

解密接近人脑的智能学习机器 ——深度学习及并行化实现

蒋 杰 金 涛
腾讯公司

关键词：深度学习 软件框架 工业界平台

深度学习的革命

人工智能是计算机领域一门具有神秘色彩的学科，科学家希望制造出代替人类思考的智能机器，艺术家将这一题材写进小说，搬上银幕，引发了人们无限的遐想。然而，作为一门严肃的学科，人工智能在过去的半个多世纪时间里发展得并不顺利。过去的很多努力还是基于某些预设规则的快速搜索和推理，离真正的智能还有相当长的距离，或者说距离创造像人类一样具有抽象学习能力的机器还很遥远。

近年来，深度学习开始直接尝试解决抽象认知的难题，并取得了突破性进展。深度学习引爆的这场革命将人工智能带上了一个新的台阶。不仅其学术意义巨大，而且实用性很强。工业界也开始在这方面进行大规模的投入，一大批产品将从中获益。

2006年，机器学习泰斗、加拿大多伦多大学教授杰弗里·辛顿 (Geoffery Hinton) 在《科学》(Science) 上发表文章^[1]，提出了基于深度信念网络 (Deep Belief Networks, DBN) 可使用非监督的逐层贪心训练算法^[2]，为训练深度神经网络带来了希望。

2012年，辛顿又带领学生在目前最大的图像数据库 ImageNet 上，对分类问题取得了惊人的进展^[3]，将 Top5 错误率由 26% 大幅降低至 15%。

2012年，由人工智能和机器学习顶级学者、斯

坦福大学人工智能实验室主任吴恩达 (Andrew Ng) 和分布式系统顶级专家杰夫·迪恩 (Jeff Dean) 领衔的梦幻阵容，开始打造“谷歌大脑 (Google Brain)”项目，用包含 1.6 万个 CPU 核的并行计算平台训练超过 10 亿个神经元的深度神经网络，在语音识别和图像识别等领域取得了突破性的进展^[4]。该系统通过分析 YouTube 上选取的视频，采用无监督的方式训练深度神经网络，可将图像自动聚类。在系统中输入“cat”后，结果在没有外界干涉的条件下，识别出了猫脸。

2012年，微软首席研究官里克·雷斯特 (Rick Rashid) 在 21 世纪的计算大会上演示了一套自动同声传译系统^[5]，将他的英文演讲实时转换成与他音色相近、字正腔圆的中文演讲。同声传译需要经历语音识别、机器翻译、语音合成三个步骤。该系统一气呵成，流畅的翻译赢得了参会者的一致赞扬，深度学习是这一系统中的关键技术。

2013年，谷歌收购了一家名为 DNN Research 的神经网络初创公司。这家公司只有三个人，杰弗里·辛顿和他的两个学生。这次收购并不涉及任何产品和服务，只是希望辛顿可以将深度学习打造为支持谷歌未来的核心技术。同年，纽约大学教授燕乐存 (Yann LeCun) 加盟脸谱，出任人工智能实验室主任^[6]，负责利用深度学习探寻用户图片等，希望在未来能给用户提供更智能化的产

品使用体验。

2013年,百度成立了百度研究院及下属的深度学习研究所,将深度学习应用于语音识别和图像识别、检索以及广告点击率预估(Click-Through-Rate Prediction, pCTR),其中图片检索达到了国际领先水平。2014年百度将吴恩达招致麾下。

腾讯也是互联网公司中的数据大户,目前即时通信QQ活跃账户数达8.29亿,QQ智能终端月活跃账户数5.21亿,即时通信QQ最高同时在线账户数达到2.06亿,微信和WeChat合并月活跃账户数达到4.38亿,QQ空间月活跃账户数达到6.45亿,QQ空间智能终端月活跃账户数4.97亿,增值服务付费注册账户数为8800万。腾讯每天的数据量是一个天文数字。目前日接入消息条数最高达1万亿,日接入数据量200TB,并发分拣业务接口1万个。为了挖掘这些数据中蕴含的巨大价值,2013年,腾讯深度学习团队构建了加速深度学习的并行化平台Mariana,包括深度神经网络的多图形处理器(GPU)数据并行框架、深度卷积神经网络的多GPU模型并行和数据并行框架以及深度神经网络的CPU集群框架。Mariana支持语音识别、图像识别,并正在积极探索在广告推荐中的应用。

如果说辛顿2006年发表在《科学》杂志上的论文^[1]只是在学术界掀起了对深度学习的研究热潮,那么近年来各大巨头公司争相跟进,将顶级人才从学术界争抢到工业界,则标志着深度学习真正进入了实用阶段,将对一系列产品和服务产生深远影响,成为它们背后强大的技术引擎。

目前,深度学习在几个主要领域都取得了突破性进展:在语音识别领域,深度学习用深层模型替换声学模型中的混合高斯模型(Gaussian Mixture Model, GMM),降低了30%左右的错误率;在图像识别领域,通过构造深度卷积神经网络(Convolutional Neural Networks, CNN)^[3],将Top5错误率进一步降低到11%;在自然语言处理领域,深度学习基本获得了与其他方法水平相当的结果,但可以免去繁琐的特征提取步骤。

深层模型的基本结构

深度学习采用的模型为深层神经网络(Deep Neural Networks, DNN)模型,即包含多个隐藏层(hidden layer, 隐含层)的神经网络。深度学习利用模型中的隐藏层,通过特征组合的方式,逐层将原始输入转化为浅层特征、中层特征、高层特征,直至最终的任务目标。

深度学习源于人工神经网络的研究。一个神经元如图1所示^[15]。

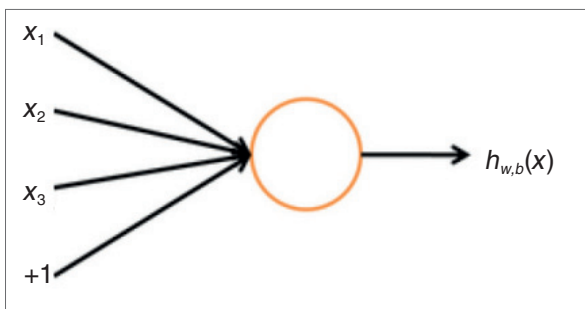


图1 神经元结构

这个神经元接受三个输入 x_1, x_2, x_3 , 神经元输出为

$$h_{w,b}(x) = f(\sum_{i=1}^3 W_i x_i + b)$$

其中, W_i 和 b 为神经元的参数, $f(z)$ 称为激活函数,一种典型的激活函数为Sigmoid函数,即

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

其图像为图2。

神经网络则是多个神经元组成的网络,一个简单的神经网络如图3所示。

使用圆圈来表示神经网络的输入,标上“+1”的圆圈称为偏置节点,也就是截距项。神经网络最左边的一层叫做输入层(本例中,有3个输入单元,偏置单元不计);最右的一层叫做输出层(本例中,输出层有2个节点);中间的节点叫做隐藏层(本例中,有2个隐藏层,分别包含3个和2个神经元,偏置单元同样不计),之所以叫隐藏层,是因为无法

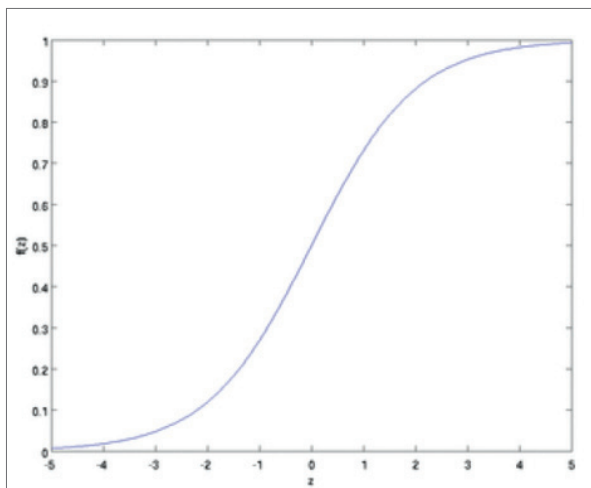


图2 Sigmoid函数图像

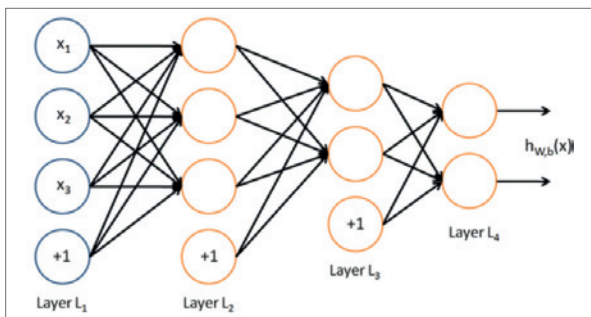


图3 一个简单的神经网络

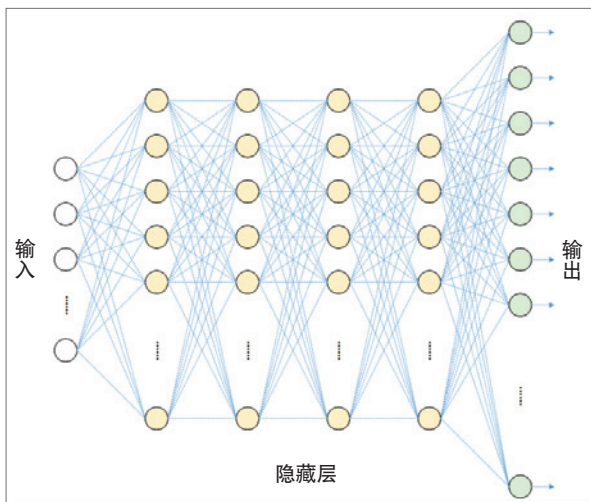


图4 一种典型的深层神经网络模型

在训练样本集中观测到它们的值。神经网络中的每一条连线对应一个连接参数，连线个数对应网络

的参数个数（本例共有 $4 \times 3 + 4 \times 2 + 3 \times 2 = 26$ 个参数）。求解此神经网络，需要 $(x(i), y(i))$ 的样本集，其中 $x(i)$ 是 3 维向量， $y(i)$ 是 2 维向量。

图 4 是一个用于语音识别的深层神经网络。具有 1 个输入层、4 个隐藏层和 1 个输出层，相邻两层的神经元全部连接。

选择深层模型的原因

为什么要构造包含这么多隐藏层的深层网络结构呢？其背后是有一些理论依据的。

天然层次化的特征

对于很多训练任务来说，特征具有天然的层次结构。以语音、图像、文本为例，层次结构如表 1 所示。

表1 几种任务领域的特征层次结构

任务领域	原始输入	从浅层特征到中层特征再到高层特征	训练目标
语音	样本	频段 声音 音调 音素 单词	语音识别
图像	像素	线条 纹理 图案 局部 物体	图像识别
文本	字母	单词 词组 短语 句子 段落 文章	语义理解

以图像识别为例，图像的原始输入是像素，相邻像素组成线条，多个线条组成纹理，进一步形成图案，图案构成了物体的局部，直至整个物体的样子。我们从中不难找到原始输入和浅层特征之间的联系，再通过中层特征，一步一步获得和高层特征的联系。

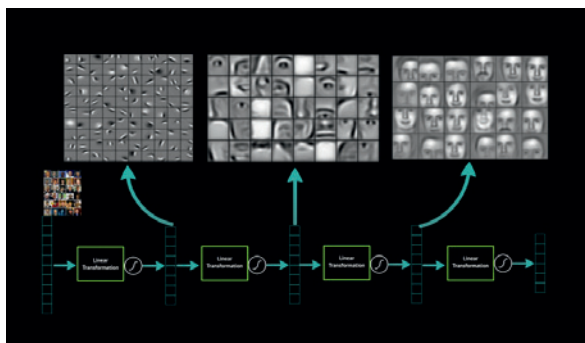


图5 人脸识别系统的多层结构和特征表示^[7]

仿生学依据

人工神经网络本身就是对人类神经系统的模拟,这种模拟具有仿生学的特征。1981年,戴维·休伯尔(David Hubel)和托斯坦·维厄瑟尔(Torsten Wiesel)发现可视皮层是分层的^[14]。人类视觉系统包含了不同的视觉神经元,这些神经元与瞳孔所受的刺激(系统输入)之间存在着某种对应关系(神经元之间的连接参数),即受到某种刺激后(对于给定的输入),某些神经元就会活跃(被激活)。这证实了人类神经网络和大脑的工作其实是不断将低级抽象传导为高级抽象的过程,高层特征是低层特征的组合,越到高层特征就越抽象。

特征的层次可表示性

特征的层次可表示性也得到了证实。文献[8]收集了很多黑白风景照,并从这些照片中找出400个 16×16 的基本碎片;然后再从照片中找到其他一些同样大小的碎片,以便将其他碎片表示为这400个基本碎片的线性组合,并使误差尽可能小,使用的碎片尽可能少;再固定其他碎片,选择更合适的基本碎片组合优化近似结果;反复迭代后,得到可以表示其他碎片的最佳基本碎片组合。文章作者发现,这些基本碎片组合都是不同物体不同方向的边缘线。

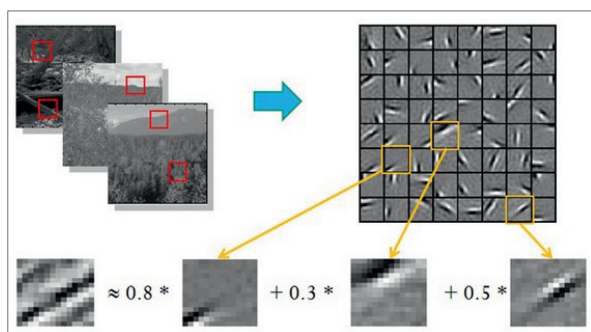


图6 初级图像特征的提取和表示(sparse coding)
(原图由吴恩达提供)

这说明通过进行有效的特征提取,可以将像素抽象成更高级的特征。类似的结果也适用于语音特征。

从浅层模型到深层模型

事实上,深层模型除了结构优势,还具有强大的表达能力,可以像人类一样有效提取高级特征,这并不是新的发现。但是,为什么深层模型直到最近几年才开始得到广泛的关注和应用呢?这需要从传统的机器学习方法和浅层学习谈起。

浅层模型及训练方法

反向传播算法(Back Propagation, BP)^[11]是一种神经网络的梯度计算方法。该算法巧妙利用了下层神经元的梯度可由上层神经元的残差导出的规律,求解的过程也正如算法的名字那样,自上而下反向逐层计算,直至获得所有参数的梯度。这种基于统计的学习方法比起传统的基于规则的方法具备很多优越性^[16]。

20世纪80~90年代,人们提出了一系列机器学习模型,应用最为广泛的是支持向量机^[12]和逻辑回归^[13],这两种模型分别可看做包含1个隐藏层和没有隐藏层的浅层模型。训练时可以利用反向传播算法计算梯度,再用梯度下降方法在参数空间中寻找最优解。浅层模型往往具有凸代价函数,其理论分析相对简单,训练方法也容易掌握,因此取得了很多成功的应用。

深层模型的训练难度

浅层模型的局限性在于有限参数和计算单元,对复杂函数的表示能力有限,对于复杂分类问题,其泛化能力受到一定的制约。深层模型恰恰可以克服浅层模型的弱点,利用反向传播和梯度下降来进行训练。深层模型目前面临三个突出问题^[10]。

局部最优 与浅层模型的代价函数不同,深层模型的每个神经元都是非线性变换的,代价函数是高度非凸函数,采用梯度下降的方法容易陷入局部最优。

梯度弥散 使用反向传播算法传播梯度时,随着传播深度的增加,梯度的幅度会急剧下降,导致浅层神经元权重的更新非常缓慢,不能有效学习,

也使得深层模型变成了前几层相对固定，只能改变最后几层的浅层模型。

数据获取 深层模型的表达能力越强大，模型所需的参数也会相应增加。对于训练如此多参数的模型，小训练数据集是无法实现的，需要海量的有标记的数据，否则只能导致严重的过拟合。

深层模型的训练方法

尽管挑战很大，辛顿并没有放弃努力，2006年他在《科学》上发表的文章中主要讲了两个观点：

1. 多隐藏层的人工神经网络具有优异的特征学习能力，学习到的特征对数据有更本质的刻画，从而有利于可视化或分类。
2. 深度神经网络的训练难度可以通过“逐层初始化”(layer-wise pre-training)来有效降低，并给出了无监督的逐层初始化方法(如图7)。

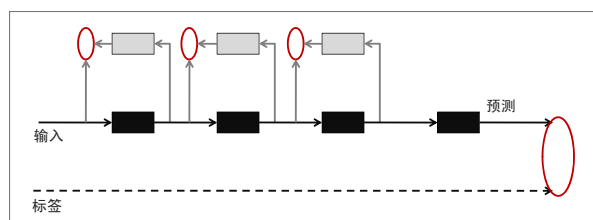


图7 逐层初始化的方法(原图由兰扎托(Marc'Aurelio Ranzato)提供)

给定原始输入后，先训练模型的第一层，即图7中左起第一个黑色框。可以将黑色框看做是一个编码器，将原始输入编码作为第一层的初级特征，可以将编码器看做模型的一种“认知”。为了验证这些特征确实是输入的一种抽象表示，且没有丢失太多信息，需要引入一个对应的解码器，即图7中左起第一个灰色框，可以看做模型的“生成”。为了让认知和生成达成一致，就要求原始输入通过编码再解码，可以大致还原为原始输入。因此，将原始输入与其编码再解码之后的误差定义为代价函数，同时训练编码器和解码器。训练收敛后，编码器就是所需的第一层模型，而不再需要解码器。这时我们得到了原始数据的第一层抽象。固定第一层模型，原始输入就映射成第一层抽象，并将其当做输入，

继续训练出第二层模型；再根据前两层模型训练出第三层模型；以此类推，直至得到最高层模型。

逐层初始化完成后，可用有标签的数据，采用反向传播算法对模型进行整体有监督的训练。这一步可看做对多层模型整体的精细调整。由于深层模型具有很多局部最优解，模型初始化的位置将在很大程度上决定最终模型的质量。“逐层初始化”的步骤就是让模型处于一个较为接近全局最优的位置，从而获得更好的效果。

浅层模型和深层模型的对比

浅层模型的一个重要特点是需要依靠人工经验来抽取样本的特征，并将其作为模型的输入，模型只用来负责分类和预测。在浅层模型中，最重要的往往不是模型的优劣，而是特征选取的优劣。因此，大多数人力都会投入到特征的开发和筛选中。这不但需要对任务问题领域有深刻的理解，还要花费大量时间反复实验摸索，由此也影响了浅层模型的训练效果。

事实上，对深层模型进行逐层初始化可看做是特征学习的过程，通过隐藏层对原始输入的一步步

表2 浅层模型和深层模型的对比

	浅层模型	深层模型
模型层数	1~2	5~10
模型表达能力	有限	强大
特征提取方式	特征工程	自动抽取特征
代价函数凸性	凸代价函数 没有局部最优点 可以收敛到全局最优	高度非凸的代价函数 存在大量的局部最优点 容易收敛到局部最优
训练难度	容易	复杂，需要较多技巧
理论	有成熟的理论基础	理论分析困难
依赖先验知识	依赖更多先验知识	依赖较少先验知识
数据需求量	多	更多
适用场景	需要简单特征的任务： 发电机故障诊断 时间序列处理 视频字幕定位提取 ……	需要高度抽象特征的任务： 语音识别 图像 自然语言处理

抽象表示,来学习原始输入的数据结构,从而找到更有用的特征,最终提高分类问题的准确性。在得到有效特征之后,可进行模型的整体训练。

深层模型的层次组件

深层模型是包含多个隐藏层的神经网络,常见的深层模型基本层次组件包括以下几种。

自编码器

自编码器^[15]可以利用一组无标签的训练数据 $\{x(1), x(2), \dots\}$ (其中 $x(i)$ 是一个 n 维向量)进行无监督的模型训练。它采用反向传播算法,使目标值接近输入值。图8是一个自编码器的示例。

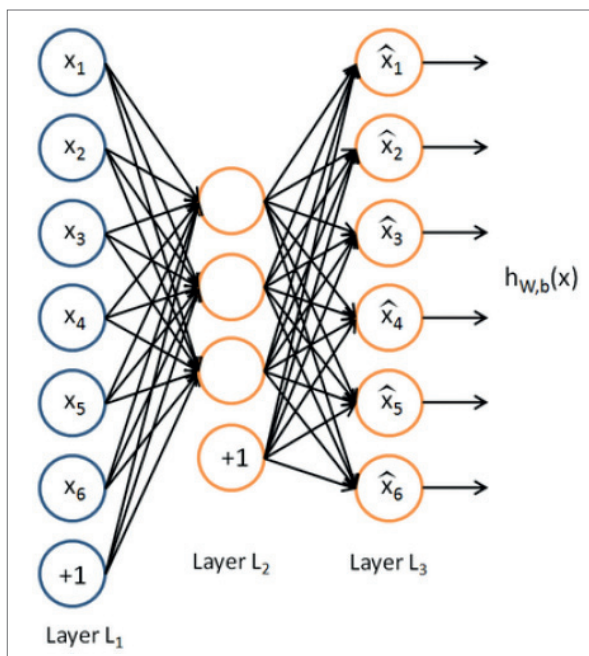


图8 自编码器

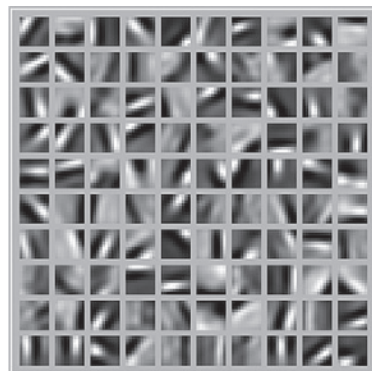
示例中的自编码器尝试训练一个恒等函数,使输出值接近输入值。这看似没有学习的意义,但由于隐藏层神经元的数目(本例中为3个)小于输入向量的维数(本例中为6维),使得隐藏层变成了输入数据的一种压缩表示或抽象的简化表示。如果网络的输入是完全随机的,则将高维向量压缩成低维

向量会变得难以实现。但训练数据往往隐含着特定的结构,自编码器可学习到这些数据的相关性,从而得到有效的压缩表示。实际训练后,如果代价函数越小,说明输入和输出的值越接近,即这个编码器的效果越好。自编码器训练完成后,实际使用时只需要它的前一层,即编码部分,解码部分就没用了。

稀疏自编码器是自编码器的一个变体,它在自编码器的基础上加入了正则化。正则化是指在代价函数中加入抑制项,使隐藏层节点的平均激活值接近于0。若有了正则化的约束,则输入数据可以用少数隐藏节点表达。之所以采用稀疏自编码器,是因为稀疏的表达往往比稠密的表达更有效,人脑神经系统也是稀疏连接,每个神经元只与少数神经元连接。

降噪自编码器是自编码器的另一种变体。通过在训练数据中加入噪声,可训练出对输入信号更加鲁棒的表达,从而提升模型的泛化能力,可以更好地应对实际预测时夹杂在数据中的噪声。

有了自编码器后,我们还想进一步了解自编码器到底学习到了什么。例如,在 10×10 的图像上训练一个稀疏自编码器,然后对于每个隐藏神经元来说,找到什么样的图像可以让其获得最大程度的激励,即是这个隐藏神经元学习到的特征。将100个隐藏神经元的特征都找出来,得到了图9中的100幅图像。图9 自编码器的隐藏神经元^[15]



可以看出,这100幅图像具备了从不同方向检测物体边缘的能力。显然,这样的能力对后续的图像识别很有帮助。

受限玻尔兹曼机

受限玻尔兹曼机(Restricted Boltzmann Machine, RBM)是一个二部图,一层是输入层(v),另一层是

隐藏层 (h)，假设所有节点都是随机二值变量节点，只能取值 0 或 1，同时假设全概率分布 $p(v, h)$ 满足玻尔兹曼分布（见图 10）。

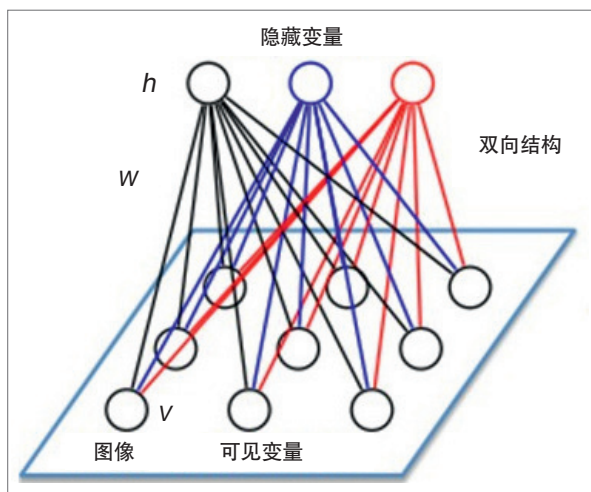


图10 受限玻尔兹曼机

由于同层节点之间没有连接，因此在已知输入层的情况下，隐藏层的各节点是条件独立的；反之，在已知隐藏层的情况下，输入层各节点也是条件独立的。同时，可以根据玻尔兹曼分布，当输入 v 时通过 $p(h|v)$ 生成隐藏层，得到隐藏层之后再通过 $p(v|h)$ 生成输入层。通过调整参数，可使输入 v 生成的 h ，再生成的 v' 与 v 尽可能接近，可以看出隐藏层 h 是输入层 v 的另外一种表示，即深层模型的基本层次组件。全部用受限玻尔兹曼机形成的深层模型为深度玻尔兹曼机 (Deep Boltzmann Machine, DBM)。如果将靠近输入层的部分替换为贝叶斯信念网络，即有向图模型，而在远离输入层的部分使用受限玻尔兹曼机，则称为深度信念网络 (Deep Belief Networks, DBN)。

卷积神经网络

前述的编码器都是全连通网络，可以完成 10×10 的图像识别，如手写体数字识别问题。然而，对于更大的图像，如 100×100 的图像，如果

要学习 100 个特征，则需要 100 万个参数，计算时间将大大增加。解决这种尺寸图像识别的有效方法是利用图像的局部性，构造一个部分联通的网络，最常见的是卷积神经网络^[19,20]，利用的是图像固有的特性，即图像局部的统计特性与其他局部相同。因此，从某个局部学习来的特征同样适用于其他的局部。

假设有一幅 100×100 的图像，准备从中学习一个 10×10 的局部图像特征的神经元，如果采用全连接方式，将 100×100 维的图像输入到这个神经元需要有 1 万个连接权重参数。而采用卷积核的方式，只需 $10 \times 10 = 100$ 个参数权重。卷积核可以看做一个 10×10 的小窗口，在图像上面上下左右移动，走遍图像中每个 10×10 的位置（共有 91×91 个位置）。每移动到一个位置，则将该位置的输入与卷积核对应位置的参数相乘再累加，得到一个输出值（输出值是 91×91 的图像）。卷积核的特点是连接数虽然很多，有 $91 \times 91 \times 10 \times 10$ 个连接，但是参数只有 $10 \times 10 = 100$ 个。参数数目大大减少，训练也变得容易了，并且不容易产生过拟合。当然，一个神经元只能提取一个特征，提取多个特征就需要多个卷积核。

图 11 揭示了对一幅 8×8 维图像使用卷积方法提取特征的示意过程。其中使用了 3×3 的卷积核，走遍图像中每个 3×3 的位置后，最终得到 6×6 维的输出图像。

深度学习的训练加速

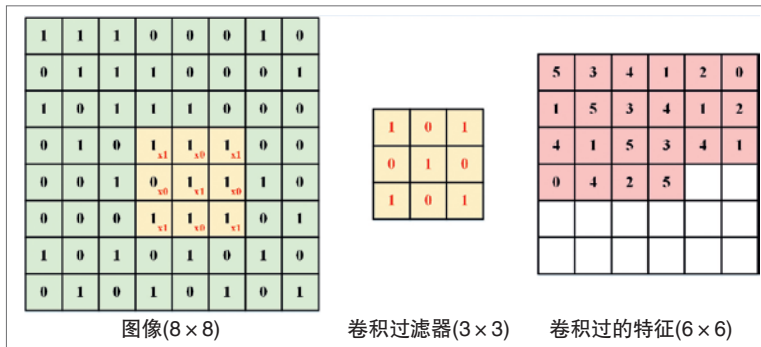


图11 8×8 图像的卷积过程示意图

深层模型训练需要各种技巧,例如网络结构的选取、神经元个数的设定、权重参数的初始化、学习率的调整、Mini-batch 的控制等等。即便对这些技巧十分精通,实践中也要多次训练,反复摸索尝试。此外,由于深层模型参数多,计算量大,训练数据的规模庞大,因此需要消耗很多计算资源。如果提高了训练速度,就可以在同样的时间内多尝试几个新主意,多调试几组参数,工作效率也会明显提升。对于大规模的训练数据和模型来说,训练加速可以将难以完成的任务变成可能。训练加速的方法如下。

GPU加速

矢量化编程是提高算法速度的一种有效方法。为了提升特定数值运算操作(如矩阵相乘、矩阵相加、矩阵-向量乘法等)的速度,数值计算和并行计算的研究人员已经努力了几十年。矢量化编程强调单一指令并行操作多条相似数据,形成单指令流多数据流的编程泛型。深层模型的算法,如反向传播算法、Auto-Encoder、卷积神经网络等,都可以写成矢量的形式。如果在单个CPU上执行,矢量运算会被展开成循环的形式,本质上还是串行执行。

GPU的众核体系结构包含几千个流处理器,可将矢量运算并行执行,大幅缩短计算时间。随着英伟达(NVIDIA)、超威(AMD)等公司GPU的大规模并行架构的不断推进,面向通用计算的图形处理器(General-Purposed GPU, GPGPU)已成为对可并行应用程序进行加速的重要手段。得益于GPU的众核体系结构,程序在该系统上的运行速度往往比单核CPU高几十倍乃至上千倍。目前GPU已经发展到了较为成熟的阶段,受益最大的是科学计算领域。典型的成功案例包括多体问题(N-Body Problem)、蛋白质分子建模、医学成像分析、金融计算、密码计算等。

利用GPU来训练深度神经网络,可以充分发

挥其数以千计的计算核心的高效并行计算能力。在使用海量训练数据的场景下,该处理器所耗费的时间大幅缩短,占用的服务器也更少。如果针对适当的深度神经网络进行合理优化,一块GPU卡相当于数十甚至上百台CPU服务器的计算能力。因此GPU已经成为业界在深度学习模型训练方面的首选解决方案。

数据并行

数据并行是指对训练数据做切分,同时采用多个模型实例,对多个分片的数据并行训练。

要完成数据并行,首先需要实现参数交换,通常由一个参数服务器来帮助完成。在训练过程中,多个训练相互独立,其结果即模型的变化量 ΔW 需要汇报给参数服务器,由参数服务器负责更新为最新的模型 $W' = W - \eta \cdot \Delta W$;然后再将最新的模型 W' 分发给训练程序,以便从新的起点开始训练(见图12)。

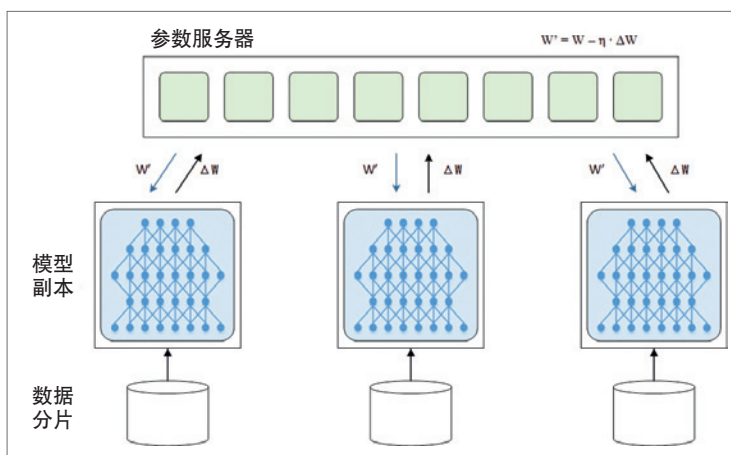


图12 数据并行的基本架构^[17]

数据并行可分为同步模式和异步模式。在同步模式中,所有训练程序同时训练一个批次的训练数据,完成后经过同步,再同时交换参数。之后,所有的训练程序就有了共同的新模型作为起点,再训练下一个批次。而异步模式中,训练程序完成一个批次的训练数据,立即和参数服务器交换参数,不考虑其他训练程序的状态。异步模式中一个训练程

序的最新结果不会立刻体现在其他训练程序中，直到他们进行下次的参数交换。

参数服务器只是一个逻辑上的概念，不一定部署为一台独立的服务器。有时候它会附属在某一个训练程序上，有时也会将参数服务器按照模型划分为不同的分片，对其分别部署。

模型并行

模型并行将模型拆分成几个分片，由几个训练单元分别持有，共同协作完成训练。当一个神经元的输入来自另一个训练单元上的神经元的输出时，产生通信开销（见图13）。

多数情况下，模型并行带来的通信开销和同步消耗超过数据并行，因此加速比也不及数据并行。但对于单机内存无法容纳的大模型来说，模型并行是一个很好的选择。令人遗憾的是，数据并行和模型并行都不能无限扩展。数据并行的训练程序太多时，不得不降低学习率，以保证训练过程的平稳；模型并行的分片太多时，神经元输出值的交换量会

急剧增加，效率也大幅下降。因此，同时进行模型并行和数据并行也是一种常见的方案。如图14所示，4个GPU分为两组，GPU0和GPU1为一组模型并行，GPU2和GPU3为另一组，每组模型并行在计算过程中交换输出值和残差。两组GPU之间形成数据并行，Mini-batch结束后交换模型权重。考虑到模型的蓝色部分由GPU0和GPU2持有，而黄色部分由GPU1和GPU3持有，因此只有同色的GPU之间需要交换权重。

计算集群

搭建CPU集群用于深度学习模型训练也是业界常用的解决方案，其优势在于利用大规模分布式计算集群的强大计算能力以及模型可分布式存储、参数可异步通信的特点，达到快速训练深层模型的目的。

CPU集群方案的基本架构包含用于执行训练任务的Worker、用于分布式存储分发模型参数服务器和用于协调整体任务的主控程序。CPU集群方案适合训练GPU内存难以容纳的大模型，以及稀疏连接神经网络。吴恩达和杰夫·迪恩在谷歌用1000台CPU服务器，完成了模型并行和Downpour SGD数据并行的深度神经网络训练^[17]。

结合GPU计算和集群计算技术，构建GPU集群正成为加速大规模深度神经网络训练的有效解决

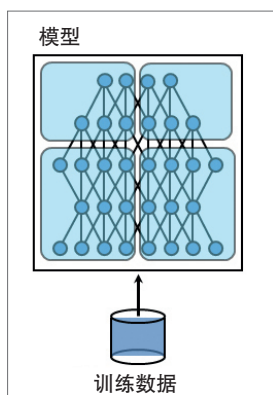


图13 模型并行的基本架构^[17]

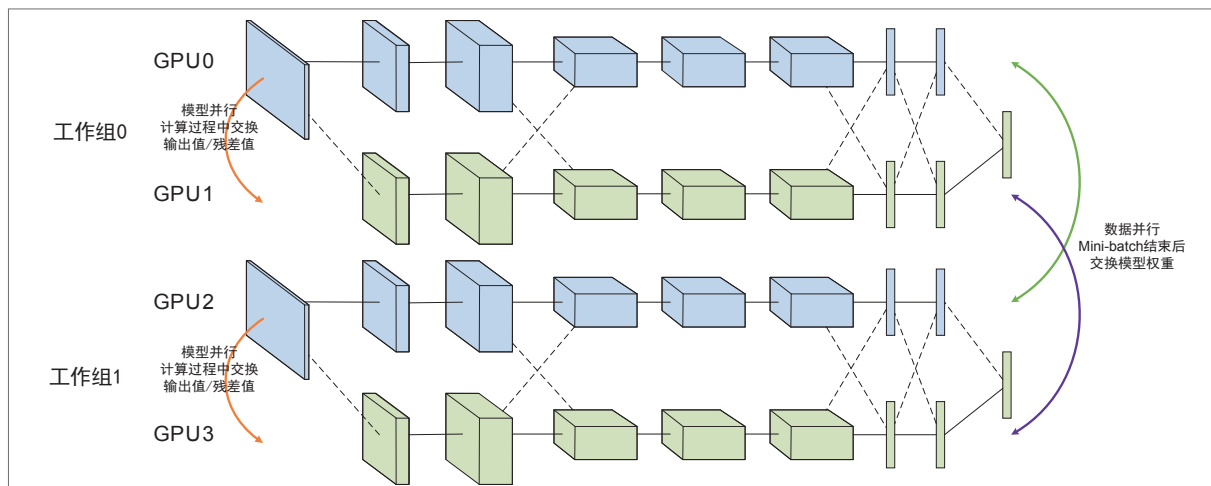


图14 4个图形处理器卡的数据并行和模型并行混合架构

方案。GPU 集群搭建在 CPU-GPU 系统之上,采用的是万兆网卡或无线带宽技术 (infiniband) 的网络通信设施以及树形拓扑等逻辑网络拓扑的结构。在发挥单节点具有较高计算能力的基础上,充分挖掘集群中多台服务器的协同计算能力,进一步加速大规模训练任务。

深度学习的软件工具及平台

目前,在深度学习系统实现方面,已有诸多较为成熟的软件工具和平台。

开源软件

在开源社区,主要有以下较为成熟的软件工具。

Kaldi 是一个提供给语音识别研究人员使用的、基于 C++ 和统一计算设备架构 (Compute Unified Device Architecture, CUDA) 的语音识别工具集^[22, 23]。Kaldi 既实现了用单个 GPU 加速的深度神经网络随机梯度下降 (Stochastic Gradient Descent, SGD) 训练,也实现了 CPU 多线程加速的深度神经网络随机梯度下降训练。

2012年发布的Cuda-convnet,基于C++/CUDA编写,采用反向传播算法的深度卷积神经网络实现^[24, 25],可支持单个 GPU 上的训练。基于其训练的深度卷积神经网络模型在 ImageNet LSVRC-2012 对图像按 1000 个类目分类,取得排名前 5 分类的 15% 错误率的结果^[3]。2014 年发布的版本可以支持多 GPU 上的数据并行和模型并行训练^[26]。

Caffe 提供了在 CPU 以及 GPU 上的快速卷积神经网络实现,同时提供训练算法,使用 NVIDIA K40 或 Titan GPU 可以 1 天完成多于 4000 万张图片的训练^[27, 28]。

Theano 提供了在深度学习数学计算方面的 Python 库。它整合了 NumPy 矩阵计算库,可以在 GPU 上运行,并提供良好的算法扩展性^[29, 30]。

OverFeat 是由美国纽约大学 CILVR 实验室开发的基于卷积神经网络系统,主要应用在图像识别和图像特征提取上^[31]。

Torch7 是一个为机器学习算法提供广泛支持的科学计算框架,其中的神经网络工具包实现了均方差代价函数、非线性激活函数和梯度下降训练神经网络的算法等基础模块,可以方便地配置实现目标多层神经网络,以开展训练实验^[32]。

工业界平台

在工业界,谷歌、脸谱、百度和腾讯等公司都实现了自己的软件框架。

谷歌的 DistBelief 系统是 CPU 集群实现的数据并行和模型并行框架,集群内使用上万个 CPU 核来训练多达 10 亿参数的深度神经网络模型。DistBelief 应用的主要算法有 Downpour SGD 和 L-BFGS,支持的目标应用有语音识别和 2.1 万类目的图像分类^[17]。

谷歌的 COTS HPC 系统是 GPU 实现的数据并行和模型并行框架,GPU 服务器间使用了无线带宽技术连接,并由 MPI (Message Passing Interface, 消息传递接口) 控制通信。该系统可以用 3 台 GPU 服务器在数天内完成对 10 亿个参数的深度神经网络训练^[33]。

脸谱实现了多 GPU 训练深度卷积神经网络的并行框架,结合数据并行和模型并行的方式来训练卷积神经网络模型,使用一台带有 4 块 NVIDIA Titan GPU 卡的服务器,可在数天内训练 ImageNet 的 1000 个分类网络^[34]。

百度搭建了 Paddle (Parallel Asynchronous Distributed Deep Learning, 并行异步分布式深度学习) 多机 GPU 训练平台^[35],将数据分布到不同机器,通过参数服务器 (parameter server) 协调各机器训练。Paddle 支持数据并行和模型并行。

腾讯深度学习平台 Mariana 是为加速深度学习模型训练而开发的并行化平台,包括深度神经网络的多 GPU 数据并行框架、深度卷积神经网络的多 GPU 模型并行和数据并行框架以及深度神经网络的 CPU 集群框架。Mariana 基于特定应用的训练场景,设计定制化的并行化训练平台,可支持语音识别和图像识别,并积极探索在广告推荐中的应用^[36]。

总结

近年来,人工智能领域掀起了深度学习的浪潮,从学术界到工业界都热情高涨。深度学习尝试解决人工智能中抽象认知的难题,在理论分析和应用方面都取得了很大的成功。深度学习是一种目前接近人脑的智能学习方法。

深度学习通过学习一种深层非线性网络结构,实现复杂函数逼近,展现了强大的学习数据集本质和高度抽象化特征的能力。逐层初始化等训练方法显著提升了深层模型的可学习性。与传统的浅层模型相比,深层模型经过若干层非线性变换,可带给模型强大的表达能力,从而具备为更复杂任务建模的条件。与人工特征工程相比,自动学习特征更能挖掘出数据中丰富的内在信息,并具备更强的可扩展性。深度学习顺应了大数据的趋势,有了充足的训练样本,复杂的深层模型可以充分发挥其潜力,挖掘出海量数据中蕴含的丰富信息。强有力的基础设施和定制化的并行计算框架,让以往不可想象的训练任务加速完成,为深度学习走向实用奠定了坚实的基础。■



蒋杰

腾讯数据平台部负责人,带领团队构建腾讯大数据平台。主要研究方向为机器学习等。zeus@tencent.com



金 津

腾讯数据平台部精准推荐中心高级工程师。主要研究方向为机器学习、深度学习、大数据中的精准广告推荐技术。xingjin@tencent.com

参考文献

- [1] Geoffrey E. Hinton, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science*. 2006 Jul 28;313(5786):504~7.
- [2] Geoffrey E. Hinton, Simon Osindero, Yee-Whye Teh, A fast learning algorithm for deep belief nets *Neural*

Compute, 18(7), 1527~54 (2006).

- [3] ImageNet Classification with Deep Convolutional Neural Networks, Alex Krizhevsky, Ilya Sutskever, Geoffrey E Hinton, NIPS 2012.
- [4] Q.V. Le, M.A. Ranzato, R. Monga, M. Devin, K. Chen, G.S. Corrado, J. Dean, A.Y. Ng. Building high-level features using large scale unsupervised learning. ICML, 2012.
- [5] Rick Rashid, Speech Recognition Breakthrough for the Spoken, Translated Word <http://www.youtube.com/watch?v=Nu-nlQqFCKg>.
- [6] NYU Deep Learning Professor LeCun Will Lead Facebook's New Artificial Intelligence Lab. <http://techcrunch.com/2013/12/09/facebook-artificial-intelligence-lab-lecun/>.
- [7] A Primer on Deep Learning, <http://www.datarobot.com/blog/a-primer-on-deep-learning/>.
- [8] Bruno A. Olshausen & David J. Field, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, 1996;381(13). http://www.cs.ubc.ca/~little/cpsc425/olshausen_field_nature_1996.pdf.
- [9] Andrew Ng. Machine Learning and AI via Brain simulations, <https://forum.stanford.edu/events/2011slides/plenary/2011plenaryNg.pdf>.
- [10] Deep Networks Overview http://ufldl.stanford.edu/wiki/index.php/Deep_Networks:_Overview.
- [11] Back propagation algorithm http://ufldl.stanford.edu/wiki/index.php/Backpropagation_Algorithm.
- [12] Support Vector Machine http://en.wikipedia.org/wiki/Support_vector_machine.
- [13] Logistic Regression http://en.wikipedia.org/wiki/Logistic_regression.
- [14] The Nobel Prize in Physiology or Medicine 1981. http://www.nobelprize.org/nobel_prizes/medicine/laureates/1981/.
- [15] Stanford deep learning tutorial. http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial.
- [16] 余凯, 深度学习-机器学习的新浪潮, Technical News 程序天下事, <http://blog.csdn.net/datoubo/article/details/8577366>.
- [17] Dean, J., Corrado, G.S., Monga, R., et al, Ng, A. Y. Large Scale Distributed Deep Networks. In Proceedings of the Neural Information Processing Systems (NIPS' 12) (Lake Tahoe, Nevada, United States, December 3-6, 2012). Curran Associates, Inc, 57 Morehouse Lane, Red Hook, NY, 2013, 1223~1232.
- [18] Geoffrey Hinton; UCL Tutorial on: Deep Belief Nets.

- [19] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time-series. In M. A. Arbib, editor, The Handbook of Brain Theory and Neural Networks. MIT Press, 1995.
- [20] Introduction to Convolutional neural network. http://en.wikipedia.org/wiki/Convolutional_neural_network.
- [21] Krizhevsky, Alex. ImageNet Classification with Deep Convolutional Neural Networks. Retrieved 17 November 2013.
- [22] Kaldi project <http://kaldi.sourceforge.net/>.
- [23] Povey, D., Ghoshal, A. Boulianne, G., et al, Vesely, K. Kaldi. The Kaldi Speech Recognition Toolkit. in Proceedings of IEEE 2011 Workshop on Automatic Speech Recognition and Understanding(ASRU 2011) (Hilton Waikoloa Village, Big Island, Hawaii, US, December 11-15, 2011). IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.
- [24] cuda-convent <https://code.google.com/p/cuda-convnet/>.
- [25] Krizhevsky, A., Sutskever, I., and Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the Neural Information Processing Systems (NIPS' 12) (Lake Tahoe, Nevada, United States, December 3–6, 2012). Curran Associates, Inc, 57 Morehouse Lane, Red Hook, NY, 2013, 1097~1106.
- [26] Krizhevsky, A. Parallelizing Convolutional Neural Networks. in tutorial of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014). (Columbus, Ohio, USA, June 23-28, 2014). 2014.
- [27] caffe <http://caffe.berkeleyvision.org/>.
- [28] Jia, Y. Q. Caffe: An Open Source Convolutional Architecture for Fast Feature Embedding. <http://caffe.berkeleyvision.org> (2013).
- [29] Theano <https://github.com/Theano/Theano>.
- [30] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley and Y. Bengio. Theano: A CPU and GPU Math Expression Compiler. Proceedings of the Python for Scientific Computing Conference (SciPy) 2010. June 30 - July 3, Austin, TX.
- [31] Overfeat <http://cilvr.nyu.edu/doku.php?id=code:start>.
- [32] Torch7 <http://torch.ch>.
- [33] Coates, A., Huval, B., Wang, T., Wu, D. J., Ng, A. Y. Deep learning with COTS HPC systems. In Proceedings of the 30th International Conference on Machine Learning (ICML' 13) (Atlanta, Georgia, USA, June 16–21, 2013). JMLR: W&CP volume 28(3), 2013, 1337~1345.
- [34] Yadan, O., Adams, K., Taigman, Y., Ranzato, M. A. Multi-GPU Training of ConvNets. arXiv:1312.5853v4 [cs.LG] (February 2014).
- [35] Kaiyu, Large-scale Deep Learning at Baidu, ACM International Conference on Information and Knowledge Management (CIKM 2013).
- [36] aaronzou, Mariana深度学习在腾讯的平台化和应用实践.
- [37] Convolutional Neural Networks (LeNet) - DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab. Retrieved 31 August 2013.
- [38] Bengio, Learning Deep Architectures for AI, http://www.iro.umontreal.ca/~bengioy/papers/ftml_book.pdf.
- [39] Deep Learning, <http://deeplearning.net/>.
- [40] Deep Learning, <http://www.cs.nyu.edu/~yann/research/deep/>.
- [41] Introduction to Deep Learning. http://en.wikipedia.org/wiki/Deep_learning.
- [42] Google的猫脸识别:人工智能的新突破. <http://www.36kr.com/p/122132.html>.
- [43] Andrew Ng's talk video: <http://techtalks.tv/talks/machine-learning-and-ai-via-brain-simulations/57862/>.
- [44] Invited talk A Tutorial on Deep Learning by Dr. Kai Yu <http://vipl.ict.ac.cn/News/academic-report-tutorial-deep-learning-dr-kai-yu>.