深度学习（Deep Learning）

深度学习中 Batch Normalization为什么效果好?

jmlr.org/proceedings/pa ...

用mxnet 做了实验，用不用bn简直是两个世界，请问大侠们，为什么BN这么有效呢?

添加评论 分享

查看全部 7 个回答

魏秀参，铁打的人儿，流水的饭

right mr、Jimmy Lee、知乎用户 等人赞同

被@王语斌邀了好久。

这里分五部分简单解释一下Batch Normalization (BN)。

1. What is BN?

顾名思义，batch normalization嘛，就是“批规范化”咯。Google在ICML文中描述的非常清晰，即在每次SGD时，通过mini-batch来对相应的activation做规范化操作，使得结果（输出信号各个维度）的均值为0，方差为1. 而最后的“scale and shift”操作则是为了让因训练所需而“刻意”加入的BN能够有可能还原最初的输入（即当 $\gamma^{(k)} = \sqrt{Var[x^{(k)}]}, \beta^{(k)} = E[x^{(k)}]$ ），从而保证整个network的capacity。

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

关于DNN中的normalization，大家都知道白化（whitening），只是在模型训练过程中进行白化操作会带来过高的计算代价和运算时间。因此本文提出两种简化方式：1）直接对输入信号的每个维度做规范化（“normalize each scalar feature independently”）；2）在每个mini-batch中计算得到mini-batch mean和variance来替代整体训练集的mean和variance. 这便是Algorithm 1.

2. How to Batch Normalize?

怎样学BN的参数在此就不赘述了，就是经典的chain rule：

$$\frac{\partial \ell}{\partial \widehat{x}_i} = \frac{\partial \ell}{\partial y_i} \cdot \gamma$$
$$\frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot (x_i - \mu_{\mathcal{B}}) \cdot \frac{-1}{2}(\sigma_{\mathcal{B}}^2 + \epsilon)^{-3/2}$$
$$\frac{\partial \ell}{\partial \mu_{\mathcal{B}}} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}$$
$$\frac{\partial \ell}{\partial x_i} = \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{2(x_i - \mu_{\mathcal{B}})}{m} + \frac{\partial \ell}{\partial \mu_{\mathcal{B}}} \cdot \frac{1}{m}$$
$$\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i} \cdot \widehat{x}_i$$
$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i}$$

3. Where to use BN?

BN可以应用于网络中任意的activation set。文中还特别指出在CNN中，BN应作用在非线性映射前，即对 $x = Wub$ 做规范化。另外对CNN的“权值共享”策略，BN还有其对应的做法（详见文中3.2节）。

4. Why BN?

好了，现在才是重头戏——为什么要用BN？BN work的原因是什么？

说到底，BN的提出还是为了克服深度神经网络难以训练的弊病。其实BN背后的insight非常简单，只是在文章中被Google复杂化了。

首先来说说“Internal Covariate Shift”。文章的title除了BN这样一个关键词，还有一个便是“ICS”。大家都知道在统计机器学习中的一个经典假设是“源空间（source domain）和目标空间（target domain）的数据分布（distribution）是一致的”。如果不一致，那么就出现了新的机器学习问题，如，transfer learning/domain adaptation等。而covariate shift就是分布不一致假设之下的一个分支问题，它是指源空间和目标空间的条件概率是一致的，但是其边缘概率不同，即：对所有 $x \in \mathcal{X}, P_s(Y|X=x) = P_t(Y|X=x)$，但是 $P_s(X) \neq P_t(X)$. 大家细想便会发现，的确，对于神经网络的各层输出，由于它们经过了层内操作作用，其分布显然与各层对应的输入信号分布不同，而且差异会随着网络深度增大而增大，可是它们所能“指示”的样本标记（label）仍然是不变的，这便符合了covariate shift的定义。由于是对层间信号的分析，也即是“internal”的来由。

那么好，为什么前面我说Google将其复杂化了。其实如果严格按照解决covariate shift的路子来做的话，大概就是上“importance weight”（ref ）之类的机器学习方法。可是这里Google仅仅说“通过mini-batch来规范化某些层/所有层的输入，从而可以固定每层输入信号的均值与方差”就可以解决问题。如果covariate shift可以用这么简单的方法解决，那前人对其的研究也真真是白做了。此外，试想，均值方差一致的分布就是同样的分布吗？当然不是。显然，ICS只是这个问题的“包装纸”嘛，仅仅是一种high-level demonstration。

那BN到底是什么原理呢？说到底还是**为了防止“梯度弥散”**。关于梯度弥散，大家都知道一个简单的栗子：$0.9^{30} \approx 0.04$。在BN中，是通过将activation规范为均值和方差一致的手段使得原本会减小的activation的scale变大。可以说是一种更有效的local response normalization方法（见4.2.1节）。

5. When to use BN?

OK，说完BN的优势，自然可以知道什么时候用BN比较好。例如，在神经网络训练时遇到收敛速度很慢，或梯度爆炸等无法训练的状况时可以尝试BN来解决。另外，在一般使用情况下也可以加入BN来加快训练速度，提高模型精度。

诚然，在DL中还有许多除BN之外的“小trick”。**别看是“小trick”，实则是“大杀器”**，正所谓“*The devil is in the details*”。希望了解其它DL trick（特别是CNN）的各位请移步我之前总结的：**Must Know Tips/Tricks in Deep Neural Networks**

以上。

****

知乎用户

SGD里有类似操作么

9 天前

魏秀参（作者） 回复 知乎用户 查看对话

惊现水哥？话说什么叫sgd的类似操作哇

9 天前

西蒙尼

这个可以有

9 天前

写下你的评论…

更多回答

清雨影，车间的装配工一枚，很多事情都略(qiang)…

（刚才被人指出答题答跑题了，答成批量梯度下降了，233333，BN没研究过，下面的回答大家当段子看吧……）原答案：比起防止陷入局部极小这种事情，深度学习也好，也别高维的数据也好，更加要解决的往往是梯度太小：滚着滚着发现四周都是平地，停了~这个时候… 显示全部

Ying Zhang，深度瞎学习中。

作者在文章中强调在训练过程中网络每层输入的分布一直在改变(internal covariate), 会使训练过程难度加大，但可以通过normalize每层的输入解决这个问题。Lecun曾在98年提过whitening每个输入层的好处，但是whitening也会有很多弊端(详见paper第2页)，二是计… 显示全部

查看全部 7 个回答

关注问题 319 人关注该问题

关于作者

魏秀参 关注他

铁打的人儿，流水的饭

被收藏 40 次

人工智能
蔡世勋 创建 | 1061 人关注

Research
赵丛 创建 | 6 人关注

技术
鲁葳 创建 | 2 人关注

算法
王冲 创建 | 2 人关注

深度学习
知乎用户 创建 | 1 人关注

相关问题 换一换

如何根据每个策略的 daily return 对不同策略进行最为有效的分类? 8 个回答

深度学习目前主要有哪些研究方向? 5 个回答

为什么 Deep Learning 最先在语音识别和图像处理领域取得突破? 26 个回答

如何向非物理专业的同学解释重整化群?

6 个回答

Extreme learning machine (ELM) 到底怎么样，有没有做的前途？ 28 个回答

回答状态

最后编辑于 2016-02-10

所属问题被浏览 **1533** 次