关于GAN的灵魂七问

作者：Augustus Odena

机器之心编译

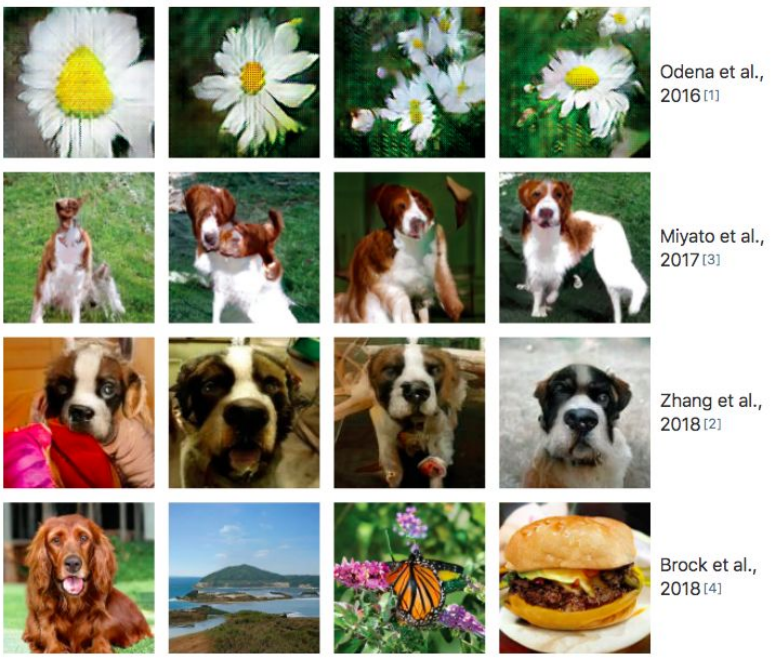
机器之心编辑部

关于生成对抗网络的七个开放性问题，个个都是灵魂追问。

生成对抗网络在过去一年仍是研究重点，我们不仅看到可以生成高分辨率（1024×1024）图像的模型，还可以看到那些以假乱真的生成图像。此外，我们还很兴奋能看到一些新的生成模型，它们能生成与 GAN 相媲美的图像，其主要代表就是流模型 Glow。

从 DeepMind 提出的 BigGAN，到英伟达的 Style-based Generator，它们生成的图像质量都令人惊叹。尽管还有很多问题没有解决，但图像生成已经能骗过一般人类了。不信的话，你可以试试区分生成的图像与真实图像。

看了上面 Style-based Generator 的生成效果，很明显感觉生成对抗网络在过去 2 年中已经取得了显著的进展。其实，从 16 年到 18 年图像合成的质量越来越高，看论文的速度都快赶不上 GAN 的发展了：



但是在另一些方面，GAN 的提升并不是那么显著。例如，关于如何评估 GAN 的效果，现在仍有很多分歧。因为目前图像合成基准已经非常多了，所以反思子领域的研究目标显得更有意义。

在这篇文章中，谷歌大脑团队的 Augustus Odena 就针对 GAN 的七大开放性问题作出了介绍。

- 问题 1：如何在 GAN 和其它生成模型之间进行挑选？
- 问题 2：GAN 能建模哪些分布？
- 问题 3：除了图像合成外，GAN 还能用于哪些地方？
- 问题 4：GAN 的全局收敛性如何？训练动态过程又是怎样的？
- 问题 5：我们该如何评估 GAN 的好坏，什么时候又该使用 GAN 这种生成模型？
- 问题 6：如何扩展训练 GAN 的批量大小？
- 问题 7：GAN 和对抗样本之间有什么关系？

Augustus 对每一个问题都做了很详细的讨论，包括问题背景、问题内容以及如何解决等等。这篇文章发布在 Distill 上，机器之心简要对六大问题做了介绍，更详细的内容与相关引用文献可阅读原文。

谷歌大脑和其他很多研究者都在致力于解决这些 GAN 的开放性研究问题。这篇文章也引用了近来非常多的生成对抗网络研究，因此并不能面面俱到地描述细节，所以读者有一定的基础、对这些问题有一定的直观了解就最好了。

如何在 GAN 和其它生成模型之间进行挑选

除了 GAN，另外两种生成模型现在也很流行：流模型和自回归模型。粗略来说，流模型将一堆可逆变换应用于先验样本，以计算观测值的精确对数似然性。另一方面，自回归模型将观测值的分布分解为条件分布，并一次处理观测值的一个组件（对于图像，可能是一次处理一个像素）。最近的研究表明，这些模型具有不同的性能特点和权衡。准确描述这些权衡并确定它们是否为模型的固有特性是一个有趣的开放性问题。

具体来说，我们先暂时把重点放在 GAN 和流模型之间计算成本的差异上，训练 GAN 和流模型的计算成本之间似乎存在巨大差异。GLOW 模型是用 40 个 GPU 花两周训练的，以生成 256x256 的名人面部图像，其使用的参数为 2 亿。相比之下，自回归 GAN 是在相似的面部数据集上用 8 个 GPU 花 4 天训练的，以生成 1024x1024 的图像，它使用了 4600 万参数。流模型大概需要 17 倍多的 GPU 天数和 4 倍多的参数来生成像素少 16 倍的图像。

为什么流模型效率更低？有两个可能的原因：首先，最大似然训练可能比对抗训练的计算难度更大。其次，归一化流可能无法有效代表特定函数。论文《Parallel WaveNet: Fast High-Fidelity Speech Synthesis》第 6.1 节对表达性做了一些小实验，但目前我们还没看到任何对这个问题的深入分析。

前面已经讨论过了 GAN 和流模型之间的权衡，那自回归模型呢？事实证明，自回归模型可以看做不可并行化的流模型（因为它们都可逆）。

因此，GAN 是并行且有效的，但不可逆；流模型是可逆且并行的，但比较低效；自回归模型是可逆且有效的，但不可并行化。

	Parallel	Efficient	Reversible
GANs	Yes	Yes	No
Flow Models	Yes	No	Yes
Autoregressive Models	No	Yes	Yes

由此引出第一个开放性问题：

> **Problem 1** What are the fundamental trade-offs between GANs and other generative models?
>
> In particular, can we make some sort of CAP Theorem [22] type statemer about reversibility, parallelism, and parameter/time efficiency?

解决这个问题的方法之一是研究更多由多种模型混合而成的模型。这种方法已经用于混合 GAN/流模型研究，但它仍然没有被充分开发。

我们也不确定最大似然训练是否一定比 GAN 训练更难。的确，在 GAN 训练损失下，将 zero mass 置于训练数据点上没有被明确禁止，但面对这种情况，强大的判别器的确会比生成器做得更好。不过，看起来 GAN 确实在实践中学习低支持度的分布。

最终，我们怀疑流模型每个参数的表达不如任意解码器函数，而且这在特定假设下是可以证明的。

GAN 能建模哪些分布？

大多数 GAN 都侧重于图像合成，具体而言，研究者会在一些标准图像数据集上训练 GAN，例如 MNIST、CIFAR-10、STL-10、CelebA 和 Imagenet 等。这些数据集也是有难易之分的，而且生成的效果也有好有坏。经验表明，CelebA 上最先进的图像合成模型生成的图像似乎比 Imagenet 上最先进的图像合成模型生成的图像更有说服力。

与任何科学一样，我们也希望有一个简单的理论来解释实验观察。理想情况下，我们可以查看数据集，并执行一些计算而不实际训练生成模型，然后就可以判断「这个数据集对于 GAN 来说比较容易建模，但是对于 VAE 来说比较难」。这些都是经验理解，不过目前在这个领域上也有一些研究。由此引出下面这个问题：

> **Problem 2** Given a distribution, what can we say about how hard it will be for a GA to model that distribution?

我们可能问「建模分布」到底是什么意思，会有一些 GAN 并不能学习到的分布吗？会不会有一些 GAN 理论上能学习的分布，但是在给定合理的计算资源下它学习的效率并不高？对于 GAN 来说，这些问题的答案和其他模型给出的会不会存在差别，现在很多都远没有解决。

Augustus 认为我们有两种策略来回答这些问题：

- 合成数据集：我们可以研究合成数据集来探讨到底哪些特征会影响数据集的可学习性。例如在论文《Are GANs Created Equal? A Large-Scale Study》中，研究者就创建了一个合成三角形的数据集。
- 修正现有的理论结果：我们可以利用现有的理论结果，并尝试修改假设以考虑数据集的不同属性。

除了图像合成外，GAN 还能用于哪些地方？

除了图像到图像的转换和领域的自适应等应用外，大多数 GAN 的成功应用都在图像合成中。而 GAN 在图像外的探索主要分为三个领域：

- 文本：文本的离散属性使其很难应用 GAN。因为 GAN 会依赖判别器的梯度信号，且它会通过生成内容反向传播给生成器，所以离散的字符难以更新。目前有两种方法解决这个困难，第一种是令 GAN 只对离散数据的连续表征起作用，第二种则是用梯度估计和实际离散的模型来训练 GAN。
- 结构化数据：GAN 能用于其它非欧氏空间的结构化数据（例如图）吗？这类数据的研究被称为几何深度学习。GAN 在这个领域的进展也不是非常显著，但其它深度学习方法取得的进步也比较有限，因此很难说是 GAN 自身的问题。
- 音频：音频是 GAN 除了图像外最成功的领域，将 GAN 应用于无监督音频合成是第一次严格的尝试，研究人员对各种实际音频操作做出了特殊的限制。

除了这些领域的尝试，图像一直是应用 GAN 最简单的领域，这就会引起一些问题：

> Problem 3 How can GANs be made to perform well on non-image data?
>
> Does scaling GANs to other domains require new training techniques, c
> does it simply require better implicit priors for each domain?

我们最终希望 GAN 能在其它连续数据上获得类似图像合成方面的成功，但它需要更好的隐式先验知识。寻找这些先验可能需要仔细思考到底哪些特征才是有意义的，并且领域中的哪些特征是可计算的。

对于结构化数据或离散数据，我们暂时还没有比较好的解决方案。一种方法可能是令生成器和判别器都采用强化学习的智能体，并以 RL 的方式进行训练。但这样又需要大量计算资源，这个问题可能还是需要基础研究的进展。

我们该如何评估 GAN 的好坏，什么时候又该使用 GAN 这种生成模型？

说到评估 GAN，目前有很多方法，但是并没有一种统一的度量方法：

- Inception Score 和 FID：这两个分数都使用预训练的图像分类器，都存在已知问题。常见的批评是这些分数测量「样本质量」而没有真正捕获「样本多样性」。
- MS-SSIM：可以使用 MS-SSIM 单独评估多样性，但该技术也存在一些问题，并没有真正流行起来。
- AIS：它建议在 GAN 的输出上应用高斯观测值模型（Gaussian observation），并使用退火重要性采样来评估该模型下的对数似然。但事实证明，当 GAN 生成器也是流模型时，这种计算方式并不准确。
- 几何分数：这种方法建议计算生成数据流形的几何属性，并将这些属性与真实数据进行比较。
- 精度和召回率：该方法尝试计算 GAN 的精度和召回率。
- 技能评级：该方法以证明，训练好的 GAN 判别器能够包含用来评估的有用信息。

这些还只是一小部分 GAN 评估方案。虽然 Inception Score 和 FID 相对比较流行，但 GAN 评估显然还不是一个确定性问题。最终，我们认为关于如何评估 GAN 的困惑源于何时使用 GAN。因此，我们将这两个问题合二为一：

> Problem 5 When should we use GANs instead of other generative models?
>
> How should we evaluate performance in those contexts?

我们应该用 GAN 来做什么？如果你想要真正的密集型模型，GAN 可能不是最好的选择。已有实验表明，GAN 学习了目标数据集的「low support」表征，这意味着 GAN（隐式地）将测试集的大部分分配为零似然度。

我们没有太担心这一点，而是将 GAN 研究的重点放在支撑集没问题甚至有帮助的任务上。GAN 可能很适合感知性的任务，如图像合成、图像转换、图像修复和属性操作等图形应用。

最后，虽然花费巨大，但也可以通过人力进行评估，这使得我们可以测量那些真正在乎的东西。通过建模预测人类答案，可以减少这种方法的成本。

如何扩展训练 GAN 的批量大小？

大的 minibatch 已经帮助扩展了图像分类任务——这些 minibatch 能帮助我们扩展 GAN 吗？对于有效地使用高度并行硬件加速器，大的 minibatch 可能非常重要。

乍一看，答案好像是肯定的——毕竟，多数 GAN 中的判别器只是个图像分类器而已。如果梯度噪声成为瓶颈，大的批量可以加速训练。然而，GAN 有一个分类器没有的独特瓶颈：训练步骤可能存在差异。因此，我们提出以下问题：

> Problem 6 How does GAN training scale with batch size?
>
> How big a role does gradient noise play in GAN training?
>
> Can GAN training be modified so that it scales better with batch size?

有证据表明，提高 minibatch 大小可以改进量化结果并减少训练时间。如果这一现象是鲁棒的，说明梯度噪声是非常重要的一个因素。然而，这一结论还没有得到系统性的验证，因此我们相信这一问题还有待解答。

交替训练步骤能否更好地利用大批量？理论上来看，最优传输 GAN 比一般 GAN 具有更好的收敛性，但需要一个大的批量，因为这种 GAN 需要对齐样本和训练数据批量。因此，最优传输 GAN 似乎是扩展到非常大的批量的潜在候选方法。

最后，异步 SGD 可以成为利用新硬件的不错备选项。在这种设定下，限制因素往往是：梯度更新是在参数的「陈旧」副本上计算的。但 GAN 实际上似乎是从在过去参数快照（snapshots）上进行的训练中获益，所以我们可能会问，异步 SGD 是否以一种特殊的方式与 GAN 训练交互。

GAN 和对抗样本之间有什么关系？

众所周知，对抗样本是图像分类任务需要克服的一大难题：人类难以察觉的干扰可以导致分类器给出错误的输出。我们还知道，有些分类问题通常可以有效学习，但鲁棒地学习却极其困难。

由于 GAN 判别器是一种图像分类器，有人可能担心其遭遇对抗样本。研究 GAN 和对抗样本的文献不在少数，但研究二者关系的文献却少得可怜。因此，我们不禁要问：

> Problem 7 How does the adversarial robustness of the discriminator affect GAN training?

我们如何开始考虑这一问题？假设有一个固定判别器 D。如果有一个生成器样本 G(z) 被正确分类为假样本，并且有一个小的扰动 p，G(z)+p 就被分类为真样本，那么 D 就有了一个对抗样本。使用一个 GAN 要考虑的是，生成器的梯度更新将产生一个新的生成器 G'，其中，G'(z) = G(z) + p。

这种担心是现实存在的吗？我们更担心一种叫做「对抗攻击」的东西。我们有理由相信这些对抗攻击发生的可能性较小。首先，在判别器再次更新之前，生成器只能进行一次梯度更新。其次，从先验分布中抽取一批样本，生成器得以优化，这批样本的每个梯度更新步都是不同的。

最后，优化是在生成器的参数空间（而不是像素空间）中进行的。然而，这些论点都没有完全排除生成器创建对抗样本的可能。这将是一个值得深度探讨且富有成果的话题。

原文链接：https://distill.pub/2019/gan-open-problems/

✂----------