斯坦福博士韩松毕业论文：面向深度学习的高效方法与硬件

机器之心
百家号 | 10-17 14:01

选自Stanford

机器之心编译

参与：路雪、蒋思源

韩松，2017 年斯坦福大学电子工程系博士毕业，师从 NVIDIA 首席科学家 Bill Dally 教授。他的研究也广泛涉足深度学习和计算机体系结构，他提出的 Deep Compression 模型压缩技术曾获得 ICLR'16 最佳论文，ESE 稀疏神经网络推理引擎获得 FPGA'17 最佳论文，对业界影响深远。他的研究成果在 NVIDIA、Google、Facebook 得到广泛应用，博士期间创立了深鉴科技，2018 年将任职 MIT 助理教授。本文对韩松博士的毕业论文做了介绍。

第一章 引言

本文，我们协同设计了适合深度学习的算法和硬件，使之运行更快更节能。我们提出的技术能够使深度学习的工作负载更加高效、紧密，然后我们设计了适合优化 DNN 工作负载的硬件架构。图 1.1 展示了本文的设计方法。打破算法和硬件栈之间的界限创造了更大的设计空间（design space），研究者获得之前从未有过的自由度，这使得深度学习能够实现更好的优化。

机器之心
百家号 | 最近更新：10-17 14:01

简介: 专业的人工智能媒体和产业服务平台

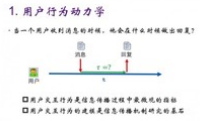
作者最新文章

业界| 吴恩达deeplearning.ai实习生招募开放：仅要求Coursera课程证书

业界| 要「自我颠覆」的Mate 10发布后，华为积极跟进3D人脸识别

报名| 北极光Lighting 智能+汽车的发展与投资机会

相关文章



更新了朋友圈&微博动态，好友何时会点赞评…
凤凰科技 10-17

那些被人悄无声息操纵了的数据

49 倍 [25,26]。我们还发现模型压缩算法能够去除冗余、防止过拟合，可以作为合适的正则化方法 [27]。

在硬件方面，压缩后的模型具备提速和降低能耗的极大潜力，因为它所需的算力和内存减少。然而，模型压缩算法使计算模式变的非常规，很难并行化。因此，我们为压缩后的模型设计了一种定制化硬件，设计模型压缩的数据结构和控制流程。该硬件加速器的能量效率比 GPU 高出 3400 倍，比之前的加速器高出一个数量级 [28]。该架构的原型在 FPGA 上，且已用于加速语音识别系统 [29]。

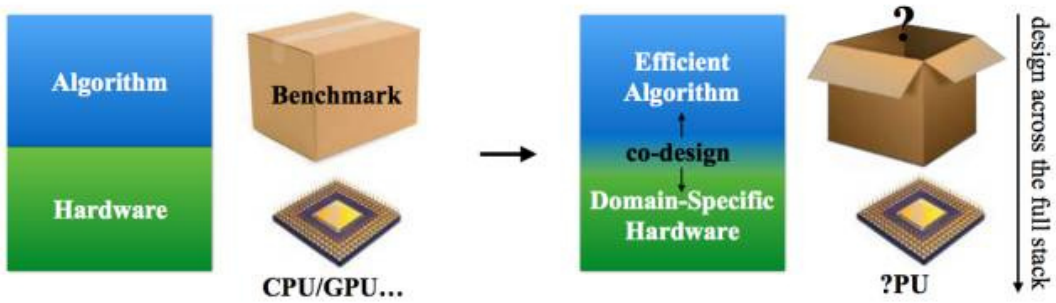


图 1.1：本文重点是协同设计适合深度学习的算法和硬件。本文回答了两个问题：哪些方法可以使深度学习算法更加高效，哪些硬件架构最适合这类算法。

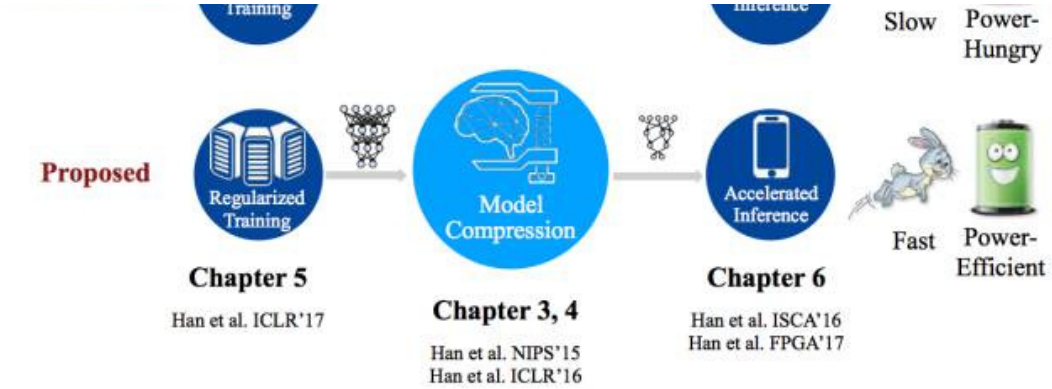


图 1.2：论文主题贡献：正则化训练、模型压缩、加速推理

第二章 背景

本章首先介绍什么是深度学习以及它的工作原理和应用；然后介绍我们实验所用的神经网络架构、数据集、在数据集上训练架构的框架。之后，我们介绍压缩、正则化和加速方面之前的研究。

第三章 神经网络剪枝

现代深度神经网络使用非常多的参数以提供足够强大的模型，因此这种方法在计算量和内存上都需要足够的资源。此外，传统的神经网络需要在训练前确定与修正架构，因此训练过程并不会提升架构的性能。而若直接选择复杂的架构，那么过多的参数又会产生过拟合问题。因此，选择适当容量（capacity）的模型和消除冗余对计算效率和准确度的提升至关重要。

为了解决这些问题，我们发展了一种剪枝方法（pruning method）来移除冗余并保证神经网络连接的有效性，这种方法能降低计算量和内存的要求以提

我们的剪枝方法移除了冗余连接，并仅通过重要的连接学习（下图 3.1）。在该图的案例中，共有三层神经网络，剪枝前第 i 层和 i+1 层间的连接为密集型连接，剪枝后第 i 层和 i+1 层间的连接为稀疏连接。当所有与神经元相联结的突触都被移除掉，那么该神经元也将移除。神经网络剪枝将密集型神经网络转化为稀疏型神经网络，并且在减少了参数与计算量的情况下完全保留预测准确度。剪枝不仅提高了推断速度，同时还降低了运行大型网络所需要的计算资源与能源，因此它可以在电池受限的移动设备上使用。剪枝同样有利于将深度神经网络储存并传递到移动应用中。

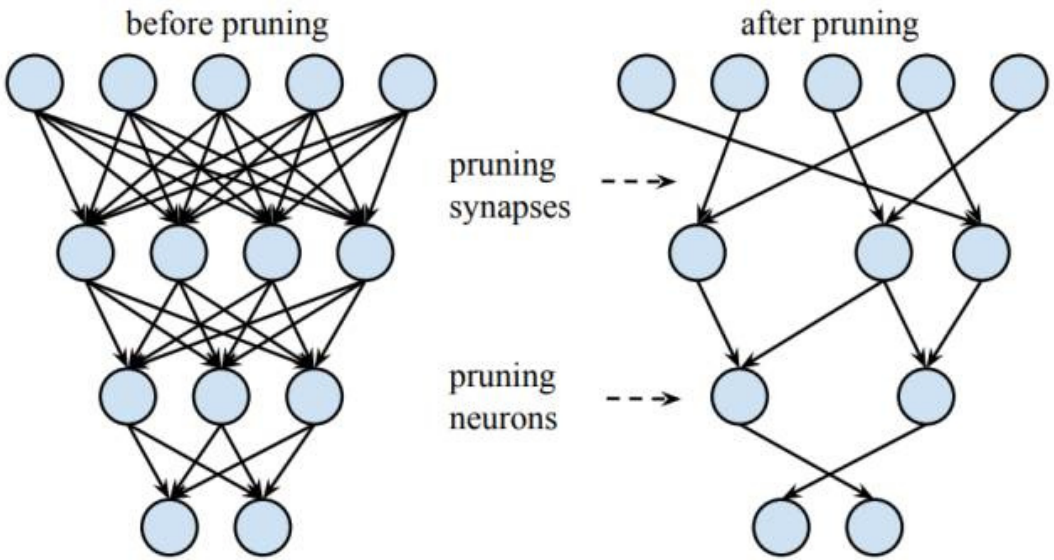


图 3.1：对深度神经网络的神经元与突触进行剪枝。

在初始化训练阶段后，我们通过移除权重低于阈值的连接而实现 DNN 模型的剪枝，这种剪枝将密集层转化为稀疏层。第一阶段需要学习神经网络的拓

地进行以减少神经网络复杂度。实际上，这种训练过程除了可以学习神经网络的权重外，还可以学习神经元间的连通性。这与人类大脑的发育过程 [109] [110] 十分相似，因为生命前几个月所形成的多余突触会被「剪枝」掉，神经元会移除不重要的连接而保留功能上重要的连接。

在 ImageNet 数据集中，剪枝方法可以将 AlexNet 的参数数量减少 9 倍（6100 万降低为 670 万）而没有准确度上的损失。VGG-16 同样有类似的现象，参数总量可以减少 13 倍左右（1.38 亿降低为 1.03 千万）而完全没有准确度损失。我们还试验了更多高效的全卷积神经网络：GoogleNet（Inception-V1）、SqueezeNet 和 ResNet-50，它们不具有或有很少的全连接层。在这些实验中，我们发现在准确度降低前它们有相似的剪枝率，即 70% 左右的全卷积神经网络参数可以被剪枝。GoogleNet 从 700 万参数降低到 200 万参数，SqueezeNet 从 120 万参数降低到 38 万参数，而 ResNet-50 从 2550 万参数降低到 747 万参数，这些网络在 ImageNet Top-1 和 Top-5 准确度上都完全没有损失。

在本章节以下部分中，我们提供了如何剪枝神经网络和再训练模型以保留预测准确度的方法。我们还展示了剪枝后模型在商业化硬件上运行所产生的速度与能源效率提升。

Network	Top-1 Error	Top-5 Error	Parameters	Pruning Rate
LeNet-300-100	1.64%	-	267K	
LeNet-300-100 Pruned	1.59%	-	**22K**	**12×**
LeNet-5	0.80%	-	431K	
LeNet-5 Pruned	0.77%	-	**36K**	**12×**
AlexNet	42.78%	19.73%	61M	
AlexNet Pruned	42.77%	19.67%	**6.7M**	**9×**
VGG-16	31.50%	11.32%	138M	
VGG-16 Pruned	31.34%	10.88%	**10.3M**	**13×**
GoogleNet	31.14%	10.96%	7.0M	
GoogleNet Pruned	31.04%	10.88%	**2.0M**	**3.5×**
SqueezeNet	42.56%	19.52%	1.2M	
SqueezeNet Pruned	42.26%	19.34%	**0.38M**	**3.2×**
ResNet-50	23.85%	7.13%	25.5M	
ResNet-50 Pruned	23.65%	6.85%	**7.47M**	**3.4×**

第四章 量化训练与深度压缩

本章节介绍了用于压缩深度神经网络的量化训练（trained quantization）技术，但它与前一章所介绍的剪枝技术相结合时，我们就能构建「深度压缩」[26]，即一种深度神经网络的模型压缩流程。深度压缩（Deep Compression）由剪枝、量化训练和可变长度编码（variable-length coding）组成，它可以压缩深度神经网络数个量级而没有什么预测准确度损失。这种大型压缩能使机器学习在移动设备上运行。

「深度压缩」是一种三阶段流程（图 4.1），它可以在保留原始准确度的情况下减小深度神经网络的模型大小。首先我们可以移除冗余连接而剪枝网络，这一过程只需要保留提供最多信息的连接（如第三章所述）。下一步需要量化权重，并令多个连接共享相同的权重。因此只有 codebook（有效权重）和索引需要储存，且每个参数只需要较少的位就能表示。最后，我们可

我们最重要的观点是，剪枝与量化训练可以在不相互影响的情况下压缩神经网络，因此可以产生惊人的高压缩率。深度压缩令存储需求变得很小（兆字节空间），所有的权重都可以在芯片上缓存而不需要芯片外的 DRAM。而动态随机储存器不仅慢同时能耗还比较高，因此深度压缩可以令模型更加高效。深度压缩是第六章高效推断机（efficient inference engine/EIE）的基础，其通过压缩模型实现了显著的速度和能源效率提升。

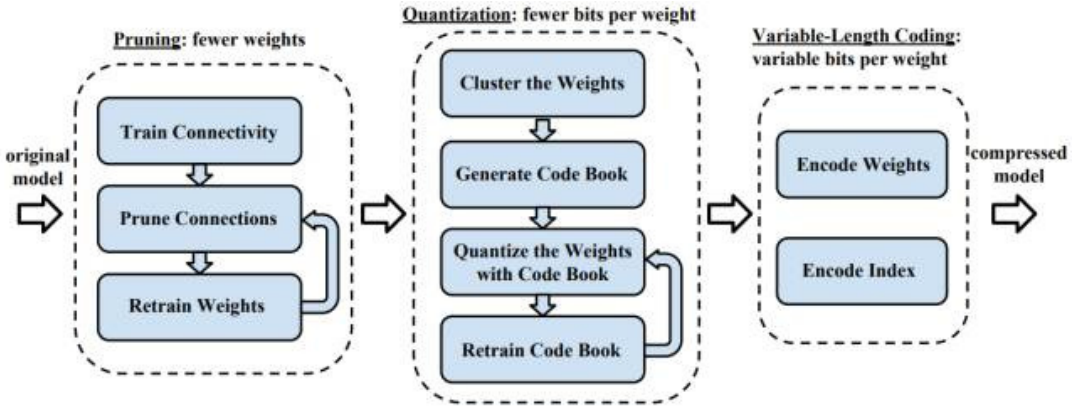


图 4.1：深度压缩的流程：剪枝、量化学习和可变长度编码

Network	Top-1 Error	Top-5 Error	Model Size	Compress Rate
LeNet-300-100	1.64%	-	1070 KB	
LeNet-300-100 Compressed	1.58%	-	**27 KB**	**40×**
LeNet-5	0.80%	-	1720 KB	
LeNet-5 Compressed	0.74%	-	**44 KB**	**39×**
AlexNet	42.78%	19.73%	240 MB	
AlexNet Compressed	42.78%	19.70%	**6.9 MB**	**35×**
VGG-16	31.50%	11.32%	552 MB	
VGG-16 Compressed	31.17%	10.91%	**11.3 MB**	**49×**
Inception-V3	22.55%	6.44%	91 MB	
Inception-V3 Compressed	22.34%	6.33%	**4.2 MB**	**22×**
ResNet-50	23.85%	7.13%	97 MB	
ResNet-50 Compressed	23.85%	6.96%	**5.8 MB**	**17×**

表 4.1：深度压缩在没有准确度损失的情况下节约了 17 倍到 49 倍的参数存储需求。

Quantization Method	1bit	2bit	4bit	6bit	8bit
Uniform (Top-1)	-	59.33%	74.52%	75.49%	**76.15%**
Uniform (Top-5)	-	82.39%	91.97%	92.60%	**92.91%**
Non-uniform -c (Top-1)	24.08%	68.41%	**76.16%**	76.13%	**76.20%**
Non-uniform -c (Top-5)	48.57%	88.49%	92.85%	**92.91%**	**92.88%**
Non-uniform -c+l (Top-1)	24.71%	69.36%	**76.17%**	**76.21%**	**76.19%**
Non-uniform -c+l (Top-5)	49.84%	89.03%	**92.87%**	**92.89%**	**92.90%**

表 4.9：使用不同更新方法比较均匀量化和非均匀量化的结果。-c 仅更新形心（centroid），-c+1 同时更新形心和标签。ResNet-50 的基线准确度分别为 76.15% 和 92.87%。所有结果都经过再训练。

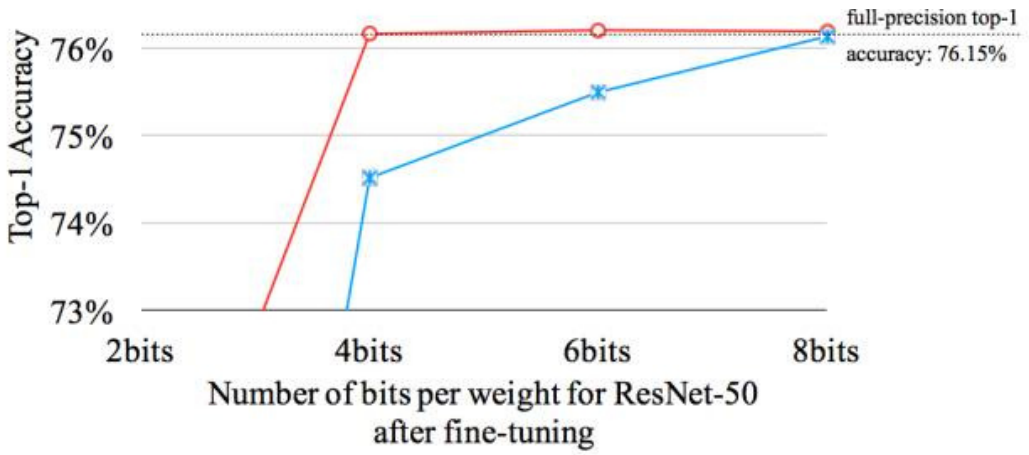


图 4.10：非均匀量化的表现要好于均匀量化。

图 4.10 和表 4.9 比较了均匀量化和非均匀量化的性能。非均匀量化指的是相邻编码的距离不为常数。量化训练是非均匀量化的一种形式，因为其不同编码的距离并不相同。对于非均匀量化（本研究），基线 ResNet-50 所有层级的参数可以压缩为 4 比特而没有准确度损失。然而对于均匀量化，基线 ResNet 所有层的参数只能压缩到 8 比特而没有准确度损失（压缩到 4 比特会产生 1.6% 的 Top-1 准确度损失）。非均匀量化可以很好的捕捉到权重的不均匀分布，而均匀量化不能很好的实现这一点。

第五章 DSD: Dense-Sparse-Dense Training

现代高性能硬件的出现使得训练复杂、模型容量巨大的 DNN 模型变得更加简单。复杂模型的优势是它们对数据的表达能力很强并且能捕捉到特征和输出之间的高度非线性的关系。而复杂模型的劣势在于，比起训练数据中所需要的模式，它们更容易捕捉到噪声。这些噪声并不会在测试数据中生成，从而使模型产生过拟合和高方差。

时优化。为了解决这个问题，我们提出了 dense-sparse-dense（DSD）训练流，以正则化深度神经网络，防止过拟合并达到更高的准确度。

传统的训练方法通常是同时训练所有的参数，而 DSD 训练法会周期性的修剪和恢复神经连接，训练过程中的有效连接数量是动态变化的。剪枝连接允许在低维空间中进行优化，捕捉到鲁棒性特征；恢复连接允许增大模型的容量。传统的训练方法只在训练开始的时候将所有权重初始化一次，而 DSD 训练法允许连接在周期性剪枝和恢复的中有多于一次的机会执行初始化。

DSD 的一个优势是最后的神经网络仍然拥有和初始的密集模型同样的架构和维度，因此 DSD 训练不会产生任何额外的推断成本。使用 DSD 模型进行推断不需要指定专门的硬件或专门的深度学习框架。实验证明 DSD 可以可以提高多种 CNN、RNN 和 LSTM 在图像分类、生成文字描述和语音识别任务的性能。在 ImageNet 上，DSD 提升了 GoogleNet Top-1 准确度 1.1%、VGG-16 Top-1 准确度 4.3%、ResNet-18 Top-1 准确度 1.2%、ResNet-50 Top-1 准确度 1.1%。在 WSJ』93 数据集上，DSD 把 DeepSpeech 和 DeepSpeech2 的错误率（WER）分别降低了 2.0% 和 1.1%。在 Flickr-8K 数据集上，DSD 将 NeuralTalk BLEU 的分数提高了 1.7 以上。

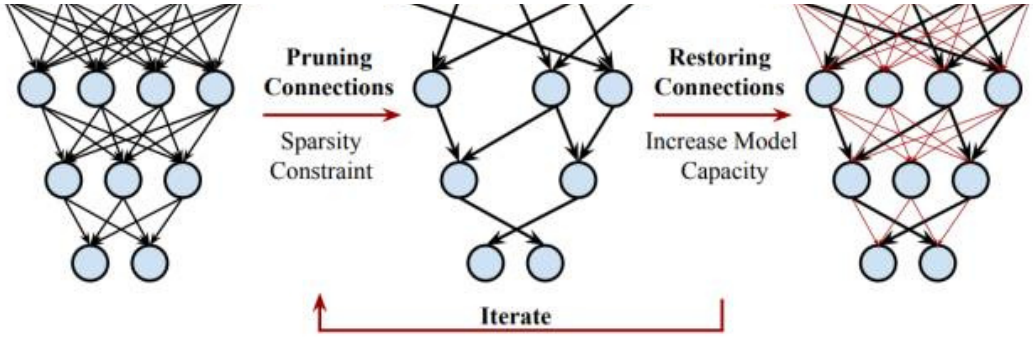


图 5：DSD（Dense-Sparse-Dense）训练法中迭代进行剪枝和恢复权重的过程。

第六章 EIE：用于稀疏神经网络的高效推断机

6.1 介绍

第三、四、五章介绍了三种提高深度学习效率的方法，本章着重介绍高效实现这些方法的硬件，「高效推断机」（EIE）[28]。该机器可以在稀疏的压缩模型上直接执行推断，节省内存带宽，实现大幅加速和能耗节约。

通过剪枝和量化训练 [25] [26] 实现的深度压缩能够大幅降低模型大小和读取深度神经网络参数的内存带宽。但是，在硬件中利用压缩的 DNN 模型是一项具有挑战性的任务。尽管压缩减少了运算的总数，但是它引起的计算不规则性对高效加速带来阻碍。例如，剪枝导致的权重稀疏使并行变的困难，也使优秀的密集型线性代数库无法正常实现。此外，稀疏性激活值依赖于上一层的计算输出，这只有在算法实施时才能知道。为了解决这些问题，实现在稀疏的压缩 DNN 模型上高效地运行，我们开发了一种专门的硬件加速器

EIE 是处理单元（processing element / PE）的一种可扩展数组（scalable array）。它通过在处理单元上交织（interleave）矩阵的行来分配稀疏矩阵并实现并行计算。每个处理单元在 SRAM 中存储一个网络分区，与子网络共同执行计算。EIE 利用了静态权重稀疏性、动态激活向量稀疏性、相对索引（relative indexing）、共享权重和极窄权重（4 比特/extremely narrow weights）。

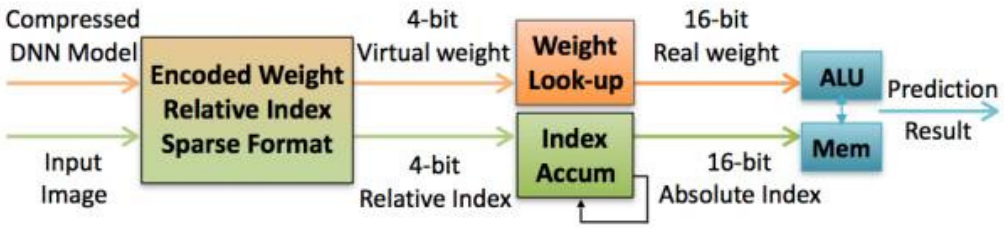


图 6.1：压缩 DNN 模型在 EIE 上运行。

EIE 架构如图 6.1 所示。EIE 以压缩稀疏列（compressed sparse column，CSC）格式存储权重不为零的稀疏权重矩阵 W。EIE 只在权重和激活值都不为零的情况下执行乘法。EIE 以游程编码（run-length encoded）格式存储每个权重的地址索引。在量化训练和共享权重之后，每个权重只占用 4 比特，它们可访问由 16 个寄存器实现的查找表以解码成 16 比特权重。

为评估 EIE 的性能，我们创建了行为级仿真和 RTL 模型，然后将 RTL 模型综合、布局布线，以提取准确的能量和时钟频率。将 EIE 在九个 DNN 基准上进行评估，它的速度分别是未压缩 DNN 的 CPU 和 GPU 实现的 189 和 13 倍。EIE 在稀疏网络上的处理能力为 102 GOPS/s，相当于在同等准确度

稀疏权重：EIE 是第一个用于稀疏和压缩深度神经网络的加速器。直接在稀疏压缩模型上运行可使神经网络的权重适应芯片上 SRAM，比访问外部 DRAM 节省 120 倍的能耗。通过跳过零权重，EIE 节省了 10 倍的计算周期。

稀疏激活值：EIE 利用激活函数的动态稀疏性来节约算力和内存。EIE 通过避免在 70% 的激活函数上的计算节约了 65.16% 的能量，这些激活函数在典型深度学习应用中的值为零。

权重编码：EIE 是第一个用非统一量化、极窄权重（每个权重 4 比特）利用查找表执行推断的加速器。与 32 比特浮点相比，它获取权重节约了 8 倍的内存占用，与 int-8 相比，它节约了 2 倍的内存占用。

并行化：EIE 引入了在多个处理单元上分配存储和算力的方法，以并行化稀疏层。EIE 还引入架构改变以达到负载平衡和优秀的扩展性。

第七章 结论

深度神经网络改变了大量 AI 应用，也正在改变我们的生活。但是，深度神经网络需要大量的计算量和内存。因此，它们很难部署到计算资源和能源预算有限的嵌入式系统中。为了解决该问题，我们提出了改善深度学习效率的方法和硬件。

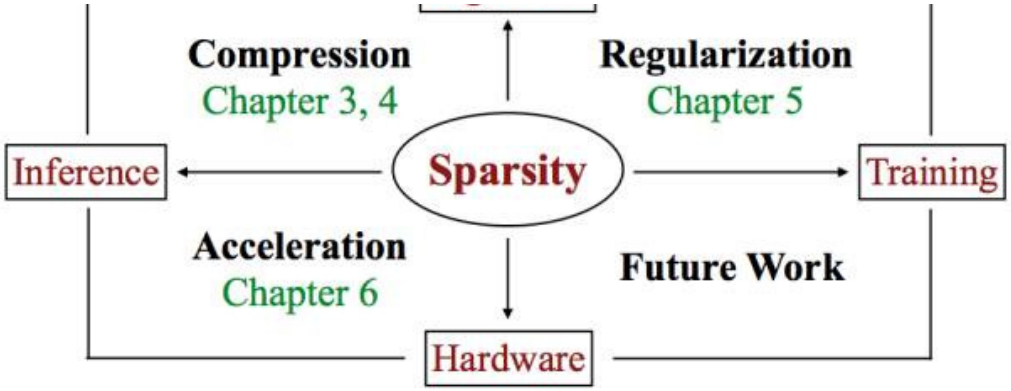


图 7.1：论文总结

本文从三方面研究如何提高深度学习的效率：利用深度压缩实现更小的模型大小、利用 DSD 正则化实现更高的预测准确度，以及利用 EIE 加速实现快速、能耗低的推断（图 7.1）。这三个方面遵循相同的原则：利用神经网络的稀疏性进行压缩、正则化和加速。

论文地址：https://stacks.stanford.edu/file/druid:qf934gh3708/EFFICIENT%20METHODS%20augmented.pdf

百度AI实战营·深圳站将于 10 月 19 日在深圳科兴科学园国际会议中心举行，AI 开发者与希望进入 AI 领域的技术从业者请点击「阅读原文」报名，与百度共同开创人工智能时代。

> 本文仅代表作者观点，不代表百度立场。系作者授权百家号发表，未经许可不得转载。