原 Deep Learning（深度学习）学习笔记整理系列之（三）

分类： Deep Learning 机器学习 Linux驱动 2013-04-09 00:37 3757人阅读

Deep Learning（深度学习）学习笔记整理系列

zouxy09@qq.com

http://blog.csdn.net/zouxy09

作者：**Zouxy**

version 1.0 2013-04-08

声明：

1）该Deep Learning的学习系列是整理自网上很大牛和机器学习专家所无私奉献的资料的。具体引用的资料请看参考文献。具体的版本声明也参考原文献。

2）本文仅供学术交流，非商用。所以每一部分具体的参考资料并没有详细对应。如果某部分不小心侵犯了大家的利益，还望海涵，并联系博主删除。

3）本人才疏学浅，整理总结的时候难免出错，还望各位前辈不吝指正，谢谢。

4）阅读本文需要机器学习、计算机视觉、神经网络等等基础（如果没有也没关系了，没有就看看，能不能看懂，呵呵）。

5）此属于第一版本，若有错误，还需继续修正与增删。还望大家多多指点。大家都共享一点点，一起为祖国科研的推进添砖加瓦（呵呵，好高尚的目标啊）。请联系：zouxy09@qq.com

目录：

接上

好了，到了这一步，终于可以聊到Deep learning了。上面我们聊到为什么会有Deep learning（让机器自动学习良好的特征，而免去人工选取过程。还有参考人的分层视觉处理系统），我们得到一个结论就是Deep learning需要多层来获得更抽象的特征表达。那么多少层才合适呢？用什么架构来建模呢？怎么进行非监督训练呢？

五、**Deep Learning**的基本思想

假设我们有一个系统S，它有n层（S1,…Sn），它的输入是I，输出是O，形象地表示为： I =>S1=>S2=>…..=>Sn => O，如果输出O等于输入I，即输入I经过这个系

统变化之后没有任何的信息损失（呵呵，大牛说，这是不可能的。信息论中有个“信息逐层丢失”的说法（信息处理不等式），设处理a信息得到b，再对b处理得到c，那么可以证明：a和c的互信息不会超过a和b的互信息。这表明信息处理不会增加信息，大部分处理会丢失信息。当然了，如果丢掉的是没用的信息那多好啊），保持了不变，这意味着输入I经过每一层Si都没有任何的信息损失，即在任何一层Si，它都是原有信息（即输入I）的另外一种表示。现在回到我们的主题Deep Learning，我们需要自动地学习特征，假设我们有一堆输入I（如一堆图像或者文本），假设我们设计了一个系统S（有n层），我们通过调整系统中参数，使得它的输出仍然是输入I，那么我们就可以自动地获取得到输入I的一系列层次特征，即S1，…, Sn。

对于深度学习来说，其思想就是对堆叠多个层，也就是说这一层的输出作为下一层的输入。通过这种方式，就可以实现对输入信息进行分级表达了。

另外，前面是假设输出严格地等于输入，这个限制太严格，我们可以略微地放松这个限制，例如我们只要使得输入与输出的差别尽可能地小即可，这个放松会导致另外一类不同的Deep Learning方法。上述就是Deep Learning的基本思想。

六、浅层学习（**Shallow Learning**）和深度学习（**Deep Learning**）

浅层学习是机器学习的第一次浪潮。

20世纪80年代末期，用于人工神经网络的反向传播算法（也叫Back Propagation算法或者BP算法）的发明，给机器学习带来了希望，掀起了基于统计模型的机器学习热潮。这个热潮一直持续到今天。人们发现，利用BP算法可以让一个人工神经网络模型从大量训练样本中学习统计规律，从而对未知事件做预测。这种基于统计的机器学习方法比起过去基于人工规则的系统，在很多方面显出优越性。这个时候的人工神经网络，虽也被称作多层感知机（Multi-layer Perceptron），但实际是种只含有一层隐层节点的浅层模型。

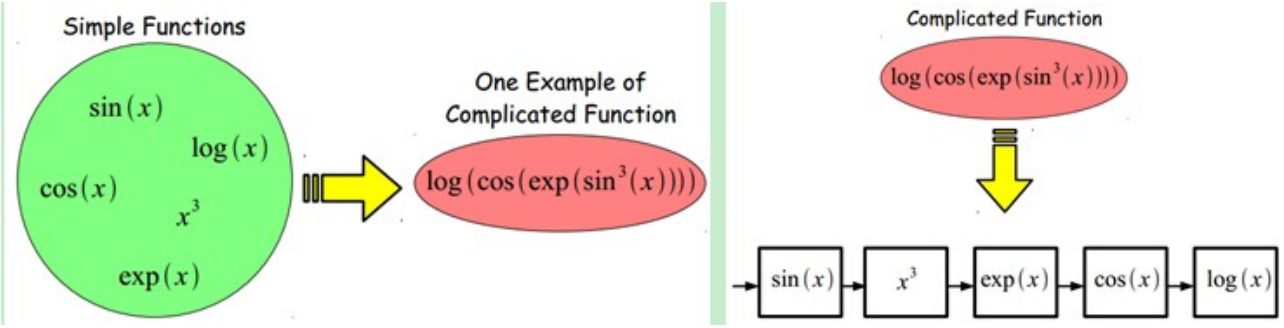
20世纪90年代，各种各样的浅层机器学习模型相继被提出，例如支撑向量机（SVM，Support Vector Machines）、 Boosting、最大熵方法（如LR，Logistic Regression）等。这些模型的结构基本上可以看成带有一层隐层节点（如SVM、Boosting），或没有隐层节点（如LR）。这些模型无论是在理论分析还是应用中都获得了巨大的成功。相比之下，由于理论分析的难度大，训练方法又需要很多经验和技巧，这个时期浅层人工神经网络反而相对沉寂。

深度学习是机器学习的第二次浪潮。

2006年，加拿大多伦多大学教授、机器学习领域的泰斗Geoffrey Hinton和他的

学生RuslanSalakhutdinov在《科学》上发表了一篇文章，开启了深度学习在学术界和工业界的浪潮。这篇文章有两个主要观点：1）多隐层的人工神经网络具有优异的特征学习能力，学习得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）深度神经网络在训练上的难度，可以通过“逐层初始化”（layer-wise pre-training）来有效克服，在这篇文章中，逐层初始化是通过无监督学习实现的。

当前多数分类、回归等学习方法为浅层结构算法，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。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。（多层的好处是可以用较少的参数表示复杂的函数）



深度学习的实质，是通过构建具有很多隐层的机器学习模型和海量的训练数据，来学习更有用的特征，从而最终提升分类或预测的准确性。因此，“深度模型”是手段，“特征学习”是目的。区别于传统的浅层学习，深度学习的不同在于：1）强调了模型结构的深度，通常有5层、6层，甚至10多层的隐层节点；2）明确突出了特征学习的重要性，也就是说，通过逐层特征变换，将样本在原空间的特征表示变换到一个新特征空间，从而使分类或预测更加容易。与人工规则构造特征的方法相比，利用大数据来学习特征，更能够刻画数据的丰富内在信息。

七、Deep learning与Neural Network

深度学习是机器学习研究中的一个新的领域，其动机在于建立、模拟人脑进行分析学习的神经网络，它模仿人脑的机制来解释数据，例如图像，声音和文本。深度学习是无监督学习的一种。

深度学习的概念源于人工神经网络的研究。含多隐层的多层感知器就是一种深度学习结构。深度学习通过组合低层特征形成更加抽象的高层表示属性类别或特征，以发现数据的分布式特征表示。

Deep learning本身算是machine learning的一个分支，简单可以理解为neural network的发展。大约二三十年前，neural network曾经是ML领域特别火热的一个方

向，但是后来确慢慢淡出了，原因包括以下几个方面：

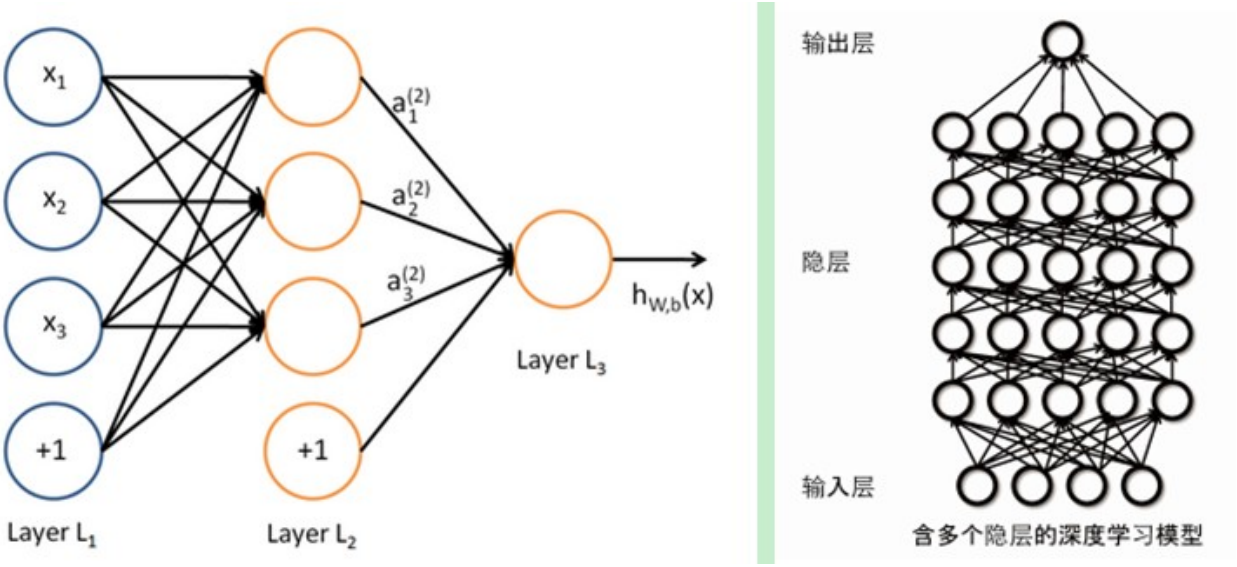
1）比较容易过拟合，参数比较难tune，而且需要不少trick；

2）训练速度比较慢，在层次比较少（小于等于3）的情况下效果并不比其它方法更优；

所以中间有大约20多年的时间，神经网络被关注很少，这段时间基本上是SVM和boosting算法的天下。但是，一个痴心的老先生Hinton，他坚持了下来，并最终（和其它人一起Bengio、Yann.lecun等）提成了一个实际可行的deep learning框架。

Deep learning与传统的神经网络之间有相同的地方也有很多不同。

二者的相同在于deep learning采用了神经网络相似的分层结构，系统由包括输入层、隐层（多层）、输出层组成的多层网络，只有相邻层节点之间有连接，同一层以及跨层节点之间相互无连接，每一层可以看作是一个logistic regression模型；这种分层结构，是比较接近人类大脑的结构的。



而为了克服神经网络训练中的问题，DL采用了与神经网络很不同的训练机制。传统神经网络中，采用的是back propagation的方式进行，简单来讲就是采用迭代的算法来训练整个网络，随机设定初值，计算当前网络的输出，然后根据当前输出和label之间的差去改变前面各层的参数，直到收敛（整体是一个梯度下降法）。而deep learning整体上是一个layer-wise的训练机制。这样做的原因是因为，如果采用back propagation的机制，对于一个deep network（7层以上），残差传播到最前面的层已经变得太小，出现所谓的gradient diffusion（梯度扩散）。这个问题我们接下来讨论。

八、**Deep learning**训练过程

8.1、传统神经网络的训练方法为什么不能用在深度神经网络

BP算法作为传统训练多层网络的典型算法，实际上对仅含几层网络，该训练方法就已经很不理想。深度结构（涉及多个非线性处理单元层）非凸目标代价函数中普遍存在的局部最小是训练困难的主要来源。

BP算法存在的问题：

（1）梯度越来越稀疏：从顶层越往下，误差校正信号越来越小；

（2）收敛到局部最小值：尤其是从远离最优区域开始的时候（随机值初始化会导致这种情况的发生）；

（3）一般，我们只能用有标签的数据来训练：但大部分的数据是没标签的，而大脑可以从没有标签的的数据中学习；

8.2、**deep learning**训练过程

如果对所有层同时训练，时间复杂度会太高；如果每次训练一层，偏差就会逐层传递。这会面临跟上面监督学习中相反的问题，会严重欠拟合（因为深度网络的神经元和参数太多了）。

2006年，hinton提出了在非监督数据上建立多层神经网络的一个有效方法，简单的说，分为两步，一是每次训练一层网络，二是调优，使原始表示x向上生成的高级表示r和该高级表示r向下生成的x'尽可能一致。方法是：

1）首先逐层构建单层神经元，这样每次都是训练一个单层网络。

2）当所有层训练完后，Hinton使用wake-sleep算法进行调优。

将除最顶层的其它层间的权重变为双向的，这样最顶层仍然是一个单层神经网络，而其它层则变为了图模型。向上的权重用于“认知”，向下的权重用于“生成”。然后使用Wake-Sleep算法调整所有的权重。让认知和生成达成一致，也就是保证生成的最顶层表示能够尽可能正确的复原底层的结点。比如顶层的一个结点表示人脸，那么所有人脸的图像应该激活这个结点，并且这个结果向下生成的图像应该能够表现为一个大概的人脸图像。Wake-Sleep算法分为醒（wake）和睡（sleep）两个部分。

1）**wake**阶段：认知过程，通过外界的特征和向上的权重（认知权重）产生每一层的抽象表示（结点状态），并且使用梯度下降修改层间的下行权重（生成权重）。也就是“如果现实跟我想象的不一样，改变我的权重使得我想象的东西就是这样的”。

2）**sleep**阶段：生成过程，通过顶层表示（醒时学得的概念）和向下权重，生成底层

的状态，同时修改层间向上的权重。也就是“如果梦中的景象不是我脑中的相应概念，改变我的认知权重使得这种景象在我看来就是这个概念”。

deep learning训练过程具体如下：

1）使用自下上升非监督学习（就是从底层开始，一层一层的往顶层训练）：

采用无标定数据（有标定数据也可）分层训练各层参数，这一步可以看作是一个无监督训练过程，是和传统神经网络区别最大的部分（这个过程可以看作是feature learning过程）：

具体的，先用无标定数据训练第一层，训练时先学习第一层的参数（这一层可以看作是得到一个使得输出和输入差别最小的三层神经网络的隐层），由于模型capacity的限制以及稀疏性约束，使得得到的模型能够学习到数据本身的结构，从而得到比输入更具有表示能力的特征；在学习得到第n-1层后，将n-1层的输出作为第n层的输入，训练第n层，由此分别得到各层的参数；

2）自顶向下的监督学习（就是通过带标签的数据去训练，误差自顶向下传输，对网络进行微调）：

基于第一步得到的各层参数进一步fine-tune整个多层模型的参数，这一步是一个有监督训练过程；第一步类似神经网络的随机初始化初值过程，由于DL的第一步不是随机初始化，而是通过学习输入数据的结构得到的，因而这个初值更接近全局最优，从而能够取得更好的效果；所以deep learning效果好很大程度上归功于第一步的feature learning过程。