·体系结构与软件技术· 文章编号：1000-3428(2019)01-0017-06 文献标志码：A 中图分类号：TP399

面向移动平台的轻量级卷积神经网络架构

胡 挺[1,2]，祝永新[1]，田 犁[1]，封松林[1]，汪 辉[1]

（1. 中国科学院上海高等研究院，上海 201210；2. 中国科学院大学，北京 100049）

摘 要： 针对深度神经网络在移动平台上存在准确度低、过拟合等问题，提出一种轻量级的卷积神经网络架构。将 3×3 的深度可分离卷积替换 SqueezeNet 网络模型基本模块 Fire 中的标准 3×3 卷积核，并构建 SparkNet 的网络结构，替换模型卷积得到网络变形结构。实验结果表明，与 SqueezeNet 网络结构相比，该架构可以提高网络模型的计算速度，有效降低网络模型规模并减少参数数量。

关键词： 深度学习；卷积神经网络；深度可分离卷积；神经网络压缩；轻量级

中文引用格式： 胡挺，祝永新，田犁，等. 面向移动平台的轻量级卷积神经网络架构[J]. 计算机工程，2019，45(1)：17-22.

英文引用格式： HU Ting，ZHU Yongxin，TIAN Li，et al. Lightweight convolutional neural network architecture for mobile platforms[J]. Computer Engineering，2019，45(1)：17-22.

Lightweight Convolutional Neural Network Architecture for Mobile Platforms

HU Ting[1,2]，ZHU Yongxin[1]，TIAN Li[1]，FENG Songlin[1]，WANG Hui[1]

（1. Shanghai Advanced Research Institute，Chinese Academy of Sciences，Shanghai 201210，China；
2. University of Chinese Academy of Sciences，Beijing 100049，China）

【Abstract】 For the problem that the deep neural network has low accuracy and over-fitting on the mobile platforms，a lightweight Convolutional Neural Network（CNN） architecture is proposed. The 3 × 3 depthwise separable convolution replaces the standard 3 × 3 convolution kernel in the SqueezeNet network model basic module Fire，constructs the SparkNet network structure，and replaces the model convolution to obtain the network deformation structure. Experimental results show that compared with the SqueezeNet network structure，the architecture can improve the calculation speed of the network model，effectively reduce the network model size and reduce the number of parameters.

【Key words】 deep learning；Convolutional Neural Network（CNN）；depthwise separable convolution；neural network compression；lightweight

DOI：10.19678/j.issn.1000-3428.0049905

0 概述

随着移动设备的普及，深度神经网络在移动平台上具有极大的应用价值。虽然移动设备具有强大的计算能力和丰富的存储资源，但是在移动设备上运行深度神经网络仍然是一个巨大的挑战。

为提高卷积神经网络（Convolutional Neural Network，CNN）模型的准确度，研究人员不断增加神经网络的层数，这使得 CNN 模型越来越复杂，模型参数急剧增加。例如，AlexNet 网络模型[1]的参数数量达到 61M，VGGNet 网络模型和 GoogleNet 网络模型的参数数量更多。CNN 模型的参数数量越多，其运行所需要的计算资源和存储资源也越多。因此，使用特定的方法，可以保持网络准确度的同时压缩网络模型。对于给定的准确度水平，复杂度小的网络模型具有如下优点：在分布式训练平台上对模型进行训练时，较小的网络模型各节点之间的数据交换更少；云服务器向客户端部署网络模型时，需要的带宽更小；较小的网络模型能够便利地部署在移动平台上。

针对上述研究，本文基于深度可分离卷积，提出一种轻量级的高效低延时 CNN 架构。在 SparkNets 基础上，利用变形结构将卷积神经网络部署在移动平台上，运用 CNN 模型中的冗余在移动设备上进行高效的硬件实现。

基金项目： 国家重点研发计划（2017YFA0206104）；上海市科学技术委员会科研计划项目（16511108701）；上海市张江管委会公共服务平台项目（2016-14）。

作者简介： 胡 挺（1992—），男，硕士研究生，主研方向为深度学习、目标检测、网络压缩；祝永新，研究员、博士；田 犁，副研究员、博士；封松林、汪 辉（通信作者），研究员、博士。

收稿日期： 2017-12-28 **修回日期：** 2018-02-01 **E-mail：** hut@sari.ac.cn

1 相关工作

CNN 模型存在大量的冗余,这种冗余会对计算资源和存储资源造成巨大的浪费。近年来,卷积神经网络压缩引起越来越多研究人员的关注,主要分为 3 类:网络剪裁,网络参数量化和直接训练轻量级的网络。

网络裁剪广泛应用于压缩网络模型,这种方法在降低网络复杂度的同时具有防止过拟合的作用。文献[2]提出基于偏置参数衰减的网络裁剪方法。文献[3-4]认为基于目标函数海森矩阵的网络裁剪方法比基于模的网络裁剪方法更精确,然而计算二次偏差计算量非常大。文献[5]以神经元输出值的方差作为相关性界限,提出改进的相关性剪枝方法,取得较好的效果。文献[6]在目标函数后加入一个可调参数的惩罚函数项,将单一惩罚函数项泛化为一类随参数规律变化的新的惩罚函数,得到更好的剪枝效果。文献[7]利用灰色关联分析方法,提出一种新的剪枝算法,通过剪枝和并枝2 个阶段实现网络结构的优化,提高了网络的数值效能。文献[8]提出网络压缩方法,该方法具有潜在的网络不可恢复破坏的风险,且训练效率较低。

深度神经网络模型要达到高精度,其参数的位数无需太高。参数量化方法的原理是降低表征网络参数甚至是特征图所用数字的位数。文献[9]提出一种用 8 位定点整数表示网络特征图的方法。文献[10]提出使用向量量化的方法压缩神经网络。文献[11]提出 HashNets 利用哈希方程将网络参数随机分配到哈希篮子里面,在同一个篮子里面的参数可以共享一个值。目前,BinaryConnect、Binarized Neural Networks 和 XNOR-Net 能够以 32 倍的比例压缩网络模型大小[12-14],但是这些方法会给网络的准确度造成一定损伤[15]。参数量化方法和网络裁剪方法是正交的,因此,这 2 种方法相结合可以达到更高的压缩比例。

对于给定的准确度水平,可以直接设计一种更加轻量级的网络结构。轻量级的网络结构所需要的参数数量,计算量和交换数据所需要的带宽更小。因此,轻量级的网络结构更适合应用到存储资源和计算资源均有限的移动硬件平台上。文献[16]利用瓶颈方法设计一种达到 AlexNet 相同精确度水平的轻量级网络结构,这种网络结构模型的大小只有 AlexNet 网络模型的 1/50。文献[17]基于深度可分离卷积提出一种 MobileNets 网络结构,该卷积方法的计算量只有标准卷积方法的 1/9。

2 SparkNets 架构

基于深度可分离卷积,本文设计一种轻量级的 CNN 架构及其 2 种变形架构。

2.1 深度可分离卷积

深度可分离卷积是一种标准卷积的分解形式,这种卷积结构将标准卷积分解为在每个通道平面上的卷积和通道间的卷积。图 1 为标准的卷积结构和深度分离卷积结构。标准的卷积结构将过滤和组合输入到一组输出,而可分离卷积结构将卷积分为 2 层,分别为过滤和组合。这种卷积结构不仅可以极大地降低卷积神经网络的复杂度,减小网络模型的参数数量和计算量,而且不会造成明显的精确度损失。MobileNets 使用 3 ×3 的深度可分离卷积在几乎可忽略的精度损失的前提下,达到比标准卷积神经网络模型参数减少 8 倍 ~9 倍的效果。

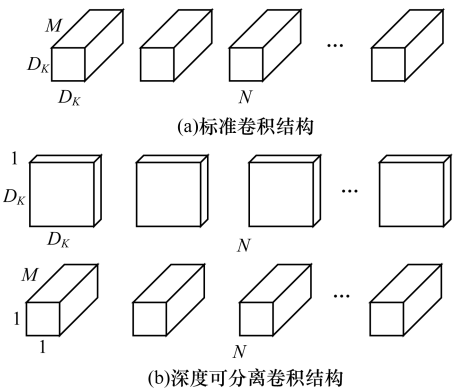


图 1 标准卷积结构和深度可分离卷积结构

2.2 网络结构

本文所提出的网络结构主要受 SqueezeNet 网络模型的启发。SqueezeNet 网络模型的核心模块是一个名为 Fire 的基本模块,如图 2 所示。Fire 模块利用以下2 个重要的策略:

1)用大量 1 ×1 的卷积模块替代 3 ×3 的卷积模块。由于 1 ×1 卷积模块的参数数量只有 3 ×3 卷积模块参数数量的 1/9,这种替换可以极大地减小网络模型的参数数量。

2)使用一种由一定数量 1 ×1 卷积核组成的压缩层降低输入特征图的通道数量,这样可以大大降低卷积计算时的计算量。

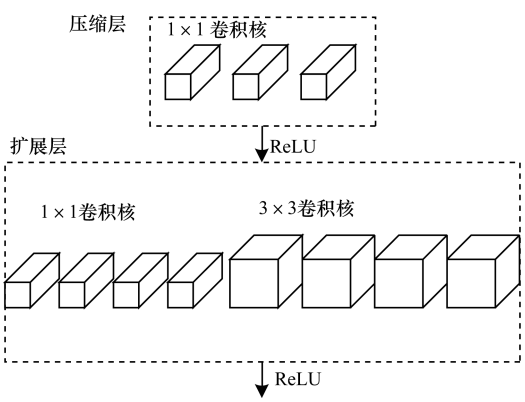


图 2 Fire 模块结构示意图

通过运用这 2 个策略,SqueezeNet 在保证网络精确度没有损失的前提下,减小网络模型的规模。本文提出的方法通过将 3×3 的深度可分离卷积替换 SqueezeNet 网络模型基本模块 Fire 中的标准 3×3卷积核,构建一种新的卷积神经网络基本模块,将新提出的基本模块命名为 Spark,表示比 Fire 更小但是蕴含着更大的力量,如图 3 所示。以 Spark 模块为基础构建的卷积神经网络 SparkNets 比同等精确度的 SqueezeNet 的参数数量降低了 3 倍,比同等精确度的标准卷积神经网络模型规模至少降低 100 倍。

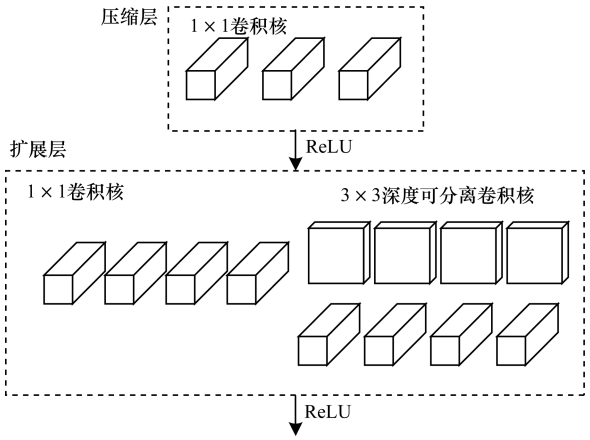


图 3　Spark 模块结构示意图

Spark 模块将 $D_F \times D_F \times M$ 大小的特征图作为输入,$D_G \times D_G \times N$ 大小的特征图作为输出。其中,D_F 和 D_G 分别是输入特征图和输出特征图的图像尺寸,M 和 N 分别是输入特征图和输出特征图的通道数量。设定模块中扩展层的 1×1 卷积核和 3×3 深度分离卷积核的数量相等,则 Spark 模块的计算量为:

$$C_{\text{Spark}} = M \times S \times D_F \times D_F + S \times N \times D_F \times D_F + D_K \times D_K \times S \times D_F \times D_F \quad (1)$$

其中,S 表示 Spark 模块中压缩层中 1×1 卷积核的数量。Spark 模块中参数的数量为:

$$N_{\text{Spark}} = M \times S + S \times N + D_K \times D_K \times S \quad (2)$$

Fire 模块的计算量为:

$$C_{\text{Fire}} = M \times S \times D_F \times D_F + S \times \frac{N}{2} \times D_F \times D_F + D_K \times D_K \times S \times \frac{N}{2} \times D_F \times D_F \quad (3)$$

Fire 模块的参数数量为:

$$N_{\text{Fire}} = M \times S + S \times \frac{N}{2} + D_K \times D_K \times S \times \frac{N}{2} \quad (4)$$

通过替换 Fire 模块中的标准卷积模块,Spark 模块的计算量与 Fire 模块的计算量比值以及参数数量的比值均为:

$$R = \frac{C_{\text{Spark}}}{C_{\text{Fire}}} = \frac{N_{\text{Spark}}}{N_{\text{Fire}}} = 1 - \frac{(N-2)D_K^2 - N}{(D_K^2+1)N + 2M} \quad (5)$$

当使用 3×3 深度分离卷积替换标准 3×3 卷积时,Spark 模块的计算量是 Fire 模块的 3/11 左右。而 SqueezeNet 与标准的卷积神经网络模型相比,计算量降低 50 倍。因此,本文基于 Spark 模块构建的 SparkNets 与标准的卷积神经网络相比,计算量降低至少 100 倍。

图 4 所示为 SparkNet 的网络架构。SparkNet 由一个标准的卷积层开始,在其上有一定数量的 Spark 模块,最后以一个标准的卷积层结束。除去最后一层,每个层之间插入批量化层(Batchnorm Layer,BL)和 ReLU 激活层。Spark 模块间插入最大池化层。对于网络准确度而言,因为较低位置网络层的影响比较高位置网络层的影响大,所以第一层使用标准的卷积层而不是 Spark 模块,同时可以避免网络准确度的损失。

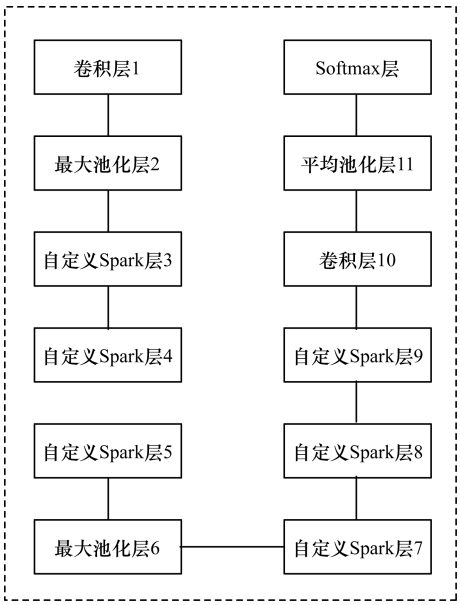


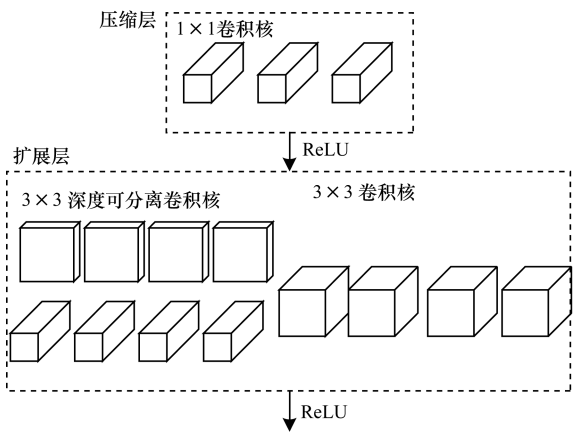
图 4　SparkNets 网络架构

在 CNN 中,最后一层卷积层的输出一般作为一个全连接层的输入。但是,全连接层通常会导致过拟合,更重要的是它是参数集中的层,拥有的参数数量较大,这种特性导致全连接层不适合迁移到移动设备上,全局平均层是一个很好的替代。根据文献[18],全局平均层没有任何参数,可以避免过拟合的发生,而且更适合迁移到移动设备上。因此,在 SparkNet 中,用全局平均层替代全连接层。

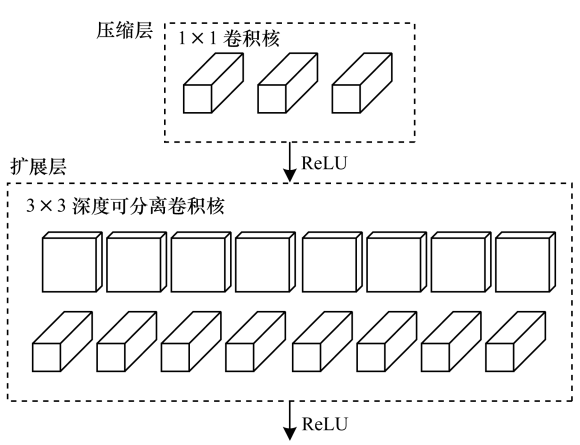
2.3　网络变形

本文不仅对 Fire 模块扩展层中的 3×3 标准卷积进行替换,而且尝试替换其他卷积。不同的替换方式,产生不同的网络基本模块的变形。当扩展层中的 1×1 卷积被替换时,得到第 1 种基本模块的变形 Spark_v1。当扩展层中的 1×1 卷积和 3×3 卷积同时被替换时,得到了第 2 种基本模块的变形 Spark_v2。这 2 种基本模块的变形结构如图 5所示。当压缩层

和扩展层中的所有卷积全部被替换时，得到 MobileNets 的一种变形结构。



(a)网络变形结构Spark_v1



(b) 网络变形结构Spark_v2

图 5 2 种网络变形结构

3 实验结果与分析

本文在 4 种图像分类基准数据集上对 SparkNet 进行实验评测，这 4 种图像分类数据集分别为 MNIST[19]、CIFAR-10[20]、CIFAR-100[20]和 SVHN[21]。

3.1 MNIST 数据集

MNIST 数据集是一个由 60 000 张二值训练图像和 10 000 张二值测试图像组成的图像分类数据集。其图像均为 28×28 像素大小的二值手写数字。本文在数据集上训练一个由 3 层 Spark 模块和 1 层 Softmax 层组成的 SparkNets 模型，不同网络的对比结果如表 1 所示。

表 1 不同网络结构在 MNIST 数据集上的对比结果

网络结构	测试集错误率/%	参数数量
LeNet-5 Pruned[8]	0.77	36 000
LeNet-5 Ref[19]	0.80	431 000
LeNet-5 Compressed[22]	0.74	11 000
SqueezeNet	1.06	11 000
SparkNet	2.23	4 000

从表 1 可以看出，在 SparkNet 与 SqueezeNet 保持同等准确度水平的条件下，网络模型降低 3 倍。与 LeNet-5 网络模型相比，SparkNet 的模型参数数量减少 108 倍。虽然 LeNet-5 Pruned 模型和 LeNet-5 Compressed 模型的错误率比 SparkNet 更低，但是这 2 种模型的训练复杂度较高，参数数量较大。因此，与这 2 种模型相比，SparkNet 应用到移动平台时具有显著的优势。

3.2 CIFAR-10 数据集

CIFAR-10 数据集是一个由 50 000 张彩色训练图像和 10 000 张彩色测试图像组成的图像分类数据集，其图像为 32×32 像素大小的属于 10 个物体类别的彩色图像。实验通过随机水平翻转图像对训练数据进行扩增处理。本文网络结构在 CIFAR-10 数据集上的实验对比结果如表 2 所示。在准确度方面，SparkNet 模型比其他方法具有更好的表现，与 SqueezeNet 模型相比，模型规模降低 3 倍。

表 2 不同网络结构在 CIFAR-10 数据集上的对比结果

网络结构	测试集错误率/%	参数数量
Stochastic pooling[23]	13.15	—
CNN + Spearmint[24]	14.13	—
Conv + maxout + dropout[25]	11.68	—
SqueezeNet	9.74	8 800
SparkNet	11.06	2 900

3.3 CIFAR-100 数据集

CIFAR-100 数据集和 CIFAR-10 数据集的组成方式基本一致，只是 CIFAR-100 数据集中有 100 类不同的图像，每种类别的图像数量是 CIFAR-10 数据集中的 1/10，区分难度更大，对比结果如表 3 所示。SparkNet 能够以较小的网络规模达到与其他网络同等的准确度水平。

表 3 不同网络结构在 CIFAR-100 数据集上的对比结果

网络结构	测试集错误率/%	参数数量/10^6
Stochastic pooling[23]	42.51	—
Conv + maxout + dropout[25]	38.57	—
Learned pooling[26]	43.71	—
SqueezeNet	39.80	2.33
SparkNet	43.19	0.85

3.4 SVHN 数据集

SVHN 数据集是数字分类数据集，其中的数字图像取自现实街景中出现的数字。这个数据集的训练图像有 604 000 张，测试图像有 26 000 张，图像为 32×32 像素大小的彩色图像，分类难度比 MNIST 数据集更大。实验采用与 CIFAR-10 一致的图像预处理方式。如表 4 所示，SparkNet 能够以 1/3 的网络规模达到与 SqueezeNet 同等的准确度水平，证明 SparkNet 的有效性。

表 4　不同网络结构在 SVHN 数据集上的对比结果

网络结构	测试集错误率/%	参数数量/10^6
NIN + Dropout[18]	2.35	—
Stochastic pooling[23]	2.80	—
Conv + maxout + dropout[25]	2.47	—
Rectifier + dropout[27]	2.78	—
SqueezeNet	4.02	2.20
SparkNet	4.20	0.72

3.5　网络变形结构

本文对网络基本模块的变形模块所构成的网络在 CIFAR-10 数据集上进行实验。与 SqueezeNet 和 SparkNet 对比结果如表 5 所示。

表 5　网络变形结构在 CIFAR-10 数据集上的对比结果

网络结构	训练集错误率/%	测试集错误率/%	参数数量/10^6
SqueezeNet	4.69	9.74	8.8
SparkNet	7.83	11.06	2.9
SparkNet_v1	3.78	9.38	8.8
SparkNet_v2	6.32	12.71	2.9

在表 5 中,与 SqueezeNet 网络模型相比,具有同等准确度水平的 SparkNet 网络模型的参数数量降低 3 倍左右,而测试集错误率仅上升 1.32% 。第 1 种变形网络模型 SparkNet_v1 和 SqueezeNet 的网络参数数量基本相同,但是具有较好的准确度表现。SparkNet_v1 与 SqueezeNet 相比,训练集错误率下降 0.91% ,测试集错误率下降 0.36% 。2 种网络结构的区别在于 SparkNet_v1 用 3 ×3 深度可分离卷积替换扩展层中的 1 ×1 标准卷积。这说明 3 ×3 深度可分离卷积和 1 ×1 标准卷积的参数数量基本相同,但是 3 ×3 深度可分离卷积具有更强的表现能力,这使得网络的表现效果更好。当采用第 2 种变形网络模型 SparkNet_v2 时,与 SqueezeNet 相比,网络模型的训练集错误率上升 1.63% ,测试集错误率上升 2.97% 。与 SparkNet 相比,SparkNet_v2 训练集错误率更低,测试集错误率更高,说明 SparkNet_v2 具有过拟合的倾向,这种现象说明在 SparkNet 中存在的 1 ×1 卷积核具有一定的稀疏网络结构,可以用来防止过拟合。

4　结束语

本文提出一种轻量级的高效卷积神经网络架构。基于 SqueezeNet 网络模型,构建 SparkNets 网络结构,替换模型卷积得到网络变形结构。实验结果表明,与 SqueezeNet 相比,该架构的规模降低了 3 倍左右。与传统的卷积神经网络模型相比,该架构规模至少降低 100 倍。下一步将研究 SparkNets 在计算机视觉中的目标检测、图像分割、自然语言处理等领域的表现,以探究 SparkNets 的高效性和通用性。

参考文献

[1] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[C]//Proceedings of the 25th International Conference on Neural Information Processing Systems. [S.l.]:Curran Associates Inc.,2012:1097-1105.

[2] PRATT L Y. Comparing biases for minimal network construction with back-propagation[C]//Proceedings of International Conference on Neural Information Processing Systems. Cambridge,USA:MIT Press,1988:177-185.

[3] LE Y, DENKER, JOHN S, et al. Optimal brain damage[C]//Proceedings of the 2nd International Conference on Neural Information Processing Systems. Cambridge,USA:MIT Press,1989:598-605.

[4] HASSIBI B,STORK D G. Second order derivatives for network pruning:optimal brain surgeon[J]. Advances in Neural Information Processing Systems, 1993, 5:164-171.

[5] 李小夏,李孝安. 一种改进的神经网络相关性剪枝算法[J]. 电子设计工程,2013,21(8):65-67.

[6] 熊俊,王士同,潘永惠,等. 基于惩罚函数泛化的神经网络剪枝算法研究[J]. 计算机工程,2014,40(11):149-154.

[7] 赵蓉,唐楚淇,刘伟林,等. 一种新的基于灰色关联分析的 BP 神经网络剪枝算法[J]. 科技创新与应用,2016(13):17-18.

[8] TRAN J,TRAN J,TRAN J,et al. Learning both weights and connections for efficient neural networks[EB/OL]. [2017-11-20]. https://arxiv. org/abs/1506. 02626.

[9] VANHOUCKE V, SENIOR A, MAO M Z. Improving the speed of neural networks on CPUs[EB/OL]. [2017-11-20]. https://ai. google/research/pubs/pub37631.

[10] GONG Y, LIU L, YANG M, et al. Compressing deep convolutional networks using vector quantization[EB/OL]. [2017-11-20]. http://adsabs. harvard. edu/abs/2014arXiv1412. 6115G.

[11] CHEN W,TYREE S,TYREE S,et al. Compressing neural networks with the hashing trick[C]//Proceedings of the 32nd International Conference on Machine Learning. Cambridge,USA:MIT Press,2015:2285-2294.

[12] 杨志刚,吴俊敏,徐恒,等. 基于虚拟化的多 GPU 深度神经网络训练框架[J]. 计算机工程,2018,44(2):68-74,83.

[13] COURBARIAUX M, BENGIO Y, DAVID J P. Binaryconnect:training deep neural networks with binary weights during propagations[C]//Proceedings of International Conference on Neural Information Processing Systems. Cambridge,USA:MIT Press,2015:3123-3131.

[14] COURBARIAUX M, HUBARA I, SOUDRY D, et al. Binarized neural networks:training deep neural networks with weights and activations constrained to +1 or -1 [EB/OL]. [2017-11-20]. https://arxiv. org/pdf/1602.

02830v3. pdf.
[15] RASTEGARI M, ORDONEZ V, REDMON J, et al. Xnor-net: imagenet classification using binary convolutional neural networks[C]//Proceedings of the European Conference on Computer Vision. Berlin, Germany: Springer, 2016: 525-542.
[16] IANDOLA F N, HAN S, MOSKEWICZ M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size[EB/OL]. [2017-11-20]. https://arxiv.org/pdf/1602.07360v3.pdf.
[17] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications[EB/OL]. [2017-11-20]. https://arxiv.org/abs/1704.04861.
[18] LIN M, CHENQ, YAN S. Network in network[EB/OL]. [2017-11-20]. https://arxiv.org/pdf/1312.4400.pdf.
[19] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-324.
[20] KRIZHEVSKY A. Learning multiple layers of features from tiny images[EB/OL]. [2017-11-20]. https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.
[21] NETZER Y, WANG T, COATES A, et al. Reading digits in natural images with unsupervised feature learning[EB/OL]. [2017-11-20]. http://101.110.118.29/ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.
[22] HAN S, MAO H, DALLY W J. Deep compression: compressing deep neural networks with pruning, trained quantization and Huffman coding[EB/OL]. [2017-11-20]. https://arxiv.org/abs/1510.00149.
[23] ZEILER M D, FERGUS R. Stochastic pooling for regularization of deep convolutional neural networks[EB/OL]. [2017-11-20]. https://arxiv.org/abs/1301.3557.
[24] SNOEK J, LAROCHELLE H, ADAMS R P. Practical bayesian optimization of machine learning algorithms[C]//Proceedings of the 25th International Conference on Neural Information Processing Systems. [S.l.]: Curran Associates Inc., 2012: 2951-2959.
[25] GOODFELLOW I J, WARDE-FARLEY D, MIRZA M, et al. Maxout networks[J] Machine Learning, 2013, 28(3): 1319-1327.
[26] MALINOWSKI M, FRITZ M. Learnable pooling regions for image classification[EB/OL]. [2017-11-20]. https://arxiv.org/abs/1301.3516? context = cs.CV.
[27] SRIVASTAVA N. Improving neural networks with dropout[EB/OL]. [2017-11-20]. http://www.cs.toronto.edu/~nitish/msc_thesis.pdf.

编辑 赵 辉

(上接第 16 页)

[9] LIU X, THOMSEN C, PEDERSEN T B. 3XL: an efficient DBMS-based triple-store[C]//Proceedings of International Workshop on Database and Expert Systems Applications. Washington D. C., USA: IEEE Computer Society, 2012: 284-288.
[10] ATRE M, CHAOJI V, ZAKI M J, et al. Matrix "Bit" loaded: a scalable lightweight join query processor for RDF data[C]//Proceedings of the 19th International Conference on World Wide Web. New York, USA: ACM Press, 2010: 41-50.
[11] MELNIK S. Storing RDF in a relational database[EB/OL]. [2017-10-21]. http://infolab.stanford.edu/~melnik/rdf/db.html.
[12] PAN Z, HEFLIN J. DLDB: extending relational databases to support semantic Web queries[EB/OL]. [2017-10-21]. http://swat.cse.lehigh.edu/pubs/pan04a.pdf.
[13] 陶承恺. 基于属性表的 RDF 数据存储系统研究[D]. 南京: 南京大学, 2013.
[14] LIAROU E, IDREOS S, MANEGOLD S, et al. MonetDB/datacell: online analytics in a streaming column-store[J]. Proceedings of the VLDB Endowment, 2012, 5(12): 1910-1913.
[15] 阮彤, 王梦婕, 王昊奋, 等. 垂直知识图谱的构建与应用研究[J]. 知识管理论坛, 2016, 1(3): 226-234.
[16] SALEEM M, PADMANABHUNI S S, NGOMO A C N, et al. TopFed: TCGA tailored federated query processing and linking to LOD[J]. Journal of Biomedical Semantics, 2014, 5(1): 47.
[17] STRATTON M R, CAMPBELL P J, FUTREAL P A. The cancer genome[J]. Nature, 2009, 458(7239): 719-724.
[18] 俞思伟, 范昊, 王菲, 等. 基于知识图谱的智能医疗研究[J]. 医疗卫生装备, 2017, 38(3): 109-111.
[19] TELLO A, CRUZ A L, SAQUICELA V, et al. RDF-ization of DICOM medical images towards linked health data cloud[C]//Proceedings of American Congress on Biomedical Engineering CLAIB. Berlin, Germany: Springer, 2015: 757-760.
[20] HARRIS S, SEABORNE A, PRUD'HOMMEAUX E. SPARQL 1.1 query language[EB/OL]. [2017-10-21]. https://www.w3.org/TR/sparql11-query/.
[21] 佟强, 程经纬, 张富, 等. 基于查询转换的 RDF 高效查询方法[J]. 吉林大学学报(工学版), 2015, 45(5): 1550-1558.

编辑 赵 辉