兹曼机（非受限的）。

深度置信网络 (DBN)

将波尔兹曼机进行栈式叠加来构建深度信度网络 (DBN)

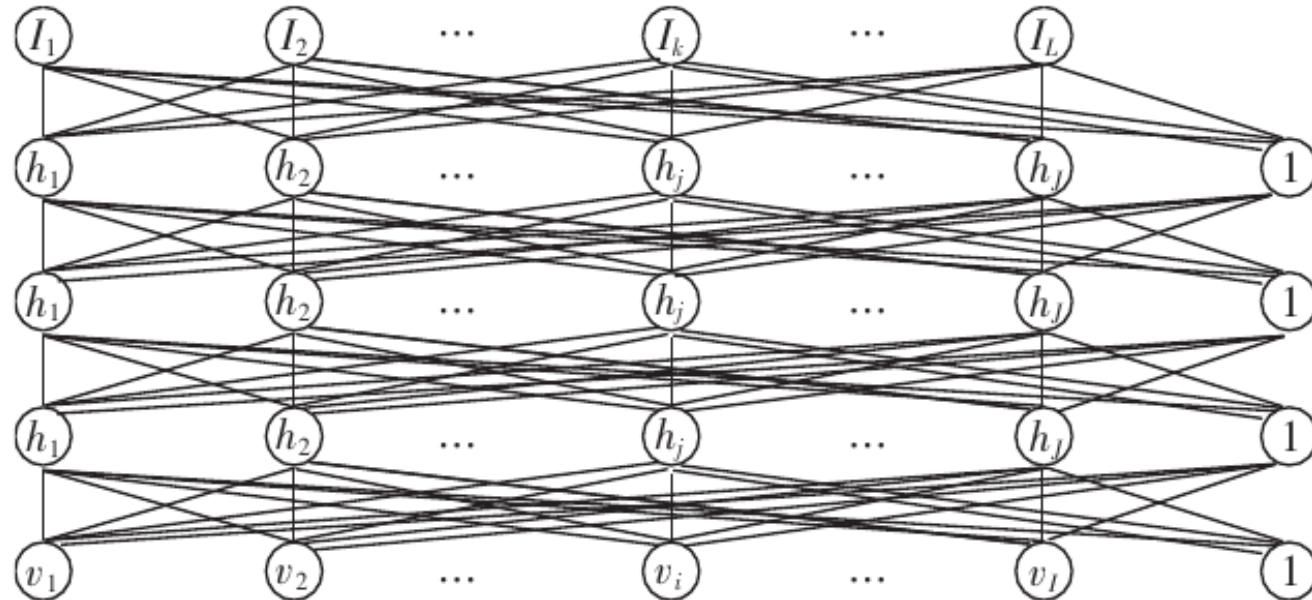


图 3 DBN 模型

属于生成型深度结构

深度置信网络

DBN解决传统BP算法训练多层神经网络的难题:

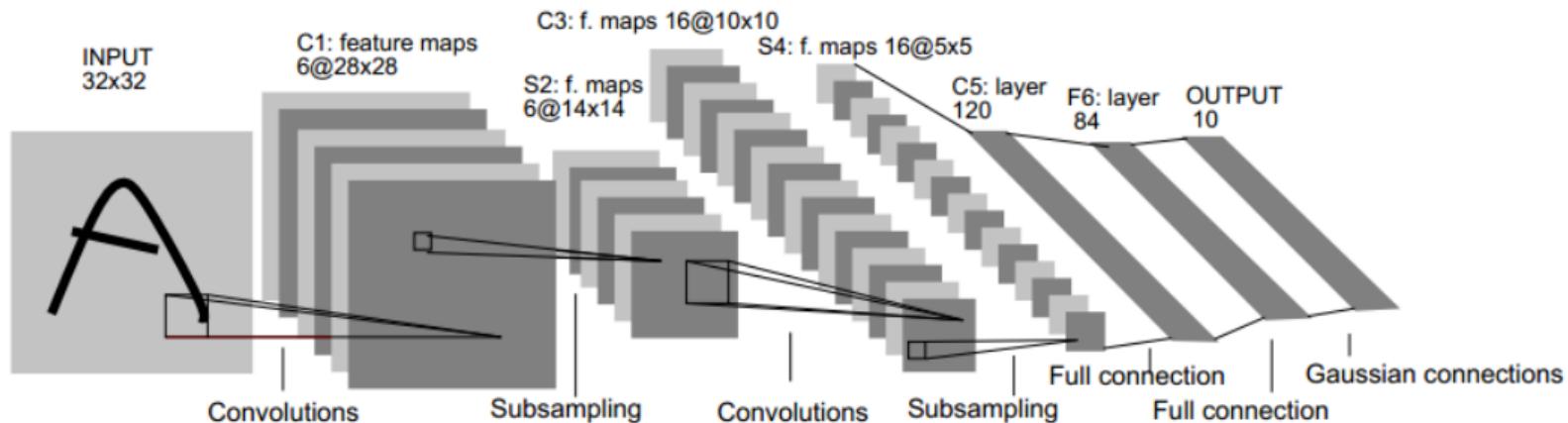
- 需要大量含标签训练样本集
- 较慢的收敛速度
- 因不合适的参数选择陷入局部最优

深度置信网络

卷积神经网络

用于模拟视觉皮层(visual cortex)。CNN在视觉识别任务中的效果很好。

卷积神经网络(CNNs)是第一个真正成功训练多层网络结构的学习算法,与DBNs不同,它属于区分性训练算法。卷积网络最初是受视觉神经机制的启发而设计的, 是为识别二维形状而设计的一个多层感知器, 这种网络结构对平移、比例缩放、倾斜或者其他形式的变形具有高度不变性。



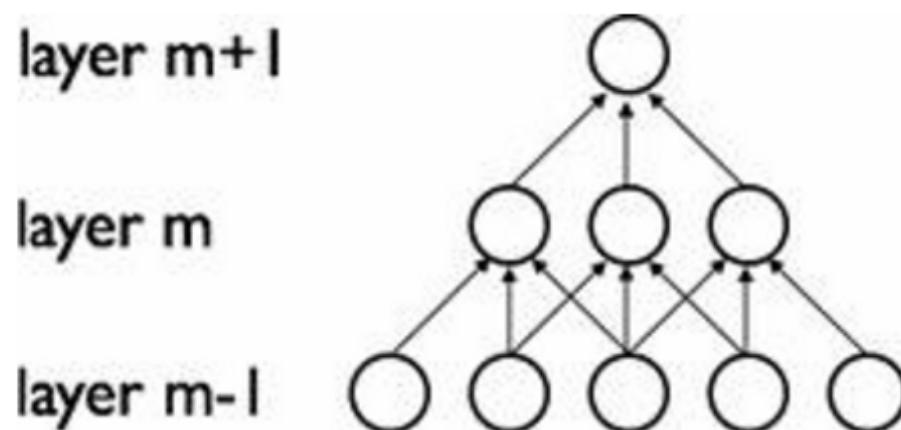
LeNet-5 (Yann LeCun)

卷积神经网络

传统的神经网络都是采用全连接的方式，即输入层到隐藏层的神经元都是全部连接的，这样做将导致参数量巨大，使得网络训练耗时甚至难以训练，而CNN则通过局部连接、权值共享等方法避免这一困难

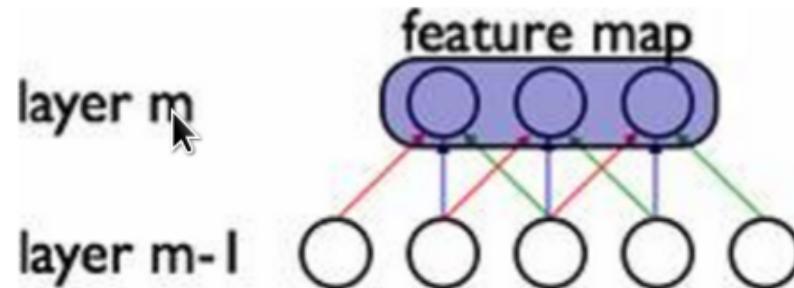
卷积神经网络

局部连接



卷积神经网络

权值共享



卷积神经网络

CNN中主要有两种类型的网络层，分别是卷积层和池化/汇集层(Pooling)

卷积层的作用是提取图像的各种特征；

汇集层的作用是对原始特征信号进行抽象，从而大幅度减少训练参数，另外还可以减轻模型过拟合的程度。

深度学习工具介绍

1. TensorFlow
2. Caffe
3. Torch

TensorFlow

TensorFlow是谷歌基于DistBelief进行研发的第二代人工智能学习系统，其命名来源于本身的运行原理。Tensor（张量）意味着N维数组，Flow（流）意味着基于数据流的计算，TensorFlow为张量从图像的一端流动到另一端的计算过程。TensorFlow是将复杂的数据结构传输至人工智能神经网络进行分析和处理的系统。

TensorFlow已被用于语音识别和图像识别等多项机器深度学习领域，TensorFlow支持CNN、RNN和LSTM算法，这都是目前在图像、语言和自然语言处理上最流行的深度神经网络模型。TensorFlow使用灵活，不仅支持深度学习，还支持一般机器学习的搭建。

TensorFlow

Caffe

Caffe的全称是Convolutional Architecture for Fast Feature Embedding，顾名思义，Caffe针对卷积架构的网络提供良好的支持，而对于循环神经网络、递归神经网络没有提供支持，在使用灵活性上不如TensorFlow。

由于卷积神经网络的强大功能，Caffe可以被用于图像分类、目标识别、图像分割等领域，同样也可用于处理非图像数据的分类、回归问题。Caffe使用简单，一般修改配置文件就可以构建网络，还提供了Python和Matlab接口，直接使用上层接口使得网络构架操作变得直接和简单。Caffe提供许多Demo使得学习Caffe变得非常轻松。

Torch

Torch诞生于2000年，但真正开始受到关注还是开始于FaceBook在2015年1月开放了大量Torch的深度学习模块和拓展。

得益于Lua的特性，Torch的优点在于快速，灵活，支持CPU模式和GPU模式，在多CPU模式下提速效果最为明显。同样是因为Lua语言书写，缺点在于没有Python接口，无法整合Python上的资源。如同Caffe一样，Torch对新型网络连接和架构的支持不如Theano和TensorFlow，深度神经网络中层的种类受到限制。

由于长时间的发展，Torch提供了很多拓展组件，功能丰富，同时Torch在语言、图像、视频等领域应用广泛，得益于Torch优异的性能，现在有许多商业公司将Torch作为人工智能研究的核心。

应用现状

1. 计算机视觉

- 典型的应用包括：人脸识别、车牌识别、扫描文字识别、图片内容识别、图片搜索等等。

2. 自然语言处理

- 典型的应用包括：搜索引擎智能匹配、文本内容理解、文本情绪判断，语音识别、输入法、机器翻译等等。

3. 社会网络分析

- 典型的应用包括：用户画像、网络关联分析、欺诈作弊发现、热点发现等等。

4. 推荐

- 典型的应用包括：虾米音乐的“歌曲推荐”，淘宝的“猜你喜欢”等等。

参考

周志华, 机器学习. 清华大学出版社有限公司, 2016.

https://en.wikipedia.org/wiki/Deep_learning

https://en.wikipedia.org/wiki/Machine_learning

孙志军, et al. "深度学习研究综述." 计算机应用研究 29.8 (2012): 2806-2810.

[An Introduction to Deep Learning: From Perceptrons to Deep Networks](#)