

**调研报告**

**二值化卷积神经网络算法**

**调研人：苗琦慧**

# 模型压缩方法

基于FPGAs的嵌入式设备通常只有几千个计算单元，远远不能处理常见深度模型中的数百万个浮点运算。复杂的模型与有限的计算资源之间存在着严重的矛盾。

乘法器是神经网络的数字实现最耗费空间和能量的组件，如果神经网络的模型太复杂，受内存空间和计算能力的限制，很难将模型应用到硬件设备上。

其实模型压缩最早可以追溯到1989年，Lecun的那篇Optimal Brain Damage（OBD）就提出可以将网络中不重要的参数剔除，达到压缩尺寸的作用，那时候连个深度网络都训练不出来，更没有现在这么发达的技术，Lecun就已经想好怎么做裁剪了，真是有先见之明，目前很多裁剪方案，都是基于OBD方法。

## 网络剪枝

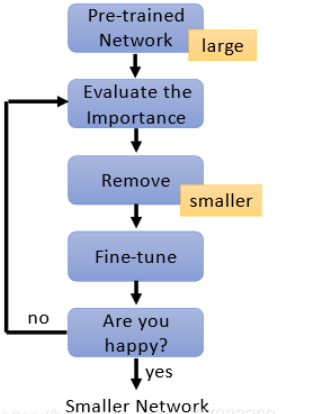
通常网络模型的参数都是过多的，比如有些weight接近0，或者neuron的输出为0，实际上用不到那么参数。所以就可以将这些多余的参数从网络中移除。

剪枝方式有两种，一种是Weight Pruning，一种是Neuron Pruning。对于Weight Pruning的实操是困难的，因为weight缺失之后会改变网络的架构，对于运算是不方便的，所以一般的操作是将剪枝的weight置为0。但是这种方式又似乎并没有使得参数减少，可操作性不大。

剪枝neuron会是比较好的方式，剪去neuron之后，其前后的weight都消失了，并不会改变计算的结构。因为参数是实际减少了，训练速度也会加快。

网络剪枝的具体步骤：

1. 预先训练一个比较庞大的模型
2. 评估每个weight和neuron的重要性：比如计算L1和L2范数来衡量Weight的数值大小，观察neuron的输出变化。
3. 按照参数重要性排序，Remove不重要的参数。不过这可能会导致准确率的下跌，所以一次不能移除太多，否则很难恢复。
4. 将缩小的模型用训练资料再Fine-tune（微调）一次，这样可以减小Remove带来的损失。
5. 如果缩小的模型没有达到要求，就重新从步骤2开始做一次。



## 参数量化

具体可分为三种方式：一是使用更少的bit的来存储一个参数；二是权值聚类，使用聚类中心来代替整个类的值，这样可以减少参数的储存；三是使用多的参数用比较少的bit来表示，使用少的参数用比较多的bit来表示，比如哈夫曼编码。

2015年，Han发表的[Deep Compression](https://arxiv.org/abs/1510.00149" \t "https://blog.csdn.net/wspba/article/details/_blank)是一篇对于模型压缩方法的综述型文章，将裁剪、权值共享和量化、编码等方式运用在模型压缩上，取得了非常好的效果，作为ICLR2016的best paper，也引起了模型压缩方法研究的热潮

参数修剪和量化主要是分别通过去除冗余/非关键参数或压缩参数空间（如从浮点权重到整数权重）来消除模型参数中的冗余。

## 知识蒸馏

基本思想是可以先训练一个规模大的Teacher network，再训练一个小的Student network去学习大的Teacher network的行为。

## 低秩参数分解

低秩因式分解应用矩阵/张量分解技术，使用小尺寸的代理参数来估计信息参数。

## 转移/紧凑卷积滤波器

基于紧凑卷积滤波器的方法依赖于精心设计的结构卷积滤波器来降低存储和计算复杂度。

# 二值量化

在现有的网络压缩技术中，基于量化的网络压缩技术是一种有前途的快速解决方案，通过以非常低的精度表示网络权重，与浮点压缩技术相比，它产生高度紧凑的模型。

最极端的量化是二值化，这是一种1位量化，其中数据只能有两个可能的值，即-1（0）或+1。另外，权重和激活都可以用1位表示，不需要占用太多内存。此外，通过二值化，重矩阵乘法运算可以被轻量级的按位XNOR运算和位计数运算所取代。因此，与其他压缩方法相比，二进制神经网络具有许多硬件友好的特性，包括节省内存、功率效率和显著的加速。

在一些研究中表明，二进制神经网络可以过滤输入噪声，并指出专门设计的bnn比全精度神经网络更鲁棒。在神经网络的前向传播和二值化过程中，噪声被连续放大，通过保持较小的噪声幅度来提高鲁棒性。

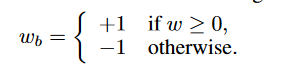
可以将二值化网络分为朴素二值化网络和基于优化的二值化网络。

## 朴素二值化网络

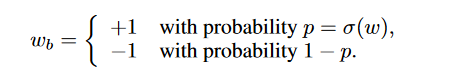
朴素二进制神经网络通过固定的二值化函数直接将神经网络中的权重和激活量化为1位。然后采用带有STE的基本后向传播策略，以标准训练方式优化深度模型。下文将依次介绍二值化函数、神经网络前馈、反馈、和参数更新以及STE技术。

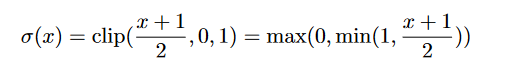
### 二值化函数

一个非常简单的二值化操作将基于sign函数，这种二值化方式称为确定性二值化：



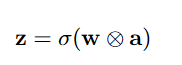
还有一种二值化方式是随机二值化，但这种方式比较占用内存，所以实际中一般采用确定二值化



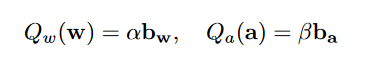


### 前馈、反馈及参数更新

在全精度卷积神经网络中，基本运算可以表示为



在前向传播过程中，二值权重和激活可以用下式得到



其中bw和ba是二元权重和二元激活的张量，具有相应的标量α和β。在文献中，符号函数广泛用于Qw和Qa 利用二值化的权重和激活，前向传播中的向量乘法可以重新表述为



通常二值化函数（如符号）是不可微的，但辛顿等人提出的直通估计量（STE）的技术可以解决训练由符号函数二值化的深度网络时出现的梯度问题。STE的功能定义如下:



通过STE，可以使用与普通全精度神经网络相同的梯度下降方法直接训练二进制神经网络。但是，当在反向传播中使用clip函数时，如果全精度激活的绝对值大于1，则不能在反向传播中更新它。因此，在实际情况下，也选择单位函数来逼近符号函数的导数。

在2016年，Courbariaux等人介绍了BinaryConnect，开创了二元神经网络的研究，这是一种在向前和向后传播期间用二进制权重训练DNN的方法，同时保持累积梯度的存储权重的精度。并且在MNIST、CIFAR-10和SVHN等任务中用BinaryConnect获得了接近最精确的结果，他们还进一步介绍了二值化神经网络（BNN），详细介绍了训练和加速技巧。从理论和实际两个方面证明了二元神经网络的实用性和加速能力。对于具有批量归一化的网络的推理加速，该方法还设计了基于移位的批量归一化和XNOR位计数等技术。

代码见XNOE\_Net\_EXAMPLE

## 基于优化的二值化网络

### 最小量化误差

优化二值化网络通常的做法是减少权重和激活的量化误差。这是一个类似于标准量化机制的简单解决方案，即量化参数应该尽可能接近全精度参数，期望二进制神经网络模型的性能将接近全精度参数。

作为早期考虑量化误差的研究，Rastegari等人提出的二元权重网络（BWN）和XNOR网络。BWN采用二进制权重和全精度激活的设置，而XNOR-Net将权重和激活都二进制化。

具体来说BWN和XNOR-Net中的权重量化过程可以表述为w≈αbw，其中α是二值化权重bw的浮点比例因子。这意味着BWN中的权重被二值化为{−α，+α}，但仍然可以带来快速计算的好处。则最小化量化误差有助于找到最佳比例因子和二进制参数。具体步骤可以参考论文研读笔记。

Mishra等人设计了宽降低精度网络（WRPN），该网络也以类似于XNOR-Net的方式最小化量化误差，但增加了每层中滤波器的数量。与直接对网络进行二值化相比，加宽和二值化相结合可以在精度和加速性之间取得很好的平衡。

为了进一步降低量化误差，高阶残差量化（HORQ）采用了基于量化残差的全精度激活的递归近似，而不是XNOR-Net中使用的一步近似。它通过每个递归步骤中近似的线性组合来产生最终的量子化激活。

以非常相似的方式，Lin设计了ABC-Net，线性组合多个二进制权重矩阵和比例因子来拟合全精度权重和激活，可以大大减少二值化造成的信息丢失。

Wang等人指出了以前方法的缺点，即分别最小化权重和激活的量化误差很难保证输出与全精度输出相似。为了解决这一问题，设计了一种两步量化（TSQ）方法。在第一步中，所有权重都是全精度值，并且所有激活都用可学习的量化函数Qa量化成低位格式。在第二步骤期间，Qa是固定的，并且如下学习低比特权重向量bw和比例因子α：



上述方法通常选择固定的二值化函数（如符号函数）。还可以采用更灵活的二值化函数，并在最小化量化误差的同时学习其参数。

了实现这个目标，Choi等人提出的参数化削波激活（PACT），具有激活函数的可学习上限。每一层的优化上界能够保证每一层的量化范围与原始分布一致。在实践中，PACT在二进制网络上表现更好，在更大的网络上可以达到接近全精度网络的精度。

在中，学习量化（LQ网）试图通过联合训练神经网络和网络中的量化器来最小化量化误差。与以前的工作不同，LQ网通过在网络训练过程中最小化量化误差来学习量化阈值和截止值，并且可以支持任意比特量化。

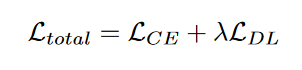
Bulat等人基于XNOR网络，将激活和权重比例因子融合成一个单一的因子，该因子通过向后传播有区别地学习，并提出了XNOR-Net++。

### 改进损失函数

一般的二值化方案只关注浮点值的精确局部逼近，而忽略了二值参数对全局损耗的影响。

Hou等提出的损失感知二值化(LAB)，其使用准牛顿算法直接最小化与二进制权重相关联的总损失。该方法利用来自由Adam优化器计算的二阶移动平均的信息，在考虑二值化特征的情况下找到最优权重。除了从量化的角度考虑任务相关的损失，设计额外的量化感知损失项目被证明是可行的。

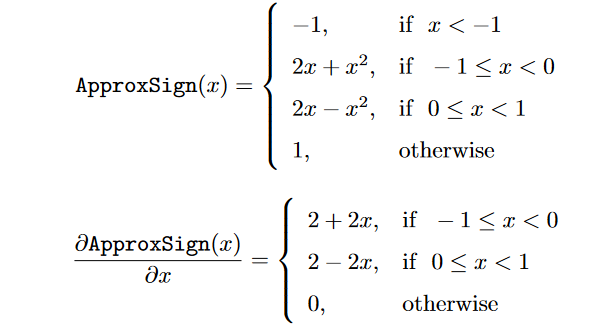
Ding等人总结了二元神经网络中由于前向二值化和后向传播所引起的退化、饱和和梯度失配等问题。为了解决这些问题，引入了分布损失，以明确地将激活分布正则化。是用于训练深度神经网络的公共交叉熵损失，是用于学习适当二值化的分布损失，λ平衡了两种类型损失的影响，损失函数如下：

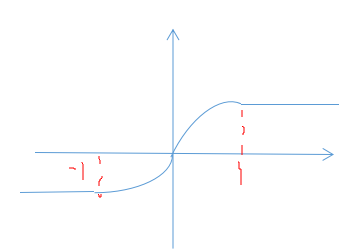


### 减小梯度误差

二值化函数的梯度（如符号）和STE之间存在明显的梯度失配。此外，它还存在[1，+1]范围之外的参数不会被更新的问题。这些问题容易导致二元网络的优化不足，导致网络性能严重下降。

Bi-Real提供了一个ApproxSign函数来替换反向传播梯度计算的符号，如下所示：





与传统的STE相比，ApproxSign具有与原始二值化函数符号相近的形状，从而可以在一定程度上控制梯度误差。

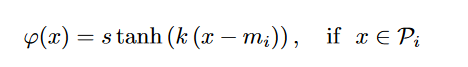
循环二进制卷积网络（CBCN）也应用了一个近似函数来解决符号函数的梯度失配问题。

BNN+直接提出了符号函数导数的改进近似，并引入了正则化函数，该函数鼓励围绕二进制值学习权重。

最近的一些方法试图在前向传播中追求良好的量化函数，这也可以减少梯度误差。

2017年，半件高斯量化（HWGQ）方法的提出，很好地解决了梯度失配问题，它对训练过程中前向传播中更常用的ReLU函数给出了低精度估计。

遵循同样的方法，在2019年，一种差分软量化（DSQ）方法被提出，用软量化函数代替传统的量化函数：



其中k确定渐近函数的形状，s是使软量化函数平滑的比例因子，mi是区间的中心。DSQ可以调整截止值和软量化函数的形状，以逐渐接近标准符号函数。

2019年ProxQuant一文指出，将量化网络训练公式化为正则化学习问题，并通过prox-gradient方法对其进行优化。ProxQuant在底层全精度向量上进行反向传播，并在随机梯度步长之间应用有效的prox算子。在此基础上，这两篇文章探讨了符号导数的平滑过渡，使用带参数v的Tanh函数和SoftSign函数来减小训练梯度误差。

Qin等人改进了BNNs，因为保留前向和后向信息对于准确的BNNs至关重要，所以提出了IR-Net，它是第一个同时考虑前向和后向信息保留的BNNs设计，包括一个自适应误差衰减估计器（EDE），以减少训练中的梯度误差，该估计器考虑了训练过程不同阶段的不同要求，并平衡了参数的更新能力和梯度误差的减少。

除了在向后或向前传播中修改二值化函数，【91】还通过混合粗梯度下降（BCGD）算法直接校准梯度。BCGD的权重更新采用全精度权重及其量化对应权重的加权平均值：



其中wt表示第t步上的全精度权重，表示btw的梯度，从而在目标值中产生足够的下降，从而加速训练。此文从理论角度进一步研究了量化神经网络的训练方法，并表明利用高精度表示的训练算法具有纯量化训练方法所缺乏的重要贪婪搜索阶段，这解释了使用低精度算法训练的困难。

# 训练技巧

牛津大学的研究人员指出，几乎所有的二值化研究都重新定位了池层的位置。池层总是紧接在卷积层之后使用，以避免二值化后最大池化导致的信息丢失。**实验表明，这种位置重排序方法在精度上有很大的提高。**除了池层外，批归一化层的位置也极大地影响了二进制神经网络训练的稳定性。在所有量化操作之前插入批归一化层以校正数据，可以使量化输入服从稳定分布（有时接近高斯分布），因此均值和方差保持在合理的范围内，训练过程变得更加平滑。

基于类似的想法，Bi-Real、WRPN、BENN等都通过结构转换来调整数据分布。

适当的超参数和特定的优化器也可以提高BNNs的性能。大多数现有的二进制神经网络模型选择自适应学习率优化器，如Adam。使用Adam可以使训练过程更好更快，其中二阶导数的平滑系数尤为关键。一文的分析表明，如果使用固定学习率优化器，例如随机梯度下降（SGD）算法，则需要采用大批量来提高性能。批量归一化动量系数的设置也很关键。在此文中，通过比较不同动量系数下的精度结果，发现批量归一化的参数需要适当设置，以适应二值化操作引起的抖动。

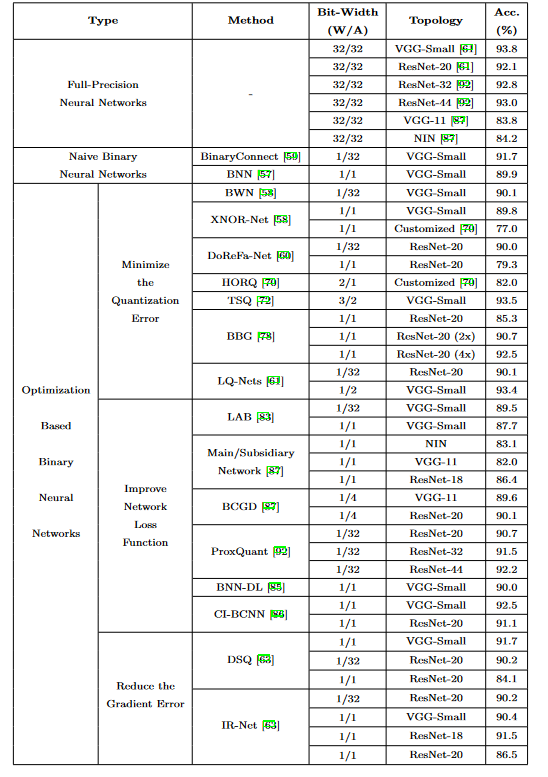
由于量化对训练有负面影响，许多研究中采用渐近量化策略，逐渐增加量化程度，以减少参数二值化带来的损失。实践表明，这种逐步量化方法有助于找到最优解。例如，INQ对参数进行分组，并逐渐增加参与量化的组的数量，以实现基于组的逐步量化。还可以先量化到较高的比特宽度，然后量化到较低的比特宽度。该策略可以避免极低位量化带来的大扰动，补偿训练过程中量化参数的梯度误差。

最后，使用更平滑的估计器成为一种常见的做法。由于直通估计量的存在，梯度误差通常存在于反向传播中，找到一个接近二值化函数的近似函数是简单实用的解决方案，这成为最近研究中广泛考虑的流行技术。根据不同的动机定制近似函数，将其形状与二值化函数的形状对齐，以取代导致梯度误差的标准剪辑函数。参考文献有：

# 模型性能对比

在BNNs中，权重和激活都是二进制的，理论上卷积运算速度提高了58倍，内存节省了32倍。

## 基于CIFAR-10数据集的二值神经网络图像分类性能研究



## 二值神经网络在ImageNet数据集上的图像分类性能

