深度神经网络的压缩研究*

韩云飞[1,2,3], 蒋同海[1,2,3†], 马玉鹏[1,2,3], 徐春香[1,2,3], 张 睿[1,2,3]
(1. 中国科学院新疆理化技术研究所, 乌鲁木齐 830011; 2. 新疆民族语音语言信息处理实验室, 乌鲁木齐 830011; 3. 中国科学院大学, 北京 100049)

摘 要: 深度神经网络中过多的参数使得自身成为高度计算密集型和内存密集型的模型,这使得深度神经网络的应用不能轻易地移植到嵌入或移动设备上以解决特殊环境下的实际需求。为了解决该问题,提出了基于网络删减、参数共享两者结合的神经网络压缩方案。首先通过删减掉权重小于阈值的网络连接,保留其重要的连接;然后使用 K-means 聚类算法将删减后每层的参数进行聚类,每簇内的各个参数共享该簇的中心值作为其权重。实验在 MNIST 数据集上完成手写数字识别功能的 LeNet-300-100 网络和修改得到的 LeNet-300-240-180-100 网络分别压缩了 9.5×和 12.1×。基于网络删减、参数共享两者结合的神经网络压缩方案为未来在特殊环境下更丰富的基于深度神经网络的智能应用提供了可行方案。

关键词: 神经网络; 压缩; 网络删减; 参数共享
中图分类号: TP183 **文献标志码**: A **文章编号**: 1001-3695(2018)10-2894-04
doi:10.3969/j.issn.1001-3695.2018.10.003

Compression of deep neural networks

Han Yunfei[1,2,3], Jiang Tonghai[1,2,3†], Ma Yupeng[1,2,3], Xu Chunxiang[1,2,3], Zhang Rui[1,2,3]
(1. *Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi* 830011, *China*; 2. *Xinjiang Laboratory of Minority Speech & Language Information Processing, Urumqi* 830011, *China*; 3. *University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: Over-parameterized deep neural networks are both computationally intensive and memory intensive, making them too difficult to deploy on embedded or mobile systems to deal with some actual needs in special environment. To address this problem, this paper proposed a neural networks compression method based on pruning and parameters sharing. First, it pruned the network connections which's weight was smaller than the threshold, and retaining the important connections. Then, it clustered the parameters of each pruned layer, and the parameters of each cluster shared the cluster's central value as their weights. In the course of experiment, LeNet-300-100 network and LeNet-300-240-180-100 network obtained by updating on prior both were trained to recognize handwritten digits on MNIST datasets, and they were compressed by 9.5× and 12.1× respectively with about 1% loss of accuracy. The neural networks compression method based on pruning and parameters sharing makes it possible for intelligent application of deep neural network in special environment in the further.

Key words: neural networks; compression; network pruning; parameters sharing

0 引言

深度神经网络在图像处理、语音处理、自然语言处理等方面有着显著的精度提升。在图像处理中,如多伦多大学的 Krizhevsky 等人[1]在 2012 年的 ILSVRC(ImageNet Large Scale Visual Recognition Competition)上,通过在像素值的基础上训练一个很大很深的 CNN(convolutional neural networks),即 AlexNet,在图像分类和图像定位中取得了最优异的成绩。牛津大学的 Simonyan 等人[2]在 2014 年的 ILSVRC 上,通过卷积神经网络 VGG-16 在图像分类和定位综合任务中同样取得了优异的成绩,以及 2014 年 ILSVRC 上的 GoogLenNet[3]和 2015 年的 ResNet[4]都在图像处理上取得了很大的成功。在语音识别研究中,Dong 等人[5]在 2015 年采用结合深度神经网络和隐马尔可夫模型实现了人声转换,通过 MOS(mean opinion score)评测,证明该方法能获得很好的人声转换结果。Dat 等人[6]在 2016 年对城市环境中的多通道语音处理方法进行了比较,发现通过结合深度神经网络和循环神经网络的方法可以有效地降低语音识别的错误率。与此同时,百度和科大讯飞也都基于深度神经网络在语音识别上取得了突飞猛进的性能提升。在自然语言处理研究中,自然语言属于人类认知过程中产生的认知抽象实体,而语音和图像属于较为原始的输入,因此自然语言并没有像语音和图像那样适合采用深度神经网络模型。但是,深度神经网络还是为自然语言处理带来了新的机遇。Collobert 等人[7,8]在 2008 年和 2011 年,通过建立一个深度神经网络结构模型从而在自然语言处理的多个任务中获得良好的成绩,实现了一个高效的具备多功能的自然语言处理工具。

目前,深度神经网络在工业界有着极为丰富的应用,如百度的语音识别和微软 Cortana 虚拟助理上所使用的语音识别,以及 DeepMind 和 AlchemyAPI 两家企业都在基于深度神经网络技术为客户提供人工智能相关服务,并获得了极大的成功。基于这些智能服务的应用也层出不穷,应用通过申请 API,认证后接入服务接口,将在移动端或嵌入式端上采集到的图像数据、语音数据、文本数据、非结构化数据等进行压缩或非压缩或特征提取后经网络发送给 API 提供商,由 API 提供商对数据进

收稿日期: 2017-05-08; **修回日期**: 2017-07-26 **基金项目**: 中国科学院科技服务网络计划(STS 计划)资助项目(KFJ-EW-STS-129);中国科学院西部之光人才培养计划资助项目(XBBS201319);中国科学院青年创新促进会资助项目;新疆维吾尔自治区引进高层次人才计划资助项目

作者简介:韩云飞(1990-),男,博士研究生,主要研究方向为物联网技术;蒋同海(1963-),男(通信作者),研究员,博士,主要研究方向为物联网技术(jth@ms.xjb.ac.cn);马玉鹏(1979-),男,副研究员,博士,主要研究方向为物联网技术;徐春香(1982-),女,副研究员,博士,主要研究方向为物联网技术;张睿(1990-),男,博士研究生,主要研究方向为物联网技术.

行计算后将结果再返回给移动端。在大多数情况下,该方案是第三方应用最优的选择,其优点主要有:a)节省自行开发智能模块成本;b)节省智能模块的储存、运行和维护成本。但是考虑到实际生产环境,如深山作业、窨井作业、隧道作业、矿井作业、偏远及荒野地区作业和信号屏蔽区作业等,第三方应用在无网络连接下无法与服务提供商进行数据交互,这将导致智能应用无法正常使用。针对此类特殊情况,可以通过将深度神经网络应用移植到嵌入式设备或移动设备上来解决。

深度神经网络模型中过多的参数使得其自身成为高度计算密集型和内存密集型的模型,这使得移植深度神经网络的应用到嵌入式或移动设备上将遇到三个较大的难点:a)模型大,如 AlexNet 的模型大小就超过 200 MB[9],VGG-16 的模型大小超过了 500 MB[2];b)计算量大,表现良好的深度神经网络模型都有着成千上万的参数,运行一次智能服务需要进行成千上万次的计算才能获得其结果;c)耗电量大,大量的内存访问和 CPU 计算资源使用将导致巨大的耗电量。对于硬件资源有限的嵌入式或移动设备而言,完整的深度神经网络模型几乎无法直接移植进行离线方式使用,智能应用无法正常工作,这也是本文研究的主要出发点。

本文研究目标是要合理地减少深度神经网络中的参数,从而减小模型大小、减少计算量、减少耗电,最终使得基于深度神经网络模型的应用能够移植部署在嵌入式或移动设备上。本文采用网络删减结合参数共享对深度神经网络进行压缩,首先对原始神经网络进行训练,删减掉权重小于阈值的连接,将其权重置零,保留信息量较大的连接权重不变,为了尽量保持其网络准确率不变,在删减后的网络上再次进行训练微调;然后,针对每一层的参数进行 K-means 聚类,落于每个相同簇的多个参数共享中心值作为权重,在尽量保持准确率不变的前提下,优化 k 的取值。压缩流程如图 1 所示。

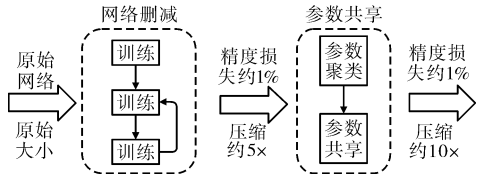

图 1　压缩流程

最终,网络删减结合参数共享可以在可控的精度损失下有效地压缩神经网络模型的参数,两者共同作用可以提高神经网络模型的压缩率,将 LeNet-300-100 网络压缩了 9.5 ×,将基于前者得到的 LeNet-300-240-180-100 网络压缩了 12.1 ×。压缩后的深度神经网络模型参数存储需求较低,为移植到嵌入式设备或移动设备上提供了有效解决方案。

1　相关工作

深度神经网络通常有着过多的参数,存在着极大的参数冗余现象[10],这将严重浪费设备的内存和计算资源,加大了嵌入式或移动设备的能耗。目前,针对参数冗余问题,研究人员也提出了不同的解决方法。Kim 等人[11]在 2015 年提出一种简单有效的方式来压缩 CNN 模型,首先进行变分贝叶斯矩阵分解的秩选择,然后再进行核张量 Tucker 分解,最后再次对模型进行调整。通过分别将 AlexNet、VGG-S、GoogLeNet 和 VGG-16 进行压缩并移植到智能手机上,该方法得到了有效证明。Soulie 等人[12]在 2016 年提出一种在模型学习阶段进行压缩的方法,首先在全连接层损失函数上增加额外的归一项,使得权重趋向于二进制值,然后对输出层进行粗粒度的量化,该方法在 MNIST 和 CIFAR10 两个公开数据集上也取得了很好的结果。Sau 等人[13]在 2016 年使用基于教师—学生框架来进行深度神经网络的模型压缩,不仅降低了储存复杂度,同时还有效降低了运行复杂度和训练复杂度。Kadetotad 等人[14]在 2016 年使用针对全连接的权重矩阵进行粗粒度的稀疏化技术,然后再利用低精度的定点格式来降低权重精度和大小,从而实现对深度神经网络的压缩,达到降低储存和降低计算量的目的。

深度神经网络压缩近期引起了强烈的关注,就目前所存在的压缩方式来看,可以大概分为四种方式[13]:参数共享方法(parameter sharing)、网络删减方法(network pruning)、暗知识方法(dark knowledge)和矩阵分解方法(matrix decomposition)。参数共享的主要思路就是多个参数共享一个值,其具体实现的方法也各不相同,Vanhoucke 和 Hwang 等人[15,16]通过定点(fixed-point)方法来降低参数精度,从而使得值相近的参数共享同一值;Chen 等人[17]提出一种基于哈希算法的方法,将参数映射到相应的哈希桶内,在同一个哈希桶内的参数则共享同一值;Gong 等人[18]通过使用 K-means 聚类算法将全部的参数进行聚类,每簇中参数共享中心值,在系统精度保持在 1% 的损失下,该方法压缩率达到 24 ×。网络删减是一种用来对深度网络进行网络广泛使用的方法。早期的研究中,网络删减用来降低网络复杂度,防止过拟合[19~21]。近年来,Han 等人[22,23]针对训练的模型结果,删减掉一定阈值下的网络连接来对网络进行压缩,然后再基于参数共享和 Huffman 编码进一步对网络压缩。在基于暗知识的深度神经网络压缩方面,Sau 等人[13]基于教师—学生学习框架对网络进行压缩,在 MNIST[24]数据集上进行测试,结果表明该方法同时降低了模型的储存和运行复杂度。基于矩阵分解理论,文献[25~27]都采用低秩分解来对神经网络不同层中的参数进行压缩;Denton 等人[28]在卷积神经网络上使用矩阵分解方法加速了卷积层的计算过程,同时也有效减少了全连接神经网络的参数,压缩了神经网络。

深度神经网络的压缩主要表现在储存、训练复杂度以及运行复杂度[13]三个方面。以上所介绍的四种方法中,各有利弊。参数共享方法、网络删减方法、暗知识方法和矩阵分解方法都可以有效降低模型的储存,但是在训练复杂度和运行复杂度上却没有重要影响;暗知识方法虽然在三个方面都有着比较好的表现,但是就其准确率来看,较其他方法变化较大。本文主要基于网络删减和参数共享针对深度神经网络模型的储存复杂度展开探索与研究。

2　网络删减压缩

网络删减是一种在深度网络压缩中广泛使用的方法,早期的研究中,网络删减用来降低网络复杂度,防止过拟合[19~21]。Han 等人[22,23]针对深度网络结构模型,通过网络删减对神经网络进行压缩,结果显示该方法可有效压缩网络存储。首先通过训练深度神经网络,从而获得理想精度下的各个参数权重;然后删减掉所有绝对值小于阈值权重的连接;最后重新训练删减后的神经网络。重新训练有两种不同的方式,一是采用稀疏训练方式,即训练时,只更新删减后非零权重,零参数不参与更新,保留零值,直到训练完成;二是采用全局训练方式,即训练时,将删减后的权重作为神经网络的初始参数,进行所有参数的迭代更新,然后再次对网络进行删减,重复该过程,直到训练完成。其中第二种方式简单有效,所以本文实验采用第二种方式。

删减后的权重是一个典型的稀疏矩阵,从而完成对网络权重的压缩。针对稀疏矩阵的存储,本文使用按行存储(compressed sparse row,CSR)和按列存储(compressed sparse column,CSC)格式。如果矩阵的列数较大,则用 CSR 格式;如果矩阵的行数较大,则用 CSC 格式。

CSR 是一种比较标准的稀疏矩阵存储格式,需要三类数据来进行表示:数值、列号以及行偏移。数值为矩阵中所有非零

数值;列号为数值对应的所在列号;行偏移表示某一行的第一个非零元素在数值里面的起始偏移位置,行偏移最后要补上总的非零值个数。CSC 是与 CSR 相对应的一种方式,同样需要三类数据来表示:数值、行号以及列偏移。数值为矩阵中所有非零数值;行号为数值对应的所在行号;列偏移表示某一列的第一个非零元素在数值里面的起始偏移位置,列偏移最后补上总的非零值个数。CSR 存储格式如图 2 所示,稀疏矩阵的行列存储格式需要储存 $2a+n+1$ 个数值,其中 $2a$ 为非零值个数,n 为行或者列数。

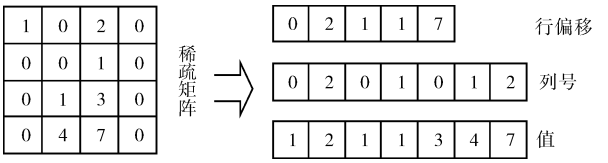

图 2 CSR 储存格式示意图

为了进一步压缩网络参数大小,将 64 位的数值直接压缩为 32 位数值。基于数值精度的压缩是一种十分有效的压缩方法,压缩率可计算,压缩效果明显。更多的基于数值精度的压缩方法可参见文献[15,16]。

网络删减方法最终将 LeNet-300-100 网络[26] 在损失了 0.82% 的精度下压缩了 4.9×,将修改得到的 LeNet-300-240-180-100 网络在损失了 1% 的精度下压缩了 4.7×。

3 参数共享压缩

参数共享的主要思路就是多个参数共享一个值,其具体实现参数共享的方法也各不相同。本文使用基于 K-means 聚类算法的参数共享压缩方法。首先,使用 K-means 聚类算法将删减后的参数进行聚类,获得每簇中心;然后每个参数根据所在的簇,更新该簇的中心值,即每簇内的多个参数共享一个值;最后构建整个参数所对应的码表,即聚类后参数的簇标签和各簇的中心值。

K-means 聚类算法是一种自下而上的聚类方法,其算法简单,速度快。将删减后的参数使用 K-means 方法对每层参数进行聚类,聚类的簇数为 k。因为每个簇的中心值都是代表该簇中各个连接的参考参数,所以将簇内的各个参数共享簇的中心值作为其权重值。

假设神经网络一层中参数集为 S,参数个数为 m,共聚为 n 簇,各簇中心为簇的中心值。a)读入参数集,假设聚类簇数为 k,随机取出 k 个参数值作为初始中心,其中 $1\leqslant k\leqslant m$;b)将参数聚入到距离最近的簇中;c)重新计算簇的中心向量;d)重复上述步骤,直到收敛,取每个簇的中心值作为簇内各个参数的权重值。

最终得到参数的簇标签和各簇的中心值,参数的簇标签最大值小于等于 k,所以参数的簇标签比参数值本身存储要小。由于中心值数量远小于参数权重数量,在 k 值远小于权重数量下,参数可以得到有效的压缩。在参数需要参与计算时,根据簇标签和中心值即可复原参数。

为进一步压缩,本实验将可压缩为 16 或 8 位整型和浮点型数值计算的部分值尽可能压缩,如在稀疏矩阵中的行号、列号和行偏移、列偏移,以及 K-means 聚类后的簇标签。该方法可有效地将模型参数进行压缩。

最终,LeNet-300-100 网络在损失了 1.32% 的精度下压缩了约 9.5×,LeNet-300-240-180-100 网络在损失了 1% 的精度下压缩了 12.1×。进行参数共享压缩后的模型精度与之前模型精度相比较,几乎没有再次损失。

4 实验

神经网络压缩实验对象为 LeNet-300-100 网络和修改得到的 LeNet-300-240-180-100 网络,LeNet-300-240-180-100 网络是在 LeNet-300-100 网络上增加了 240 个隐含单元和 180 个隐含单元的两层隐含层,两个网络均为完全连接网络。实验内容为手写数字识别,采用的数据是由纽约大学的 LeCun 等人维护的 MNIST 手写数字数据库,其中包含 60 000 个训练样本及10 000 个测试样本。

a)训练 LeNet-300-100 和 LeNet-300-240-180-100 网络。LeNet-300-100 网络经过 78 万次迭代后,错误率为 1.78%,网络的参数大小为2.04 Mb;LeNet-300-240-180-100 网络经过 38 万次迭代后,错误率为 1.87%,网络的参数大小为 2.83 Mb。

b)针对以上两个模型进行网络删减压缩实验。在精度损失 1% 的条件下,LeNet-300-100 网络在第一次阈值为 0.099 下删减后,经过再次训练,然后进行第二次删减,其阈值为 0.003,为进一步压缩,将 64 bit 数值压缩为 32 bit 数值,压缩后的大小为 424 Kb,最后的错误率为 2.60%,压缩了 4.9×;LeNet-300-240-180-100 网络在第一次阈值为 0.043 下删减后,经过再次训练,然后进行第二次删减,其阈值为 0.009,同样将 64 bit 数值压缩为 32 bit 数值,压缩后的大小为 623 Kb,最后的错误率为 2.87%,压缩了 4.7×。

c)将网络删减后的两模型进行参数共享压缩。通过 K-means 将参数进行聚类后生成参数的码表,其码表的存储与参数本身比较要小。LeNet-300-100 网络在参数共享压缩中,通过实验其聚类簇数选择为 128,由于 K-means 聚类时其聚类中心不稳定,将导致模型的准确率不稳定。为此实验随机进行五次,取其结果均值作为参考结果,实验错误率为 3.10%,误差为 1.32%,压缩后大小为 365 Kb。为进一步压缩,将可压缩的 32 bit 的数值直接压缩为 16 或 8 bit 数值,最后的大小为 220 Kb,最终将参数压缩了 9.5×。LeNet-300-240-180-100 网络在参数共享压缩中,聚类簇数选择为 64,实验随机进行五次后,错误率为 2.87%,误差为 1%,压缩后大小为 625 Kb。为进一步压缩,同样将可压缩的 32 bit 的数值直接压缩为 16 或 8 bit 数值,最后的大小为 239 Kb,最终将参数压缩了 12.1×。实验结果如表 1 所示。根据实验结果可以得出,网络删减作为神经网络进行整个压缩的第一步,起到了很大的压缩作用。

表 1 压缩结果

模型及参数		LeNet-300-100	LeNet-300-240-180-100
未压缩	错误率	1.78%	1.87%
	大小	2.04 Mb	2.83 Mb
网络删减压缩后	错误率	2.60%	2.87%
	大小	424 Kb	623 Kb
	压缩比	4.9×	4.7×
参数共享压缩后	错误率	3.1%	2.87%
	大小	220 Kb	239 Kb
	压缩比	9.5×	12.1×

5 讨论

如图 3 所示,展示了 LeNet-300-100 和 LeNet-300-240-180-100 网络在进行第二次网络删减时,在不同删减阈值下的精度损失。为保障精度损失在 1% 内,LeNet-300-100 网络最大可选择 0.003 作为网络删减阈值,同理,LeNet-300-240-180-100 网络最大可选择 0.009 作为网络删减阈值,与此同时可以采用 0.007 作为最佳删减阈值。除此之外,由图 3 可以看出,LeNet-300-100 网络随着删减程度的增加,其精度损失也陡然增加,但是隐含层较多的 LeNet-300-240-180-100 网络在同样条件下的精度损失变化却显得比较稳定。随着神经网络结构越深,其拥有越多的隐含神经元,获取的隐含信息越丰富,承载的信息量也就越多,当其结构被破坏时,准确率的稳定性说明网络的鲁棒性较强。

图 4 展示了两个网络结构在不同删减阈值下的存储大小。

理论上,当进行删减时,阈值越大,删减的网络连接越多,参数的稀疏矩阵中零值也就越多,其存储也就越小。由实验所得数据可以看到,两个网络结构的存储都随着删减阈值的增加而减小,与理论结论相同。此外,两个网络在 0.001 ~ 0.003 的网络删减中,存储减小比较明显,可以充分说明在神经网络中,其权重较小的连接即不重要的连接比较多,反之其权重较大的连接即重要的连接比较少。该现象与大自然很多现象雷同,神经网络如树一样,有着很多枝蔓,而树的主干却很少,修剪掉枝蔓对树的影响并不大,但是主干的损坏对树的功能和生命却至关重要。神经网络本就是模拟人的神经网络,与大自然有着巧妙的关联,还需要进一步探讨以揭示神经网络的本质。

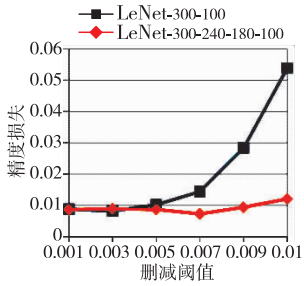

图 3　网络删减对精度损失的影响

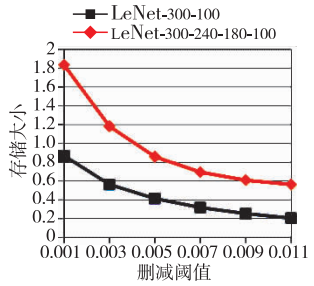

图 4　网络删减对存储大小的影响

图 5 展示了两个网络结构在不同聚类簇数的精度损失。由于 K-means 聚类使用随机中心点算法,所以中心会更新变化,为保证实验结果合理,取五次实验的结果均值进行实验,与此同时,取实验的结果方差作为网络稳定性的衡量标准。LeNet-300-100 网络选择 128 作为聚类簇数,一方面由图 5 可以看出随后的情况其精度损失将显著增加,另一方面是其结果方差较小,最后的结果方差也最小,网络较稳定。LeNet-300-240-180-100 网络则选择 64 作为聚类簇数,由于该条件下精度损失最小,最后的结果方差也最小。同图 3 所得结论类似,随着聚类簇数的减少,越深的神经网络结构,其鲁棒性较强,精度稳定性越强。此外,神经网络中有着较大值的重要连接其权重也相近,如自然界的大树,主干都有着相近的粗细程度。

图 6 展示了两个网络结构在不同聚类簇数下的存储大小。

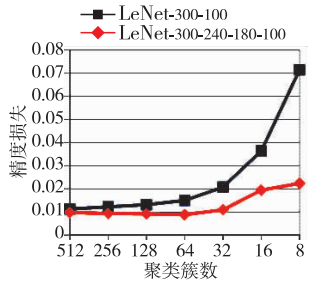

图 5　聚类簇数对精度损失的影响

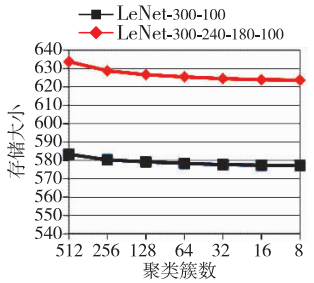

图 6　聚类簇数对存储大小的影响

理论上,当进行参数共享压缩时,当聚类簇数越小,其中心越少,其存储也就越小。根据图 6 所示实验所得数据可以得到,两个网络结构的存储都随着聚类簇数的减小而减小,与理论结论相同。但是与网络删减压缩相比,本实验中基于 K-means 聚类的参数共享实现的压缩程度比网络删减压缩的要弱。

6　结束语

通过网络删减压缩,删减神经网络中权重小于一定阈值的连接,将参数转换为稀疏矩阵进行存储,将 64 bit 数值降低为 32 bit 数值,然后结合参数共享方法,使用 K-means 聚类算法将参数进行聚类,然后在同一簇的参数共享簇中心值;另外尽可能地将 32 bit 数值降低为 16 或 8 bit,最终使得 LeNet-300-100 网络以及基于前者修改得到的 LeNet-300-240-180-100 网络在 MNIST 手写数字数据库中分别在精度损失 1.32% 和 1% 的情况下压缩 9.5 × 和 12.1 × 。

深度神经网络的压缩使得模型的存储缩小,可以有效移植到嵌入式设备和移动设备上,为基于深度神经网络的智能应用在特殊条件下的利用提供了可能性,降低了其硬件需求,有效扩展了其应用范围,为未来在特殊环境下更丰富的基于深度神经网络的智能应用提供了有效可行的解决方案。基于网络删减结合参数共享的压缩方案可有效压缩深度神经网络模型,希望能将较大的深度神经网络应用于嵌入式设备和移动设备,本文虽然在较浅层的网络上证明压缩方案可以压缩 10 × 左右,由于实验条件受限并没有在较深层的神经网络如 AlexNet 和 VGG-16 上进行相应实验来测试其压缩率;除此之外,并没有在典型的嵌入式设备如 ARM 板和移动设备如 Android 手机中进行测试,其运行效率及能耗问题有待进一步检测。本文主要针对储存复杂度问题进行了探索和研究,希望将来在较深层的神经网络上及计算力、电力上都能进行有效测试,观察压缩后的深度神经网络综合表现。除此之外,希望通过神经网络的压缩能更进一步地帮助理解神经网络连接的本质。

参考文献:

[1] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C]//Proc of the 25th International Conference on in Neural Information Processing Systems. [S. l.]:Curran Associates, 2012: 1097-1105.

[2] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [J/OL]. (2015-04-10). http://org/abs/.1409.1556.

[3] Szegedy C, Liu Wei, Jia Yangqing, *et al*. Going deeper with convolutions [C]//Computer Vision and Pattern Recognition. 2015: 1-9.

[4] He Kaiming, Zhang Xiangyu, Ren Shaoqing, *et al*. Deep residual learning for image recognition [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2016: 770-778.

[5] Dong Minghui, Yang Chenyu, Lu Yanfeng, *et al*. Mapping frames with DNN-HMM recognizer for non-parallel voice conversion [C]// Proc of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference. Picataway, NJ: IEEE Press, 2015: 488-494.

[6] Dat T H, Dennis J, Ren Lengyi, *et al*. A comparative study of multi-channel processing methods for noisy automatic speech recognition in urban environments [C]//Proc of IEEE International Conference on Acoustics, Speech and Signal Processing. Picataway, NJ: IEEE Press, 2016: 6465-6469.

[7] Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning [C]//Proc of the 25th International Conference on Machine Learning. New York: ACM Press, 2008: 160-167.

[8] Collobert R, Weston J, Bottou L, *et al*. Natural language processing (almost) from scratch [J]. Journal of Machine Learning Research, 2011, 12(11): 2493-2537.

[9] LéCun Y, Bottou L, Bengio Y, *et al*. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[10] Denil M, Shakibi B, Dinh L, *et al*. Predicting parameters in deep learning [C]//Proc of the 26th International Conference on Neural Information Processing Systems. [S. l.]:Curran Associates, 2013: 2148-2156.

[11] Kim Y D, Park E, Yoo S, *et al*. Compression of deep convolutional neural networks for fast and low power mobile applications [J/OL]. (2016-02-24). http://arxiv.org/abs/1511.06530.

[12] Soulie G, Gripon V, Robert M, *et al*. Compression of deep neural networks on the fly [C]//Proc of International Conference on Artificial Neural Networks. Cham:Springer, 2015: 153-160.

[13] Sau B B, Balasubramanian V N. Deep model compression: distilling knowledge from noisy teachers [J/OL]. (2016-10-30). https://arxiv.org/pdf/1610.09650v1.pdf.

(下转第 2903 页)

Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2010: 384-394.

[6] Pang Bo, Lee L, Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques [C]//Proc of ACL Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2002: 79-86.

[7] Mikolov T, Karafiát M, Burget L, *et al*. Recurrent neural network based language model [C]//Proc of the 11th Annual Conference of the International Speech Communication Association. 2010: 1045-1048.

[8] Son L H, Allauzen A, Yvon F. Measuring the influence of long range dependencies with neural network language models [C]//Proc of the NAACL-HLT Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT. Stroudsburg, PA: Association for Computational Linguistics, 2012: 1-10.

[9] Oualil Y, Singh M, Greenberg C, *et al*. Long-short range context neural networks for language modeling [C]//Proc of Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2016: 1473-1481.

[10] 陈龙, 管子玉, 何金红,等. 情感分析研究进展[J]. 计算机研究与发展, 2017,54(6):1150-1170.

[11] Graves A, Fernández S, Schmidhuber J. Bidirectional LSTM networks for improved phoneme classification and recognition [C]//Proc of International Conference on Artificial Neural Networks. Berlin: Springer-Verlag, 2005: 753-753.

[12] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures [J]. Neural Networks, 2005, 18(5/6): 602-610.

[13] Zhou Peng, Shi Wei, Tian Jun, *et al*. Attention-based bidirectional long short-term memory networks for relation classification [C]//Proc of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2016: 207-212.

[14] Yang Zichao, Yang Diyi, Dyer C, *et al*. Hierarchical attention networks for document classification [C]//Proc of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016: 1480-1489.

[15] Vu N T, Adel H, Gupta P, *et al*. Combining recurrent and convolutional neural networks for relation classification [C]//Proc of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016:534-539.

[16] 何炎祥, 孙松涛, 牛菲菲,等. 用于微博情感分析的一种情感语义增强的深度学习模型[J]. 计算机学报, 2017, 40(4): 773-790.

[17] 蔡慧苹, 王丽丹, 段书凯. 基于 word embedding 和 CNN 的情感分类模型[J]. 计算机应用研究, 2016, 33(10): 2902-2905, 2909.

[18] 夏从零, 钱涛, 姬东鸿. 基于事件卷积特征的新闻文本分类[J]. 计算机应用研究, 2017, 34(4): 991-994.

[19] Zhou Peng, Qi Zhenyu, Zheng Suncong, *et al*. Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling [C]//Proc of the 26th International Conference on Computational Liguistics. 2016: 3485-3495.

[20] Lai Siwei, Xu Liheng, Liu Kang, *et al*. Recurrent convolutional neural networks for text classification [C]//Proc of National Conference of the American Association for Artificial Intelligence. Palo Alto, CA: AAAI Press, 2015: 2267-2273.

[21] Zeiler M D. ADADELTA: an adaptive learning rate method [J/OL]. (2012-12-22). http://arxiv.org/abs/1212.5701.

[22] Blunsom P, Grefenstette E, Kalchbrenner N. A convolutional neural network for modelling sentences [C]//Proc of the 52nd Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2014:655-665.

[23] Tai K S, Socher R, Manning C D. Improved semantic representations from tree-structured long short-term memory networks [C]//Proc of the 53rd Annual Meeting of the Association for Computational Linguistics & the 7th International Joint Conference on Natural Languages Processing. Stroudsburg, PA: Association for Computational Linguistics, 2015: 1556-1566.

(上接第 2897 页)

[14] Kadetotad D, Arunachalam S, Chakrabarti C, *et al*. Efficient memory compression in deep neural networks using coarse-grain sparsification for speech applications [C]//Proc of the 35th International Conference on Computer-Aided Design. New York: ACM Press, 2016: Article No 78.

[15] Vanhoucke V, Senior A, Mao M Z. Improving the speed of neural networks on CPUs [C]//Proc of Deep Learning & Unsupervised Feature Learning Workshop. 2011:1-8.

[16] Hwang K, Sung W. Fixed-point feedforward deep neural network design using weights +1, 0, and −1[C]//Proc of IEEE Workshop on Signal Processing Systems. Piscataway, NJ: IEEE Press, 2014: 1-6.

[17] Chen Wenlin, Wilson J T, Tyree S, *et al*. Compressing neural networks with the hashing trick [C]//Proc of the 32nd International Conference on Machine Learning. 2015: 2285-2294.

[18] Gong Yunchao, Liu Liu, Yang Ming, *et al*. Compressing deep convolutional networks using vector quantization [J/OL]. (2014-12-18). https://arxiv.org/abs/1412.6115.

[19] Hanson S J, Pratt L Y. Comparing biases for minimal network construction with back-propagation [C]//Proc of the 1st International Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press, 1989: 177-185.

[20] LeCun Y, Denker J S, Solla S A. Optimal brain damage [C]//Advances in Neural Information Processing Systems. San Francisco: Morgan Kaufmann Publishers Inc, 1990: 598-605.

[21] Hassibi B, Stork D G. Second order derivatives for network pruning: optimal brain surgeon [C]//Advances in Neural Information Processing Systems. San Francisco: Morgan Kaufmann Publishers Inc, 1992: 164-171.

[22] Han Song, Pool J, Tran J, *et al*. Learning both weights and connections for efficient neural networks [C]//Proc of the 28th International Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press, 2015: 1135-1143.

[23] Han Song, Mao Huizi, Dally W J. Deep compression: compressing deep neural networks with pruning, trained quantization and Huffman coding [J/OL]. (2016-02-15). https://arxiv.org/abs/1510.00149.

[24] LéCun Y, Bottou L, Bengio Y, *et al*. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[25] Sainath T N, Kingsbury B, Sindhwani V, *et al*. Low-rank matrix factorization for deep neural network training with high-dimensional output targets [C]//Proc of IEEE International Conference on Acoustics, Speech and Signal Processing. Picataway, NJ: IEEE Press, 2013: 6655-6659.

[26] Denil M, Shakibi B, Dinh L, *et al*. Predicting parameters in deep learning [J/OL]. (2014-10-27). https://arxiv.org/abs/1306.0543.

[27] Nakkiran P, Alvarez R, Prabhavalkar R, *et al*. Compressing deep neural networks using a rank-constrained topology [C]//Proc of the 16th Annual Conference of the International Speech Communication Association. 2015: 1473-1477.

[28] Denton E, Zaremba W, Bruna J, *et al*. Exploiting linear structure within convolutional networks for efficient evaluation [C]//Proc of the 27th International Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press, 2014: 1269-1277.