

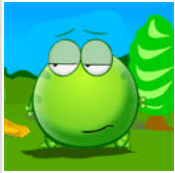
网络资源是无限的

目录视图

摘要视图

RSS 订阅

个人资料



fengbingchun



访问：2252591次
积分：25003
等级：

BLOG > ?

排名：第202名

原创：341篇 转载：144篇
译文：0篇 评论：1434条

文章分类

- Android (9)
- ActiveX (18)
- Bar Code (16)
- Caffe (20)
- C# (5)
- Cimg (4)
- Contour Detection (9)
- CxImage (6)
- Code::Blocks (3)
- Cloud Computing (1)
- C/C++ (82)
- CUDA (10)
- CMake (3)
- Design Patterns (25)
- Database/Dataset (4)
- Deep Learning (9)
- Eclipse (3)
- Emgu CV (1)
- Eigen (1)
- FFmpeg (1)
- Feature Extraction (1)
- FreeType (1)
- Face (8)
- GPU (3)
- Git (3)
- GCC (1)
- GDAL (5)

CSDN学院招募微信小程序讲师啦

程序员简历优化指南！

【观点】移动原生App开发 PK HTML 5开发

云端应用征文大赛，秀绝招，赢无人机！

卷积神经网络(CNN)基础介绍

2016-01-16 20:31

11397人阅读

评论(10)

收藏

举报

分类：

Caffe (19)

Neural Network (12)

Deep Learning (8)

版权声明：本文为博主原创文章，未经博主允许不得转载。

本文是对卷积神经网络的基础进行介绍，主要内容包括卷积神经网络概念、卷积神经网络结构、卷积神经网络求解、卷积神经网络LeNet-5结构分析、卷积神经网络注意事项。

一、卷积神经网络概念

上世纪60年代，Hubel等人通过对猫视觉皮层细胞的研究，提出了感受野这个概念，到80年代，Fukushima在感受野概念的基础之上提出了神经认知机的概念，可以看作是卷积神经网络的第一个实现网络，神经认知机将一个视觉模式分解成许多子模式（特征），然后进入分层递阶式相连的特征平面进行处理，它试图将视觉系统模型化，使其能够在即使物体有位移或轻微变形的时候，也能完成识别。

卷积神经网络(Convolutional Neural Networks, CNN)是多层感知机(MLP)的变种。由生物学家休博尔和维瑟尔在早期关于猫视觉皮层的研究发展而来。视觉皮层的细胞存在一个复杂的构造。这些细胞对视觉输入空间的子区域非常敏感，我们称之为感受野，以这种方式平铺覆盖到整个视野区域。这些细胞可以分为两种基本类型，简单细胞和复杂细胞。简单细胞最大程度响应来自感受野范围内的边缘刺激模式。复杂细胞有更大的接受域，它对来自确切位置的刺激具有局部不变性。

通常神经认知机包含两类神经元，即承担特征提取的采样元和抗变形的卷积元，采样元中涉及两个重要参数，即感受野与阈值参数，前者确定输入连接的数目，后者则控制对特征子模式的反应程度。卷积神经网络可以看作是神经认知机的推广形式，神经认知机是卷积神经网络的一种特例。

CNN由纽约大学的Yann LeCun于1998年提出。CNN本质上是一个多层感知机，其成功的原因关键在于它所采用的局部连接和共享权值的方式，一方面减少了的权值的数量使得网络易于优化，另一方面降低了过拟合的风险。CNN是神经网络中的一种，它的权值共享网络结构使之更类似于生物神经网络，降低了网络模型的复杂度，减少了权值的数量。该优点在网络的输入是二维图像时表现的更为明显，使图像可以直接作为网络的输入，避免了传统识别算法中复杂的特征提取和数据重建过程。在二维图像处理上有众多优势，如网络能自行抽取图像特征包括颜色、纹理、形状及图像的拓扑结构；在处理二维图像问题上，特别是识别位移、缩放及其它形式扭曲不变性的应用上具有良好的鲁棒性和运算效率等。

CNN本身可以采用不同的神经元和学习规则的组合形式。

CNN具有一些传统技术所没有的优点：良好的容错能力、并行处理能力和自学习能力，可处理环境信息复杂，背景知识不清楚，推理规则不明确情况下的问题，允许样品有较大的缺损、畸变，运行速度快，自适应性能好，具有较高的分辨率。它是通过结构重组和减少权值将特征抽取功能融合进多层感知器，省略识别前复杂的图像特征抽取过程。

CNN的泛化能力要显著优于其它方法，卷积神经网络已被应用于模式分类，物体检测和物体识别等方面。利用卷积神经网络建立模式分类器，将卷积神经网络作为通用的模式分类器，直接用于灰度图像。

CNN是一个前馈式神经网络，能从一个二维图像中提取其拓扑结构，采用反向传播算法来优化网络结构，求

- HTML (3)
- Image Recognition (8)
- Image Processing (18)
- Image Registration (13)
- ImageMagick (3)
- Java (5)
- Linux (20)
- Log (2)
- Makefile (2)
- Mathematical Knowledge (6)
- Multi-thread (4)
- Matlab (33)
- MFC (8)
- MinGW (3)
- Mac (1)
- Neural Network (13)
- OCR (9)
- Office (2)
- OpenCL (2)
- OpenSSL (7)
- OpenCV (86)
- OpenGL (2)
- OpenGL ES (3)
- OpenMP (3)
- Photoshop (1)
- Python (4)
- Qt (1)
- SIMD (14)
- Software Development (4)
- System architecture (2)
- Skia (1)
- SVN (1)
- Software Testing (4)
- Shell (2)
- Socket (3)
- Target Detection (2)
- Target Tracking (2)
- VC6 (6)
- VS2008 (16)
- VS2010 (4)
- VS2013 (3)
- vigra (2)
- VLC (5)
- VLFeat (1)
- wxWidgets (1)
- Watermark (4)
- Windows7 (6)
- Windows Core Programming (9)
- XML (2)

Free Codes

- pu4n
- freecode
- Peter's Functions
- CodeProject
- SourceCodeOnline
- Computer Vision Source Code
- Codesoso
- Digital Watermarking
- SourceForge
- HackChina
- oschina

解网络中的未知参数。

CNN是一类特别设计用来处理二维数据的多层神经网络。CNN被认为是第一个真正成功的采用多层次结构网络的具有鲁棒性的深度学习方法。CNN通过挖掘数据中的空间上的相关性，来减少网络中的可训练参数的数量，达到改进前向传播网络的反向传播算法效率，因为CNN需要非常少的数据预处理工作，所以也被认为是一种深度学习的方法。在CNN中，图像中的小块区域（也叫做“局部感知区域”）被当做层次结构中的底层的输入数据，信息通过前向传播经过网络中的各个层，在每一层中都由过滤器构成，以便能够获得观测数据的一些显著特征。因为局部感知区域能够获得一些基础的特征，比如图像中的边界和角落等，这种方法能够提供一定程度对位移、拉伸和旋转的相对不变性。

CNN中层次之间的紧密联系和空间信息使得其特别适用于图像的处理和理解，并且能够自动的从图像抽取丰富的相关性特性。

CNN通过结合局部感知区域、共享权重、空间或者时间上的降采样来充分利用数据本身包含的局部性等特征，优化网络结构，并且保证一定程度上的位移和变形的不变性。

CNN受视觉神经机制的启发而设计,是为识别二维或三维信号而设计的一个多层感知器,这种网络结构对平移、缩放、倾斜等变形具有高度不变性。

CNN可以用来识别位移、缩放及其它形式扭曲不变性的二维或三维图像。CNN的特征提取层参数是通过训练数据学习得到的，所以其避免了人工特征提取，而是从训练数据中进行学习；其次同一特征图的神经元共享权值，减少了网络参数，这也是卷积网络相对于全连接网络的一大优势。共享局部权值这一特殊结构更接近于真实的生物神经网络使CNN在图像处理、语音识别领域有着独特的优越性，另一方面权值共享同时降低了网络的复杂性，且多维输入信号（语音、图像）可以直接输入网络的特点避免了特征提取和分类过程中数据重排的过程。

CNN是一种特殊的深层的神经网络模型，它的特殊性体现在两个方面，一方面它的神经元的连接是非全连接的，另一方面同一层中某些神经元之间的连接的权重是共享的（即相同的）。它的非全连接和权值共享的网络结构使之更类似于生物神经网络，降低了网络模型的复杂度（对于很难学习的深层结构来说，这是非常重要的），减少了权值的数量。

CNN是一种深度的监督学习下的机器学习模型，具有极强的适应性，善于挖掘数据局部特征，提取全局训练特征和分类，它的权值共享结构网络使之更类似于生物神经网络，在模式识别各个领域都取得了很好的成果。

稀疏连接：在BP神经网络中，每一层的神经元节点是一个线性一维排列结构，层与层各神经元节点之间是全连接的。卷积神经网络中，层与层之间的神经元节点不再是全连接形式，利用层间局部空间相关性将相邻每一层的神经元节点只与和它相近的上层神经元节点连接，即局部连接。这样大大降低了神经网络架构的参数规模。

权重共享：在卷积神经网络中，卷积层的每一个卷积滤波器重复的作用于整个感受野中，对输入图像进行卷积，卷积结果构成了输入图像的特征图，提取出图像的局部特征。每一个卷积滤波器共享相同的参数，包括相同的权重矩阵和偏置项。共享权重的好处是在对图像进行特征提取时不用考虑局部特征的位置。而且权重共享提供了一种有效的方式，使要学习的卷积神经网络模型参数数量大大降低。

最大池采样：它是一种非线性降采样方法。在通过卷积获取图像特征之后是利用这些特征进行分类。可以用所有提取到的特征数据进行分类器的训练，但这通常会产生极大的计算量。所以在获取图像的卷积特征后，要通过最大池采样方法对卷积特征进行降维。将卷积特征划分为数个n*n的不相交区域，用这些区域的最大(或平均)特征来表示降维后的卷积特征。这些降维后的特征更容易进行分类。

最大池采样在计算机视觉中的价值体现在两个方面：(1)、它减小了来自上层隐藏层的计算复杂度；(2)、这些池化单元具有平移不变性，即使图像有小的位移，提取到的特征依然会保持不变。由于增强了对位移的鲁棒性，最大池采样方法是一个高效的降低数据维度的采样方法。

Softmax回归：它是在逻辑回归的基础上扩张而来，它的目的是为了解决多分类问题。在这类问题中，训练样本的种类一般在两个以上。Softmax回归是有监督学习算法，它也可以与深度学习或无监督学习方法结合使用。

CNN是一种带有卷积结构的深度神经网络，通常至少有两个非线性可训练的卷积层，两个非线性的固定卷积层（又叫Pooling Layer）和一个全连接层，一共至少5个隐含层。

卷积神经网络中，输入就是一幅幅的图像，权值W就是卷积模板，一般是卷积层和下采样层交替，最后是全连接的神经网络。

局部区域感知能够发现数据的一些局部特征，比如图片上的一个角，一段弧，这些基本特征是构成动物视觉的

关闭

[libsvm](#)
[joys99](#)
[CodeForge](#)
[cvchina](#)
[tesseract-ocr](#)
[sift](#)
[TIRG](#)
[imgSeek](#)
[OpenSURF](#)

Friendly Link

[OpenCL](#)
[Python](#)
[poesia-filter](#)
[TortoiseSVN](#)
[imgSeek](#)
[Notepad](#)
[Beyond Compare](#)
[CMake](#)
[VIGRA](#)
[CodeGuru](#)
[vchome](#)
[aforgenet](#)
[CVLAB](#)
[Doxygen](#)
[Coursera](#)
[OpenMP](#)

Technical Forum

[Matlab China](#)
[OpenCV China](#)
[The CImg Library](#)
[Open Computer Vision Library](#)
[CxImage](#)
[ImageMagick](#)
[ImageMagick China](#)
[OpenCV_China](#)
[Subversion China](#)

基础。

CNN中每一层的由多个map组成，每个map由多个神经单元组成，同一个map的所有神经单元共用一个卷积核（即权重），卷积核往往代表一个特征，比如某个卷积核代表一段弧，那么把这个卷积核在整个图片上滚一下，卷积值较大的区域就很有可能是一段弧。注意卷积核其实就是权重，我们并不需要单独去计算一个卷积，而是一个固定大小的权重矩阵去图像上匹配时，这个操作与卷积类似，因此我们称为卷积神经网络，实际上，BP也可以看作一种特殊的卷积神经网络，只是这个卷积核就是某层的所有权重，即感知区域是整个图像。权重共享策略减少了需要训练的参数，使得训练出来的模型的泛华能力更强。

CNN一般采用卷积层与采样层交替设置，即一层卷积层接一层采样层，采样层后接一层卷积...这样卷积层提取出特征，再进行组合形成更抽象的特征，最后形成对图片对象的描述特征，CNN后面还可以跟全连接层，全连接层跟BP一样。

CNN的最大特点就是稀疏连接（局部感受）和权值共享。稀疏连接和权值共享可以减少所要训练的参数，减少计算复杂度。

卷积神经网络是一个多层的神经网络，每层由多个二维平面组成，而每个平面由多个独立神经元组成。

卷积网络在本质上是一种输入到输出的映射，它能够学习大量的输入与输出之间的映射关系，而输入和输出之间的精确的数学表达式，只要用已知的模式对卷积网络加以训练，网络就具有输入输出对之间的映射能力。卷积网络执行的是有导师训练，所以其样本集是由形如：（输入向量，理想输出向量）的向量对构成的。所有这些向量对，都应该是来源于网络即将模拟的系统的实际“运行”结果。它们可以是实际运行系统中采集来的。在开始训练前，所有的权都应该用一些不同的小随机数进行初始化。“小随机数”用来保证网络不会因权值过大而进入饱和状态，从而导致训练失败；“不同”用来保证网络可以正常地学习。实际上，如果用相同的数去初始化权矩阵，则网络无能力学习。训练算法与传统的BP算法差不多。

二、卷积神经网络结构

卷积神经网络整体架构：卷积神经网络是一种多层的监督学习神经网络，隐含层的卷积层和池采样层是实现卷积神经网络特征提取功能的核心模块。该网络模型通过采用梯度下降法最小化损失函数对网络中的权重参数逐层反向调节，通过频繁的迭代训练提高网络的精度。卷积神经网络的低隐层是由卷积层和最大池采样层交替组成，高层是全连接层对应传统多层感知器的隐含层和逻辑回归分类器。第一个全连接层的输入是由卷积层和子采样层进行特征提取得到的特征图像。最后一层输出层是一个分类器，可以采用逻辑回归，Softmax回归甚至是支持向量机对输入图像进行分类。

卷积神经网络结构包括：卷积层，降采样层，全链接层。每一层有多个特征图，每个特征图通过一种卷积滤波器提取输入的一种特征，每个特征图有多个神经元。

卷积层：使用卷积层的原因是卷积运算的一个重要特点是，通过卷积运算，可以使原信号特征增强，并且降低噪音。

降采样层：使用降采样的原因是，根据图像局部相关性的原理，对图像进行子采样可以减少计算量，同时保持图像旋转不变性。

采样的目的主要是混淆特征的具体位置，因为某个特征找出来后，它的具体位置已经不重要了，我们只需要这个特征与其他的相对位置，比如一个“8”，当我们得到了上面一个“o”时，我们不需要知道它在图像的具体位置，只需要知道它下面又是一个“o”我们就可以知道是一个“8”了，因为图片中“8”在图片中偏左或者偏右都不影响我们认识它，这种混淆具体位置的策略能对变形和扭曲的图片进行识别。

全连接层：采用softmax全连接，得到的激活值即卷积神经网络提取到的图片特征。

卷积层的map个数是在网络初始化指定的，而卷积层的map的大小是 $n \times n$ 的，假设上一层的map大小是 $n \times n$ 、卷积核的大小是 $k \times k$ ，则该层的map大小是 $(n-k+1) \times (n-k+1)$ 。

采样层是对上一层map的一个采样处理，这里的采样方式是对上一层map的相邻小区域进行聚合统计，区域大小为 $scale \times scale$ ，有些实现是取小区域的最大值，而ToolBox里面的实现是采用 2×2 小区域的均值。注意，卷积的计算窗口是有重叠的，而采样的计算窗口没有重叠，ToolBox里面计算采样也是用卷积(conv2(A,K,'valid'))来实现的，卷积核是 2×2 ，每个元素都是 $1/4$ ，去掉计算得到的卷积结果中有重叠的部分。

CNN的基本结构包括两种特殊的神经元层，其一为卷积层，每个神经元的输入与前一层的局部相连，并提取该局部的特征；其二是池化层，用来求局部敏感性与二次特征提取的计算层。这种两次特征提取结构减少了特征分辨

关闭

Technical Blog

邹宇华

深之JohnChen

HUNNISH

周伟明

superdort

carson2005

OpenHero

Netman(Linux)

wqvbjhc

yang_xian521

gnuipc

gnuipc

千里8848

CVART

tornadomeet

gotosuc

onezeros

hellogv

abcjennifer

czy_sparrow

评论排行

Windows7 32位机上, O (120)

tiny-cnn开源库的使用(MI (93)

Ubuntu 14.04 64位机上2 (89)

tesseract-ocr3.02字符识 (63)

Windows7上使用VS201 (47)

tesseract-ocr (42)

图像配准算法 (41)

Windows 7 64位机上Op (36)

OpenCV中resize函数五 (34)

小波矩特征提取matlab代 (30)

最新评论

Tesseract-OCR 3.04在Windows
fengbingchun: @iliked: 没有密码, 那个commit只是提示是从哪个commit fork过来的, 无需管那个

Tesseract-OCR 3.04在Windows
iliked: 问一下, 你第一句中的commit的那个密码, 怎么用啊

卷积神经网络(CNN)的简单实现(
fengbingchun: @hugl950123: 是需要opencv的支持, 你在本地opencv的环境配好了吗, 配好了就应该没...

卷积神经网络(CNN)的简单实现(
hugl950123: @fengbingchun: 博主请问一下, test_CNN_predict()函数是不是需要open...

卷积神经网络(CNN)的简单实现(
hugl950123: @fengbingchun: 博主请问一下, test_CNN_predict()函数是不是需要open...

卷积神经网络(CNN)的简单实现(
hugl950123: @fengbingchun: 谢谢, 能够成功运行了现在

卷积神经网络(CNN)的简单实现(
fengbingchun:

@hugl950123: NN中一共有四个工程, 它们之间没有任何关系, 都是独立的, 如果要运行这篇文章的...

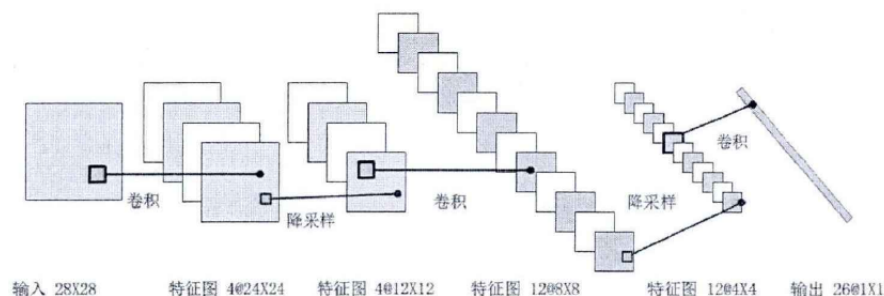
率, 减少了需要优化的参数数目。

CNN是部分连接网络, 其最底层是特征提取层(卷积层), 接着是池化层(Pooling), 然后可以继续增加卷积、池化或全连接层。用于模式分类的CNN, 通常在最后层使用softmax。

一般情况下, CNN的结构形式是: 输入层--> Conv层 --> Pooling层 --> (重复Conv、Pooling层) ... --> FC(Full-connected)层 --> 输出结果。通常输入层大小一般为2的整数倍, 如32, 64, 96, 224, 384等。通常卷积层使用较小的filter, 如3*3, 最大也就5*5。Pooling层用于对卷积结果进行降低维度, 例如选择2*2的区域对卷积层进行降低维度, 则选择2*2区域的最大值作为输出, 这样卷积层的维度就降为之前一半。

一般地, CNN的基本结构包括两层, 其一为特征提取层, 每个神经元的输入与前一层的局部接受域相连, 并提取该局部的特征。一旦该局部特征被提取后, 它与其它特征间的位置关系也随之确定下来; 其二是特征映射层, 网络的每个计算层由多个特征映射组成, 每个特征映射是一个平面, 平面上所有神经元的权值相等。特征映射结构采用影响函数核小的sigmoid函数作为卷积网络的激活函数, 使得特征映射具有位移不变性。此外, 由于一个映射面上的神经元共享权值, 因而减少了网络自由参数的个数。卷积神经网络中的每一个卷积层都紧跟着一个用来求局部平均与二次提取的计算层, 这种特有的两次特征提取结构减小了特征分辨率。

对于图像识别任务, 卷积神经网络的结构一般如下图所示。输入层读入经过简单的规则化(统一大小、亮度、颜色)的图像。每一层中的单元将前一层中的一组小的局部邻近的单元作为输入。这种局部连接观点来源于早期Hubel和Wiesel从猫科动物的视觉系统中发现的局部感知、方向选择神经元相一致。通过局部感知场, 神经元能够抽取一些基本的视觉特征, 比如有向边、结束点、边角等等。这些特征然后被更高层的神经元所使用。并且, 适用于某个局部的基础特征抽取器同样也倾向于适用于整个图像。通过利用这种特征, 卷积神经网络利用一组分布于图像各个不同位置但具有相同权值向量的单元, 来获取图像的特征并构成一幅特征图(Feature Map)。在每个位置, 来自不同特征图的单元得到各自不同类型的特征。一个特征图中的不同单元被限制为对输入图中的各个不同位置的局部数据进行同样的操作。这种操作等同于将输入图像对于一个小的核进行卷积。一个卷积层中通常包含多个具有不同权值向量的特征图, 使得在同一个位置能够获得多种不同的特征。如下图, 第一个隐含层包含4个特征图, 每个特征图由5*5的局部感知区域构成。一旦一个特征被检测到, 只要其相对于其他特征的相对位置没有改变, 那么其在图像中的绝对位置就变得不是特别重要。因此, 每个卷积层后面跟随着一个降采样层。降采样层进行局部平均和降采样的操作, 降低特征图的分辨率, 同时降低了网络输出对于位移和变形的敏感程度。第二个隐含层进行2*2的平均化降采样的操作。后续的卷积层和降采样层都交替分布连接, 构成一个“双金字塔”结构: 特征图的数目逐渐增多, 而且特征图的分辨率逐渐降低。



卷积神经网络结构示意图

由于所有权重都是通过反向传播算法训练得到, 卷积神经网络可以被看作自动合成其自身的特征抽取器。

一般情况下卷积神经网络中, 卷积层和降采样层交替连接在一起, 用于降低计算时间并逐步建立起更高的空间和数据结构不变性, 并通过比较小的降采样系数使得这些特性得以维持。

关闭

CNN的分类模型与传统模型的不同点在于其可以直接将一幅二维图像输入模型中, 接着在输出端即给出分类结果。其优势在于不需复杂的预处理, 将特征抽取, 模式分类完全放入一个黑匣子中, 通过不断的优化来获得网络所需参数, 在输出层给出所需分类, 网络核心就是网络的结构设计与网络的求解。这种求解结构比以往多种算法性能更高。

CNN是一个多层的神经网络, 每层由多个二维平面组成, 而每个平面由多个独立神经元组成。网络中包含简单单元(S-元)和复杂单元(C-元), S-元聚合在一起组成S-面, S-面聚合在一起组成S-层。C-元、C-面和C-层之间存在类似的关系。网络的中间部分由S-层与C-层串接而成, 输入级只含一层, 它直接接受二维视觉模式。样本特征提取步骤已嵌入到卷积神经网络模型的互联结构中。

卷积神经网络(CNN)的简单实现(hugl950123: @fengbingchun:下的是新的,我在CNN.cpp文件中每个函数都设置了断点,还是没有变化=...

卷积神经网络(CNN)的简单实现(fengbingchun: @hugl950123:你用的是GitHub上最新的吗?既然能编译过,在Debug下断点,应该很快...

卷积神经网络(CNN)的简单实现(hugl950123: 博主,请问我按照您的代码成功编译后执行结果窗口一闪而过,并且里面什么内容也没有,应该如何解决,能不能...

阅读排行

C#中 OpenFileDialog 的 (47141)
tesseract-ocr3.02字识别 (34575)
举例说明使用MATLAB C (25987)
OpenCV中resize函数 (24317)
利用cvMinAreaRect2求 (24277)
Windows 7 64位机上搭建 (22586)
opencv 检测直线、线段、 (20776)
OpenCV运动检测跟踪(b (20475)
图像配准算法 (19237)
有效的rtsp流媒体测试地: (19143)

文章存档

2017年01月 (18)
2016年12月 (11)
2016年11月 (8)
2016年10月 (7)
2016年09月 (16)

展开



程序员有前途吗



一般, S为特征提取层, 每个神经元的输入与前一层的局部感受野相连接, 并提取该局部的特征, 一旦该局部特征被提取, 它与其它特征之间的位置关系就被确定; C是特征映射层, 网络的每个计算层由多个特征映射组成, 每个特征映射为一个平面, 平面上所有神经元的权值相同。特征映射结构采用影响函数核小的Sigmoid函数作为卷积网络的激活函数, 使得特征映射具有位移不变性。由于每个映射面上的神经元权值共享, 减少了网络的自由参数数目, 降低了网络参数选择的复杂度。CNN中的每一个特征提取层(S-层) 都跟着一个用来求局部平均与二次提取的计算层(C-层), 这种特有的两次特征提取结构使网络在识别时对输入样本有较高的畸变容忍能力。

CNN网络除了输入输出层, 还有中间的卷积层, 抽样层与全连接层, 将原始图像直接输入到输入层, 原始图像的大小决定了输入向量的尺寸, 神经元提取图像的局部特征, 每个神经元都与前一层的局部感受野相连, 通过交替出现的抽样层(S) 与卷积层(C) 和最后的全连接层, 在输出层给出网络的输出。在卷积层和抽样层中有若干个特征图, 每一层有多个平面, 每层中各平面的神经元提取图像中特定区域的局部特征, 如边缘特征, 方向特征等, 在训练时不断修正S-层神经元的权值。同一层面上的神经元权值相同, 这样可以有相同程度的位移、旋转不变性。由于权值共享, 所以从一个平面到下个平面的映射可以看做是卷积运算, S-层可看作是模糊滤波器, 起到二次特征提取的作用。隐层与隐层之间空间分辨率递减, 每层所含的平面数递增, 这样可用于检测更多的特征信息。

卷积层中, 前一层的特征图与一个可学习的核进行卷积, 卷积的结果经过激活函数后的输出形成这一层的神经元, 从而构成该层特征图。卷积层与抽样层间隔出现, 卷积层每一个输出的特征图可能与前一层的几个特征图建立关系。每个特征图可以有不同卷积核。卷积层主要的任务就是从不同的角度来选择前一层的特征图, 使其具有位移不变性。卷积的本质就是对前一层的特征图进行处理, 来得到这一层的特征图。抽样层主要作用是降低网络的空间分辨率, 通过降低图像的空间分辨率来消除偏移和图像的扭曲。

隐层的参数个数和隐层的神经元个数无关, 只和滤波器的大小和滤波器种类的多少有关。隐层的神经元个数, 它和原图像, 也就是输入的大小(神经元个数)、滤波器的大小和滤波器在图像中的滑动步长都有关。

三、卷积神经网络求解

CNN通过三个方法来实现识别图像的位移、缩放和扭曲不变性, 即局域感受野、权值共享和次抽样。局域感受野指的是每一层网络的神经元只与上一层的一个小邻域内的神经元连接, 通过局域感受野, 每个神经元可以提取初级的视觉特征, 如方向线段, 端点和角点等; 权值共享使得CNN具有更少的参数, 需要相对少的训练数据; 次抽样可以减少特征的分辨率, 实现对位移、缩放和其它形式扭曲的不变性。卷积层之后通常用一个次抽样层来减少计算时间、建立空间和结构上的不变性。

构造好网络之后, 需要对网络进行求解, 如果像传统神经网络一样分配参数, 则每一个连接都会有未知参数。而CNN采用的是权值共享, 这样一来通过一幅特征图上的神经元共享同样的权值就可以大大减少自由参数, 这可以用来检测相同的特征在不同角度表示的效果。在网络设计中通常都是抽样层与卷积层交替出现, 全连接层的前一层通常为卷积层。

在CNN中, 权值更新是基于反向传播算法。

CNN在本质上是一种输入到输出的映射, 它能够学习大量的输入与输出之间的映射关系, 而不需要任何输入和输出之间的精确的数学表达式, 只要用已知的模式对卷积网络加以训练, 网络就具有输入输出对之间的映射能力。卷积网络执行的是监督训练, 所以其样本集是由形如: 输入向量, 理想输出向量的向量对构成的。所有这些向量对, 都应该是来源于网络即将模拟系统的实际“运行”结构, 它们可以是实际运行系统中采集来。在开始训练前, 所有的权都应该用一些不同的随机数进行初始化。“小随机数”用来保证网络不会因权值过大而进入饱和状态, 从而导致训练失败; “不同”用来保证网络可以正常地学习。实际上, 如果用相同的数去初始化权矩阵, 则网络无学习能力。

训练算法主要包括四步, 这四步被分为两个阶段:

第一阶段, 向前传播阶段:

- (1)、从样本集中取一个样本, 输入网络;
- (2)、计算相应的实际输出; 在此阶段, 信息从输入层经过逐级的变换, 传送到输出层。这个过程也是网络在完成训练后正常执行时执行的过程。

第二阶段, 向后传播阶段:

- (1)、计算实际输出与相应的理想输出的差;
- (2)、按极小化误差的方法调整权矩阵。

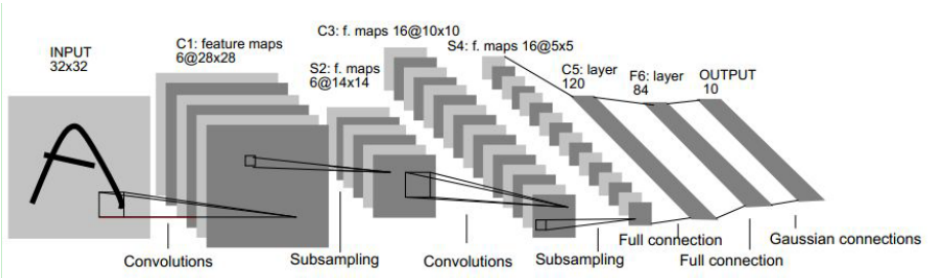
关闭

这两个阶段的工作一般应受到精度要求的控制。

网络的训练过程如下：

- (1)、选定训练组，从样本集中分别随机地寻求N个样本作为训练组；
- (2)、将各权值、阈值，置成小的接近于0的随机值，并初始化精度控制参数和学习率；
- (3)、从训练组中取一个输入模式加到网络，并给出它的目标输出向量；
- (4)、计算出中间层输出向量，计算出网络的实际输出向量；
- (5)、将输出向量中的元素与目标向量中的元素进行比较，计算出输出误差；对于中间层的隐单元也需要计算出误差；
- (6)、依次计算出各权值的调整量和阈值的调整量；
- (7)、调整权值和调整阈值；
- (8)、当经历M后，判断指标是否满足精度要求，如果不满足，则返回(3)，继续迭代；如果满足就进入下一步；
- (9)、训练结束，将权值和阈值保存在文件中。这时可以认为各个权值已经达到稳定，分类器已经形成。再一次进行训练，直接从文件导出权值和阈值进行训练，不需要进行初始化。

四、卷积神经网络LeNet-5结构分析



CNN是一种带有卷积结构的深度神经网络，通常至少有两个非线性可训练的卷积层，两个非线性的固定卷积层（又叫Pooling Layer或降采样层）和一个全连接层，一共至少5个隐层。

CNN的结构受到著名的Hubel-Wiesel生物视觉模型的启发，尤其是模拟视觉皮层V1和V2层中Simple Cell和Complex Cell的行为。

LeNet-5手写数字识别结构(上图)分析：

- 1. 输入层：N个32*32的训练样本

输入图像为32*32大小。这要比Mnist数据库中最大的字母还大。这样做的原因是希望潜在的明显特征如笔画断点或角点能够出现在最高层特征监测子感受野的中心。

- 2. C1层：

输入图片大小：	32*32
卷积窗大小：	5*5
卷积窗种类：	6
输出特征图数量：	6
输出特征图大小：	28*28 (32-5+1)
神经元数量：	4707 (28*28)*6)
连接数：	122304 (28*28*5*5*6)+(28*28*6)
可训练参数：	156 5*5*6+6

关闭

用6个5×5的过滤器进行卷积，结果是在卷积层C1中，得到6张特征图，特征图的每个神经元与输入图片中的5×5的邻域相连，即用5×5的卷积核去卷积输入层，由卷积运算可得C1层输出的特征图大小为(32-5+1)×(32-5+1)=28×28。

3. S2层：

输入图片大小：	(28*28)*6
卷积窗大小：	2*2
卷积窗种类：	6
输出下采样图数量：	6
输出下采样图大小：	14*14 (28/2)*(28/2)
神经元数量：	1176 (14*14)*6
连接数：	5880 2*2*14*14*6+14*14*6
可训练参数：	12 1*6+6

卷积和子采样过程：

(1)、卷积过程包括：用一个可训练的滤波器 f_x 去卷积一个输入的图像（第一阶段是输入的图像，后面的阶段就是卷积特征map了），然后加一个偏置 b_x ，得到卷积层 C_x 。

卷积运算一个重要的特点就是，通过卷积运算，可以使原信号特征增强，并且降低噪音。

(2)、子采样过程包括：每邻域四个像素求和变为一个像素，然后通过标量 W_{x+1} 加权，再增加偏置 b_{x+1} ，然后通过一个sigmoid激活函数，产生一个大概缩小四倍的特征映射图 S_{x+1} 。

利用图像局部相关性的原理，对图像进行子抽样，可以减少数据处理量同时保留有用信息。

卷积之后进行子抽样的思想是受到动物视觉系统中的“简单的”细胞后面跟着“复杂的”细胞的想法的启发而产生的。

降采样后，降采样层S2的输出特征图大小为 $(28 \div 2) \times (28 \div 2) = 14 \times 14$ 。

S2层每个单元的4个输入相加，乘以一个可训练参数，再加上一个可训练偏置。结果通过sigmoid函数计算。可训练系数和偏置控制着sigmoid函数的非线性程度。如果系数比较小，那么运算近似于线性运算，下采样相当于模糊图像。如果系数比较大，根据偏置的大小下采样可以被看成是有噪声的“或”运算或者有噪声的“与”运算。每个单元的2*2感受野并不重叠，因此S2中每个特征图的大小是C1中特征图大小的1/4（行和列各1/2）。

从一个平面到下一个平面的映射可以看作是作卷积运算，S-层可看作是模糊滤波器，起到二次特征提取的作用。隐层与隐层之间空间分辨率递减，而每层所含的平面数递增，这样可用于检测更多的特征信息。

4. C3层：

输入图片大小：	(14*14)*6
卷积窗大小：	5*5
卷积窗种类：	16
输出特征图数量：	16
输出特征图大小：	10*10 (14-5+1)
神经元数量：	1600 (10*10)*16
连接数：	151600 1516*10*10
可训练参数：	1516 6*(3*25+1)+6*(4*25+1)+3*(4*25+1)+1*(6*25+1)

C3层也是一个卷积层，它同样通过5x5的卷积核去卷积S2层，然后得到的特征map就只有10x10个神经元，但是它有16种不同的卷积核，所以就存在16个特征map了。这里需要注意的一点是：C3中的每个特征map是连接到

关闭

S2中的所有6个或者几个特征map的，表示本层的特征map是上一层提取到的特征map的不同组合（这个做法也并不是唯一的）。

C3中每个特征图由S2中所有6个或者几个特征map组合而成。为什么不把S2中的每个特征图连接到每个C3的特征图呢？原因有2点。第一，不完全的连接机制将连接的数量保持在合理的范围内。第二，也是最重要的，其破坏了网络的对称性。由于不同的特征图有不同的输入，所以迫使他们抽取不同的特征。

5. S4层：

输入图片大小：	(10*10)*16	
卷积窗大小：	2*2	
卷积窗种类：	16	
输出下采样图数量：	16	
输出下采样图大小：	(5*5)*16	
神经元数量：	400	(5*5)*16
连接数：	2000	(2*2*5*5*16)+(5*5*16)
可训练参数：	32	(1+1)*16

S4层是一个下采样层，由16个5*5大小的特征图构成。特征图中的每个单元与C3中相应特征图的2*2邻域相连接，跟C1和S2之间的连接一样。S4层有32个可训练参数（每个特征图1个因子和一个偏置）和2000个连接。

6. C5层：

输入图片大小：	(5*5)*16	
卷积窗大小：	5*5	
卷积窗种类：	120	
输出特征图数量：	120	
输出特征图大小：	1*1	(5-5+1)
神经元数量：	120	(1*120)
连接数：	48120	5*5*16*120*1+120*1
可训练参数：	48120	5*5*16*120+120

C5层是一个卷积层，有120个特征图。每个单元与S4层的全部16个单元的5*5邻域相连。由于S4层特征图的大小也为5*5（同滤波器一样），故C5特征图的大小为1*1，这构成了S4和C5之间的全连接。之所以仍将C5标示为卷积层而非全相联层，是因为如果LeNet-5的输入变大，而其他的保持不变，那么此时特征图的维数就会比1*1大。C5层有48120个可训练连接。

7. F6层：

输入图片大小：	(1*1)*120	
卷积窗大小：	1*1	
卷积窗种类：	84	
输出特征图数量：	1	
输出特征图大小：	84	
神经元数量：	84	
连接数：	10164	120*84+84
可训练参数：	10164	120*84+84

关闭

F6层有84个单元（之所以选这个数字的原因来自于输出层的设计），与C5层全相连。有10164个可训练参数。如同经典神经网络，F6层计算输入向量和权重向量之间的点积，再加上一个偏置。然后将其传递给sigmoid函数产生单元的一个状态。

8. OUTPUT层：

输入图片大小： 1*84

输出特征图数量： 1*10

最后，输出层由欧式径向基函数（EuclideanRadial Basis Function）单元组成，每类一个单元，每个有84个输入。换句话说，每个输出RBF单元计算输入向量和参数向量之间的欧式距离。输入离参数向量越远，RBF输出的越大。一个RBF输出可以被理解为衡量输入模式和与RBF相关联类的一个模型的匹配程度的惩罚项。用概率术语来说，RBF输出可以被理解为F6层配置空间的高斯分布的负log-likelihood。给定一个输入模式，损失函数应能使得F6的配置与RBF参数向量（即模式的期望分类）足够接近。这些单元的参数是人工选取并保持固定的（至少初始时候如此）。这些参数向量的成分被设为-1或1。虽然这些参数可以以-1和1等概率的方式任选，或者构成一个纠错码，但是被设计成一个相应字符类的7*12大小（即84）的格式化图片。这种表示对识别单独的数字不是很有用，但是对识别可打印ASCII集中的字符串很有用。

使用这种分布编码而非更常用的“1 of N”编码用于产生输出的另一个原因是，当类别比较大的时候，非分布编码的效果比较差。原因是大多数时间非分布编码的输出必须为0。这使得用sigmoid单元很难实现。另一个原因是分类器不仅用于识别字母，也用于拒绝非字母。使用分布编码的RBF更适合该目标。因为与sigmoid不同，他们在输入空间的较好限制的区域内兴奋，而非典型模式更容易落到外边。

RBF参数向量起着F6层目标向量的角色。需要指出这些向量的成分是+1或-1，这正好在F6 sigmoid的范围内，因此可以防止sigmoid函数饱和。实际上，+1和-1是sigmoid函数的最大弯曲的点处。这使得F6单元运行在最大非线性范围内。必须避免sigmoid函数的饱和，因为这将会导致损失函数较慢的收敛和病态问题。

五、卷积神经网络注意事项

1. 数据集的大小和分块

数据驱动模型一般依赖于数据集的大小，CNN和其它经验模型一样，能适用于任意大小的数据集，但用于训练的数据集应该足够大，能够覆盖问题域中所有已知可能出现的问题。设计CNN的时候，数据集中应该包含三个子集：训练集、测试集、验证集。训练集应该包含问题域中的所有数据，并在训练阶段用来调整网络权值。测试集用来在训练过程中测试网络对于训练集中未出现的数据的分类性能。根据网络在测试集上的性能情况，网络的结构可能需要做出调整，或者增加训练循环的次数。验证集中的数据同样应该包含在测试集训练集中没有出现过的数据，用于在确定网络结构后能够更加好的测试和衡量网络的性能。Looney等人建议，数据集中的65%用于训练，25%用于测试，剩余的10%用于验证。

2. 数据预处理

为了加速训练算法的收敛速度，一般都会采用一些数据预处理技术，这其中包括：去除噪声、输入数据降维、删除无关数据等。数据的平衡化在分类问题中异常重要，一般认为训练集中的数据应该相对于标签类别近似于平均分布，也就是每一个类别标签所对应的数据量在训练集中是基本相等的，以避免网络过于倾向于表现某些分类的特点。为了平衡数据集，应该移除一些过度富余的分类中的数据，并相应的补充一些相对样例稀少的分类中的数据。还有一个办法就是复制一部分这些样例稀少分类中的数据，并在这些输入数据中加入随机噪声。

3. 数据规则化

将数据规则化到一个统一的区间（如[0,1]）中具有很重要的优点：防止数据中存在较大数值的数据造成数值较小的数据对于训练效果减弱甚至无效化。一个常用的方法是将输入和输出（sigmoid函数等）相对应的区间。

[关闭](#)

4. 网络权值初始化

CNN的初始化主要是初始化卷积层和输出层的卷积核（权重）和偏置。

网络权值初始化就是将网络中的所有连接权值（包括阈值）赋予一个初始值。如果初始权值向量处在误差曲面的一个相对平缓的区域的时候，网络训练的收敛速度可能会异常缓慢。一般情况下，网络的连接权值和阈值被初始化为一个具有0均值的相对小的区间内均匀分布，比如[-0.30, +0.30]这样的区间内。

5. BP算法的学习速率

如果学习速率 η 选取的比较大则会在训练过程中较大幅度的调整权值 w ，从而加快网络训练的速度，但这会造成网络在误差曲面上搜索过程中频繁抖动且有可能使得训练过程不能收敛，而且可能越过一些接近优化 w 。同样，比较小的学习速率能够稳定的使得网络逼近于全局最优，但也有可能陷入一些局部最优区域。对于不同的学习速率设定都有各自的优缺点，而且还有一种自适应的学习速率方法，即 η 随着训练算法的运行过程而自行调整。

6. 收敛条件

有几个条件可以作为停止训练的判定条件，训练误差、误差梯度和交叉验证。一般来说，训练集的误差会随着网络训练的进行而逐步降低。

7. 训练方式

训练样例可以有两种基本的方式提供给网络训练使用，也可以是两者的结合：逐个样例训练(EET)、批量样例训练(BT)。在EET中，先将第一个样例提供给网络，然后开始应用BP算法训练网络，直到训练误差降低到一个可以接受的范围，或者进行了指定步骤的训练次数。然后再将第二个样例提供给网络训练。EET的优点是相对于BT只需要很少的存储空间，并且有更好的随机搜索能力，防止训练过程陷入局部最小区域。EET的缺点是如果网络接收到的第一个样例就是劣质（有可能是噪音数据或者特征不明显）的数据，可能使得网络训练过程朝着全反方向进行搜索。相对的，BT方法是在所有训练样例都经过网络传播后才更新一次权值，因此每一次学习周期就包含了所有的训练样例数据。BT方法的缺点也很明显，需要大量的存储空间，而且相比EET更容易陷入局部最小区域。而随机训练(ST)则是相对于EET和BT一种折衷的方法，ST和EET一样也是一次只接受一个训练样例，但只进行一次BP算法并更新权值，然后接受下一个样例重复同样的步骤计算并更新权值，并且在接受训练集最后一个样例后，重新回到第一个样例进行计算。ST和EET相比，保留了随机搜索的能力，同时又避免了训练样例中最开始几个样例如果出现劣质数据对训练过程的过度不良影响。

以上整理的内容主要摘自：

1. 《Gradient-Based Learning Applied to Document Recognition》，1998，Yann LeCun
2. 《卷积神经网络在图像识别上的应用研究》，2012，硕论，浙江大学
3. 《卷积神经网络的研究与应用》，2013，硕论，南京林业大学
4. 《卷积神经网络及其应用》，2014，硕论，大连理工大学
5. 《基于卷积神经网络的深度学习算法与应用研究》，2014，硕论，浙江工商大学

顶 踩
3 0

上一篇 BP神经网络公式推导及实现(MNIST)

下一篇 tiny-cnn开源库的使用(MNIST)

我的同类文章

关闭

Caffe (19) Neural Network (12) Deep Learning (8)

- | | | | |
|-------------------------------|--------------------|---|--------------------|
| • Caffe中Layer注册机制 | 2017-01-10 阅读 87 | • windows7下解决caffe check... 2017-01-09 阅读 121 | |
| • cifar数据集介绍及到图像转... | 2016-12-10 阅读 245 | • 深度学习开源库tiny-dnn的使... | 2016-12-04 阅读 465 |
| • 卷积神经网络(CNN)代码实... | 2016-12-03 阅读 2188 | • 一步一步指引你在Windows... | 2016-03-26 阅读 1381 |
| • Windows7 64bit VS2013 Ca... | 2016-03-26 阅读 1550 | • 卷积神经网络(CNN)的简单... | 2016-03-06 阅读 7518 |
| • tiny-cnn执行过程分析(MNIST) | 2016-01-31 阅读 3557 | • tiny-cnn开源库的使用(MNIST) | 2016-01-24 阅读 9464 |

[更多文章](#)[代码解析](#)[什么是学习综述](#)[layer](#)[python map函数](#)[数学之美 p](#)[1元拾壹里](#)[参考知识库](#)[猜你在找](#)[C++ 单元测试（GoogleTest）](#)[深度学习DL与卷积神经网络CNN学习笔记随笔-01-CNN](#)[《C语言/C++学习指南》加解密密篇（安全相关算法）](#)[深度学习DL与卷积神经网络CNN学习笔记随笔-01-CNN](#)[C语言系列之 字符串压缩算法与结构体初探](#)[深度学习DL与卷积神经网络CNN学习笔记随笔-01-CNN](#)[C语言系列之 递归算法示例与 Windows 趣味小项目](#)[深度学习DL与卷积神经网络CNN学习-01-CNN基础知识](#)[C语言系列之 字符串相关算法](#)[CNN笔记通俗理解卷积神经网络](#)[数学之美 pdf](#)[兼职工作](#)[layer](#)[什么是学习综述](#)[代码解析](#)[高考书法](#)[查看评论](#)4楼 [wmf1991yeah](#) 2016-10-13 19:16发表

你好，我在用MATLAB训练CNN的时候，目标函数与精度曲线不太正常，随着迭代次数的增加，误差不怎么变，而且还比较高呢，这是为什么呢？

Re: [fengbingchun](#) 2016-10-13 21:24发表

回复wmf1991yeah：不好判断，影响因素比较多，正负样本因素、参数设置导致等都有可能吧。

Re: [wmf1991yeah](#) 2016-10-14 21:47发表

回复fengbingchun：多谢，问题已经找出来了，是数据的问题，各个类别样本的数据数量最好保持平衡，二是归一化的方法是对多通道的整个图像一起归一化，而不是各个通道分别归一化

3楼 [wanto4rom](#) 2016-09-05 17:25发表

输出特征图大小中的+1 是不是假设填充是0并且Step为1的情况呢？因为我要跨越多个的话，那么应该就是 $(\text{数据Size} - \text{感受野Size} + 2 * \text{填充数量}) / \text{step} + 1$ 。

Re: [fengbingchun](#) 2016-09-06 08:14发表

回复wanto4rom：是step为1

Re: [wanto4rom](#) 2016-09-06 11:33发表

回复fengbingchun：多谢，我去自己实践一下，看看各个参数对结果的影响和速度的影响

Re: [fengbingchun](#) 2016-09-06 13:02发表

回复wanto4rom：赞

2楼 [pingye9](#) 2016-03-06 20:55发表

厉害厉害~

1楼 [冰水比水冰](#) 2016-01-17 19:47发表

能不能举个例子然后说一下具体实现呢，太高深了实在是没看完。。。。

Re: [fengbingchun](#) 2016-01-18 08:32发表

回复luoyhang003：这个会有的，公式推导和C++实现，正在实现中.....

[关闭](#)

您还没有登录,请[登录](#)或[注册](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

- 全部主题
- Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
- VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
- BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
- Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack FTC
- coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
- Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
- Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2016, CSDN.NET, All Rights Reserved



关闭