第 2 卷◆第 1 期◆版本 1.0◆2024 年
文章类型：论文|刊号（ISSN）：2972-4007(P) / 2972-4015(O)

为何缩放法则不能解决大语言模型的实际问题

文木源
GPT DESK PTE LTD
DOI:10.12238/deitar.v2i1.6833

[摘　要] 缩放法则,尽管在AI训练中广泛使用,但并不能解决所有问题。这种方法主要关注单一的损失函数,使用静态的训练集和验证集,这可能使问题过于简化,有时甚至不可接受。在实际应用中,考虑部署成本和执行多样任务的能力变得至关重要,尤其是那些在训练数据中较少出现的任务。此外,对齐的过程及其对不同规模模型的影响仍然是未知数。在数据的大小和质量之间需要做出权衡,提高数据集质量或创造更多数据都是可行的方法。部署后,现实世界的数据增加为模型提供了新的学习机会,如人类反馈强化学习。当模型在低频任务上表现不佳时,这突显了模型对齐阶段的挑战。这通常是因为基础模型在数据集中缺乏某些任务的代表性。解决方案之一是调整基本模型以平衡任务频率,通过修改训练数据集以更公平地代表低频任务。另一种方法是少样本学习,即在基础训练阶段针对每项任务使用少量示例。这两种策略都旨在通过丰富训练的多样性来增强模型的整体性能和适应性。
[关键词] 缩放法则；模型对齐；任务频率；数据质量；小样本学习；性能适应性
中图分类号：TV149.2　文献标识码：A

Why Scaling Rules Do Not Solve Practical Problems with Large Language Models

Muyuan Wen
GPT DESK PTE LTD

[Abstract] Scaling Laws, although widely used in AI training, do not solve all problems. This approach mainly focuses on a single loss function and uses static training and validation sets, which may oversimplify the problem and sometimes even be unacceptable. In practical applications, it becomes critical to consider deployment costs and the ability to perform diverse tasks, especially those that appear less frequently in the training data. Furthermore, the process of alignment and its impact on models of different scales remain unknown. There is a trade-off between the size and quality of the data, and improving the quality of the data set or creating more data are possible approaches. Once deployed, the increase in real-world data provides the model with new learning opportunities, such as reinforcement learning with human feedback. It highlights the challenges in the model alignment stage when the model performs poorly on low-frequency tasks. This is usually because the underlying model lacks representation of certain tasks in the dataset. One solution is to adjust the base model to balance task frequency by modifying the training data set to more fairly represent low-frequency tasks. Another approach is few-shot learning, which uses a small number of examples for each task in the base training phase. Both strategies aim to enhance the overall performance and adaptability of the model by enriching training diversity.
[Key words] Scaling Laws; Model Alignment; Task Frequency; Data Quality; Few-shot Learning; Performance Adaptability

在大语言模型的调优过程中,缩放法则并不能解决所有问题,因为它们专注于具有单一损失指标的训练过程,并使用静态训练和验证集。这种方法简化了问题,有时会导致不令人满意的结果。

在实际应用中,我们必须考虑部署成本和任务的多样性,而不仅仅是单词预测,特别是在训练数据中很少出现的任务。基于大小、初始训练样本和步骤的对齐对各种模型的影响尚不清楚。平衡数据数量和质量至关重要,需要战略性地在提高数据集质

量或生成更多数据之间做出选择。此外，模型部署后现实世界数据的增加，以及人类反馈强化学习等技术的后续应用，引发了这样的问题：这些因素如何不同地影响各种模型规模，以及多少对齐可以解决问题。

1 缩放法则

缩放定律的主要目标是找出计算资源、数据集大小和模型参数之间的关系。它有助于解决以下问题：模型应该有多大，或者大语言模型需要多少数据，或者大语言模型应该有多少参数，等等。

为了便于大家形象地了解这一问题，我们来看下面的图1至图4及附在图上的说明：

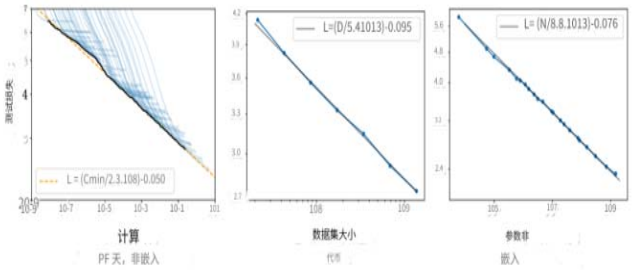

图 1 随着我们增加模型大小、数据集大小和用于训练的计算量[2]，语言建模性能平稳提高。为了获得最佳性能，所有三个因素必须同时扩大。当不受其他两个因素的瓶颈时，经验绩效与每个单独的因素都具有幂律关系。

注：以上图示翻译自《模型的缩放定律》https://arxiv.org/pdf/2001.08361.pdf

该图示呈现了三个线图，每个图都说明了不同因素对语言模型测试损失的影响。测试损失是用于评估模型预测与实际结果匹配程度的指标，较低的测试损失表明更好的性能。

1.1第一张图(计算)：显示了计算资源(以PF天衡量，代表PetaFlop天，计算能力的单位)和测试损失之间的关系。各种线代表不同的模型配置或实验，突出显示的虚线作为参考。如该图所示，随着计算资源的增加，测试损失呈幂律下降，这表明更多的计算能力可以显着提高模型性能。

1.2第二张图(数据集大小)：这张图说明了数据集大小的增加(以标记为单位)如何与测试损失的减少相关。标有“L=(D/5.4•10^13)^-0.095”的线性线表示特定的缩放法则，其中“D”代表数据集大小。这一趋势表明，在遵循幂律关系的基础上，更大的数据集往往会带来更好的模型性能。

1.3第三张图(参数)：这张图显示了参数数量(非嵌入)对测试损失的影响。我们再次观察到参数数量和测试损失之间的幂律关系，参数越多，测试损失就越低，这意味着模型性能得到提高。

总之，随着模型大小、数据集大小和用于训练的计算量的增加，语言建模性能也会提高。然而，为了获得最佳性能，这些因素必须一起扩大。如果这三个因素中的任何一个没有按比例增加，它都可能成为瓶颈，限制扩展其他因素所带来的好处。最重要的是，语言建模的性能与这三个关键因素存在幂律关系，而平衡的缩放对于实现最佳结果至关重要。

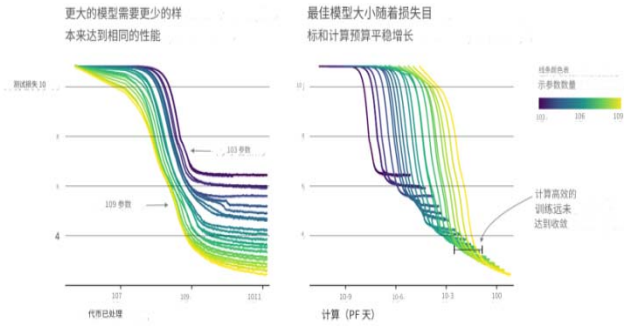

图 2 我们展示了一系列语言模型训练运行，模型大小从 103 到 109 个参数不等（不包括嵌入）。

注：以上图示翻译自《模型的缩放定律》https://arxiv.org/pdf/2001.08361.pdf

左图显示了处理的令牌数量(模型训练数据量的度量)和测试损失之间的关系。由图可见，随着参数数量的增加(由曲线向右移动表示)，模型需要处理更少的令牌来实现更低的测试损失。这表明较大的模型样本效率更高，可以用更少的数据达到更好的性能。

右图显示出针对所使用的计算资源的测试损失(以PF天为单位)。由图可风，随着计算资源的增加，测试损失会减少。值得注意的是，对于每条曲线(模型大小)，都有一个收益递减点，其中额外的计算不会显着降低测试损失。这意味着对于给定的损失目标和计算预算存在最佳模型大小。

以上是不同大小的语言模型的训练运行的效果，较大模型的效率更突出，因为需要更少的数据样本即可达到相同的性能水平。既然存在与损失目标和可用计算资源相关的最佳模型大小，那么，如果计算预算有限，模型并不总是越大越好。语言模型的效率和性能受到其参数数量和可用计算资源的影响，但为了获得最佳结果，需要保持平衡。

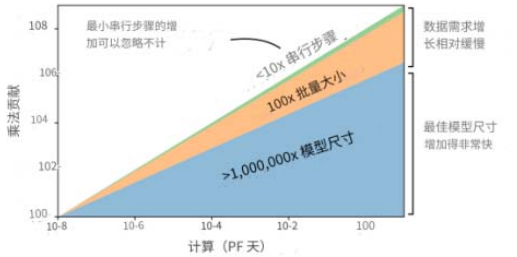

图 3 随着更多计算的可用，我们可以选择分配多少资源来训练更大的模型、使用更大的批次以及训练更多的步骤。我们通过计算量增加十亿倍来说明这一点。为了获得最佳的计算效率训练，大部分增加应该用于增加模型大小。需要相对少量地增加数据以避免重复使用。在数据的增加中，大部分可用于通过更大的批量大小来提高并行性，而所需的串行训练时间仅增加非常小的一部分。

注：以上图示翻译自《模型的缩放定律》https://arxiv.org/pdf/2001.08361.pdf

这张双对数图展示了计算资源(以PF天为单位)与“乘法贡献”之间的关系，而“乘法贡献”指的是模型的综合效益，即尺寸、批次大小和串行步骤数对模型训练过程的效率或有效性的影响。

文章类型：论文|刊号（ISSN）：2972-4007(P) / 2972-4015(O)

随着计算资源的增加，模型的最佳大小似乎会迅速增加。这表明，凭借更强大的计算能力，可以更有效地训练更大的模型。

增加批次大小（在训练的每一步中一起处理的样本数量）的影响会大幅增长，但它的增长并不如模型大小那么急剧。这表明较大的批次大小有助于提高效率，但会出现收益递减的情况。

此外，随着计算量的增加，串行步骤（顺序处理步骤）的数量略有增加。这表明随着计算能力的增强，串行处理的需求只会略有增长。

最后，数据需求的增长相对缓慢，这意味着所需的数据量不需要像计算量或模型大小一样快速增长来保持模型的效率。

当我们有更多的计算资源时，我们可以选择如何分配：是训练更大的模型，使用更大的批次大小，还是训练更多的步骤？对于更高效的训练来说，大部分资源的增加应该用于更大的模型和更大的批次大小，而不是简单地更多数据或更长的训练时间。上面图示中说明了计算量增长十亿倍的策略，强调相对较小的数据增长就足以避免过于频繁地重用数据（可能导致过拟合），并且数据的增长应该主要用于增强计算能力。此外，我们应该通过更大的批次大小而不是更长的训练来实现并行性。

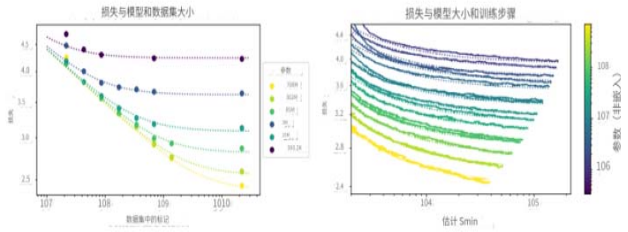

图 4 左：根据公式 (1.5)，提前停止的测试损失 L(N, D) 随着数据集大小 D 和模型大小 N 的变化而变化。右图：在初始瞬态期之后，所有模型大小 N 的学习曲线都可以用方程 (1.6) 拟合，该方程以 Smin 为参数，Smin 是大批量训练时的步数（详细信息请参见第 5.1 节）。

注：以上图示翻译自《模型的缩放定律》https://arxiv.org/pdf/2001.08361.pdf

左图显示了与数据集中令牌数量相关的损失，不同的曲线代表不同参数大小的模型（范围从393.2K到708M参数）。随着数据集中标记数量的增加，所有模型的损失都会减少。然而，较大的模型（具有更多参数）在相同的数据集大小下往往具有较低的损失，这表明它们在利用数据方面更有效。

右图则显示了与估计Smin相比的损失，这可能代表达到特定性能阈值所需的最小训练步骤数。右侧的色标表示模型中参数的数量。随着训练步骤数的增加，所有大小的模型的损失都会减少。参数数量较多的模型比较小的模型更快地达到较低的损失水平。

由图示可见，数据集的大小和模型的大小（就参数而言）都是模型性能的重要预测因素（通过损失来衡量）。此外，损失和这些变量之间的关系可以通过特定的方程来描述，表明随着数据量或训练步骤数量的增加，损失如何减少存在着可预测的模式。

这些真知灼见来自OpenAI的一篇著名论文，它提出了一些构建大语言模型的重要见解，涉及如何获得最佳大语言模型、如何通过计算资源预算和数据集规模来估计模型大小和损失。这些规则即使在现在也非常有用，并吸引了许多后续研究，例如 DeepMind 的 Chinchilla。

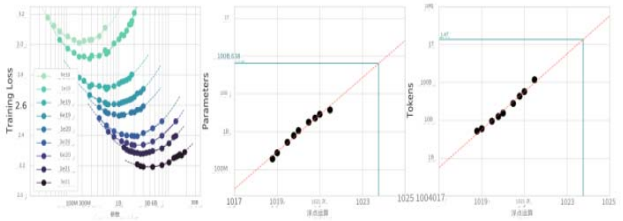

图3| IsoFLOP 曲线。对于各种模型大小，我们选择训练令牌的数量，以使最终的 FLOP 是一个常数。余弦周期长度设置为与目标 FLOP 计数匹配。我们发现损失有一个明显的谷值，这意味着对于给定的 FLOP 预算，有一个最佳模型可以训练（左）。利用这些山谷的位置，我们可以预测较大模型的最佳模型大小和令牌数量（中心和右侧）。在绿色中，我们显示了使用 Gopher 的计算预算训练的最佳模型的参数和标记的估计数量。

注：该图翻译自《训练计算最优大型语言模型》https://arxiv.org/pdf/2203.15556.pdf

这是来自Deep Mind的一篇论文，他们在这篇论文中提出了Chinchilla。这篇论文对缩放定律也做出了很大贡献。著名的开源大语言模型LLaMa系列跟踪了这项研究，并得到了相当积极的反馈。所有基准测试都有效，我们似乎找到了模型大小、数据大小和计算成本的最佳值。他们大致符合以下公式：

isoFLOPs=iso-FLOPs=相同的FLOPs

2 缩放法则并不能解决所有问题

为什么缩放法则并不能解决所有问题？缩放法则专注于训练事物，其目标是单一损失，训练集和验证集是静态的。这些事情并不意味着缩放法则失效了，但它们使问题变得简单，有时甚至令人无法接受。

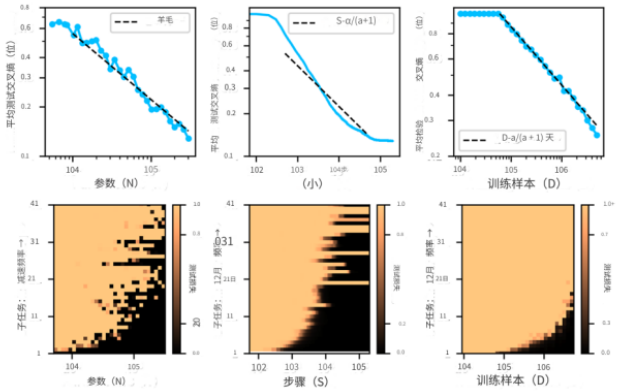

图 2：顶部：神经网络在损失方面表现出幂律缩放。在多任务稀疏奇偶校验数据集上训练时的参数 N、训练时间 S 和训练样本 D（用于多时期训练）。这里 a = 0.4，我们绘制线 ∝ N-α、α S-a/(a+1)、∝ D-a/(a+1)。底部：按子任务细分的神经缩放。单个子任务的缩放行为表现出涌现，其中子任务突然学习到超过特定的规模。均值测试损失的幂律神经标度对网络性能的大量质变进行平均（按子任务分解时），随着越来越多的子任务（按频率分布的幂律分布）将损失驱至零，实现了第 2 节讨论了神经缩放机制。

注：该图翻译自《神经缩放的量化模型》https://arxiv.org/pdf/2303.13506.pdf

在现实世界的需求中，我们需要了解部署成本，并且我们的任务不仅仅是预测下一个单词，我们需要它来执行任务，并且任务在训练数据集中的频率可能较低。我们需要进行对齐，并且我们不知道对齐对模型大小/基础训练样本/基础训练步骤上的不同基础模型有何影响。我们还需要在数据大小和数据质量之间进行权衡，以某种方式，我们可以努力提高数据集质量，或者有意地制造更多数据，我们需要选择目标。此外，一旦我们的模型部署完毕，现实世界的数据量就会增加，我们可以在上面做人类反馈强化学习之类的事情。我们也会进一步了解到这些指标在不同规模的模型上有何不同，通过对齐可以解决多少问题。

当模型在低频任务上表现不佳但与缩放定律保持良好一致时，它凸显了模型对齐阶段需要解决的挑战。出现此问题的原因是，基础模型通常是在某些任务代表性不足的数据集上进行训练的。因此，模型在这些领域的熟练程度是有限的。

一种可能的解决方案是调整基本模型以平衡任务频率。这意味着有意修改训练数据集，以确保更平等地表示不太频繁的任务。这种方法可以帮助模型在更广泛的任务中形成更平衡的理解和表现。

另一种方法是在基础训练阶段除了纯文本之外还训练模型的少量任务。少样本学习涉及针对每项任务使用少量示例来训练模型，教其从有限的数据中进行概括。这种方法可能会增强模型在训练数据中没有大量体现的任务上表现良好的能力，因为它学会了充分利用有限的信息。

这两种策略都旨在通过更多样化的任务和学习场景来丰富其训练，从而减轻基础模型的局限性，提高其整体性能和适应性。

[参考文献]

[1]陈欣.例谈数学模型在解决实际问题中的应用[J].模型世界,2022(16):200-202.

[2]徐春宇.利用函数模型解决实际问题的策略[J].中学生数理化(高一使用),2022(1):18-19.