**深度学习研究综述**

# 摘要

机器学习是通过计算模型和算法从数据中学习规律的科学，在各种需要从复杂数据中挖掘规律的领域中有大量应用，是如今人工智能领域核心的技术之一。同时多种深度神经网络在大量机器学习问题上取得令人瞩目的成果，形成了机器学习领域亮眼的一个新分支。近年来，中美等国家、谷歌等高科技公司纷纷加大对人工智能的投入尤其是深度学习领域。本文对深度学习代表性方法的背景原理，经典网络及模型进行了综述，在此基础上，进一步分析了不断涌现出来的新型卷积神经网络和循环神经网络。然后本文总结了深度学习在人工智能众多领域中的应用，包括语音处理、计算机视觉和自然语言处理等。本文组织结构为：深度学习发展历史，基本概念核心算法(吕光奥)，基本网络结构以及经典模型(牟鑫)，改进的网络结构，深度学习的典型应用，优化方法与现有瓶颈(白璠)

# 1.

# 引言

从现象中发现规律，是人类智能核心的能力之一，人们也很早就开始研究如何用数学方法 来分析数据中的规律。从1930年代 Fisher 线性 判别和 1950 年代感知器算法开始，诞生了模式识别学科，研究从数据中学习分类信息的数学方法，形成了早的机器学习研究。“机器学习” 这个术语也是 1950 年代末提出来的，初并不专指从数据中学习，更多地包括了机器推理等典人工智能问题，直到 20世纪后期才逐渐被用来专指从数据中学习。现在，这两个术语的含义已经非常接近，模式识别专指对数据的分类，机器学习则学习数据中的各种规律尤其是分类规律，而“深度学习”是机器学习中新发展起来 的一类方法的总称。

很多模式识别方法和统计学习方法，如线性判别、近邻法、罗杰斯特回归、决策树、支持向 量机等，已经在很广泛的问题上取得了成功，如 广告点击率预]、希格斯子信号识别[、基于基因表达的疾病分型等。这些统计学习方法往往是直接根据特征对样本进行分类，不进行特征变换或只进行一次特征变换或选择。与深度学习方法相比，这些方法中特征变换较少，或者依赖于上游处理来对特征进行变换，所以被有些人称作“浅层模型”或“浅层学习方法”。

这些浅层模型在很多应用上取得了成功，但是也存在很大局限，即模型的效果非常依赖于上游提供的特征。一方面，构造特征的过程是很难的，需要对问题有丰富的先验知识和对原始数据详尽的了解；另一方面，在先验知识不充分的情况下，需要人为构建的特征数目庞大，如某些广告点击率预测算法中人工构造的特征维数高达数亿维。

深度学习是一种深层的机器学习模型，其深度体现在对特征的多次变换上。常用的深度学习模型为多层神经网络，神经网络的每一层都会将输入进行非线性映射，通过多层非线性映射的堆叠，可以在深层神经网络中计算出非常抽象的特征来帮助分类。比如在用于图像分析的卷积神经 网络中，将原始图像的像素值直接输入，第一层神经网络可以视作是边缘的检测器，而第二层神经网络则可以检测边缘的组合，得到一些基本模块，再深层的网络会将这些基本模块进行组合，终检测出待识别目标。深度学习的出现使得人们在很多应用中不再需要单独对特征进行选择与变换，而是将原始数据输入到模型中，由模型通过学习给出适合分类的特征表示

2016年3月，“人工智能”一词被写入中国“十三五” 规划纲要，2016年10 月美国政府发布《美国国家人工智能研究与发展战略规划》文件．Google、Microsoft、Facebook、百度、腾讯、阿里巴巴等各大互联网公司也纷纷加大对人工智能的投入．各类人工智能创业公司层出不穷，各种人工智能应用逐渐改变人类的生活．深度学习是目前人工智能的重点研究领域之一，应用于人工智能的众多领域，包括语音处理、计算机视觉、自然语言处理等。

当前，深度学习是机器学习领域热门的分支，并且有多个高度集成化的方法平台可以让使用者无需对方法原理充分了解就可以搭建程序进行实验和应用。本文接下来的章节尝试对 典型的深度学习方法原理进行综述，对深度学习经典网络模型进行介绍以及需要研究的问题和展望。

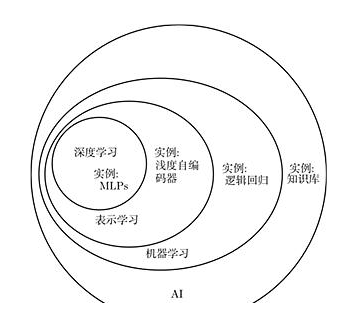
图1.1 深度学习的范围概念

图1.1展示了深度学习即是一种表示学习，也是一种机器学习，可以用于许多的AI方法。

# 深度学习发展历史

深度学习有着悠久而丰富的历史，但随着许多不同哲学观点的渐渐消逝，与之对应的名称也渐渐尘封。深度学习看似是一个全新的领域，事实上，深度学习的历史可以追溯到 20 世纪 40 年代，只不过因为在目前流行的前几年它还是相对冷门的，同时也因为它被赋予了许多不同的名称（其中大部分已经不再使用），最近才成为众所周知的“深度学习”。这个领域已经更换了很多名称，它反映了不同的研究人员和不同观点的影响。一般认为，到目前为止，深度学习经历了三次发展浪潮，20世纪40年代到60年代的深度学习雏形出现在控制论中，20世纪80到90年代，以联结主义为代表，2006年以后以深度学习为名复苏。目前，随着可用的训练数据量不断增加，深度学习变得更加有用。随着时间的推移，针对深度学习的计算机软硬件基础设施都有所改善，深度学习模型的规模也随之增长。随着时间的推移，深度学习已经解决日益复杂的应用，并且精度不断提高。

## 2.1控制论

我们今天知道的一些最早的学习算法，旨在模拟生物学习的计算模型，即大脑怎样学习或为什么能学习的模型。其结果是深度学习以人工神经网络之名而淡去。当时，深度学习模型被认为是受生物大脑所启发而设计出来的系统。尽管有些机器学习的神经网络有时被用来理解大脑功能，但它们一般都没有设计成生物功能的真实模型。深度学习的神经观点受两个主要思想启发：一个想法是，大脑作为例子证明智能行为是可能的，因此，概念上，建立智能的直接途径是逆向大脑背后的计算原理，并复制其功能；另一种看法是，理解大脑和人类智能背后的原理也非常有趣，因此机器学习模型除了解决工程应用的能力，如果能让人类对这些基本的科学问题有进一步的认识，也将会很有用。

第1次浪潮开始于20世纪40年代到20世纪60年代的控制论，随着生物学习理论的发展和第一个模型的实现，能实现单个神经元的训练。第2次浪潮开始于 1980—1995 年间的联结主义方法，可以使用反向传播训练具有一两个隐藏层的神经网络。当前第 3 次浪潮，也就是深度学习，大约始于 2006 年，并且于 2016 年以图书的形式出现。另外，前两次浪潮类似地出现在书中的时间比相应的科学活动晚得多。现代术语“深度学习”超越了目前机器学习模型的神经科学观点。它诉诸于学习多层次组合这一更普遍的原理，这一原理也可以应用于那些并非受神经科学启发的机器学习框架。现代深度学习最早的前身是从神经科学的角度出发的简单线性模型。这些模型设计为使用一组 n 个输入（x1,··· ,xn），并将它们与一个输出 y 相关联。这些模型希望学习一组权重 （w1,··· ,wn），并计算它们的输出f（x,w） = x1w1 + ··· + xnwn。

**控制论的原理：**

第一次神经网络研究浪潮称为控制论。神经元是脑功能的早期模型。该线性模型通过检验函数f（x,w）的正负来识别两种不同类别的输入。显然，模型的权重需要正确设置后才能使模型的输出对应于期望的类别。这些权重可以由操作人员设定。20 世纪 50 年代，感知机成为第一个能根据每个类别的输入样本来学习权重的模型。

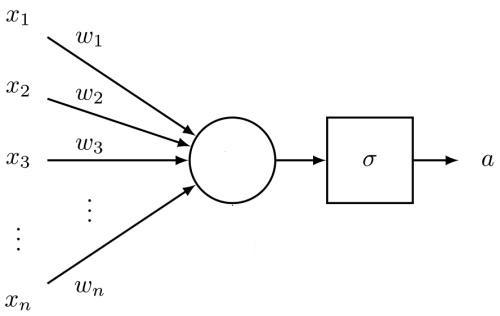


图2.1单个神经元

大约在同一时期，自适应线性单元简单地返回函数f（x）本身的值来预测一个实数，并且它还可以学习从数据预测这些数。这些简单的学习算法大大影响了机器学习的现代景象。用于调节ADALINE权重的训练算法是被称为随机梯度下降的一种特例。稍加改进后的随机梯度下降算法仍然是当今深度学习的主要训练算法。基于感知机和 ADALINE 中使用的函数f（x,w）的模型称为线性模型。尽管在许多情况下，这些模型以不同于原始模型的方式进行训练，但仍是目前最广泛使用的机器学习模型。

**线性模型的局限性：**

线性模型有很多局限性。最著名的是，它们无法学习异或（XOR）函数，即f（[0,1],w）= 1和f（[1,0],w）= 1，但f（[1,1],w）= 0和f（[0,0],w）= 0。观察到线性模型这个缺陷的批评者对受生物学启发的学习普遍地产生了抵触。这导致了神经网络热潮的第一次大衰退。现在，神经科学被视为深度学习研究的一个重要灵感来源，但它已不再是该领域的主要指导。如今神经科学在深度学习研究中的作用被削弱，主要原因是我们根本没有足够的关于大脑的信息来作为指导去使用它。要获得对被大脑实际使用算法的深刻理解，我们需要有能力同时监测（至少是）数千相连神经元的活动。我们不能够做到这一点，所以我们甚至连大脑最简单、最深入研究的部分都还远远没有理解。

**机器模拟大脑的失败原因：**

我们能够从神经科学得到一些粗略的指南。仅通过计算单元之间的相互作用而变得智能的基本思想是受大脑启发的。新认知机受哺乳动物视觉系统的结构启发，引入了一个处理图片的强大模型架构，它后来成为了现代卷积网络的基础。目前大多数神经网络是基于一个称为整流线性单元的神经单元模型。原始认知机受我们关于大脑功能知识的启发，引入了一个更复杂的版本。我们知道，真实的神经元计算着与现代整流线性单元非常不同的函数，但更接近真实神经网络的系统并没有导致机器学习性能的提升。此外，虽然神经科学已经成功地启发了一些神经网络架构，但我们对用于神经科学的生物学习还没有足够多的了解，因此也就不能为训练这些架构用的学习算法提供太多的借鉴。现代深度学习从许多领域获取灵感，特别是应用数学的基本内容，如线性代数、概率论、信息论和数值优化。尽管一些深度学习的研究人员引用神经科学作为灵感的重要来源，然而其他学者完全不关心神经科学。值得注意的是，了解大脑是如何在算法层面上工作的尝试确实存在且发展良好。这项尝试主要被称为“计算神经科学”，并且是独立于深度学习的领域。研究人员在两个领域之间来回研究是很常见的。深度学习领域主要关注如何构建计算机系统，从而成功解决需要智能才能解决的任务，而计算神经科学领域主要关注构建大脑如何真实工作的、比较精确的模型。

## 2.2联结主义

20 世纪 80 年代，神经网络研究的第二次浪潮在很大程度上是伴随一个被称为联结主义并行分布处理潮流而出现的。联结主义是在认知科学的背景下出现的。认知科学是理解思维的跨学科途径，即它融合多个不同的分析层次。20 世纪 80 年代初期，大多数认知科学家研究符号推理模型。尽管这很流行，但符号模型很难解释大脑如何真正使用神经元实现推理功能。联结主义的中心思想是，当网络将大量简单的计算单元连接在一起时可以实现智能行为。这种见解同样适用于生物神经系统中的神经元，因为它和计算模型中隐藏单元起着类似的作用。

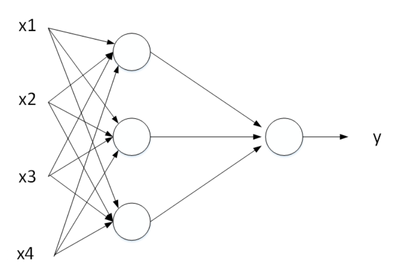


图2.2多层神经元

在 20 世纪 80 年代的联结主义期间形成的几个关键概念在今天的深度学习中仍然是非常重要的。其中一个概念是分布式表示。其思想是：系统的每一个输入都应该由多个特征表示，并且每一个特征都应该参与到多个可能输入的表示。例如，假设我们有一个能够识别红色、绿色或蓝色的汽车、卡车和鸟类的视觉系统，表示这些输入的其中一个方法是将 9 个可能的组合：红卡车、红汽车、红鸟、绿卡车等使用单独的神经元或隐藏单元激活。这需要 9 个不同的神经元，并且每个神经必须独立地学习颜色和对象身份的概念。改善这种情况的方法之一是使用分布式表示，即用 3 个神经元描述颜色，3 个神经元描述对象身份。这仅仅需要6个神经元而不是9个，并且描述红色的神经元能够从汽车、卡车和鸟类的图像中学习红色，而不仅仅是从一个特定类别的图像中学习。

联结主义潮流的另一个重要成就是反向传播在训练具有内部表示的深度神经网络中的成功使用以及反向传播算法的普及。20世纪90年代，研究人员在使用神经网络进行序列建模的方面取得了重要进展如今，LSTM 在许多序列建模任务中广泛应用，包括 Google 的许多自然语言处理任务。神经网络研究的第二次浪潮一直持续到20世纪90年代中期。基于神经网络和其他AI技术的创业公司开始寻求投资，其做法野心勃勃但不切实际。当AI研究不能实现这些不合理的期望时，投资者感到失望。同时，机器学习的其他领域取得了进步。比如，核方法和图模型都在很多重要任务上实现了很好的效果。这两个因素导致了神经网络热潮的第二次衰退，并一直持续到2007年。

在此期间，神经网络继续在某些任务上获得令人印象深刻的表现。加拿大高级研究所 （CIFAR）通过其神经计算和自适应感知（NCAP）研究计划帮助维持神经网络研究。该计划联合了分别由 Geoffrey Hinton、Yoshua Bengio和 Yann LeCun 领导的多伦多大学、蒙特利尔大学和纽约大学的机器学习研究小组。这个多学科的CIFAR NCAP 计划还包括了神经科学家、人类和计算机视觉专家。在那个时候，人们普遍认为深度网络是难以训练的。现在我们知道，20 世纪80年代就存在的算法能工作得非常好，但是直到2006年前后都没有体现出来。这可能仅仅由于其计算代价太高，而以当时可用的硬件难以进行足够的实验。

## 2.3深度神经网络

神经网络研究的第三次浪潮始于2006年的突破。Geoffrey Hinton表明名为“深度信念网络”的神经网络可以使用一种称为“贪婪逐层预训练”的策略来有效地训练。其他CIFAR附属研究小组很快表明，同样的策略可以被用来训练许多其他类型的深度网络，并能系统地帮助提高在测试样例上的泛化能力。神经网络研究的这一次浪潮普及了“深度学习”这一术语，强调研究者现在有能力训练以前不可能训练的比较深的神经网络，并着力于深度的理论重要性上。

此时，深度神经网络已经优于与之竞争的基于其他机器学习技术以及手工设计功能的 AI 系统。神经网络的第三次发展浪潮仍在继续，尽管深度学习的研究重点在这一段时间内发生了巨大变化。第三次浪潮已开始着眼于新的无监督学习技术和深度模型在小数据集的泛化能力，但目前更多的兴趣点仍是比较传统的监督学习算法和深度模型充分利用大型标注数据集的能力。

既然人工神经网络的第一个实验在20世纪50年代就完成了，为什么深度学习直到最近才被认为是关键技术？确实，要从一个深度学习算法获得良好的性能需要一些技巧。幸运的是，随着训练数据的增加，所需的技巧正在减少。目前在复杂的任务中达到人类水平的学习算法，与20世纪80年代努力解决玩具问题的学习算法几乎是一样的，尽管我们使用这些算法训练的模型经历了变革，即简化了极深架构的训练。

最重要的新进展是，现在我们有了这些算法得以成功训练所需的资源，日益增加的训练数据量，有能力训练规模更大的神经网络。

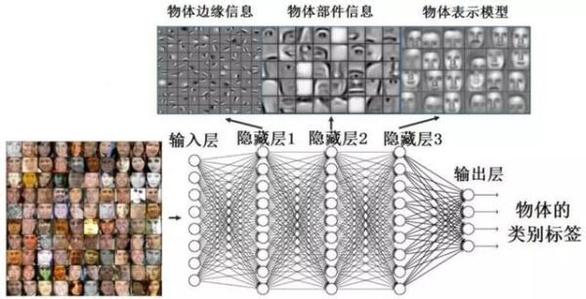


图2.3深度学习的学习过程

深度学习的发展过程中大事件：

1、IBM的深蓝1997年击败世界冠军。

2、2012年，AlexNet采用ReLU激活函数，从根本上解决了梯度消失问题，并采用GPU极大的提高了模型的运算速度。

3、由斯坦福大学著名的吴恩达教授和世界顶尖计算机专家Jeff Dean共同主导的深度神经网络——DNN技术在图像识别领域取得了惊人的成绩，在ImageNet评测中成功的把错误率从26％降低到了15％。

4、2014年，Facebook基于深度学习技术的DeepFace项目，在人脸识别方面的准确率已经能达到97%以上，跟人类识别的准确率几乎没有差别。

5、2016年，随着谷歌公司基于深度学习开发的AlphaGo以4:1的比分战胜了国际顶尖围棋高手李世石，深度学习的热度一时无两。

6、2017年，基于强化学习算法的AlphaGo升级版AlphaGo Zero横空出世。其采用“从零开始”、“无师自通”的学习模式，以100:0的比分轻而易举打败了之前的AlphaGo。

# 深度学习概念及算法

## 3.1机器学习拟合问题

机器学习中的逼近目标函数过程。监督式机器学习通常理解为逼近一个目标函数f()，此函数映射输入变量X到输出变量Y，Y=f(X)。这种特性描述可以用于定义分类和预测问题和机器学习算法的领域。

从训练数据中学习目标函数的过程中，我们必须考虑的问题是模型在预测新数据时的泛化性能。泛化好坏是很重要的，因为我们收集到的数据只是样本，其带有噪音并且是不完全的。在机器学习中，我们描述从训练数据学习目标函数的学习过程为归纳性的学习。归纳与特别的样本中学习到通用的概念有关，而这就是监督式机器学习致力于解决的问题。这与推演不同，其主要是另一种解决问题和寻求从通常的规则中找寻特别的内容。泛化即是，机器学习模型学习到的概念在它处于学习的过程中时模型没有遇见过的样本时候的表现。好的机器学习模型的模板目标是从问题领域内的训练数据到任意的数据上泛化性能良好。这让我们可以在未来对模型没有见过的数据进行预测。

在机器学习领域中，当我们讨论一个机器学习模型学习和泛化的好坏时，我们通常使用术语：过拟合和欠拟合。过拟合和欠拟合是机器学习算法表现差的两大原因。

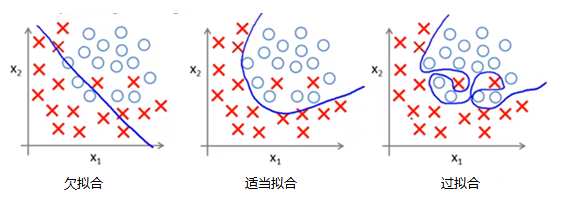


图3.1拟合问题

在统计学中，拟合指的是你逼近目标函数的远近程度。这个术语同样可以用于机器学习中，因为监督式机器学习算法的目标也是逼近一个未知的潜在映射函数，其把输入变量映射到输出变量。

1）机器学习中的过拟合指的是模型对于训练数据拟合程度过当的情况。当某个模型过度的学习训练数据中的细节和噪音，以至于模型在新的数据上表现很差，我们称过拟合发生了。这意味着训练数据中的噪音或者随机波动也被当做概念被模型学习了。而问题就在于这些概念不适用于新的数据，从而导致模型泛化性能的变差。过拟合更可能在无参数非线性模型中发生，因为学习目标函数的过程是易变的具有弹性的。同样的，许多的无参数器学习算法也包括限制约束模型学习概念多少的参数或者技巧。例如，决策树就是一种无参数机器学习算法，非常有弹性并且容易受过拟合训练数据的影响。这种问题可以通过对学习过后的树进行剪枝来解决，这种方法就是为了移除一些其学习到的细节。

2）机器学习中的欠拟合指的是模型在训练和预测时表现都不好的情况。一个欠拟合的机器学习模型不是一个良好的模型并且由于在训练数据上表现不好这是显然的。欠拟合通常不被讨论，因为给定一个评估模型表现的指标的情况下，欠拟合很容易被发现。矫正方法是继续学习并且试着更换机器学习算法。虽然如此，欠拟合与过拟合形成了鲜明的对照。

3）机器学习中好的拟合，选择一个正好介于欠拟合和过拟合之间的模型。这就是我们学习的目标，但是实际上很难达到。为了理解这个目标，我们可以观察正在学习训练数据机器学习算法的表现。我们可以把这个过程划分为分别是训练过程和测试过程。随着时间进行，算法不断地学习，模型在训练数据和测试数据上的错误都在不断下降。但是，如果我们学习的时间过长的话，模型在训练数据上的表现将继续下降，这是因为模型已经过拟合并且学习到了训练数据中的不恰当的细节以及噪音。同时，测试数据集上的错误率开始上升，也即是模型的泛化能力在下降。这个完美的临界点就处于测试集上的错误率开始上升时，此时模型在训练集和测试集上都有良好的表现。

过拟合和欠拟合可以导致很差的模型表现。但是到目前为止大部分机器学习实际应用时的问题都是过拟合。过拟合是个问题因为训练数据上的机器学习算法的评价方法与我们最关心的实际上的评价方法，也就是算法在位置数据上的表现是不一样的。通常有两种手段可以帮助限制过拟合，找到一个完美的临界点：重采样方法和验证集方法。

（1）使用重采样来评价模型效能。保留一个验证数据集。最流行的重采样技术是k折交叉验证。指的是在训练数据的子集上训练和测试模型k次，同时建立对于机器学习模型在未知数据上表现的评估。

（2）验证集只是训练数据的子集，你把它保留到你进行机器学习算法的最后才使用。在训练数据上选择和调谐机器学习算法之后，我们在验证集上在对于模型进行评估，以便得到一些关于模型在未知数据上的表现的认知。对于机器学习，使用交叉验证在未知数据上进行验证模型效能是一种良好的标准。如果你拥有数据，使用验证集也是一种良好的实践。

## 3.2深度学习和经典机器学习的区别

深度学习虽然属于机器学习，但是和传统的机器学习还是有很大的区别。主要在以下几方面：

1、数据依赖性

深度学习与传统机器学习之间最重要的区别随着数据规模的增加而表现出来。当数据很小时，深度学习算法表现不佳。这是因为深度学习算法需要大量数据才能完美理解。另一方面，在这种情况下，传统的机器学习算法及其手工制作的规则占主导地位。

2、硬件依赖性

深度学习算法在很大程度上依赖于高端机器，这与传统的机器学习算法相反，后者可以在低端机器上运行。这是因为深度学习算法的要求包括GPU，它是其工作的一个组成部分。深度学习算法固有地执行大量矩阵乘法运算。可以使用GPU有效地优化这些操作，因为GPU是为此目的而构建的。

3、特征工程

特征工程是将领域知识放入特征提取器的创建过程中的过程，以降低数据的复杂性并使模式对于学习算法更加可见。就时间和专业知识而言，这个过程既困难又昂贵。在机器学习中，大多数应用的功能需要由专家识别，然后根据域和数据类型进行手动编码。例如，特征可以是像素值，形状，纹理，位置和方向。大多数机器学习算法的性能取决于识别和提取特征的准确程度。

深度学习算法尝试从数据中学习高级特征。这是深度学习的一个非常独特的部分，是传统机器学习的重要一步。因此，深度学习减少了为每个问题开发新特征提取器的任务。就像CNN将尝试学习低层特征，例如早期层中的边缘和线条，然后是人脸的部分面部，然后是面部的高级表示。

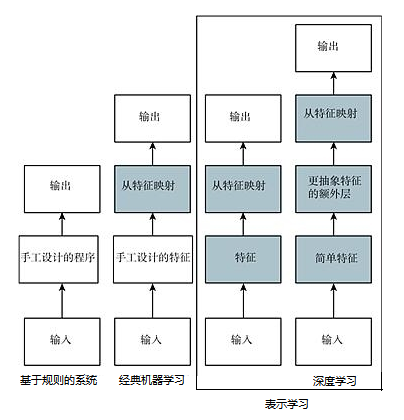


图3.2 算法学习过程

4、问题解决方法

当使用传统的机器学习算法解决问题时，通常建议将问题分解为不同的部分，单独解决它们并将它们组合起来得到结果。相反，深度学习主张端到端地解决问题。

例如：假设有多个对象检测任务。任务是确定对象是什么以及它在图像中的位置。在典型的机器学习方法中，可以将问题分为两个步骤：对象检测和对象识别。首先，可以使用像grabcut这样的边界框检测算法来浏览图像并查找所有可能的对象。然后，在所有已识别的对象中，将使用对象识别算法（如带有HOG的SVM）来识别相关对象。相反，在深度学习方法中，您将完成端到端的过程。例如，在YOLO网（这是一种深度学习算法）中，传入一个图像，它将给出该位置以及对象的名称。

5、执行时间

通常，深度学习算法需要很长时间才能进行训练。这是因为深度学习算法中有如此多的参数，训练它们需要比平时更长的时间。最先进的深度学习算法ResNet大约需要两周时间才能从头开始训练。虽然机器学习相对需要更少的时间进行训练，但从几秒到几小时不等。这在测试时间完全颠倒了。在测试时，深度学习算法运行时间要少得多。然而，如果将其与k近邻（一种机器学习算法）进行比较，则测试时间会随着数据大小的增加而增加。虽然这不适用于所有机器学习算法，但其中一些算法的测试时间也很短。

6、可解释性

最后，我们将可解释性作为比较机器学习和深度学习的一个因素。假设我们使用深度学习来为论文提供自动评分。它在得分方面的表现非常出色，接近人类表现。但这是一个问题。它没有揭示为什么它给出了这个分数。事实上，你可以通过数学方法找出深层神经网络的哪些节点被激活，但我们不知道神经元应该建模的是什么以及这些神经元层共同做了什么。所以我们无法解释结果。另一方面，像决策树这样的机器学习算法为我们提供了清晰的规则，为什么它选择了它所选择的内容，因此特别容易理解其背后的推理。因此，决策树和线性/逻辑回归等算法主要用于工业中的可解释性。

## 3.3激活函数

为了解决非线性的分类或回归问题，我们的激活函数必须是非线性的函数，另外我们使用基于梯度的方式来训练模型，因此激活函数也必须是连续可导的。

目前主流的激活函数有Logistic、Tanh以及Relu函数

1. Logistic函数（又称为sigmoid函数）

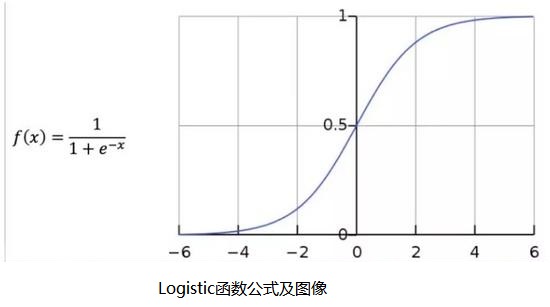


图3.3logistic函数公式及图像

Logistic函数在定义域上单调递增，值域为（0,1），越靠近两端，函数值的变化越平缓。因为Logistic函数简单易用，以前的神经网络经常使用它作为激活函数，但是由于Logistic函数存在一些缺点，使得现在的神经网络已经很少使用它作为激活函数了。它的缺点之一是容易饱和，从函数图像可以看到，Logistic函数只在坐标原点附近有很明显的梯度变化，其两端的函数变化非常平缓，这会导致我们在使用反向传播算法更新参数的时候出现梯度消失的问题，并且随着网络层数的增加问题会越严重。

2、Tanh函数（双曲正切激活函数）

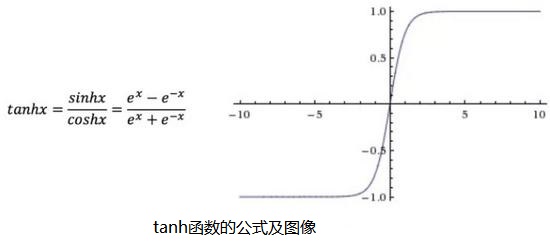


图3.4tanh函数的公式及图像

Tanh函数很像是Logistic函数的放大版，其值域为（-1,1）。在实际的使用中，Tanh函数要优于Logistic函数，但是Tanh函数也同样面临着在其大部分定义域内都饱和的问题。

3、ReLU函数（又称修正线性单元或整流线性单元）

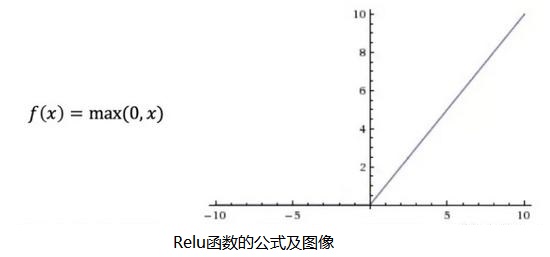


图3.5 Relu函数的公式及图像

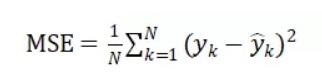
ReLU激活函数是目前最受欢迎，也是使用最多的激活函数，它的收敛速度相较于Logistic函数和Tanh函数要快很多，ReLU函数在轴左侧的值恒为零，这使得网络具有一定的稀疏性，从而减小参数之间的依存关系，缓解过拟合的问题，并且ReLU函数在轴右侧的部分导数是一个常数值1，因此其不存在梯度消失的问题。但是ReLU函数也有一些缺点，例如ReLU的强制稀疏处理虽然可以缓解过拟合问题，但是也可能产生特征屏蔽过多，导致模型无法学习到有效特征的问题。

## 3.4损失函数

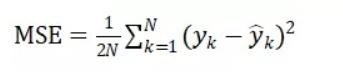
损失函数（LossFunction）又称为代价函数（Cost Function），它是神经网络设计中的一个重要部分。损失函数用来表征模型的预测值与真实类标之间的误差，深度学习模型的训练就是使用基于梯度的方法最小化损失函数的过程。损失函数的选择与输出单元的选择也有着密切的关系。

1. 均方误差损失函数

均方误差（MeanSquared Error，MSE）是一个较为常用的损失函数，我们用预测值和实际值之间的距离（即误差）来衡量模型的好坏，为了保证一致性，我们通常使用距离的平方。在深度学习算法中，我们使用基于梯度的方式来训练参数，每次将一个批次的数据输入到模型中，并得到这批数据的预测结果，再利用这批预测结果和实际值之间的距离更新网络的参数。均方误差损失函数将这一批数据的误差的期望作为最终的误差值，均方误差的公式如下：

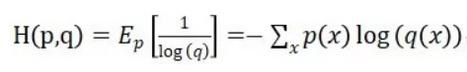


上式中为样本数据的实际值，为模型的预测值。为了简化计算，我们一般会在均方误差的基础上乘以，作为最终的损失函数：

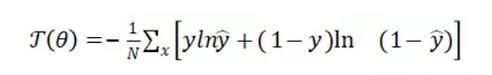


2、交叉熵损失函数

交叉熵（Cross Entropy）损失函数使用训练数据的真实类标与模型预测值之间的交叉熵作为损失函数，相较于均方误差损失函数其更受欢迎。假设我们使用均方误差这类二次函数作为代价函数，更新神经网络参数的时候，误差项中会包含激活函数的偏导。在前面介绍激活函数的时候我们有介绍，Logistic等激活函数很容易饱和，这会使得参数的更新缓慢，甚至无法更新。交叉熵损失函数求导不会引入激活函数的导数，因此可以很好地避免这一问题，交叉熵的定义如下：



上式中为样本数据的真实分布，为模型预测结果的分布。以二分类问题为例，交叉熵损失函数的形式如下：



上式中为真实值，为预测值。对于多分类问题，我们对每一个类别的预测结果计算交叉熵后求和即可。

# 基本网络及经典模型

## 4.1卷积神经网络

卷积神经网络（Convolutional Neural Networks, CNN）是一类包含卷积计算且具有深度结构的前馈神经网络（Feedforward Neural Networks），是深度学习（deep learning）的代表算法之一。由于卷积神经网络能够进行平移不变分类（shift-invariant classification），因此也被称为“平移不变人工神经网络（Shift-Invariant Artificial Neural Networks, SIANN）”。

卷积神经网络仿造生物的视知觉（visual perception）机制构建，可以进行监督学习和非监督学习，其隐含层内的卷积核参数共享和层间连接的稀疏性使得卷积神经网络能够以较小的计算量对格点化（grid-like topology）特征，例如像素和音频进行学习、有稳定的效果且对数据没有额外的特征工程（feature engineering）要求。

第一个卷积神经网络是1987年由Alexander Waibel等提出的时间延迟网络（Time Delay Neural Network, TDNN）。TDNN是一个应用于语音识别问题的卷积神经网络，使用FFT预处理的语音信号作为输入，其隐含层由2个一维卷积核组成，以提取频率域上的平移不变特征。由于在TDNN出现之前，人工智能领域在反向传播算法（Back-Propagation, BP）的研究中取得了突破性进展，因此TDNN得以使用BP框架内进行学习。在原作者的比较试验中，TDNN的表现超过了同等条件下的隐马尔可夫模型（Hidden Markov Model, HMM），而后者是二十世纪80年代语音识别的主流算法。

1988年，Wei Zhang提出了第一个二维卷积神经网络：平移不变人工神经网络（SIANN），并将其应用于检测医学影像。独立于Zhang (1988)，Yann LeCun在1989年同样构建了应用于图像分类的卷积神经网络，即LeNet的最初版本。LeNet包含两个卷积层，2个全连接层，共计6万个学习参数，规模远超TDNN和SIANN，且在结构上与现代的卷积神经网络十分接近。LeCun (1989)对权重进行随机初始化后使用了随机梯度下降（Stochastic Gradient Descent, SGD）进行学习，这一策略被其后的深度学习研究广泛采用。此外，LeCun (1989)在论述其网络结构时首次使用了“卷积”一词，“卷积神经网络”也因此得名。

LeCun (1989)的工作在1993年由贝尔实验室（AT&T Bell Laboratories）完成代码开发并被大量部署于NCR（National Cash Register Coporation）的支票读取系统。但总体而言，由于数值计算能力有限和学习样本不足，这一时期为各类图像处理问题设计的卷积神经网络停留在了研究阶段，没有得到广泛应用。

在LeNet的基础上，1998年Yann LeCun及其合作者构建了更加完备的卷积神经网络LeNet-5并在手写数字的识别问题中取得成功。LeNet-5沿用了LeCun (1989) 的学习策略并在原有设计中加入了池化层对输入特征进行筛选。LeNet-5及其后产生的变体定义了现代卷积神经网络的基本结构，其构筑中交替出现的卷积层-池化层被认为有效提取了输入图像的平移不变特征。LeNet-5的成功使卷积神经网络的应用得到关注，微软在2003年使用卷积神经网络开发了光学字符读取（Optical Character Recognition, OCR）系统。其它基于卷积神经网络的应用研究也得到展开，包括人像识别、手势识别等。

2006年后，随着深度学习理论的完善，尤其是逐层学习和参数微调（fine-tuning）技术的出现，卷积神经网络开始快速发展，在结构上不断加深，各类学习和优化理论得到引入。自2012年的AlexNet开始，各类卷积神经网络多次成为ImageNet大规模视觉识别竞赛（ImageNet Large Scale Visual Recognition Challenge, ILSVRC）的优胜算法，包括2013年的ZFNet、2014年的VGGNet、GoogLeNet和2015年的ResNet。

### 4.1.1 卷积神经网络的结构

卷积神经网络的输入层可以处理多维数据，常见地，一维卷积神经网络的输入层接收一维或二维数组，其中一维数组通常为时间或频谱采样；二维数组可能包含多个通道；二维卷积神经网络的输入层接收二维或三维数组；三维卷积神经网络的输入层接收四维数组。由于卷积神经网络在计算机视觉领域有广泛应用，因此许多研究在介绍其结构时预先假设了三维输入数据，即平面上的二维像素点和RGB通道。

与其它神经网络算法类似，由于使用梯度下降进行学习，卷积神经网络的输入特征需要进行标准化处理。具体地，在将学习数据输入卷积神经网络前，需在通道或时间/频率维对输入数据进行归一化，若输入数据为像素，也可将分布于[0,255]的原始像素值归一化至[0,1]区间。输入特征的标准化有利于提升算法的运行效率和学习表现。

卷积神经网络的隐含层包含卷积层、池化层和全连接层3类常见构筑，在一些更为现代的算法中可能有Inception模块、残差块（residual block）等复杂构筑。在常见构筑中，卷积层和池化层为卷积神经网络特有。卷积层中的卷积核包含权重系数，而池化层不包含权重系数，因此在文献中，池化层可能不被认为是独立的层。以LeNet-5为例，3类常见构筑在隐含层中的顺序通常为：输入-卷积层-池化层-卷积层-池化层-全连接层-输出。

卷积层的功能是对输入数据进行特征提取，其内部包含多个卷积核，组成卷积核的每个元素都对应一个权重系数和一个偏差量（bias vector），类似于一个前馈神经网络的神经元（neuron）。卷积层内每个神经元都与前一层中位置接近的区域的多个神经元相连，区域的大小取决于卷积核的大小，在文献中被称为“感受野（receptive field）”，其含义可类比视觉皮层细胞的感受野。卷积核在工作时，会有规律地扫过输入特征，在感受野内对输入特征做矩阵元素乘法求和并叠加偏差量。在线性卷积的基础上，一些卷积神经网络使用了更为复杂的卷积，包括平铺卷积（tiled convolution）、反卷积（deconvolution）和扩张卷积（dilated convolution）。平铺卷积的卷积核只扫过特征图的一部份，剩余部分由同层的其它卷积核处理，因此卷积层间的参数仅被部分共享，有利于神经网络捕捉输入图像的旋转不变（shift-invariant）特征。反卷积或转置卷积（transposed convolution）将单个的输入激励与多个输出激励相连接，对输入图像进行放大。由反卷积和向上池化层（up-pooling layer）构成的卷积神经网络在图像语义分割（semantic segmentation）领域有重要应用，也被用于构建包卷积自编码器（Convolutional AutoEncoder, CAE）。扩张卷积在线性卷积的基础上引入扩张率以提高卷积核的感受野，从而获得特征图的更多信息，在面向序列数据使用时有利于捕捉学习目标的长距离依赖（long-range dependency）。使用扩张卷积的卷积神经网络主要被用于自然语言处理（Natrual Language Processing, NLP）领域，例如机器翻译、语音识别等。

在卷积层进行特征提取后，输出的特征图会被传递至池化层进行特征选择和信息过滤。池化层包含预设定的池化函数，其功能是将特征图中单个点的结果替换为其相邻区域的特征图统计量。池化层选取池化区域与卷积核扫描特征图步骤相同，由池化大小、步长和填充控制。

卷积神经网络中的全连接层等价于传统前馈神经网络中的隐含层。全连接层通常搭建在卷积神经网络隐含层的最后部分，并只向其它全连接层传递信号。特征图在全连接层中会失去3维结构，被展开为向量并通过激励函数传递至下一层。

在一些卷积神经网络中，全连接层的功能可部分由全局均值池化（global average pooling）取代，全局均值池化会将特征图每个通道的所有值取平均，即若有7×7×256的特征图，全局均值池化将返回一个256的向量，其中每个元素都是7×7，步长为7，无填充的均值池化。

卷积神经网络中输出层的上游通常是全连接层，因此其结构和工作原理与传统前馈神经网络中的输出层相同。对于图像分类问题，输出层使用逻辑函数或归一化指数函数（softmax function）输出分类标签。在物体识别（object detection）问题中，输出层可设计为输出物体的中心坐标、大小和分类。在图像语义分割中，输出层直接输出每个像素的分类结果。

### 4.1.2 LeNet-5模型

LeNet-5是一个应用于图像分类问题的卷积神经网络，其学习目标是从一系列由32×32×1灰度图像表示的手写数字中，识别和区分0-9。LeNet-5的隐含层由2个卷积层、2个池化层构筑和2个全连接层组成，按如下方式构建：

1.(3×3)×1×6的卷积层（步长为1，无填充），2×2均值池化（步长为2，无填充），tanh激励函数

2.(5×5)×6×16的卷积层（步长为1，无填充），2×2均值池化（步长为2，无填充），tanh激励函数

3.2个全连接层，神经元数量为120和84。

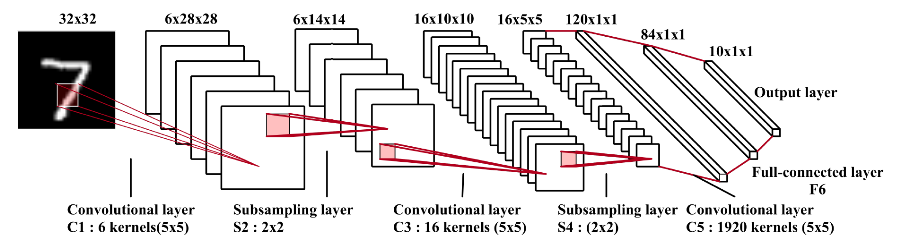


图4.1 LeNet-5模型架构图

从现代深度学习的观点来看，LeNet-5规模很小，但考虑LeCun et al. (1998)的数值计算条件，LeNet-5是完整的深度学习算法。LeNet-5使用双曲正切函数作为激励函数，使用均方差（Mean Squared Error, MSE）作为误差函数并对卷积操作进行了修改以减少计算开销，这些设置在随后的卷积神经网络算法中已被更优化的方法取代。

### 4.1.3 AlexNet模型

AlexNet是2012年ILSVRC图像分类和物体识别算法的优胜者，也是LetNet-5之后对现代卷积神经网络产生重要影响的算法。AlexNet的隐含层由5个卷积层、3个池化层和3个全连接层组成，按如下方式构建：

1.(11×11)×3×96的卷积层（步长为4，无填充，ReLU），3×3极大池化（步长为2、无填充），LRN

2.(5×5)×96×256的卷积层（步长为1，相同填充，ReLU），3×3极大池化（步长为2、无填充），LRN

3.(3×3)×256×384的卷积层（步长为1，相同填充，ReLU）

4.(3×3)×384×384的卷积层（步长为1，相同填充，ReLU）

5.(3×3)×384×256的卷积层（步长为1，相同填充，ReLU），3×3极大池化（步长为2、无填充）

6.3个全连接层，神经元数量为4096、4096和1000

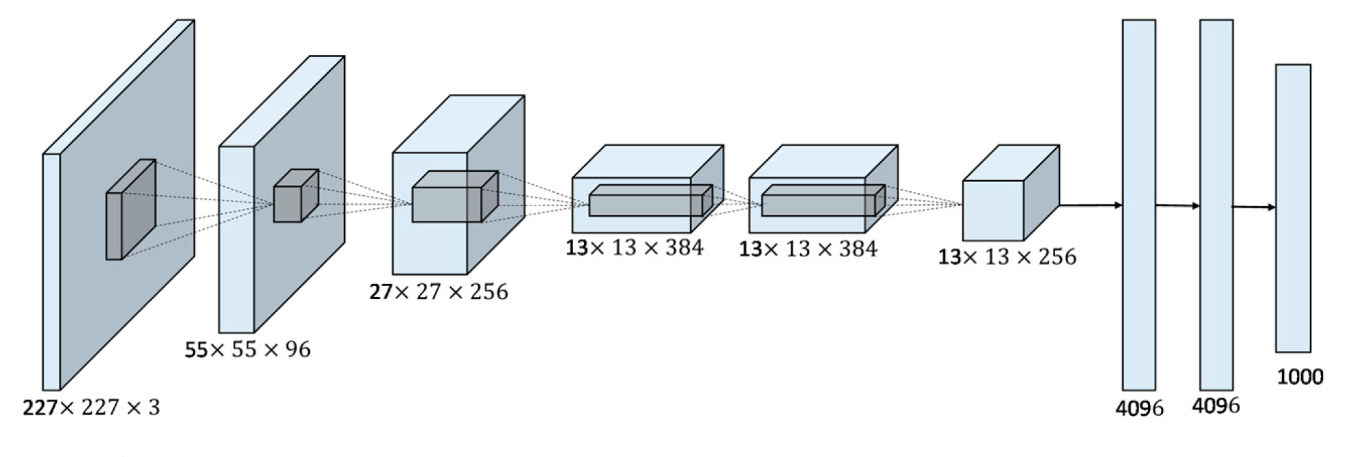


图4.2 AlexNet模型架构图

AlexNet在卷积层中选择ReLU作为激励函数，使用了随机失活，和数据增强（data data augmentation）技术，这些策略在其后的卷积神经网络中被广泛使用。AlexNet也是首个基于GPU进行学习的卷积神经网络，Krizhevsky (2012) 将AlexNet按结构分为两部分，分别在两块GPU设备上运行。此外AlexNet的1-2部分使用了局部响应归一化（local response normalization, LRN），在2014年后出现的卷积神经网络中，LRN已由分批归一化取代。

### 4.1.4 VGGNet模型

VGGNet是牛津大学视觉几何团队（Visual Geometry Group, VGG）开发的一组卷积神经网络算法，包括VGG-11、VGG-11-LRN、VGG-13、VGG-16和VGG-19。其中VGG-16是2014年ILSVRC物体识别算法的优胜者，其规模是AlexNet的2倍以上并拥有规律的结构，这里以VGG-16为例介绍其构筑。VGG-16的隐含层由13个卷积层、3个全连接层和5个池化层组成，按如下方式构建：

1.(3×3)×3×64的卷积层（步长为1，相同填充，ReLU），(3×3)×64×64的卷积层（步长为1，相同填充，ReLU），2×2极大池化（步长为2、无填充）

2.(3×3)×64×128的卷积层（步长为1，相同填充，ReLU），(3×3)×128×128的卷积层（步长为1，相同填充，ReLU），2×2极大池化（步长为2、无填充）

3.(3×3)×128×256的卷积层（步长为1，相同填充，ReLU），(3×3)×256×256的卷积层（步长为1，相同填充，ReLU），(3×3)×256×256的卷积层（步长为1，相同填充，ReLU），2×2极大池化（步长为2、无填充）

4.(3×3)×256×512的卷积层（步长为1，相同填充，ReLU），(3×3)×512×512的卷积层（步长为1，相同填充，ReLU），(3×3)×512×512的卷积层（步长为1，相同填充，ReLU），2×2极大池化（步长为2、无填充）

5.(3×3)×512×512的卷积层（步长为1，相同填充，ReLU），(3×3)×512×512的卷积层（步长为1，相同填充，ReLU），(3×3)×512×512的卷积层（步长为1，相同填充，ReLU），2×2极大池化（步长为2、无填充）

6.3个全连接层，神经元数量为4096、4096和1000

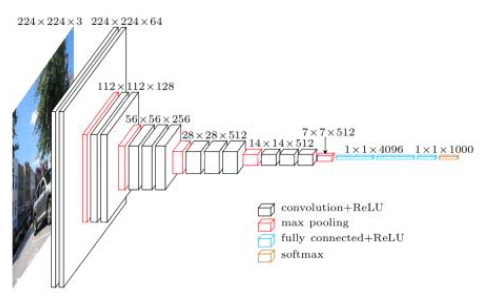


图4.3 VggNet模型架构图

VGGNet构筑中仅使用3×3的卷积核并保持卷积层中输出特征图尺寸不变，通道数加倍，池化层中输出的特征图尺寸减半，简化了神经网络的拓扑结构并取得了良好效果。

### 4.1.5 GoogLeNet模型

GoogLeNet是2014年ILSVRC图像分类算法的优胜者，是首个以Inception模块进行堆叠形成的大规模卷积神经网络，共有四个版本，即Inception v1、Inception v2、Inception v3、Inception v4。这里以Inception v1为例介绍。首先，Inception v1的Inception模块被分为四部分：

1.N1个(1×1)×C的卷积核

2.B3个(1×1)×C的卷积核（BN，ReLU），N3个(3×3)×96的卷积核（步长为1，相同填充，BN，ReLU）

3.B5个(1×1)×C的卷积核（BN，ReLU），N5个(5×5)×16的卷积核（步长为1，相同填充，BN，ReLU）

4.3×3的极大池化（步长为1，相同填充），Np个(1×1)×C的卷积核（BN，ReLU）

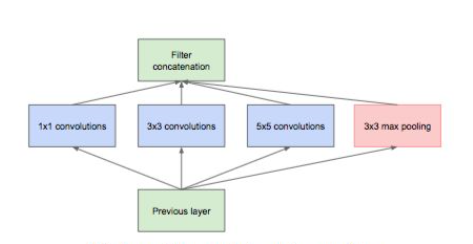


图4.4 Inception模块架构图

在此基础上，对3通道的RGB图像输入，Inception v1按如下方式构建 [41] ：

1.(7×7)×3×64的卷积层（步长为2，无填充，BN，ReLU），3×3的极大池化（步长为2，相同填充），LRN

2.(3×3)×64×192的卷积层（步长为1，相同填充，BN，ReLU），LRN，3×3极大池化（步长为2，相同填充）

3.Inception模块（N1=64，B3=96，N3=128，B5=16，N5=32，Np=32）

4.Inception模块（N1=128，B3=128，N3=192，B5=32，N5=96，Np=64）

5.3×3极大池化（步长为2，相同填充）

6.Inception模块（N1=192，B3=96，N3=208，B5=16，N5=48，Np=64）

7.旁枝：5×5均值池化（步长为3，无填充）

8.Inception模块（N1=160，B3=112，N3=224，B5=24，N5=64，Np=64）

9.Inception模块（N1=128，B3=128，N3=256，B5=24，N5=64，Np=64）

10.Inception模块（N1=112，B3=144，N3=288，B5=32，N5=64，Np=64）

11.旁枝：5×5均值池化（步长为3，无填充）

12.Inception模块（N1=256，B3=160，N3=320，B5=32，N5=128，Np=128）

13.Inception模块（N1=384，B3=192，N3=384，B5=48，N5=128，Np=128）

14.全局均值池化，1个全连接层，神经元数量为1000，权重40%随机失活

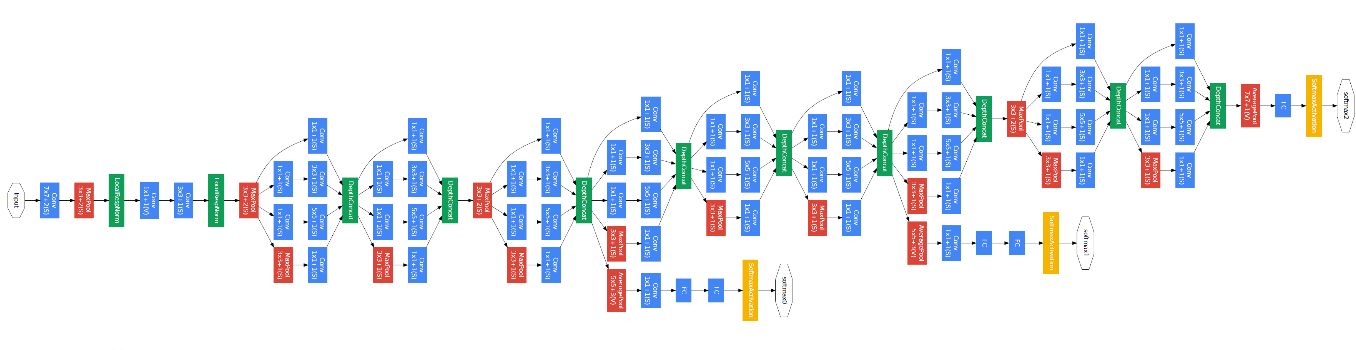


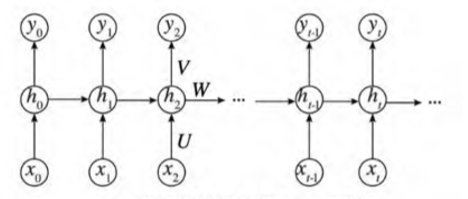
图4.5 GoogLeNet模型架构图

GoogLeNet中Inception模块的提出促进了卷积神经网络的发展，并启发了一些更为现代的算法，例如2017年提出的Xception。Inception v1的另一特色是其两个旁枝输出，旁枝和主干的所有输出会通过指数归一化函数得到结果，对神经网络起正则化的作用。

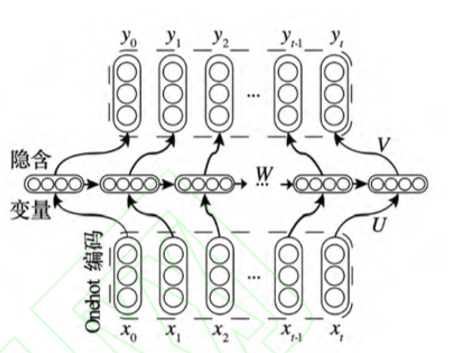
## 4.2递归神经网络

递归神经网络（recursive neural network， RNN）有别于前面所提到的前馈类型的神经网络，其主要目的是对序列型数据进行建模，例如语音 识别、语言翻译、自然语言理解、音乐合成等序列数据，希望在推断过程中保留序列上下文的信息，所以其隐节点中存在反馈环，即当前时刻的隐含节点值不仅与当前节点的输入有关，也与前一时刻的隐含节点值有关。

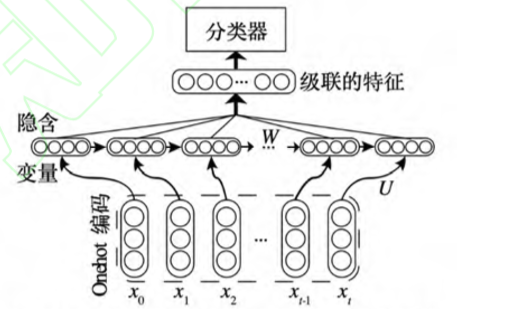
递归结构神经网络结构如下图所示：



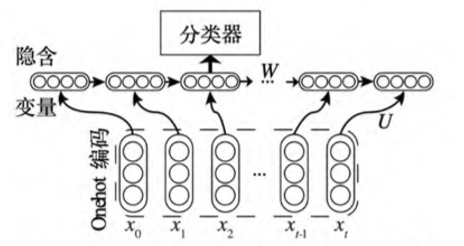
网络的输入为序列型数据，记为｛x1,x2…xt-1,xt…｝下标 t 是时刻,每一时刻的输入数据都为一个向量。在处理文本等非数值的序列数据时，需要将非数值的输入（如文本中的单词）转变为向量表示，常用的表示方法包括独热（one-hot）编码或用 word2vec 将单词表示为高维向量等，如下图所示。也可在输入层与隐含节点之间加入一层映射层，来训练针对当前任务的词向量。



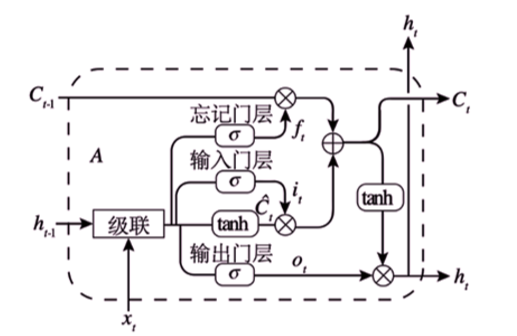
每一时刻都有一个隐含状态｛h1,h2…ht-1,ht…｝，这些隐含状态中就纪录了当前时刻之前的序列中所包含的信息，每一时刻的隐含节点需要综合之前时刻的信息以及当前时刻输入中包含的信息，将二者结合起来传递给下一时刻。对于不同类型的问题，递归神经网络可以采用不同种类的输出，比如对于序列的分类问题，可以将所有时刻的隐含状态收集到一起作为序列特征输入到分类器中进行分类，如下图所示



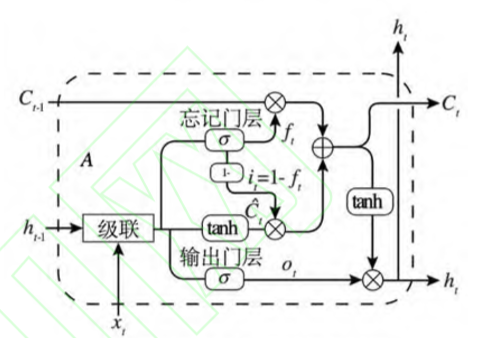
而对于序列生成或语言模型问题，每一时刻都应有相应的输出，如下图，可以将每一时刻的隐含状态作为特征进行分类



为了在减少参数数目的同时使递归神经网 络能处理不同长度的输入序列，网络中的参数针 对输入序列的每个时刻都是相等的，这也使得参 数的梯度计算比前馈神经网络略复杂了一些。对于递归神经网络的训练，需要使用随时间的反向算法计算参数的梯度，其本质上仍然利用梯度的链式传递法则。将RNN沿时间展开后，可以视为一个很深层的前馈神经网络，所以存在严重的梯度消散现象，导致RNN无法学习到数据中的长程依赖关系。为了减轻RNN中的梯度消散现象，可以从单元结构与优化两方面着手改进。单元结构方面，长短时记忆模型（Long Short-Term Memory – LSTM）添加了额外的隐含状态来记忆序列的信息，使用三个门来控制当前时刻的输入对记忆的影响，如下图所示。



通过这样的改造，记忆能够更加通畅地在时间序列中传递，从而记住更久远之前的信息。长短时 记忆模型被发明之后，也出现了诸多变种。实用的是门控递归单元（Gated Recurrent Unit）。门控递归单元合并了长短时记忆模型中的两种隐含状态，如下图所示，将控制门的个数减少到了两个，使得收敛所需的时间有所下降，经过实验验证，门控递归单元相比长短时记忆模型 几乎没有性能的损失。改进优化方面，研究表明在参数初始化合适的情况下，递归神经网络也能较好的学习到长程依赖关系。



上面介绍的递归神经网络只能用于处理输出数据定长的情况。对于某些实际问题，如语言翻译、问答系统等，对给定输入需要给出序列的输出。针对这一类问题，人们提出了 seq2seq及 encoder-decoder模型。两种模型都使用了两个递归神经网络，一个用于收集输入序列中的信息，将输入序列的信息用向量进行表示，比如用后一个时刻的隐状态作为输入序列的向量表示，另一个递归神经网络则用于生成序列。 每一时刻都要综合输入序列的信息以及已产生序列中的信息来决定下一个单词的概率分布，利用采样决定生成的单词，然后可将生成的单词重新输入网络得到新的概率分布，如此循环即可生成整条序列。 在很多序列相关的问题中，输出往往只与输入的某些片段有较强的联系。比如在机器翻译问题中，输出单词的大信息量来自于与输出单词意义相同的词，如将“knowledge is power”翻译为 “知识就是力量”，其中“力量”一词之所以被生成 完全是从“power”中获取的信息。人们为建模这 种关系引入了注意力机制，即输出序列的每个词都只将注意力放在输入序列的一个区域而 不是完整的输入序列中。该机制能够大大提高递归神经网络的效果，被广泛使用于各类序列学习 任务中。分级注意记忆（hierarchical attentive memory）进一步将节点组织成二叉树的形式， 加快了搜索效率，且能够增强训练数据与测试数据长度不一致的情况下的推广能力。 在网络结构方面，双向的递归神经网络使用从前向后以及从后向前两条链对时序数据建模，用于刻画序列的上下文信息而不仅仅是过去时刻的信息。深层递归神经网络对递归神 经网络进行叠加，将一层递归神经网络的隐含状态序列作为下一层递归神经网络的输入，可以学 习到更深层次的特征。

# 5.深度学习的优化技巧

各种深度神经网络已经在大量应用中展现了出色的效果，一些典型的模型和算法已经比较成型并且有一些公开的框架可以使用，这大大方便和加快了各种深度学习方法的应用。但针对一个实际的问题，深度学习的求解过程中存在大量的技巧需要摸索，有效地使用一些技巧能够改善网络的收敛性以及网络的推广能力。

深度学习的参数求解本质上是一个优化问题，不同的优化方法各有优劣。常用的优化方法大体上可以按照其收敛性分为一阶优化算法与二阶优化算法。一阶优化算法是以目标函数 相对于待优化参数的一阶导数（梯度）作为优化的依据，二阶优化算法则同时考虑了二阶导数信 息。一阶优化算法中常用的当属以BP算法为代表的梯度下降法及其变种。梯度下降法每轮迭代中都计算参数的梯度，并将参数向负梯度方向移动一段距离来更新参数值。根据每次计算梯度时取用的样本数不同分为梯度下降、随机梯度下降、批量梯度下降等。梯度下降每轮迭代计算所有样本的梯度平均，这样可以保证每次移动都必定能优化目标函数，但梯度计算耗时长。相对地， 随机梯度下降每次只选取一个样本计算梯度，速度快而且有一定的跳出局部优的能力，但目标函数将会波动剧烈。为了缓解随机梯度下降梯度变化剧烈的问题，人们引入了动量（momentum） 机制。动量的引入使得计算得到的梯度只起到微调参数更新方向的作用，类似于小球滚下山坡，方向相似的梯度能够让动量快速积累且参数更新方向不太发生变化，有利于目标函数的收敛。批量梯度下降法综合了随机梯度下降法与梯度 下降法的优点，选取

训练集中的一部分计算梯度和，以此平衡计算速度与算法稳定性。 在梯度下降类算法中，学习率（亦称步长） 即每一轮学习时参数更新幅度的选择是很关键的一点，学习率过低算法收敛过慢，而学习率过高则容易不收敛。因此，人们发展出一些自适应的梯度下降法，能够在学习过程中根据历史的参数更新信息自动调节学习率，如 AdaGrad、 RMSProp、AdaDelta、Adam等。 二阶优化算法考虑了目标函数的二阶导数信息，也就是目标函数在当前参数附近的曲率，使得参数的更新方向估计得更加准确，在某些问题上能够求解一阶优化算法不能解决的问题。常用的二阶优化算法包括牛顿法、共轭梯度法、 BFGS 算法、L-BFGS算法等。二阶优化算法的主要差别体现在Hessian矩阵逆的计算或近似上， 有文章指出，使用大规模集群并行化计算时， L-BFGS 与共轭梯度法能够取得比随机梯度下降 更快的收敛速度。也有文章采用不估计 Hessian 矩阵的二阶优化算法，并在自编码机、递归神经网络上取得了较好的效果。

在优化问题中，参数初始值的选择是很关键的。早期的神经网络一旦结构复杂就无法保证推广性能的一大原因是初始值选取无有效方法，深度置信网络（DBN）等采用非监督学习方法进行参数初始化（即预训练），使得深度神经网络了较好的实用性[94]。以往常用方法是用均值为0、方差较小的高斯分布或均匀分布来进行初始化，这样的初始化方法无法保证变量的方差在传播的过程中相等，会导致变量值逐渐增大或是逐减小，当变量值都很小时，都集中在 sigmoid 函数的线性区内，也就失去了层数增加的意义，而当变量值都很大时，变量都处于饱和区内，梯度减小不利于收敛。Glorot 等给出了一种方法，在使用 Sigmoid 或是 tanh 作为激活函数时，能够确保变量的方差在前向传播与反向传播的过程中都近似相等。He 等做了类似的推导，给出了 ReLU 作为激活函数时，深度神经网络的参数初始化形式。

激活函数的选择对模型的性能、收敛速度都有着很大的影响。早期被广泛使用的激活函数是Sigmoid函数，由于其在两侧的导数趋近于0， 被称为软饱和函数。软饱和性会使得网络的梯度难以向回传播，当网络的后几层很快收敛到饱和区后，网络的前几层仍然停留在随机初始化的状态而得不到训练，造成网络的推广性能较差。tanh 同样是一种软饱和的激活函数，相比 sigmoid 函数，由于其输出的均值比 sigmoid 函数更接近于0，随机梯度下降能够更趋近于自然梯度（natural gradient），故其收敛速度更快。 深度神经网络直接监督式训练的主要突破点是 采用了 ReLU 函数。它至今仍是使用广泛的激活函数，ReLU 函数在x>0时导数恒为1，梯度不会衰减，从而缓解梯度消散。ReLU 还能使得神经网络具有稀疏表达的能力，可以提升网络性能。但其在x<0梯度硬饱和，权重无法更新，且由于 ReLU 的均值恒定大于 0，故会影响网络的收敛性。为了解决神经元死亡问题，人们发展了PRELU,ELU等relu的推广，可以使网络具有更好的收敛性能。Maxout使用一个小的神经网络作为非线性单元，理论上在隐节点数目足够的情况下能够近似任意的激活函 数，且其基本不存在神经元死亡现象，但为此需要付出更大的参数数目与计算量。

过拟合或过学习是影响神经网络类方法推广能力的主要原因，对于深度神经网络，舍弃 （Dropout）法是一种常用的避免过拟合方法。 Dropout 是指在每轮训练过程中，随机地让网络的部分隐含节点不工作，即以概率将隐含节点的输出置零，这些隐含节点的参数也暂时不更新。 这种舍弃训练也属于一种正则化方法。这样每轮训练的网络结构都是不同的，终进行分类时使 用整个网络进行分类，类似于取了不同分类器平均，与集成学习中的 bagging自举聚合 （Bootstrap aggregating）方法有异曲同工之妙使用舍弃训练使网络避免某些神经元共同激活， 削弱了神经元之间的联合适应性，可以增强推广能力。也有观点认为，舍弃训练可以理解为 数据增强（data augmentation）的一种形式。 因为其易于实现，且可应用于各类不同的网络结构中，故被广泛使用。但是舍弃训练会降低网络的有效节点数目，所以应用时网络的宽度也需要相应增加，这使得在样本数目极少时表现不佳，另外，应用舍弃训练也会使得网络的训练时间上升为原来的 2~3 倍。除了舍弃训练方法，还有一些类似的方法或改进方法，比如DropConnect方法随机地将隐含节点的一些输入连接置零，理论上可以获得更好的模型平均效果，自适应舍弃（Adaptive Dropout）方法能够根据上一层的输出结果寻找优的舍弃率等等。

2015年，人们提出了批量归一化（batch normalization）算法，使用该方法可以选择较 大的初始学习率使网络快速收敛，并且可以提网络的推广性能，某些程度上可以代替舍弃训练等正则化方法。算法的核心思想很简单，机器学习的本质是学习数据的分布，归一化能够使训练 数据与测试数据分布相同，从而可以提升推广性能，而且在批量梯度下降中，归一化能够使得模型不必去适应学习每轮不同的输入数据分布，从而提升训练速度。但在训练一般的深度神经网络时，只有输入层能满足分布相同的条件，经过线性变换之后，每层隐含层的输入分布就不再稳定，受之前数层的参数影响。为了解决这一问题，使神经网络的每一层输入都拥有相同的分布，引入Batch Normalization方法。归一化可能会使得特 征的表达能力减弱，比如原本数据的分布是分布在sigmoid函数的两端，有着较强的判别能力，经过归一化之后变为分布在0附近，相当于前一层的学习结构被抹消了。

类似的这些优化策略与技巧，都是研究者针对不同的数据和实验情况提出的，面对一个特的实际问题，并没有办法事先确定哪种策略是优策略。但是了解这些策略和技巧的思路和原理将有利于我们在面对实际问题时更快找到适当的策略或者发展出新的策略。

# 6.典型应用

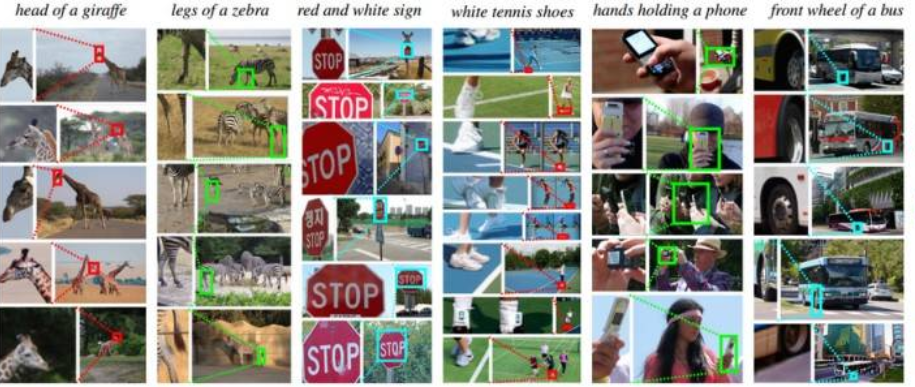
## 6.1语音处理

深度学习最先在语音处理领域取得突破性进展，无论是在标准小数据集上还是大数据集上。语音处理领域主要有两大任务: 语音识别和语音合成。深度学习广泛应用于语音识别中，Google推出端到端的语音识别系统、百度推出语音识别系统Deep Speech 2。2016年， 微软在日常对话数据上的语音识别准确率达到5.9%，首次达到人类水平。各大公司也都用深度学习来实现语音合成，包括 Google、Apple、科大讯飞等。Google DeepMind提出并行WaveNet模型来进行语音合成，百度推出产品级别的实时语音合成系统Deep Voice 3。



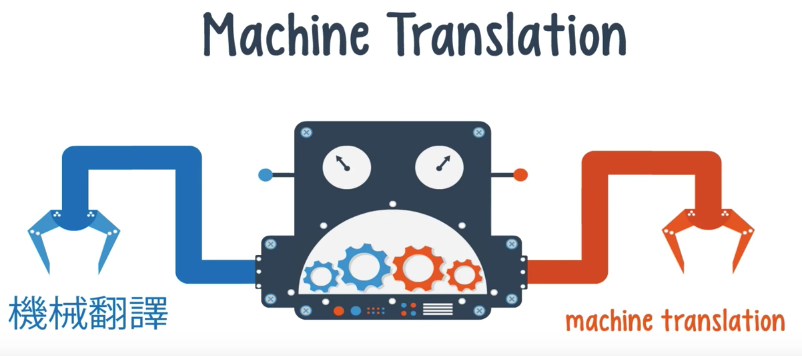
## 6.2 计算机视觉

深度学习被广泛应用于计算机视觉各种任务，包括交通标志检测和分类、人脸识别、人脸检测、图像分类、多尺度变换融合图像、物体检、图像语义分割、实时多人姿态估计、行人检测、场景识别、物体跟踪、端到端的视频分类、视频里的人体动作识别等另外，还有一些很有意思的应用，如 给黑白照片自动着彩色、将涂鸦变成艺术画、艺术 风格转移、去掉图片里的马赛克等。牛津大学和Google DeepMind还共同提出了LipNet来读唇语，准确率达到93%，远超人类52%的平均水平。



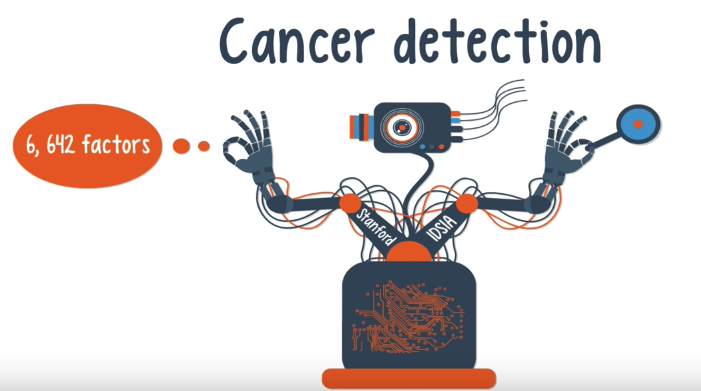
## 6.3自然语言处理

NEC Labs America最早将深度学习应用于自然语言处理领域。目前处理自然语言时通常先用 word2vec单词转化成词向量，其可以作为单词的特征。自然语言处理领域各种任务广泛用到了深度学习技术，包括词性标注、依存关系语法分析、命名体识、语义角色标注、只用字母的分布式表示来学习语言模型、 用字母级别的输入来预测单词级别的输、Twitter 情感分析、中文微博情感分析、文章分类、机器翻译、阅读理解、自动问答、对话系统等。



## 6.4其他

在生物信息学方面，深度学习能够被用来预测药物分子的活动，预测人眼停留部位，预测非编码 DNA 基因突变对基因表达与疾病的影响。金融行业积累了大量的数据，故深度学习在金融方面也有众多应用，包括金融市场预测、证券投资组合、保险流失预测等，也相应涌现出一批金融科技创业公司。另外，基于深度学习的实时发电调度算法能在满足实时发电任务前提下，使机组总污染物排放量降低，达到节能减排的目的。深度学习还可以诊断电动潜油柱塞泵的故障，避免故障事故的发生，有效延长检泵周期。深度学习应用强非线性、复杂的化工过程软测量建模中也能获得很好的精度。



# 7.瓶颈

虽然深度学习使得诸多领域取得突破性进展，但是深度学习仍然存在一些问题，攻克解决这些问题是学者们的进一步研究方向，本文针对这些问题也探讨了相应的可能解决思路。问题被分为训练问题、落地问题、功能问题和领域问题，训练问题指的是深度学习训练时间太长等问题，落地问题指的是限制了深度学习实际落地应用的问题，功能问题指的是深度学习目前还不能很好完成的任务，领域问题指的是针对计算机视觉和自然语言处理特定领域的问题。

## 7.1训练问题

训练时间太长，需要大量训练计算资源。在 WMT'14 英语到法语的数据集上，Google用96块NVIDIA K80 GPU需要训练6天来得到基本的机器翻译模型，另外还需要3天来微调进一步改进模型。这还只是训练一次模型，再加上需要调各种超参数，总的训练时间非常长，而且96块K80 GPU成本也非常高。深度学习模型需要的计算资源太多了，训练时间也太长了，需要全新的硬件、算法、系设计来加速模型的训练。如硬件方面不只采用具有通用 性的GPU，还可以采用高性能低功耗的可编程可配置FPGA 芯片、为了某种特定需求而专门定制的ASIC芯片。Google为TensorFlow深度学习框架专门设计了ASIC芯TPU 二代 Cloud TPU。自动驾驶领域也倾向于采用ASIC芯片，Nvidia发布针对L级全自动驾驶的AI 处理器 DrivePX Pegasus。

梯度消失问题，训练难度大。深层模型的训练难度非常大，网络层数太多的模型存在梯度消失问题激活函数例如Sigmoid函数的导数是个很小的数，多个很小的数连乘之后几乎为0，则梯度无法从输出层传到输入层。可以从训练方法、技巧、网络结构等方面来缓解梯度消失问题。训练方法上: Hinton 等提出先预训练后微调的训练方法，先无监督逐层预训练，再用反向传播算法对整个网络进行微调训练，预训练每次只训练一层隐藏层避免了梯度消失问题技巧上: 用 relu来代替 Sigmoid作为激活函数、使用dropout、批标准化( batch normalization)，这 些技巧都能缓解梯度消失问题。网络结构上: LSTM 采用门机制，加了输入门、输出门、遗忘门来控制允许多少信息通过; highway networks加入携带门( carry gate) 和转化门( transform gate) 使得输出由直接输入和转化后的输入两部分组成; resNet在在两层或多层之间直接加上线性连通通路，保证梯度能通过线性通路传到底层。

依赖大规模带标签训练数据。深度学习严重依赖大规模带标签数据来训练模型。人工数据标注耗时耗力，代价高昂。某些特定领域如疑难杂症等，几乎不可能收集到足够多的带标签数据。因此无监督学习是深度学习接下来的一个重要研究方向，目前已经有一些成果，包括 Goodfellow 等提出的生成对抗网络( generative adversarial nets，GANs) 、微软提出的新的学习范式对偶学习。

分布式训练问题。深度学习模型在单机上训练时间过长，尤其是当训练数据太多时，而且规模过大的模型也无法放入一台机器里所以需要进行大规模分布式训练深度学习模型．。分布式并行训练可以分为数据并行、模型并行和混合并行。混合并行里同时使用了数据并行和模型并行，例如可以在不同机器间使用数据并行，同一台机器上使用模型并行。数据并行是目前多数分布式系统的首选，各种数据并行方法的区别在于是参数平均法还是更新式方法、是同步更新还是异步更新、是中心化同步还是布式同步。参数平均法是传输各节点的参数到参数服器，然后所有参数求平均得到全局参数，更新式方法的不是参数，而是参数的更新量。微软提出延迟补偿的异步更新方法，在同步更新与异步更新中寻找合适的平衡，并开源了参数服务器框架 Multiverso。Strom去掉中心的参数服务器，平等地在各节点间传输参数更新量，高度压缩节点间的更新使得网络通信减少了3 个数量级。

## 7.2落地问题

对抗性样本攻击。Szegedy 等发现深度学习会被对抗性样本攻击，即本来正确分类的图片加上小的扰动能使深度学习模型误判成别的类别。知道神经网络结构的攻击被称为白盒攻击( white box attack) 。对抗性样本还以迁移，一个模型产生的对抗性样本也会被另一模型错误 分类，此为黑盒攻击( black box attack) 。对抗性样本攻击说明深度学习并没有那么可靠，Goodfellow 等认为这是 由于深度学习模型在高维空间中的线性性质导致的。为了防止对抗性样本攻击，可以进行对抗性训练，将对抗性样本跟普通样本都作为训练数据来训练模型，使用对抗目标函数，能使模型更加正则化。

鲁棒性差。即使深度学习的平均准确率很高，但某些测试用例上的预测效果可能很差．。高可靠性系统如无人车，远程外科手术，卫星发射等，不允许出现某个很谱的结果。需要提高深度学习的鲁棒性，保证坏的时候不至于太坏，使得深度学习的应用领域更广泛。

太多的超参数。针对实际问题，如何设计一个最适合的深度模型? 深度学习有太多的超参数，包括如何进行数据采集、生成、选择、划分。神经网络结构是用MLP、CNN还是ＲNN。神经网络层数、每层的神经元数量。权重初始化方法; 正则项系数( weight decay) ; 动量( momentum) 、学习率、learning rate decay、dropout; 迭代次数，batch。模型更新规则是用SGD还是 Adam; 分布式训练结果如何聚合等等可以用另一个神经网络来学习这些超参数，让神经网络设计变得自动化。Google DeepMind采Learning to learn 算法，用另一个网络来调整学习率，使得收敛更快。Google 发布了能自动搜索最优网络结构的Cloud AutoML。韩红桂等提出利用竞争机制来动态增加或删减隐藏层神经元，动态优化神经网络结构。张昭昭等提出多层自适应模块化神经网络结构设计方法。

可解释性差。深度学习模型的可解释性差，是典型的黑箱算法，模型复杂，通常包含上亿个参数。线上应用模型后，如果对某个用户造成严重影响，无法确定是哪个参数出了问题，从而无法针对性地调整某个参数来解决此用户的问题。而且模型的可解释性问题限制了其在医学、自动驾驶、军事、航天等重要领域的应用。可以尝试将深度学习与符号学习、可解释性算法进行结合，使其既有深度学习强大的表达能力，又有一定的可解释性微软用图学习机( Graph Engine) 来统一机器学习与知识图谱。Stanford 博士生 Wu 等将深度学习与可解释性决策树算法结合起来，用树正则化方法来提高深度学习的可解释。

模型太大。深度学习模型本身非常大，不方便放在GPU中，更不方便在移动端使用。特别对于语言模型来说，词表非常大，输出神经元很多，导致模型非常大。所以需要在保证准确率的前提下，进行模型压缩，将模型变小。模型压缩方法主要有参数修剪和共享、低秩分解、压缩卷积滤波器、知识精炼四类。参修剪和共享是去除对准确率没有提升的冗余参数，根据减 少信息冗余或参数空间冗余的方式，参数修剪和共享又可以细分为量化和二进制、剪枝和共享、设计结构化矩阵三类。低秩分解是用矩阵分解或张量分解来评估最具信息量 的参数; 压缩卷积滤波器是设计特殊结构的卷积滤波器来减少存储和计算的复杂度; 知识精炼是提取大型模型里的知识来训练更小更紧凑的模型。

## 7.3功能问题

不能像人类一样进行小样本学习。深度学习需要大规模的训练数据，样本利用率不高，但是人类的学习只需要极少几个样本。其实小孩在学习过程中，利用了大人传授的知识。目前还缺乏统一的框架向深度学习模型提供领域先验知识。要像人类一样进行小样本学习，可以尝试将深度学习与知识图谱、逻辑推理、符号学习等结合在一起，同时利用好数据与知识。

不能很好完成逻辑推理任务。深度学习目前只是在图像识别、语音识别等感知层面的任务有较好的表现，缺乏逻辑推理能力，无法很好地完成需要逻辑推理的任务。可以考虑将深度学习与擅长逻辑推理的符号学习、存储了知识的知识图谱结合; 另外还可以给深度学习模型增加记忆模块，如神经图灵机与记忆网络等。

无法同时处理多任务。人脑是多才多艺的，能同时识别语音，识别图像，理解文字等。而目前深度学习模型都是针对某一特定任务用特定数据集训练的，训练得到的模型也只能完成这一特定任务。为了能完成多任务，进步实现通用人工智能，可以尝试将不同功能的神经网络以 某种方式连接成更大的神经网络。Google通过稀疏门 矩阵将多个多层感知器子网络组合成超大网络，用反向传播同时训练所有子网络。

## 7.4领域问题

图像理解问题。深度学习目前在图像识别等感知任务有较好的表现，但在图像理解如视觉关系理解、图片内容问答、视觉注意点预测等方面成果还并不多。视觉关系理解首先需要检测出关键对象，然后预测对象之间的关系。图片内容问答是根据给定的图片，回答相应的问题。视觉注意点预测是对于给定的图片，预测人最感兴趣图片的哪一部分。这些都需要对图像内容有很好的理解图像。

自然语言处理问题。语言其实是比语音、图像更高级的非自然信号，是完全由人脑产生和处理的符号系统深度学习在自然语言处理上的效果还不像语音、图像那么显著，但是深度学习是受人脑启发得到的算法，相信深度学习接下来会在自然语言处理领域有更多的成果。

参考文献

[1]Are All People Married?Determining Obligatory Attributes in Knowledge Bases

[2]Neural Sentiment Classification with User and Product Attention

[3]Hierarchical Attention Networks for Document Classification

[4]Improving Review Representations with User Attention and Product Attention for Sentiment Classification

[5]Automatic Academic Paper Rating Based on Modularized Hierarchical Convolutional Neural Network

[6]Coevolutionary Recommendation Model: Mutual Learning between Ratings and Reviews

[7]A Joint Embedding Method for Entity Alignment of Knowledge Bases

[8]Bootstrapping Entity Alignment with Knowledge Graph Embedding

[9]Efficient Pruning of Large Knowledge Graphs

[10]Estimating Rule Quality for Knowledge Base Completion with the Relationship between Coverage Assumption

[11]Joint Extraction of Entities and Relations Based on a Novel Graph Scheme

[12] 孙志军, 薛磊, 许阳明, et al. 深度学习研究综述[J]. 计算机应用研究, 2012, 29(8):2806-2810.

[13] 尹宝才, 王文通, 王立春. 深度学习研究综述[J]. 北京工业大学学报, 2015(1):48-59.

[14] 刘钰鹏. 深度学习研究概述[J]. 信息与电脑(理论版), 2016(3):52-53.

[15] 张新钰, 高洪波, 赵建辉, et al. 基于深度学习的自动驾驶技术综述[J]. 清华大学学报(自然科学版), 2018(4).

[16] 孙瑜阳 . 深度学习及其在图像分类识别中的研究综述[J]. 信息技术与信息化, 2018.