ChatGPT背后的经济账

ChatGPT能否取代Google、百度这样的传统搜索引擎？为什么中国不能很快做出ChatGPT？当前，对这些问题的探讨大多囿于大型语言模型（LLM）的技术可行性，忽略或者非常粗糙地估计了实现这些目标背后的经济成本，从而造成对LLM的开发和应用偏离实际的误判。

本文作者从经济学切入，详细推导了类ChatGPT模型搜索的成本、训练GPT-3以及绘制LLM成本轨迹的通用框架，为探讨LLM成本结构和其未来发展提供了可贵的参考视角。

原文地址 https://sunyan.substack.com/p/the-economics-of-large-language-models

动机

LLM的惊人表现引发了人们的广泛猜想，这些猜想主要包括LLM可能引发的新兴商业模式和对现有模式的影响。

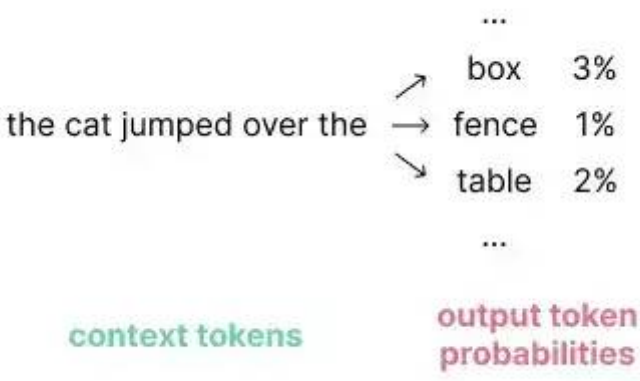
搜索是一个有趣的机会，2021年，仅谷歌就从搜索相关的广告中获得了超1000亿美元的收入[1]。ChatGPT（一个使用LLM的聊天机器人，它可以生成高质量的答案，以回答类似于搜索的查询）的“病毒性”传播已经引发了许多关于搜索领域潜在影响的思考，其中一个就是LLM如今的经济可行性：

- 一位声称是谷歌员工的人在HackerNews上表示，要想实施由LLM驱动的搜索，需要先将其成本降低10倍。
- 与此同时，微软预计将在3月份推出LLM版本的Bing[3]，而搜索初创公司如You.com已经将该技术嵌入到了他们的产品之中[4]。
- 最近，《纽约时报》报道，谷歌将在今年推出带有聊天机器人功能的搜索引擎[5]。

更广泛的问题是：**将LLM纳入当前产品和新产品的经济可行性如何？**在本文中，我们梳理了当今LLM的成本结构，并分析其未来可能的发展趋势。

2
重温LLM工作原理
尽管后续章节的技术性更强，但这篇文章对机器学习熟悉程度不做要求，即使不熟悉这方面内容的人也可以放心阅读。为了说明LLM的特殊之处，现做一个简要复习。

语言模型在给定上下文的情况下，对可能输出的token作出预测：



自回归语言模型（Autoregressive Language Model）输入上下文和输出内容的图示（在实践中，token通常是子词：即“happy”可能被分解为两个token，例如“hap”、“-py”）

为了生成文本，语言模型根据输出token的概率重复采样新token。例如，在像ChatGPT这样的服务中，模型从一个初始prompt开始，该prompt将用户的查询作为上下文，并生成token来构建响应（response）。新token生成后，会被附加到上下文窗口以提示下一次迭代。

语言模型已经存在了几十年。当下LLM性能的背后是数十亿参数的高效深度神经网络（DNN）驱动。参数是用于训练和预测的矩阵权重，浮点运算（FLOPS）的数值通常与参数数量（parameter count）成比例。这些运算是在针对矩阵运算优化的处理器上计算的，例如GPU、TPU和其他专用芯片。

随着LLM参数量呈指数增长，这些操作需要更多的计算资源，这是导致LLM成本增加的潜在原因。

3
LLM驱动搜索的成本

本节，我们将估算运行LLM驱动搜索引擎的成本。应该如何实施这样的搜索引擎仍是一个活跃的研究领域，我们这里主要考虑两种方法来评估提供此类服务的成本范围：

- ChatGPT Equivalent：一个在庞大训练数据集上训练的LLM，它会将训练期间的知识存储到模型参数中。在推理过程中（使用模型生成输出），LLM无法访问外部知识[6]。
    - 这种方法有如下两大缺点：
        - 容易“幻想”事实。
        - 模型知识滞后，仅包含最后训练日期之前的可用信息。
- 2-Stage Search Summarizer：一种架构上类似的LLM，可以在推理时访问Google或Bing等传统搜索引擎。在这种方法的第一阶段，我们通过搜索引擎运行查询以检索前K个结果。在第二阶段，通过LLM运行每个结果以生成K个响应，该模型再将得分最高的响应返回给用户[7]。
    - 相比ChatGPT Equivalent，这种方法的优点是：
        - 能够从检索到的搜索结果中引用其来源。
        - 能获取最新信息。

然而，对于相同参数数量的LLM，这种方法需要更高的计算成本。使用这种方法的成本也增加了搜索引擎的现有成本，因为我们在现有搜索引擎的结果上增加了LLM。

一阶近似：基础模型API

最直接的成本估算方法是参考市场上现有基础模型API的标价，这些服务的定价包括成本的溢价部分，这部分是供应商的利润来源。一个代表性的服务是OpenAI，它提供基于LLM的文本生成服务。

OpenAI的Davinci API由GPT-3的1750亿参数版本提供支持，与支持ChatGPT的GPT-3.5模型具有相同的参数数量[8] 。现在用该模型进行推理的价格约为0.02美元/750个单词（0.02美元/1000个token，其中1000token约等于750个单词）；用于计算定价的单词总数包括输入和输出[9]。

	Price per 1000 Tokens
Ada	$0.0004
Babbage	$0.0005
Curie	$0.0020
Davinci	$0.0200

按模型功能划分的基础模型API定价 (OpenAI)

我们这里做了一些简单假设来估计将支付给OpenAI的搜索服务费用：

- 在ChatGPT equivalent的实现中，我们假设该服务平均针对50字的prompt生成400字的响应。为了产生更高质量的结果，我们还假设模型对每个查询采样5个响应，从中选择最佳响应。因此：

$$\frac{\text{OpenAI revenue}}{\text{query}} = \left(\underbrace{\frac{\text{50 query prompt words}}{\text{query}}}_{\text{input words}} + \underbrace{\frac{\text{400 generated words}}{\text{sampled response}} \times \frac{\text{5 sampled responses}}{\text{query}}}_{\text{output words}} \right) \times \underbrace{\frac{\text{1000 tokens}}{\text{750 words}}}_{\text{words to tokens conversion}} \times \underbrace{\frac{\$0.02}{\text{1000 tokens}}}_{\text{token pricing}} = \$0.055/\text{query}$$

在2-Stage Search Summarizer的实现中，响应生成过程是相似的。然而：

    - 提示明显更长，因为它同时包含查询和搜索结果中的相关部分
    - 为每K个搜索结果生成一个单独的LLM响应
- 假设K = 10并且搜索结果中的每个相关部分平均为1000个单词：

$$\frac{\text{OpenAI revenue}}{\text{query}} = \left(\underbrace{\frac{\text{50 query prompt words}}{\text{query}} + \left(\frac{\text{1000 search result prompt words}}{\text{link}} + \frac{\text{400 generated words}}{\text{link}} \right) \times \frac{\text{10 links}}{\text{query}}}_{\text{input and output words}} \right) \times \underbrace{\frac{\text{1000 tokens}}{\text{750 words}}}_{\text{words to tokens conversion}} \times \underbrace{\frac{\$0.02}{\text{1000 tokens}}}_{\text{token pricing}} = \$0.375/\text{query}$$

假设优化的缓存命中率为30%（谷歌历史搜索缓存命中率的下限[10]）和OpenAI云服务的毛利率为75%（与典型的SaaS服务一致），我们的一阶估计意味着：

	ChatGPT Equivalent	2-Stage Search Summarizer
Query Prompt Words/Query	50	
Generated Words/Sampled Response	400	
Sampled Responses/Query	5	
Search Result Prompt Words/Link		1000
Generated Words/Link		400
# of Links/Query		10
# of Words/1000 Tokens	750	
OpenAI Revenue/1000 Tokens	$0.02	
Estimated OpenAI Revenue/Query	**$0.055**	**$0.375**
Cache Hit Rate	30%	
OpenAI Gross Margin	75%	
Estimated Cost/Query	**$0.010**	**$0.066**

按照数量级，ChatGPT Equivalent服务的预计云计算成本为0.010美元/次，与公众评论一致：

OpenAI首席执行官Sam Altman谈ChatGPT每次聊天的成本（[推特](https://twitter.com/sama/status/1599671496636780546?lang=en)）

鉴于ChatGPT Equivalent的上述缺点（即幻想事实、模型信息陈旧），在实际操作中，LLM驱动搜索引擎的开发者更可能部署2-Stage Search Summarizer变体。

2012年，谷歌搜索主管表示，其搜索引擎每月处理的搜索次数达1000亿次[11]。世界银行数据显示：全球互联网普及率已从2012年的34%上升到了2020年的60%[12]。假设搜索量按比例增长，

则预计其年均搜索量将达2.1万亿次，与搜索相关的收入将达约1000亿美元[13]，平均每次搜索的收入为0.048美元。

换句话说，2-Stage Search Summarizer的查询成本为0.066美元/次，约为每次查询收入0.048美元的1.4倍。

- 通过以下优化，预估成本大约会降至原来的1/4：1、量化（使用较低精度的数据类型）2、知识蒸馏（通过学习较大的模型去训练一个较小的模型） 3、训练更小的“计算优化”模型，该模型具有相同的性能（稍后将对此展开更详细的讨论）
- 假设云计算的毛利率约为50%，与依赖云服务提供商相比，运行自建（内部）基础设施（infrastructure in-house）会使成本降低至当前的1/2。

> 综合以上改进，降低至原有成本的1/8之后，在搜索中融入高性能LLM的成本大约占据当前查询收入的15%（现有的基础设施成本除外）。（注：成本最低可降至 0.066 美元/次 * 1/4 * 1/2， 约定于0.008美元，因此大约占每次查询收入 0.048 美元的 15%）

深度解析：云计算成本

如今，SOTA大型语言模型通常会用到可比较的模型架构（最常见的是仅包含解码器的Transformer模型），在推理过程中每个token的计算成本（以FLOPs为指标）约为2N，其中N为模型参数数量（model parameter count）[14]。

目前，NVIDIA A100是AWS最具成本效益的GPU选择，若预定1年使用该GPU，拥有8个A100的AWS P4实例的有效时薪（effective hourly rate）将达19.22美元。[15]每个A100提供峰值312 TFLOPS（万亿次浮点数/秒）FP16/FP32 混合精度吞吐量，这是LLM训练和推理的关键指标[16]。FP16/FP32混合精度是指以16位格式（FP16）执行操作，而以32位格式（FP32）存储信息。由于FP16的开销较低，混合精度不仅支持更高的FLOPS吞吐量，而且保持精确结果所需的数值稳定性也会保持不变[17]。

假设模型的FLOPS利用率为21.3%，与训练期间的GPT-3保持一致（虽然最近越来越多的模型效率得以提升，但其FLOPS利用率对于低延迟推理而言仍充满挑战）[18]。因此，对于像GPT-3这样拥有1750亿参数的模型：

$$\frac{\text{AWS cost}}{1000\text{ tokens}} = \underbrace{\overbrace{175\text{B model parameters} \times \frac{2\text{ FLOPs}}{\text{token} \cdot \text{model parameter}}}^{\text{FLOPs per token for the model}}}_{\text{FLOPs per second for each machine}} \times \underbrace{\frac{\$19.22}{\text{hour} \cdot \text{machine}} \times \frac{1\text{ hour}}{3600\text{ seconds}}}_{\text{pricing per second for each machine}}$$

$$= \$0.0035/1000\text{ tokens}$$

我们也应用了基于GCP TPU v4定价（ GCP TPU v4 pricing）相同的计算方法，并得到了相似的结果[19]：

	AWS A100 (P4 Instance)	GCP TPU v4
Model Parameters	175B	
FLOPs/Token/Model Parameter	2	
GPUs/Machine	8	
Peak FLOPS/GPU	312T	
TPUs/Machine		4
Peak FLOPS/TPU		275T
FLOPS Utilization	21.3%	
Cost/Machine/Hour (1-year reserved)	$19.22	$8.12
Seconds/Hour	3600	
Inference Cost/1000 Tokens	**$0.0035**	**$0.0034**

预估GPT-3通过云服务提供商 (AWS, GCP）每处理1000个token所需的推理成本

OpenAI的API定价为0.02美元/1000词，但我们估计其成本约为0.0035美元/1000词，占定价的20%左右。这就意味着：**对于一台一直运行的机器而言，其毛利率约为80%。**这一估算与我们之前设想的75%毛利率大致相同，进而为ChatGPT Equivalent和2-Stage Search Summarizer搜索成本估算提供了合理性验证（sanity check）。

4
训练成本如何?

另一个热门话题是GPT-3（拥有1750亿参数）或最新的LLM（如拥有2800亿参数的Gopher和拥有5400亿参数的PaLM）的训练成本。基于参数数量和token数量，我们构建了一个用于估算计算成本的框架，虽然稍作修改，但同样适用于此：

- 每个token的训练成本通常约为6N（而推理成本约为2N），其中N是LLM的参数数量[20]
- 假设在训练过程中，模型的FLOPS利用率为46.2% （而在之前的推理过程中，模型的FLOPS利用率约为21.3%），与在TPU v4芯片上进行训练的PaLM模型（拥有5400亿参数）一致[21]。

> 1750亿参数模型的GPT-3是在3000亿token上进行训练的。谷歌使用了GCP TPU v4芯片来训练PaLM模型，若我们现在也像谷歌那样做，那么如今的训练成本仅为140万美元左右。

$$\text{cost of training} = \underbrace{\overbrace{175\text{B model parameters} \times \frac{6\text{ FLOPs}}{\text{token} \cdot \text{model parameter}}}^{\text{FLOPs per token for training the model}}}_{\text{realizable FLOPs per second for each machine}} \times \underbrace{\frac{\$8.12}{\text{hour} \cdot \text{machine}} \times \frac{1\text{ hour}}{3600\text{ seconds}}}_{\text{pricing per second for each machine}} \times 300\text{B tokens}$$

$$= \frac{4\text{ TPUs}}{\text{machine}} \times \frac{275\text{ peak TFLOPS}}{\text{TPU}} \times 46.2\% \frac{\text{realizable TFLOPS}}{\text{peak TFLOPS}}$$

$$= \$1.398\text{M}$$

此外，我们还将该框架应用到一些更大的LLM模型中，以了解其训练成本。

	GPT-3 (OpenAI)	Gopher (Google DeepMind)	MT-NLG (Microsoft/Nvidia)	PaLM (Google Research)
Model Parameters	175B	280B	530B	540B
FLOPs/Token/Model Parameter		6		
TPUs/Machine		4		
Peak FLOPS/TPU		275T		
FLOPS Utilization		46.2%		
Cost/Machine/Hour (1-year reserved)		$8.12		
Seconds/Hour		3600		
Training Cost/1000 Tokens	**$0.0047**	**$0.0075**	**$0.0141**	**$0.0144**
Train Tokens	300B	300B	270B	780B
Training Cost	**$1,398,072**	**$2,236,915**	**$3,810,744**	**$11,216,529**

预估LLM在GCP TPU v4芯片上的训练成本

5
绘制成本轨迹的通用框架

为了推导LLM的推理成本/训练成本，我们总结了如下框架：

密集激活纯解码器LLM模型Transformer（Densely Activated Decoder-Only Transformer LLMs）的推理成本和训练成本（其中“N”是模型参数数量，“processor”是指TPU、GPU或其他张量处理加速器）

因此，我们假设LLM的架构相似，那么推理成本和训练成本将根据上述变量的变化而变化。虽然我们会详细考虑每个变量，但是以下部分才是关键点：

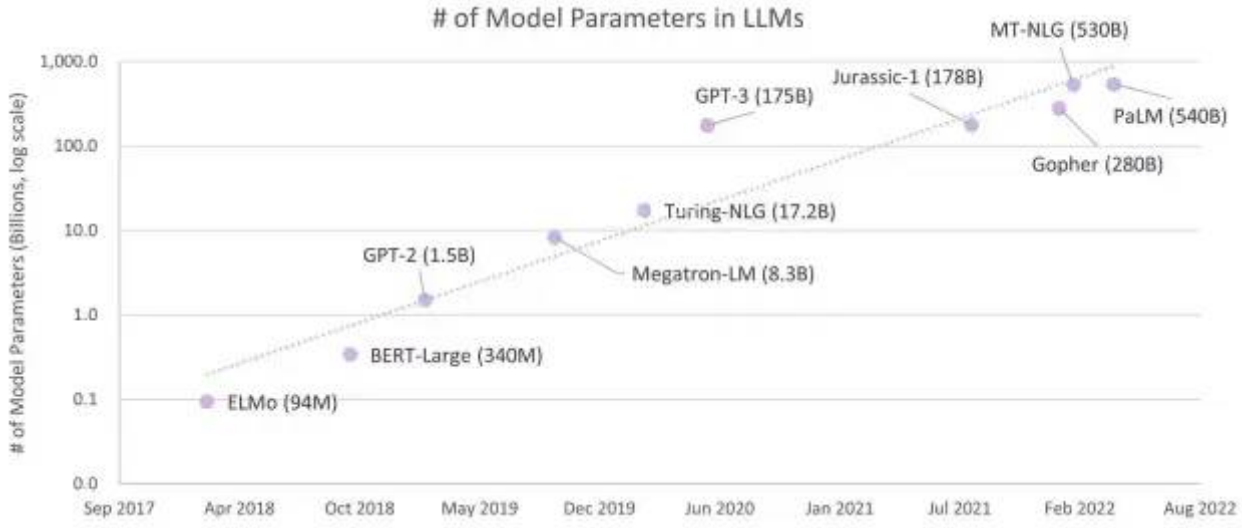
> 自2020年GPT-3发布以来，使用与GPT-3一样强大的模型进行训练和推理的成本大大降低，低于先前的五分之一。

	Cost of Inference	Cost of Training
Parameter Count ("*N*")	>60% Fewer Parameters *(Chinchilla's 70B parameters vs. GPT-3's 175B parameters with performance parity)*	
Cost/FLOP	58% Cost/FLOP Reduction *(Hardware cost and energy efficiency of H100 vs. V100, which was used to train GPT-3)*	
Model FLOPS Utilization		2.2x FLOPS Utilization *(GPT-3's 21.3% training utilization vs. PaLM's 46.2%)*
Net Reduction vs. GPT-3 in 2020	**>83%**	**>81%**

相比2020年推出的GPT-3，与其性能对等的模型的推理与训练成本降低情况总结

参数数量效率：巨型语言模型参数每年增长10倍的神话

考虑到过去5年中模型参数呈指数增长，我们普遍猜测：下一代LLM模型很可能是万亿参数（密集激活）模型：



LLM中模型参数数量的增长

虽然LLM的参数数量每年约增长10倍，但是大多数模型训练数据集的大小并没有显著变化：

$$L(N, D) = 1.69 + \frac{406.4}{N^{0.34}} + \frac{410.7}{D^{0.28}}$$

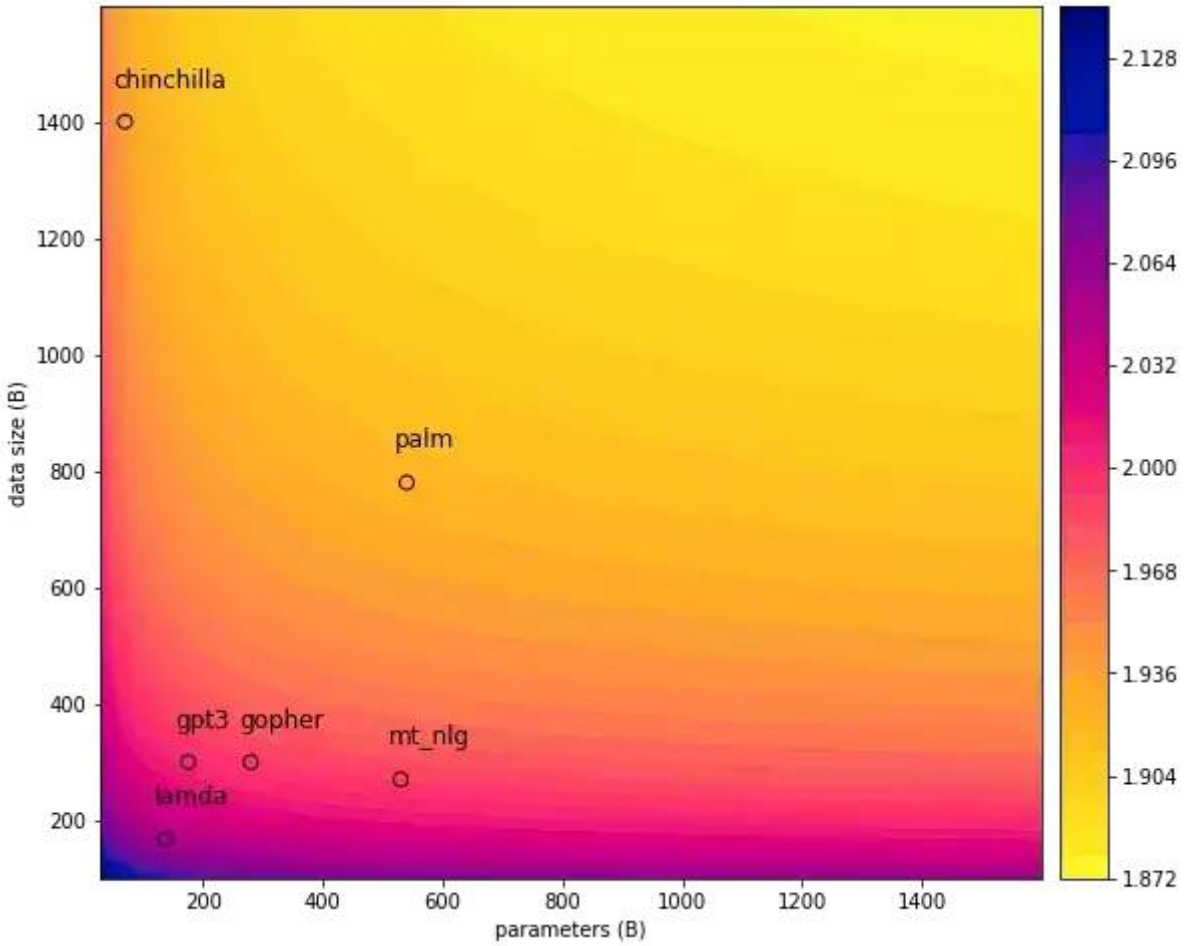
所选LLM的模型参数数量与训练token数量 (训练计算最优大语言模型)

然而，最新文献表明，假设计算资源和硬件利用率（即训练“计算最优”模型）保持不变，关注扩展参数数量（scaling parameter count）并不是性能最大化的最佳方式：
Google DeepMind的研究人员将一个参数函数（parametric function）拟合到他们的实验结果中，发现参数数量N的增速应与训练token数量D的增长速度大致相同，从而让模型损失L实现最小化（即性能最大化）：

$$L(N, D) = 1.69 + \frac{406.4}{N^{0.34}} + \frac{410.7}{D^{0.28}}$$

模型损失的参数函数 (训练计算最优大语言模型)

研究人员还训练了一个名为Chinchilla的模型（拥有700亿的参数）。虽然该模型的计算资源与Gopher（拥有2800亿参数）相同，但是该模型是在1.4万亿token上进行训练的而非3000亿token。Chinchilla的性能明显优于拥有相同FLOPs预算的大型模型，从而证明了大多数LLM过度支出了计算量和对数据的渴望 （译者注：换言之，对大多数LLM来说，使用更多的数据来训练比增大模型参数量要更加划算）。



通过训练数据大小与模型参数来预测模型损失(错误更少：Chinchilla的自然环境含义)

> 虽然Chinchilla的参数（以及推理计算需求）比GPT-3少60%，但是其性能远远优于拥有1750亿参数的GPT-3模型。

实际上，即使我们用与GPT-3相同的3000亿token数据集去训练一个万亿参数模型，仍可以预见该模型的表现不如Chinchilla：

$$\text{Loss of 1T parameter model} = L(1\text{T}, 300\text{B}) = 1.69 + \underbrace{0.03}_{\text{model parameter loss}} + \underbrace{0.25}_{\text{training token loss}} = 1.97$$

$$\text{Loss of Chinchilla} = L(70\text{B}, 1.4\text{T}) = 1.69 + \underbrace{0.08}_{\text{model parameter loss}} + \underbrace{0.16}_{\text{training token loss}} = 1.94$$

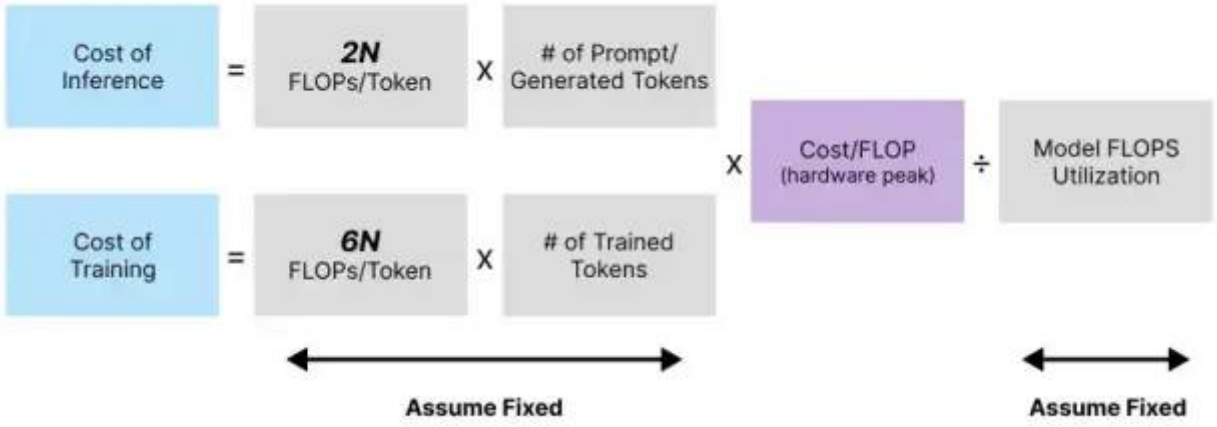
万亿参数模型相应损失项的相对量级（0.03的模型参数损失与0.25的训练token损失）也表明，通过增加模型大小获得的边际效益低于增加数据量获得的边际效益。

> 展望未来，我们不会继续扩大模型参数数量，而是将增量计算资源（incremental computational resources）转移到质量相当的更大数据集上进行训练，以获得极佳的性能。

Cost/FLOP效率

对于训练LLM而言，最重要的硬件性能指标（hardware performance metric）是可实现的混合精度FP16/FP32 FLOPS。改进硬件旨在实现成本最小化，同时使得峰值FLOPS吞吐量和模型FLOPS利用率实现最大化。

虽然这两个部分在硬件开发中密不可分，但为了让分析变得更简单，本节重点关注吞吐量，下一节再讨论利用率。



目前，我们已经通过查看云实例定价（cloud instance pricing）估算了Cost/FLOP效率。为了进行下一步探究，我们估算了运行以下机器的成本。主要包括以下两个方面：1）硬件购买（hardware purchase） 2）能源支出（energy expense）。为说明这一点，我们再来看看GPT-3(一款由OpenAI推出的模型，该模型在Microsoft Azure的10000个V100 GPU上训练了14.8天)[22]：

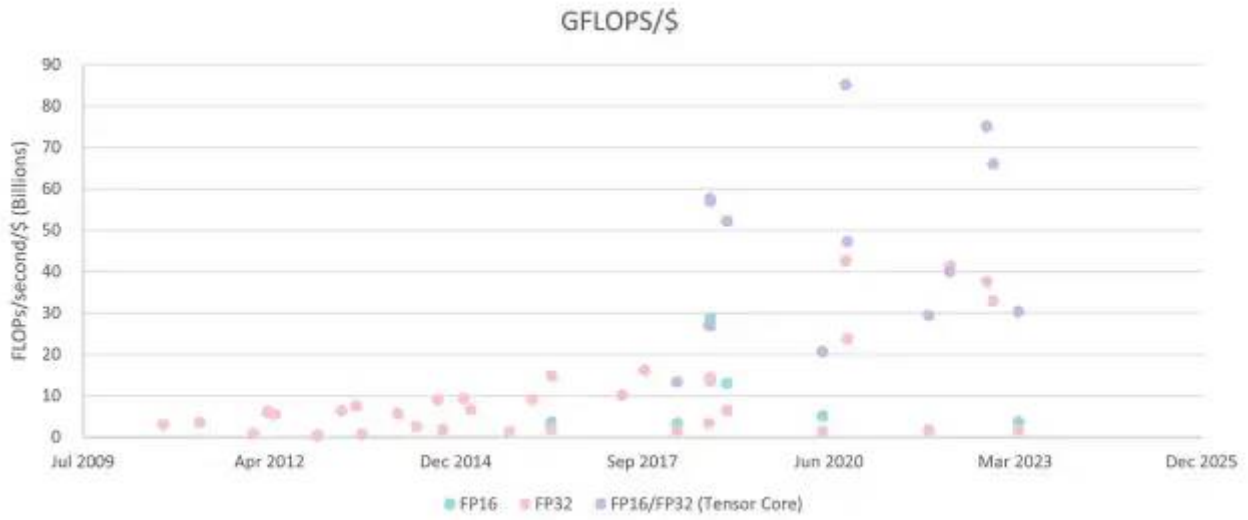
2020年用英伟达V100 GPU训练GPT-3的成本(碳排放与大型神经网络训练)

黄仁勋定律（英伟达首席执行官黄仁勋于2018年提出）指出，在硬件成本方面，GPU的增长速度比五年前快了25倍[23]。在训练LLM的背景下，GPU的性能得到了很大提升，这很大程度上得益于张量核心（Tensor Cores）（AMD采用的是矩阵核心（matrix cores））。此外，GPU不再将矢量作为计算原语，而是转为矩阵，从而实现了性能更好、效率更高的混合精度计算。

2016年，NVIDIA通过V100数据中心GPU首次推出了张量核心。与最初引入的张量核心相比，虽然这一改进不太明显，但是每一代张量核心都进一步提高了吞吐量。如今，对于用于训练LLM的数据中心GPU，我们仍能看到每一代GPU的吞吐量都提升了50%（或者说年均吞吐量提升了22%左右）。

	Launch Year	GFLOPS/$	Improvement
P100	2016	4 (FP16)	
V100	2018	13 (FP16/FP32 Tensor Core)	3.6x
A100	2020	21 (FP16/FP32 Tensor Core)	1.5x
H100 (expected)	2023	30 (FP16/FP32 Tensor Core)	1.5x

数据中心GPU FP16/FP32吞吐量/美元 (NVIDIA）

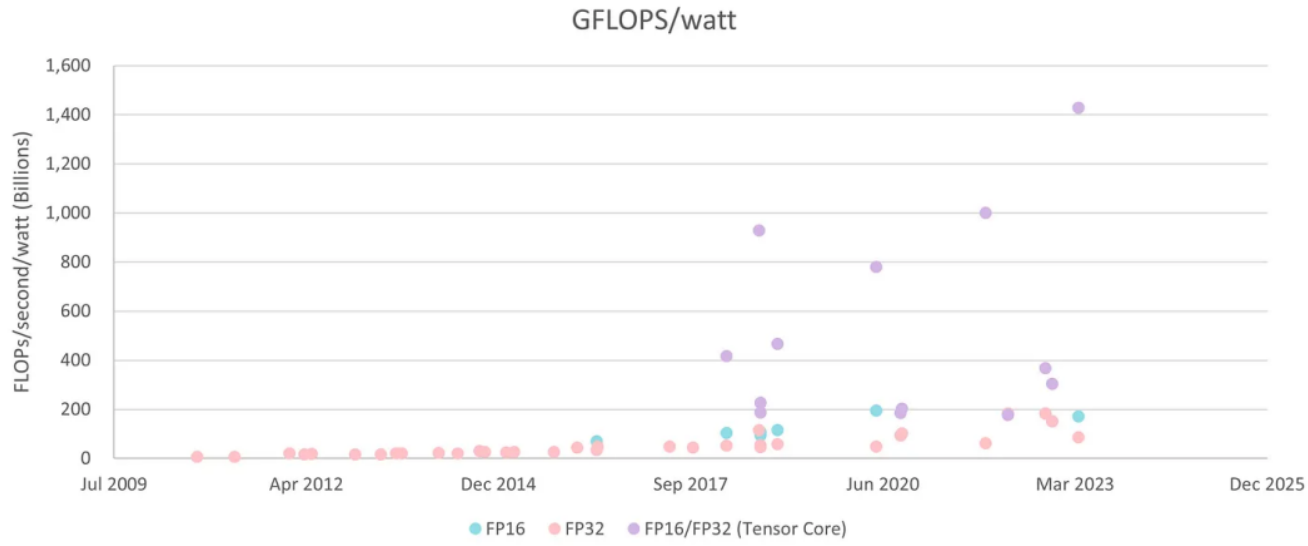


桌面GPU和数据中心GPU、按精度划分的吞吐量/美元 (英伟达，深度学习推理中的计算和能源消耗趋势)

能源效率提升得更快。现在我们可以看到，用于训练LLM的数据中心GPU的代际吞吐量/瓦特提高了80%（或者说年均吞吐量提高了34%）：

	Launch Year	GFLOPS/watt	Improvement
P100	2016	71 (FP16)	
V100	2018	417 (FP16/FP32 Tensor Core)	5.9x
A100	2020	780 (FP16/FP32 Tensor Core)	1.9x
H100 (expected)	2023	1429 (FP16/FP32 Tensor Core)	1.8x

数据中心 GPU FP16/FP32 吞吐量/瓦特 (英伟达)



按精度划分的桌面和数据中心GPU吞吐量/瓦特（英伟达，深度学习推理中的计算和能耗趋势）

> 仅从V100（用于训练 GPT-3）到即将推出的H100的改进来看，我们预计内部训练成本将降低58%（即训练成本由74.4万美元降低到31.2万美元）。

	GPT-3 Training Cost with H100 (In-House)
Hardware Training Cost with V100	$628,080
H100 Throughput/$ Improvement	2.3x
Hardware Cost (Assuming Linear Depreciation)	**$278,204**
Energy Cost with V100	$115,830
H100 Throughput/Watt Improvement	3.4x
Energy Cost	**$33,784**
Total Hardware & Energy Cost	**$311,988**

目前使用英伟达H100 GPU训练GPT-3的成本

展望未来，我们预测，**随着硬件设计的不断创新，硬件成本和能效将逐步改进。**例如，从V100到A100 GPU，NVIDIA添加了稀疏特性（sparsity features），这进一步将某些深度学习架构的吞吐量提高了2倍[24] 。NVIDIA正在H100中添加对FP8数据类型的本地支持，当与推理量化等现有技术相结合时，可以进一步提高吞吐量[25]。

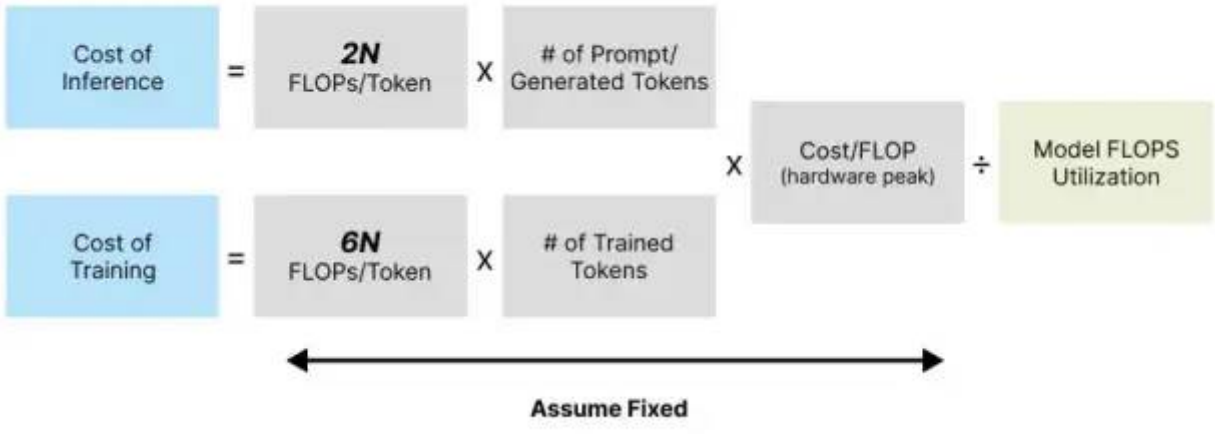
此外，TPU和其他专用芯片的出现从根本上重塑了深度学习用例的芯片架构。谷歌的TPU建立在脉动阵列结构（systolic array architecture）之上，可显著减少寄存器使用，提高吞吐量[26]。正如下一节将提到的，随着我们将训练和推理扩展到大型参数模型，最近许多硬件都着力于提高利用率。

硬件利用率提升

出于内存需求，LLM训练的主要挑战之一就是将这些模型从单个芯片扩展到多个系统和集群级别。在典型的LLM训练中，设置保存优化器状态、梯度和参数所需的内存为20N，其中N是模型参数数量[27]。

因此，BERT-Large（2018年早期的LLM之一，拥有3.4亿参数）仅需6.8GB内存，就可轻松装入单个桌面级GPU。另一方面，对于像GPT-3这样的1750亿参数模型，内存要求转换为3.5TB。同时，NVIDIA最新的数据中心 GPU（H100）仅包含80GB的高带宽内存(HBM)，这表明至少需要44个H100才能满足GPT-3的内存要求。[28]此外，即使在10000个V100 GPU上训练GPT-3也需要14.8天。

因此，即使我们增加用于训练的芯片数量，FLOPS利用率也仍然需要保持高水平，这一点至关重要。



硬件利用率的第一个维度是在单芯片层面。在单个A100 GPU上训练GPT-2模型时，硬件利用率达35.7%[29]。事实证明，**片上内存（on-chip memory）和容量是硬件利用的瓶颈之一**：处理器内核中的计算需要重复访问HBM，而带宽不足会抑制吞吐量。同样，有限的本地内存容量会迫使从延迟较高的HBM进行更频繁的读取，从而限制吞吐量[30]。

硬件利用率的第二个维度与芯片到芯片的扩展有关。训练像GPT-3这样的LLM模型需要跨多个GPU对模型和数据进行划分。正如片上存储器的带宽可能成为硬件利用的瓶颈一样，**芯片间互连的带宽也可能成为硬件利用的限制因素。**随着V100的发布，NVIDIA的NVLink实现了每个GPU 300GB/s的带宽。对于A100来说，宽带速度实现了600GB/s[31]。

硬件利用率的最后一个维度是系统到系统的扩展。一台机器最多可容纳16个GPU，因此扩展到更多数量的GPU要求跨系统的互连不能成为性能瓶颈。为此，Nvidia的Infiniband HCA在过去3年中将最大带宽提高了2倍[32]。

在第二和第三个维度上，**软件划分策略是硬件有效利用的关键考虑因素。**通过结合模型和数据并行技术，2022年使用MT-NLG的Nvidia芯片集群级别的LLM训练的模型FLOPS利用率达到了30.2%[33]，而使用GPT-3的模型FLOPS利用率在2020年只有21.3%：

	Year Trained	# of Parameters	Accelerator Chips	Model FLOPS Utilization
GPT-3	2020	175B	10,000 x Nvidia V100	21.3%
Gopher	2022	280B	4,096 x Google TPU v3	32.5%
MT-NLG	2022	530B	2,240 x Nvidia A100	30.2%
PaLM	2022	540B	6,144 x Google TPU v4	46.2%

选择LLM的模型FLOPS利用率（PaLM：使用路径扩展语言建模）

TPU等专用硬件实现了更高的效率。

> 谷歌5400亿参数的PaLM模型在TPU v4芯片上实现了46.2%的模型FLOPS利用率，是GPT-3训练利用率的2.2倍[34]

FLOPS利用率的提高得益于更高效的并行训练（使用Google的Pathways ML系统）以及从根本上TPU具有完全不同的架构。该芯片的脉动阵列结构和每个内核的显著的本地内存密度（local memory density）降低了高延迟全局内存（global memory）的读取频率。

同样地，我们可以看到**Cerebras**、**Graphcore**和SambaNova等公司在处理器中分配了更多的共享内存容量。展望未来，我们预计其他新兴创新，例如将芯片扩展到晶圆级以减少延迟/增加带宽，或通过可编程单元优化数据访问模式等将进一步推动硬件利用率的发展[35]。

6
大型语言模型即将迎来全盛时期

据《纽约时报》近日报道，谷歌宣称ChatGPT是其搜索业务的“红色警报”（code red），它的搜索量呈病毒式发展。
[36]**从经济角度来看，通过粗略估算，将高性能LLM纳入搜索将花费约15%的查询收入，这表明该技术的部署已经切实可行。**然而，谷歌的市场主导地位阻碍了它成为这方面的先行者：谷歌目前的搜索收入为1000亿美元，将高性能LLM纳入搜索会使谷歌的盈利能力减少一百多亿美元。

另一方面，也就难怪微软会计划将大语言模型纳入Bing了[37]。尽管LLM支持的搜索成本高于传统搜索，并且与谷歌相比，微软搜索引擎的市场份额要低得多，但是微软并未亏损。因此，如果微软能够成功地从谷歌手中夺取搜索市场份额，那么即使现有查询成本更高，微软仍然能够获得极高的利润。

有趣的是，**对于其他产品，通过部署LLM已经可以通过SaaS来盈利。**例如，最近估值为15亿美元、使用LLM生成文案的Jasper.ai收费为82美元/100000字（相当于1.09美元/1000个token）[38]。使用OpenAI的Davinci API 定价为 0.02美元/1000个token，即使我们对多个响应(response)进行采样，毛利率也可能远高于75%。

同样令人惊讶的是，如今在公有云中仅需约140万美元即可对GPT-3进行训练，而且即使是SOTA模型（如PaLM，约1120万美元）的训练成本也不会太高。在过去的两年半里，类似GPT-3等模型的训练成本下降了80%以上，高性能大语言模型的训练成本将进一步降低。

换句话说，**训练大语言模型并不便宜，但也没那么烧钱，训练大语言模型需要大量的前期投入，但这些投入会逐年获得回报。**更近一步，Chinchilla论文表明，在未来，**相比资金，高质量数据会成为训练LLM的新兴稀缺资源之一，因为扩展模型参数数量带来的回报是递减的。**

参考文献（请上下滑动）

<section class="" data-style="max-width: 100%; box-sizing: border-box; font-family: -apple-system, BlinkMacSystemFont, " helvetica="" neue",="" "pingfang="" sc",="" "hiragino="" sans="" gb",="" "microsoft="" yahei="" ui",="" yahei",="" arial,="" sans-serif;="" letter-spacing:="" 0.544px;="" white-space:="" normal;="" background-color:="" rgb(255,="" 255,="" 255);="" font-size:="" 16px;="" overflow-wrap:="" break-word="" !important;"="">

1. Alphabet 2021 10K
2. Comparing Google and ChatGPT
3. Microsoft and OpenAI Working on ChatGPT-Powered Bing in Challenge to Google
4. Introducing YouChat - The AI Search Assistant that Lives in Your Search Engine
5. Google Calls In Help From Larry Page and Sergey Brin for A.I. Fight
6. ChatGPT: Optimizing Langauge Models for Dialogue（实际上，ChatGPT还在基础1750亿参数语言模型之上使用了RLHF（Reinforcement Learning from Human Feedback，即从反馈中获得强化学习）机制，但为了简单起见，我们不考虑强化学习成本。）
7. Teaching language models to support answers with verified quotes
8. ChatGPT: Optimizing Langauge Models for Dialogue

9. OpenAI Pricing
10. Building Software Systems at Google and Lessons Learned
11. What's New With Google Search
12. Our World in Data: Internet
13. Alphabet 2020 10K
14. Scaling Laws for Neural Language Models（对于encoder-decoder模型，推理FLOPs约为N，而不是仅解码器模型的2N）
15. AWS EC2 P4 Instances
16. NVIDIA A100 Tensor Core GPU Architecture
17. Mixed precision training（针对FP16/FP32描述的所有内容也适用于BF16/FP32混合精度运算，这些运算在A100和其他处理器上具有类似的吞吐量）
18. PaLM: Scaling Langauge Modeling with Pathways
19. Cloud TPU pricing
20. Scaling Laws for Neural Language Models（对于encoder-decoder模型，训练FLOPS约为3N，而不是仅解码器模型的6N）
21. PaLM: Scaling Langauge Modeling with Pathways
22. Carbon Emissions and Large Neural Network Training
23. GTC 2018 Keynote with NVIDIA CEO Jensen Huang
24. NVIDIA A100 Tensor Core GPU Architecture
25. NVIDIA Hopper Architecture In-Depth
26. An in-depth look at Google's first Tensor Processing Unit (TPU)
27. Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model（假设基于使用混合精度训练的Adam优化器，每个参数占用20字节的内存）
28. NVIDIA Hopper Architecture In-Depth
29. State-of-the-Art Language Modeling Using Megatron on the NVIDIA A100 GPU
30. Which GPU(s) to Get for Deep Learning: My Experience and Advice for Using GPUs in Deep Learning
31. NVLink and NVSwitch
32. NVIDIA ConnectX InfiniBand Adapters
33. PaLM: Scaling Langauge Modeling with Pathways
34. PaLM: Scaling Langauge Modeling with Pathways
35. Cerebras Architecture Deep Dive: First Look Inside the HW/SW Co-Design for Deep Learning
    Graphcore IPU Hardware Overview
    SambaNova SN10 RDU at Hot Chips 33
36. A New Chat Bot is a 'Code Red' for Google's Search Business
37. Microsoft and OpenAI Working on ChatGPT-Powered Bing in Challenge to Google
38. Jasper.ai Pricing