

## DeepSeek V3/R1架构的 深度分析与深度思考

陈巍

2025.02

# 报告人简介

陈巍 博士 领域：大模型+AI芯片

- 陈巍博士，大模型+AI芯片专家，高级职称，中存算董事长。国际计算机学会（ACM）、中国计算机学会（CCF）专业会员，多个国际人工智能期刊审稿人。
- 主要研究方向为大模型架构、稀疏量化压缩与部署加速，存算一体与3D Chiplet处理器，相关技术成果已被广泛应用于知名IDC和互联网企业。
- 曾任领域知名人工智能（自然语言处理）企业首席科学家，中国科学院副主任（2012），多个国家科技重大专项课题负责人。中国与美国发明专利软件著作权约70+项（约50+项发明专利已授权）。著有《Sora大模型技术精要—原理、关键技术、模型架构与未来趋势》《GPT-4大模型硬核解读》《DeepSeek是否有国运级的创新？2万字解读与硬核分析DeepSeek V3/R1的架构》等。

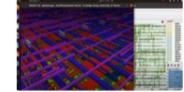
陈巍：DeepSeek V3/R1架构的深度分析与深度思考

## GPT-4/ChatGPT技术与产业分析

陈巍谈芯 · 9 篇内容

[收录内容](#) [修改介绍](#) ...

陈巍：GPT大模型攻克先进32位流水线RISC-V芯片设计难题（收录于：GPT-4/ChatGPT技术与产业分析）



▲ 赞同 46 ● 8 条评论 分享 收藏 ...

陈巍：GPT-4核心技术分析报告（2）——GPT-4的技术分析（收录于GPT-4/ChatGPT技术与产业分析）

## 《先进存算一体芯片设计》

陈巍谈芯 · 5 篇内容

[收录内容](#) [修改介绍](#) ...

## 阿里达摩院最新存算芯片技术解读



陈巍谈芯：12月3日，阿里达摩院成功研发新型架构芯片，已经被证明能够在阿里推荐系统中发挥极大的应用价值，并受到技术圈的普遍关注。据悉这款芯片与数据中心的推荐系统对于带宽/存储的需求完美匹配，大幅提升带... 阅读全文

▲ 赞同 21 ● 添加评论 分享 收藏 ...

大算力可重构存算一体芯片的架构设计与编译器开发  
存算一体芯片专场第二讲

陈巍

芯云科技  
董事长

1. 大算力与大模型计算的瓶颈与挑战
2. 存算一体的技术优势与设计难点
3. 前向云计算与边缘计算的大算力存算一体芯片
4. 可重构存算一体技术的设计挑战与编译器开发

直播时间：  
5月19日晚7点  
[添加直播助手报名](#)

2023 全球  
AI  
峰

陈巍

芯云科技  
董事长

- 面向大模型部署训练的  
存算一体架构与多处理器融合

2023.9.15  
演讲主题：  
陈巍

NPCON 2023  
AI 模型技术与应用峰会  
北京国家会议中心大酒店

陈巍

芯云科技  
董事长

- “想搞大模型，架构  
和算力怎么配置？”

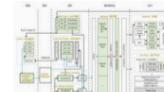
演讲主题：  
陈巍

## 《GPGPU 芯片设计：原理与实践》

陈巍谈芯 · 8 篇内容

[收录内容](#) [修改介绍](#) ...

陈巍谈芯：3（上）GPGPU流式多处理器架构之取指指令发射——《GPGPU 芯片设计：原理与实践》节选



欢迎对技术感兴趣的加入 GPU 讨论小组 这一章是讲的是所有 GPU 的核心，流式多处理器，以英伟达架构为基准。（由于某些出口管制的限制，估计 GPU 的 SM 核心也会被卡）除了编译算法和 Tensor Core 之外，... 阅读全文

▲ 赞同 122 ● 8 条评论 分享 收藏 ...

陈巍：2（上）GPGPU编程模型之CUDA——《GPGPU 芯片设计：原理与实践》节选

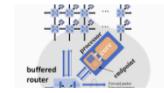
## 《AI芯片设计：原理与实践》

深入讲述AI算法、AI加速电路、AI芯片架构和知识

陈巍谈芯 · 4 篇内容

[收录内容](#) [修改介绍](#) ...

陈巍：RISC-V AI芯片Celerity史上最详细解读（上）（附开源地址）



在Hot Chips 29大会上，基于RISC-V核心的AI芯片Celerity一亮相便引起开源社区的关注。本文包括Celerity中二值化神经网络的介绍。Celerity的设计展现了两个特点：1) ... 阅读全文

▲ 赞同 71 ● 添加评论 分享 收藏 ...

2024 中国  
生成式  
AI  
大会

陈巍

芯云科技  
董事长

- 视频大模型架构对比及  
长序列模型加速

演讲主题：  
陈巍

# 大语言模型的本质

- 知识，是指人类在实践中认识（计算）客观世界的成果
- 大语言模型本质上是“对知识的压缩（存）+对输入的反馈（算）”

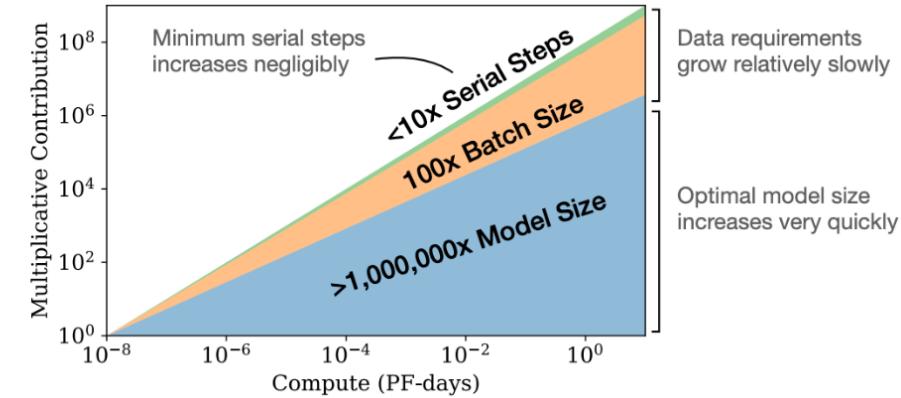
模型能力 = 模型大小  $\times$  压缩倍数  $\times$  反馈能力系数

or 模型能力 = (模型大小  $\times$  压缩倍数 + 向量数据库大小)  $\times$  反馈能力系数

- Knowledge is computing power (知识就是算力)
- 视频大模型的性能与其“知识”体量密切相关

本人2024年12月的公开预测：

- 视频大模型参数量未来会达到主流文本大模型的1-3个数量级
- 3年内日常生活算力需求至少提升1个数量级（提前实现？）
- 拥有数据最多的企业将是具身智能机器人企业



缩放定律图示（来源：OpenAI）



在2022年5月，提前半年预测大模型的产业爆发

- 1 Scaling law与Moore's law的范式共生
- 2 DeepSeek的架构提升与既要又要
- 3 V3/R1训练架构与软硬件协同进化
- 4 V3/R1训练流程与RL变幻莫测的潜力
- 5 从DeepSeek到算力竞赛的深度思考

# 回顾Scaling Laws

Scaling Laws，也称为缩放定律或规模定律，是描述模型性能如何随着**模型规模**（如参数数量）、**训练数据集大小**以及用于训练的**计算资源增加**而提升的一组经验法则

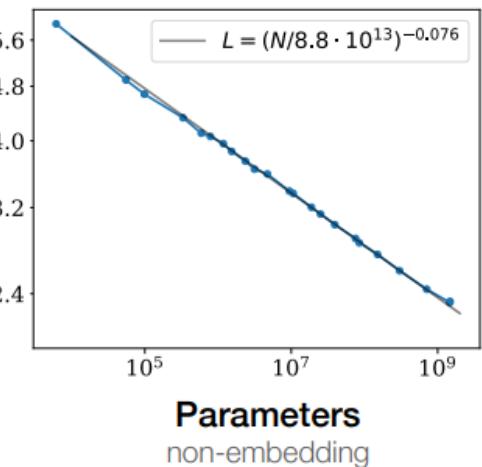
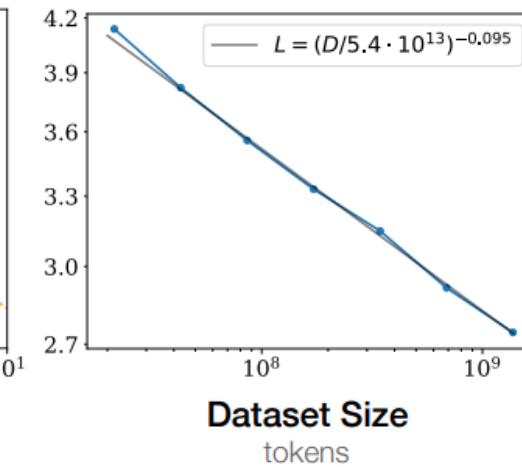
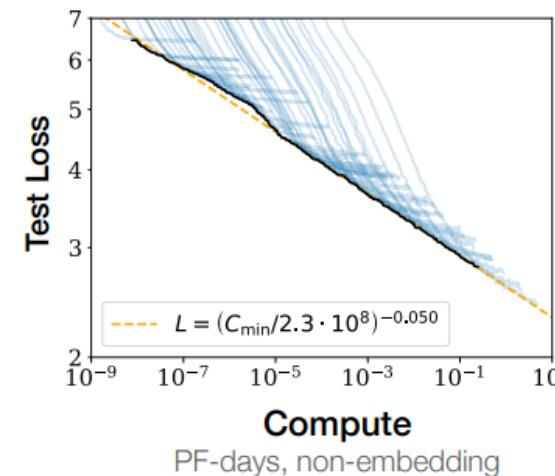
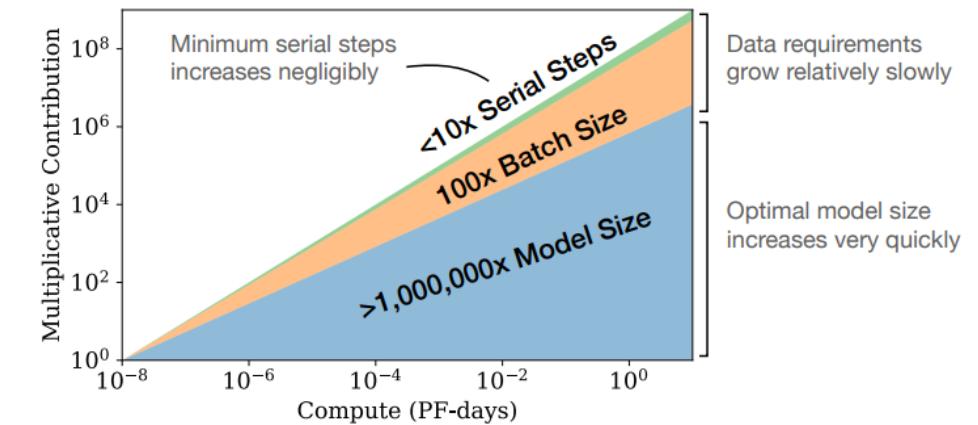
模型大小与性能的关系

数据量与性能的关系

计算量与性能的关系

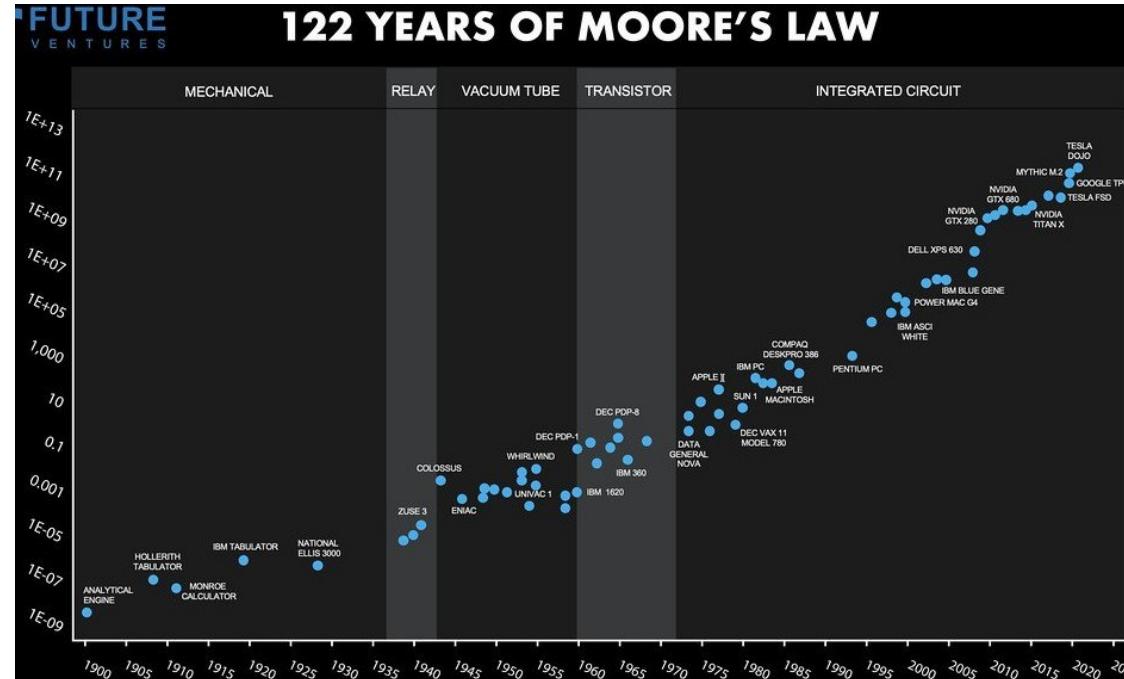
性能的可预测性

模型结构的影响



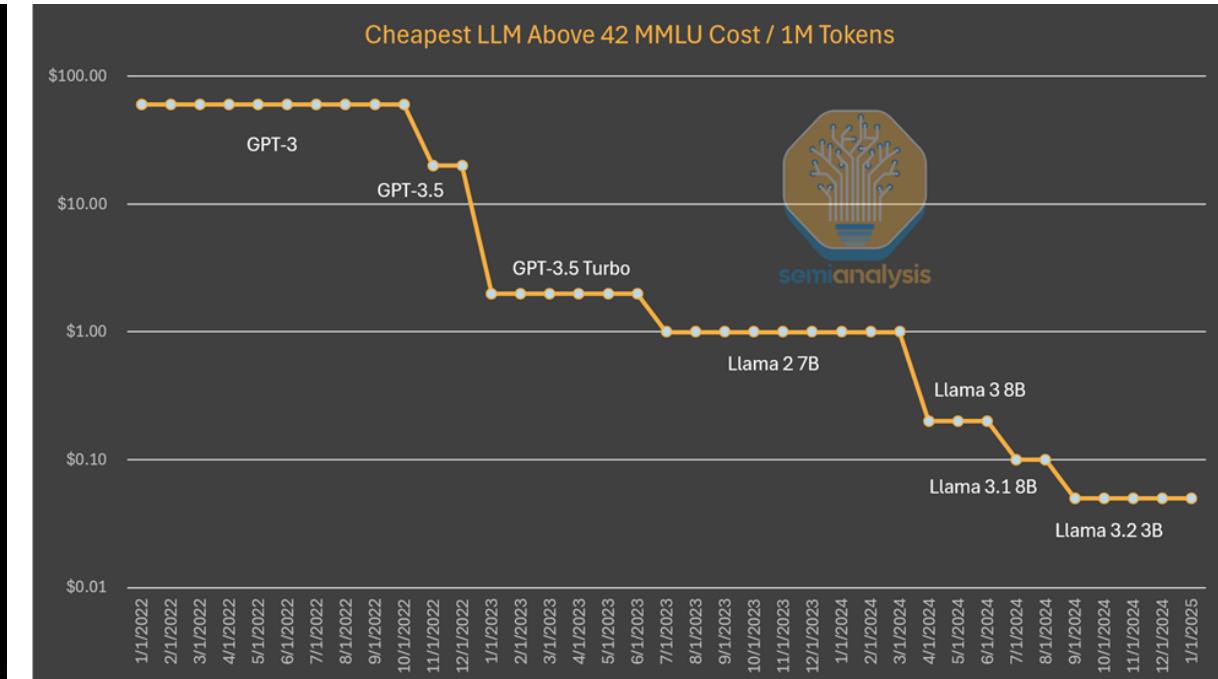
Scaling Laws图示（来源：OpenAI）

# Moore's law与模型算法的进步



摩尔定律图示 (来源: Unite.AI)

集成电路上可以容纳的晶体管数目在大约每经过18个月到24个月便会增加一倍。换言之，处理器的性能大约每两年翻一倍，同时价格下降为之前的一半

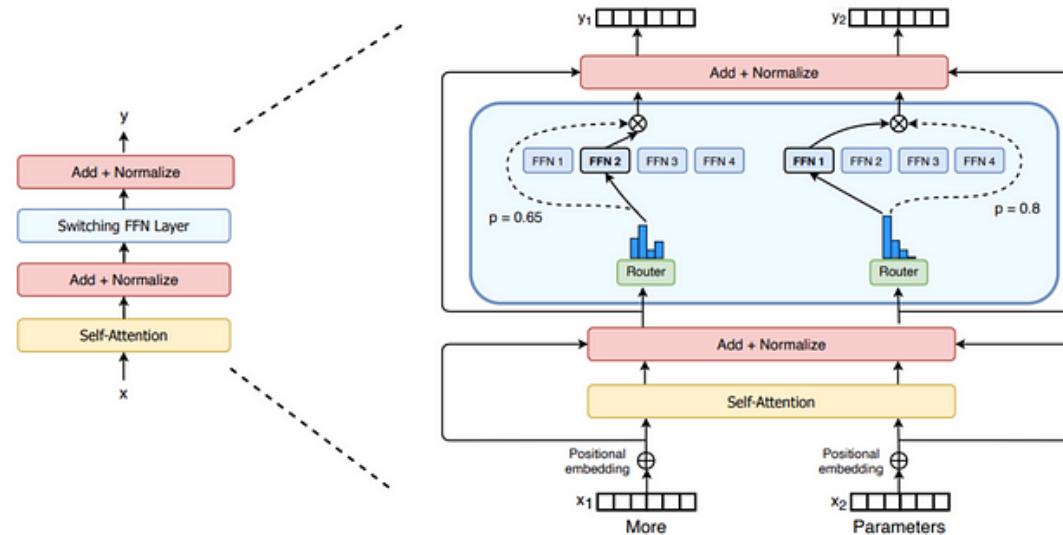


大模型成本的下降 (来源: semianalysis)

每年算法进步的速度为4倍，意味着每过一年，实现相同能力所需的计算资源减少4倍

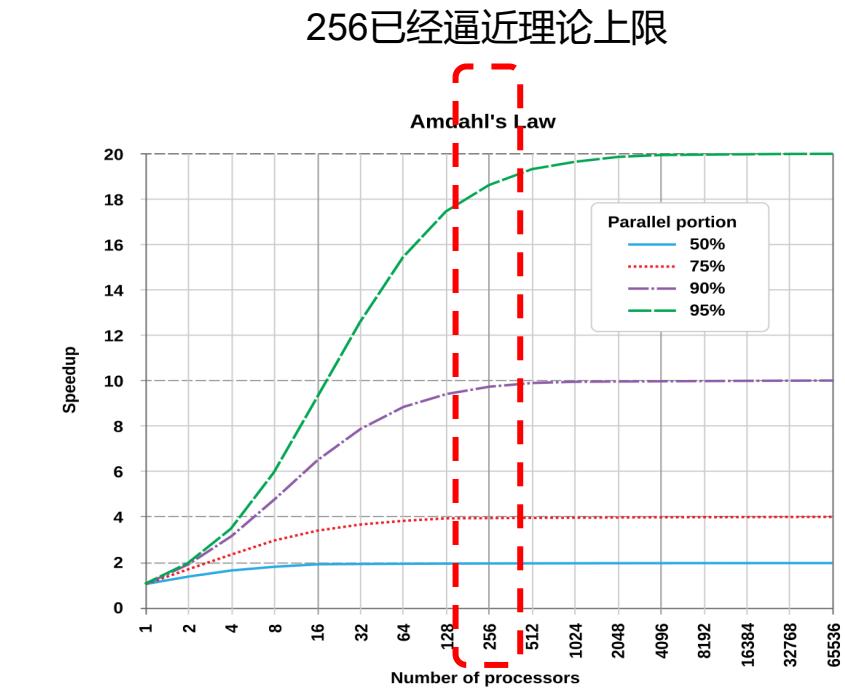
事实上等效于 (Moore's law)<sup>2</sup>

# MoE与Amdahl's Law (DeepSeek是否受到芯片大小核思路影响?)



MoE模型基本结构示意 (来源: 互联网)

MoE (Mixture of Experts) 是组合多个专家模型提升深度学习模型性能和效率的架构。其核心思想是通过引入多个专家模型 (Experts)，每个输入数据只选择和激活其中的一部分专家模型进行处理，从而减少计算量，提高训练和推理速度。MoE的概念在1991年就已提出，训练不容易收敛是MoE在大模型领域应用的主要障碍。



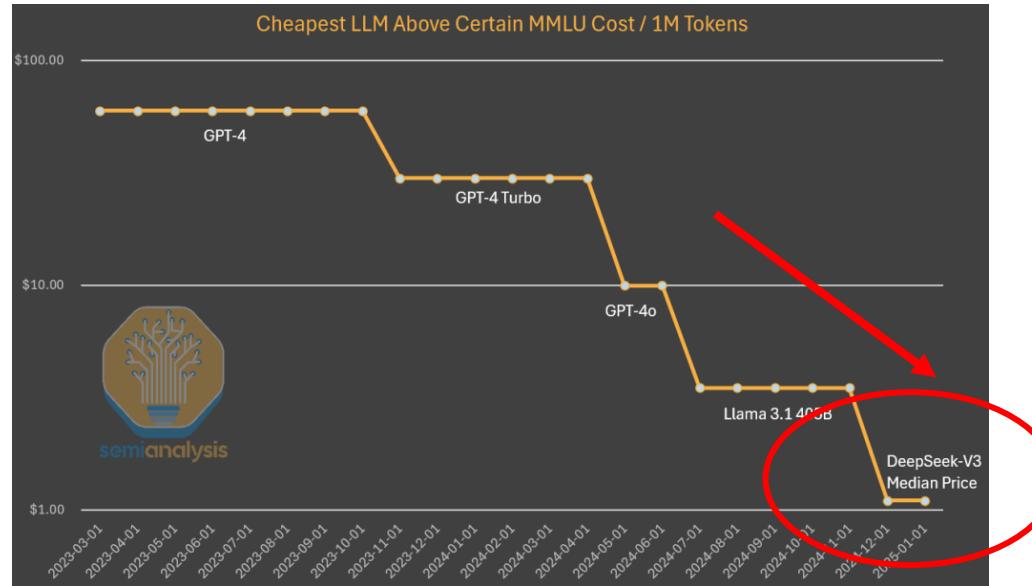
Amdahl's Law (芯片界的MoE思路)

计算负载是固定不变的。为此在一定的计算负载下，为达到实时性可利用增加处理器数来提高计算速度/性能

$$S = 1 / ((1 - a) + a/n)$$

S为 speedup，代表全局加速倍速 (原来总时间/ 加速后总时间)， $a$  为并行计算所占比例 (可以并行计算代码量 / 总代码量)， $n$  为并行节点处理器个数，可以理解为处理器的核心数

# DeepSeek的进步（对开源社区的关键贡献）



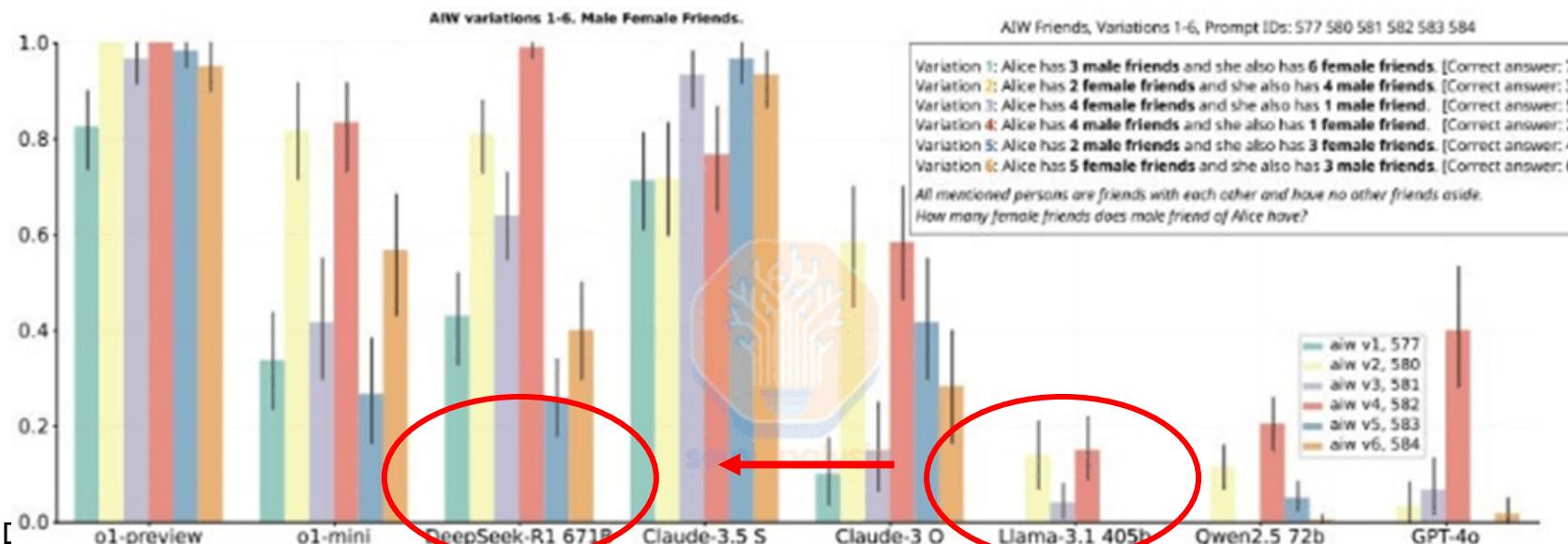
Dario Amodei (Anthropic CEO) : Shifting the curve (改变曲线)

(Moore's law)<sup>2</sup>

成本降低

以往的每次成本降低并未冲击英伟达市值

英伟达市值大降是否与DeepSeek直接关联？



Dario Amodei : Shifting the paradigm (转变范式)

Amdahl's Law

在DeepSeek R1之前这一范式转移已经发生

开源领域则是第一次

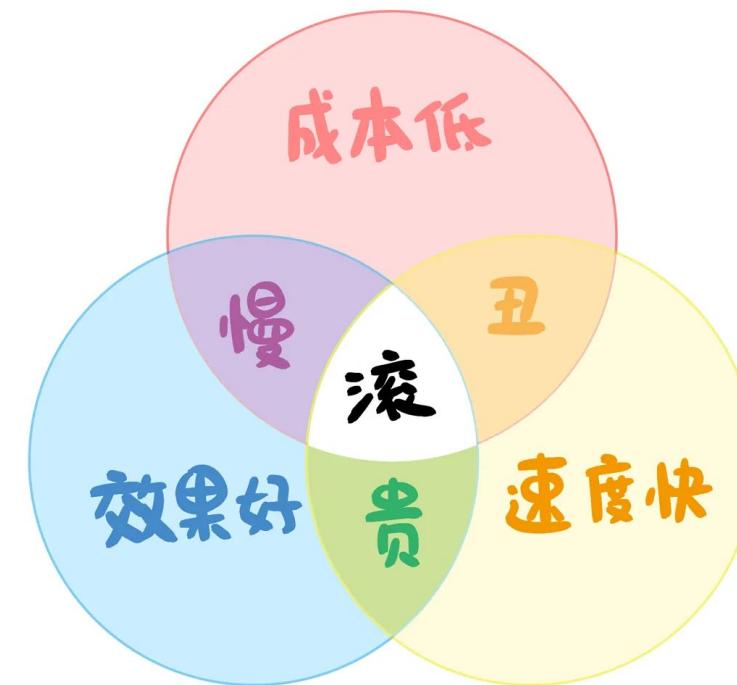
- 1 Scaling law与Moore's law的范式共生
- 2 DeepSeek的架构提升与既要又要**
- 3 V3/R1训练架构与软硬件协同进化
- 4 V3/R1训练流程与RL变幻莫测的潜力
- 5 从DeepSeek到算力竞赛的深度思考

# 既要又要与唯快不破

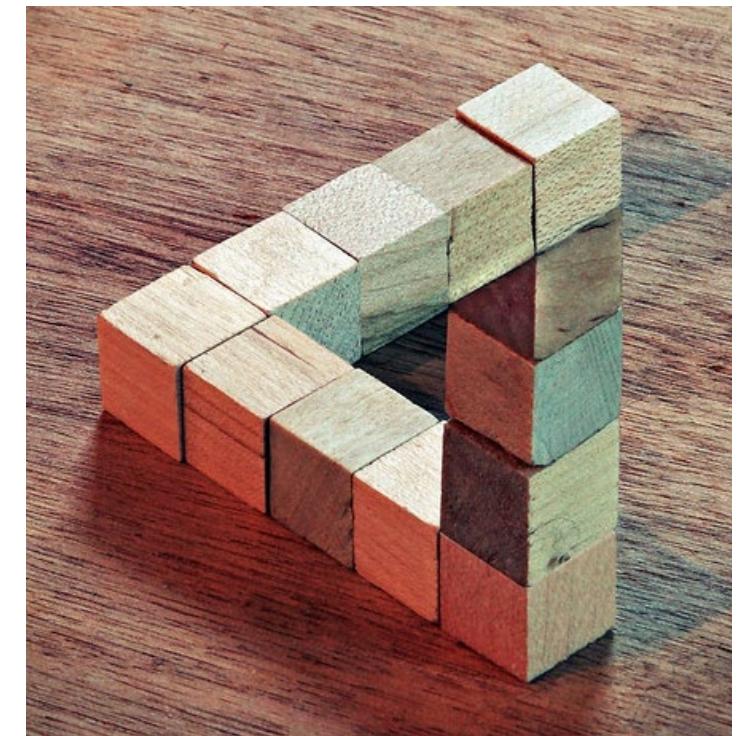
更高性能的模型

更好训练的模型

更低成本的模型



炼丹师的痛苦根源 (来源：互联网)



不可能三角 (来源：互联网)

# DeepSeek-V3/R1

## DeepSeek-V3/R1

DeepSeek-V3 是一个混合专家 (MoE) 语言模型，具有6710亿(671B)参数，其中每个Token (词元) 的计算约激活370亿(37B)参数。这个模型参数量与GPT-4大致在同一数量级

DeepSeek-R1的模型架构来自于V3，甚至可以说R1是具有推理 (Reasoning) 能力的V3

## R1模型族

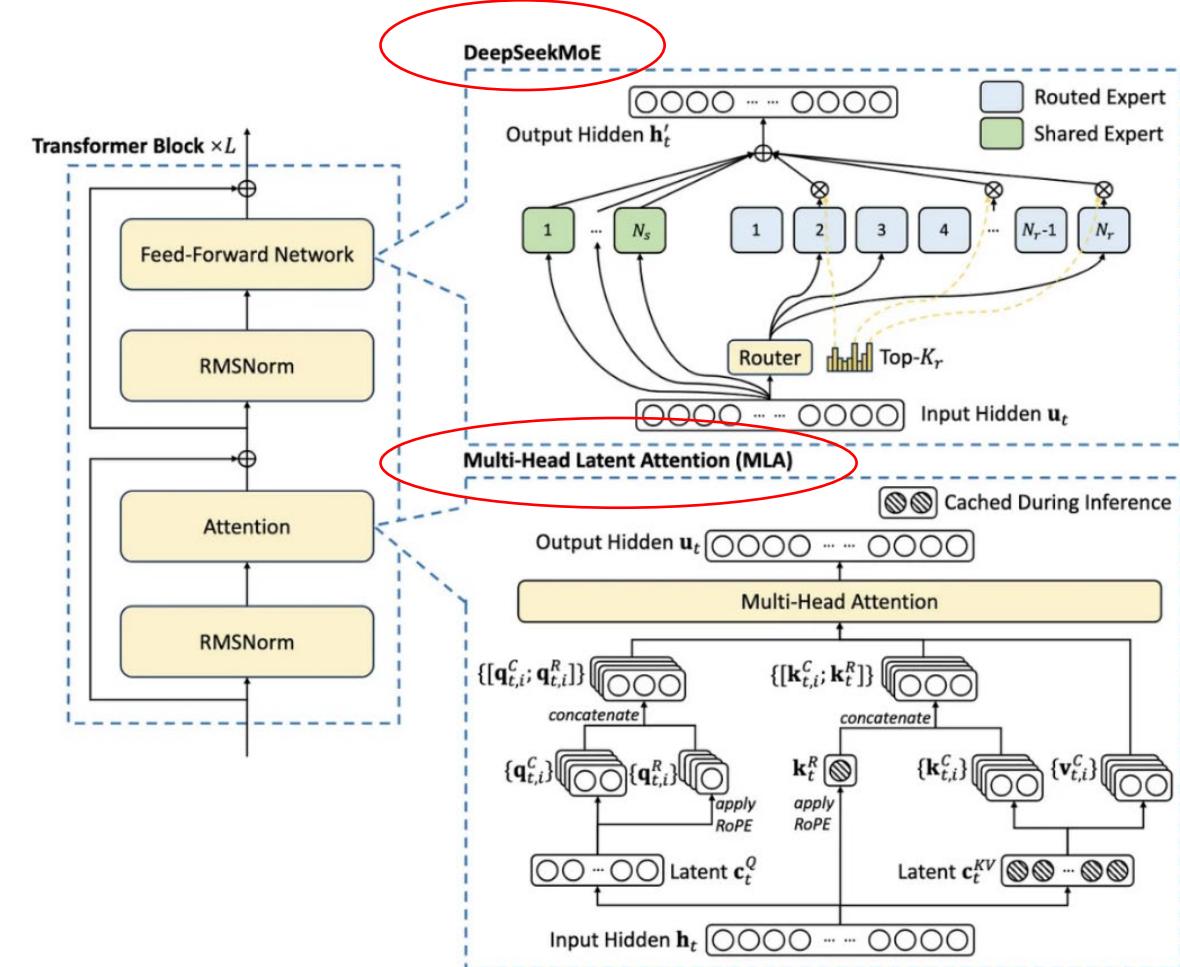
广义上的DeepSeek-R1 不是一个单一的模型，还包括了R1的初始阶段模型DeepSeek-R1-Zero，以及几个基于R1蒸馏的较小的大模型

## 主要架构特点

采用了多头潜注意力 (MLA，对传统多头注意力机制的改进)

DeepSeekMoE 架构 (对传统MoE架构的改进)

无辅助损失的负载平衡策略等创新技术



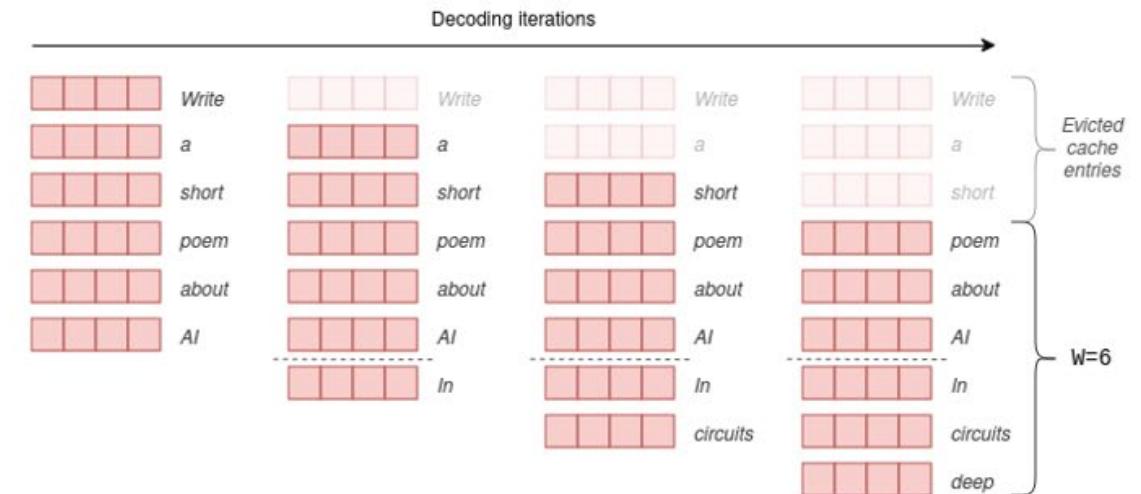
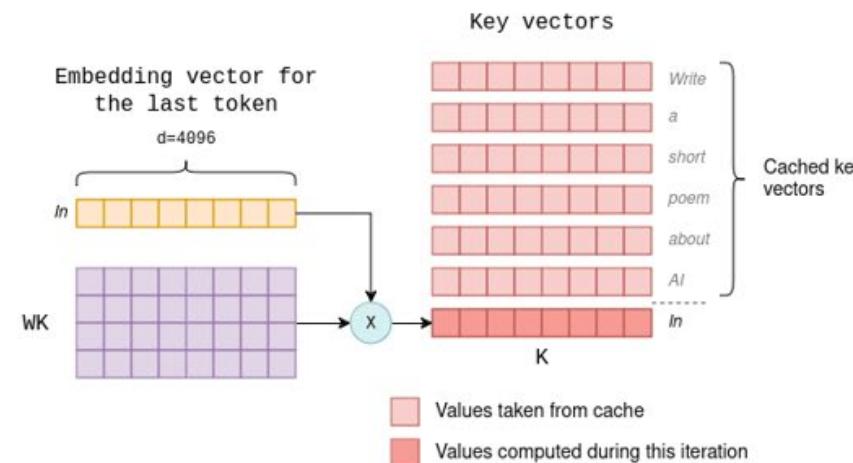
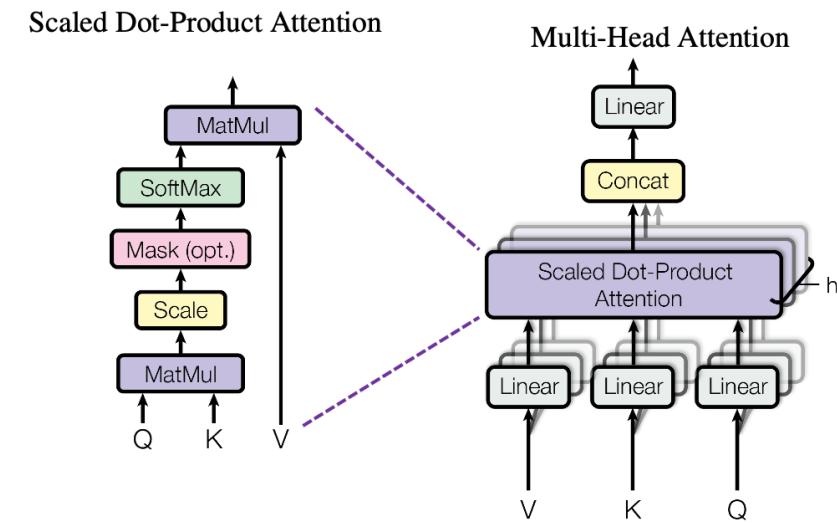
DeepSeek-V3整体架构 (来源: DeepSeek)

# MLA：从KV Cache (KV缓存) 说起

## Why KV Cache

当使用传统Transformer在推断 (Inference) 过程中计算并生成Token (词元) 时，模型需要读入所有过去Token的上下文，以决定接下来输出什么Token

目前常用的方法是缓存所有过去Token的相关内部状态，主要是注意力机制中的键 (Key) 和值 (Value) 向量。这也是键值缓存 (简称KV缓存，也就是常说的KV Cache) 名称的由来

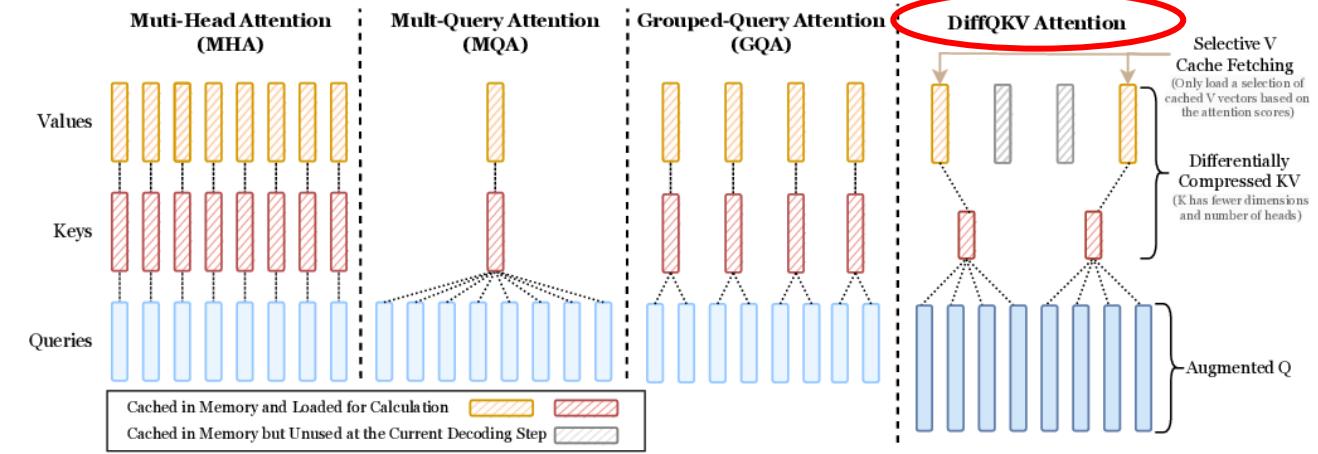


KV Cache (来源：互联网)

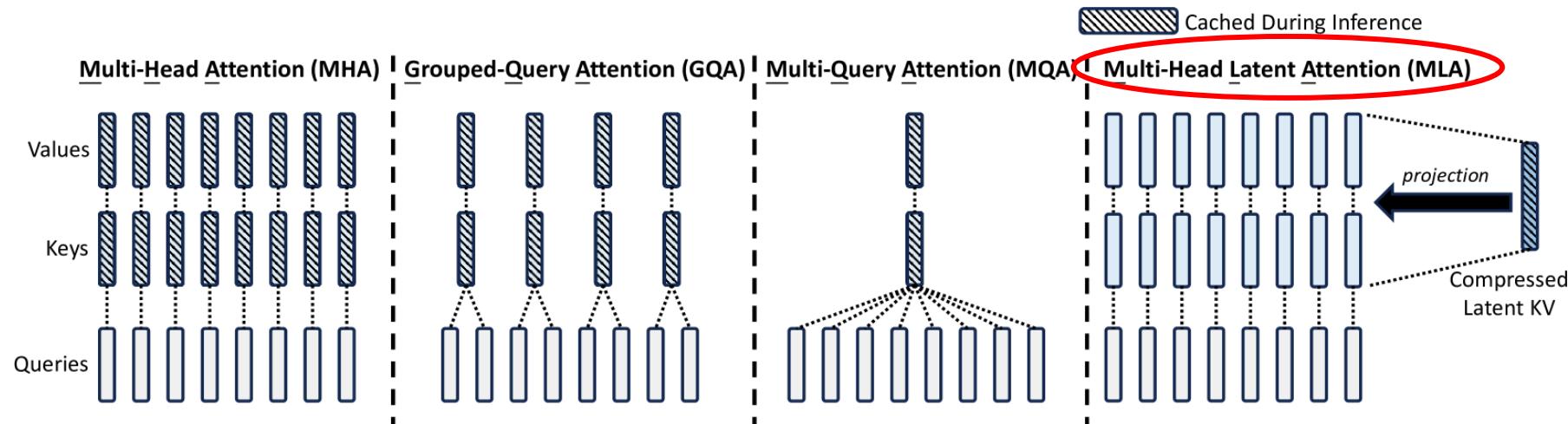
# 不同注意力机制对比

## GQA

目前开源大模型中的主流方法是分组查询注意力 (Grouped-Query Attention) 机制。在这种机制中，为每对键和值头分配多个查询头，将查询头有效的分组在一起。在Llama 3.3 70B 和 Mistral Large 2 等模型中，仅分组查询注意力机制就将 KV 缓存大小减少了大约一个数量级

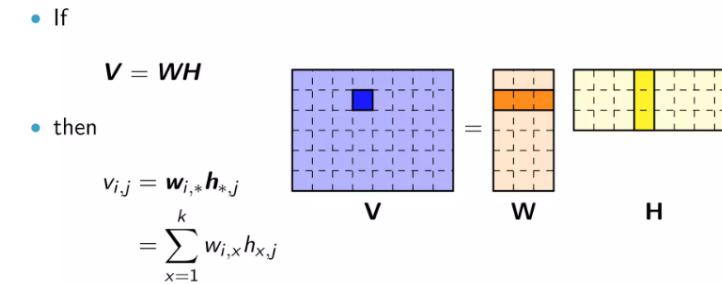
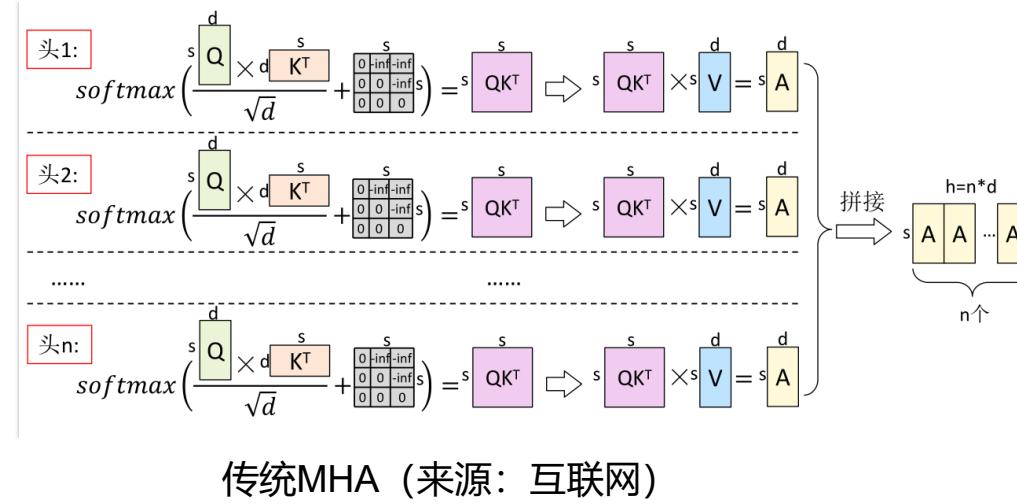


不同注意力机制的对比 (来源: 微软)



不同注意力机制的对比 (来源: DeepSeek V2)

# Multi-Head Latent Attention



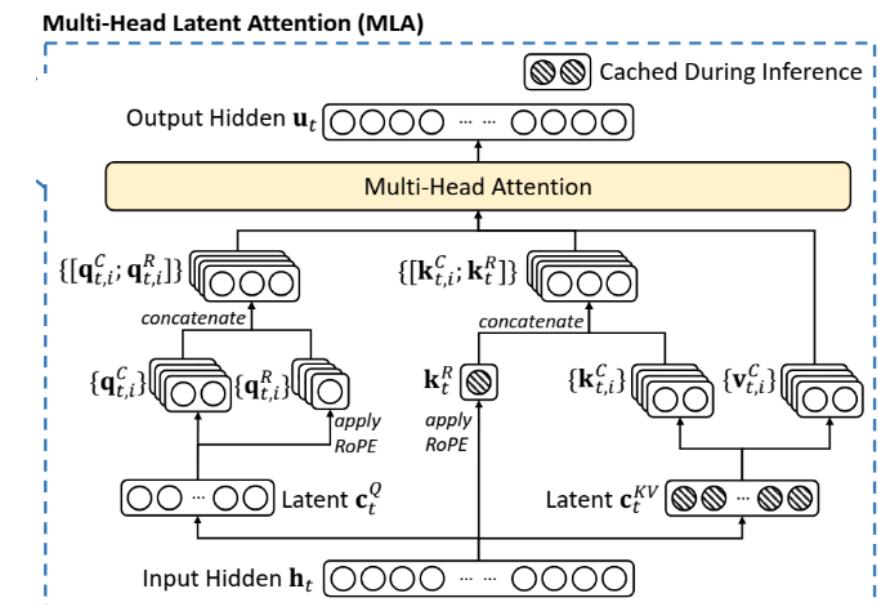
## 低秩形式

MLA的方法是将KV矩阵转换为低秩形式，将原矩阵表示为两个较小矩阵（相当于潜向量）的乘积，在推断过程中，仅缓存潜向量，而不缓存完整的键KV。这规避了分组查询注意力和多查询注意力的查询的信息损失，从而在降低KV缓存的前提下获得更好的性能

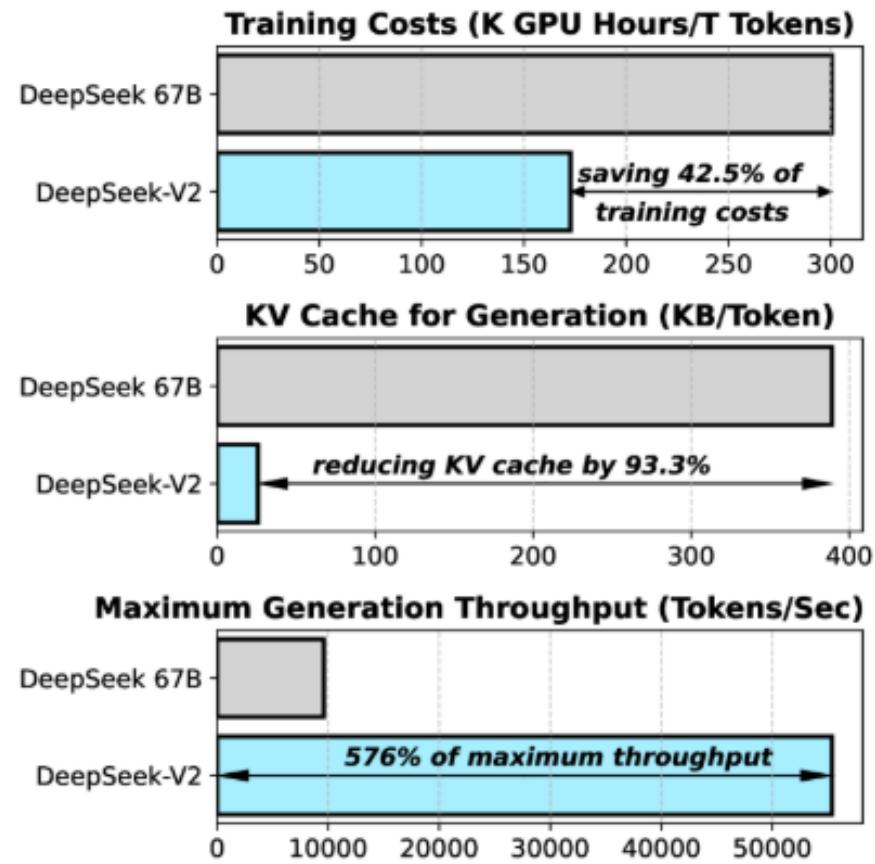
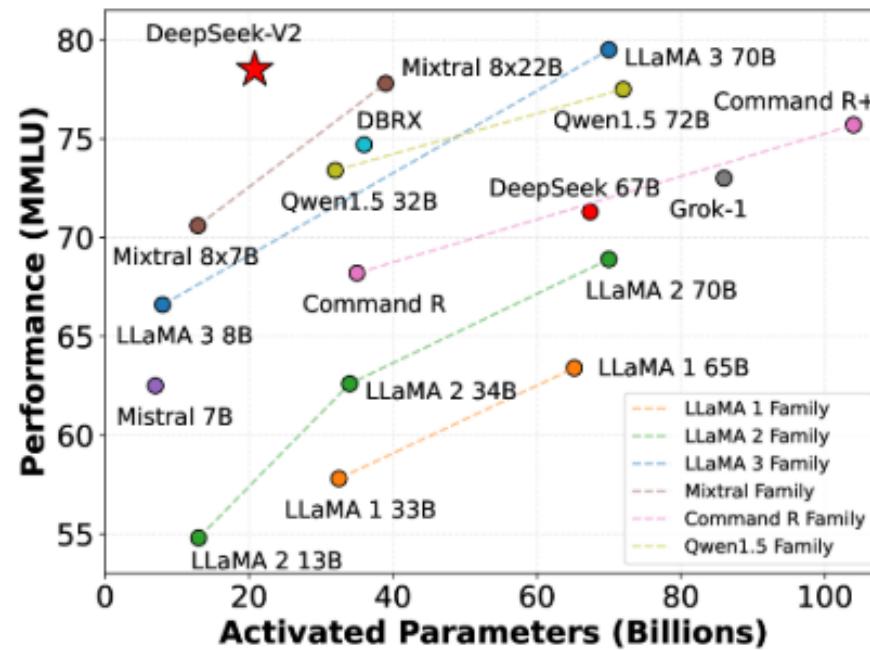
## MLA的本质

是对KV的有损压缩，提高存储信息密度的同时尽可能保留关键细节。

该技术首次在DeepSeek-V2中引入，与分组查询和多查询注意力等方法相比，MLA是目前开源模型里显著减小KV缓存大小的最佳方法



MLA (来源: DeepSeek)



MLA方法有效降低KV Cache和训练成本 (来源: DeepSeek)

# 其他KV Cache优化方法

## Linear-Time Sequence Models and Mixed Architecture

- 线性时间序列模型，例如 Mamba、RWKV，也叫KV Cache Free 方法
- 线性时间序列模型和 Transformer 结合的方法，例如 Griffin

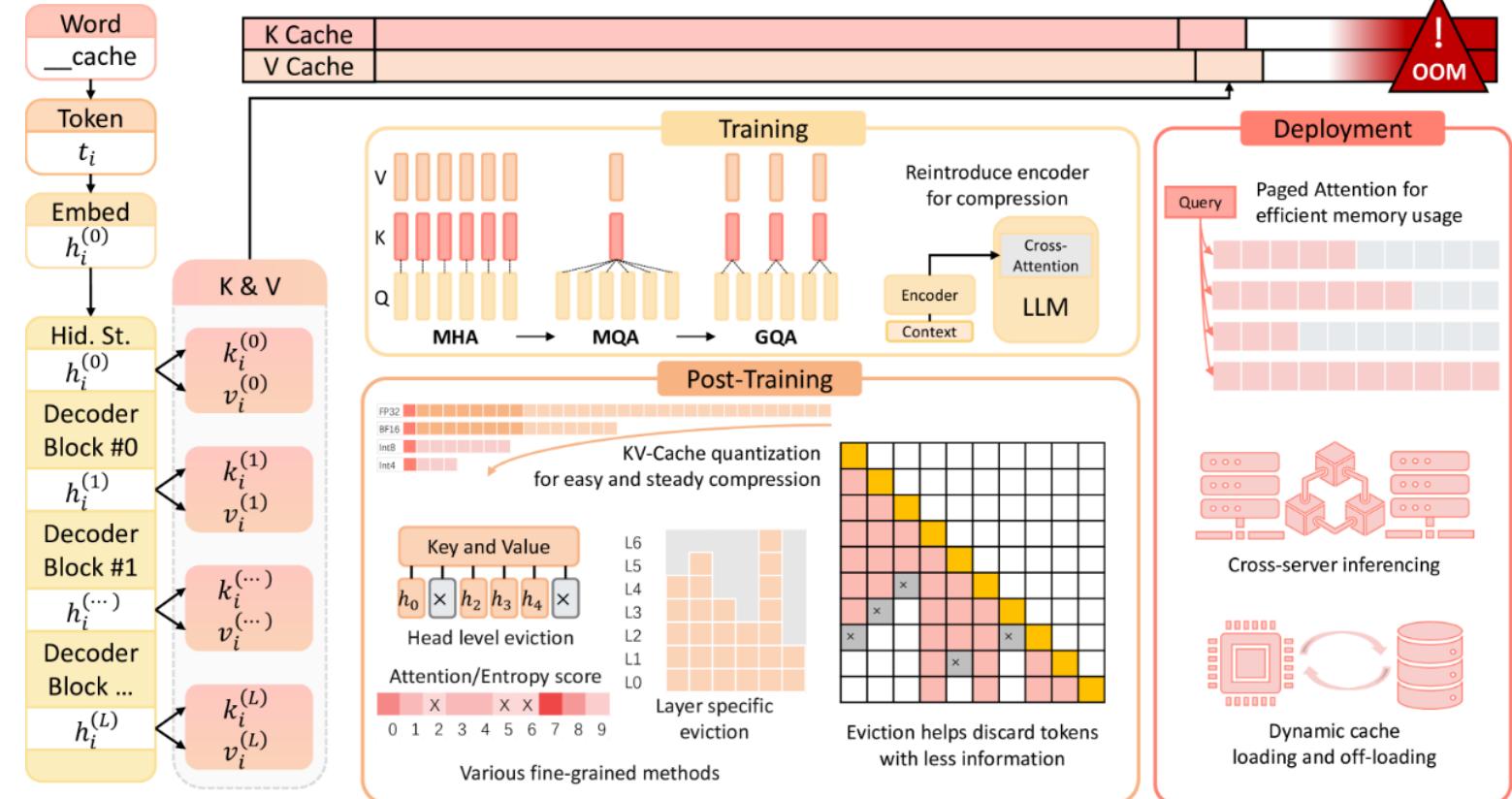
## Quantization

### Token Dropping

- 主要分为 prefill 期间丢弃令牌和在 prefill 后丢弃令牌

### Prompt Compression

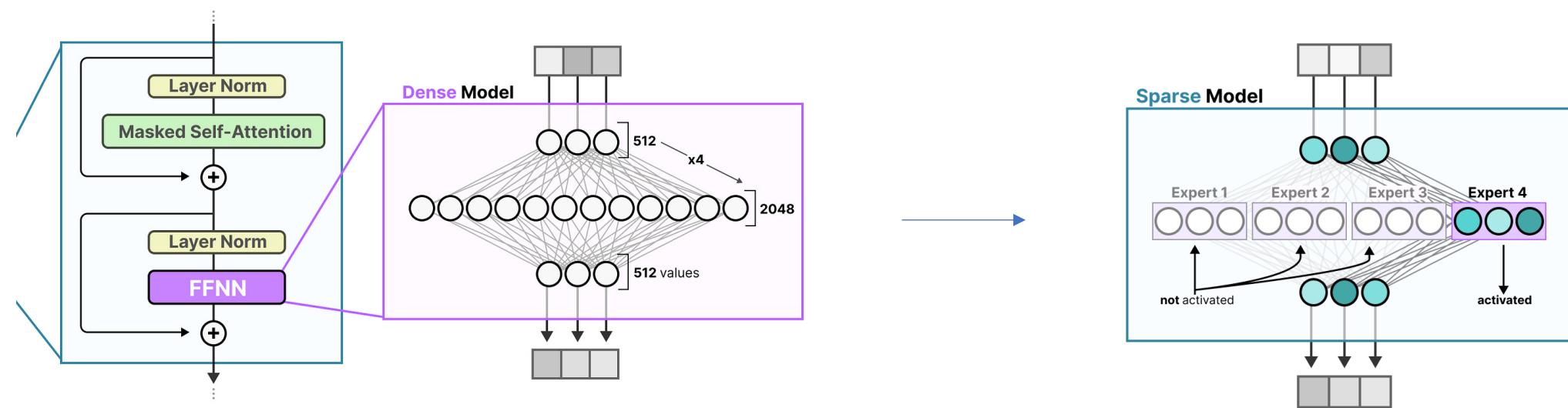
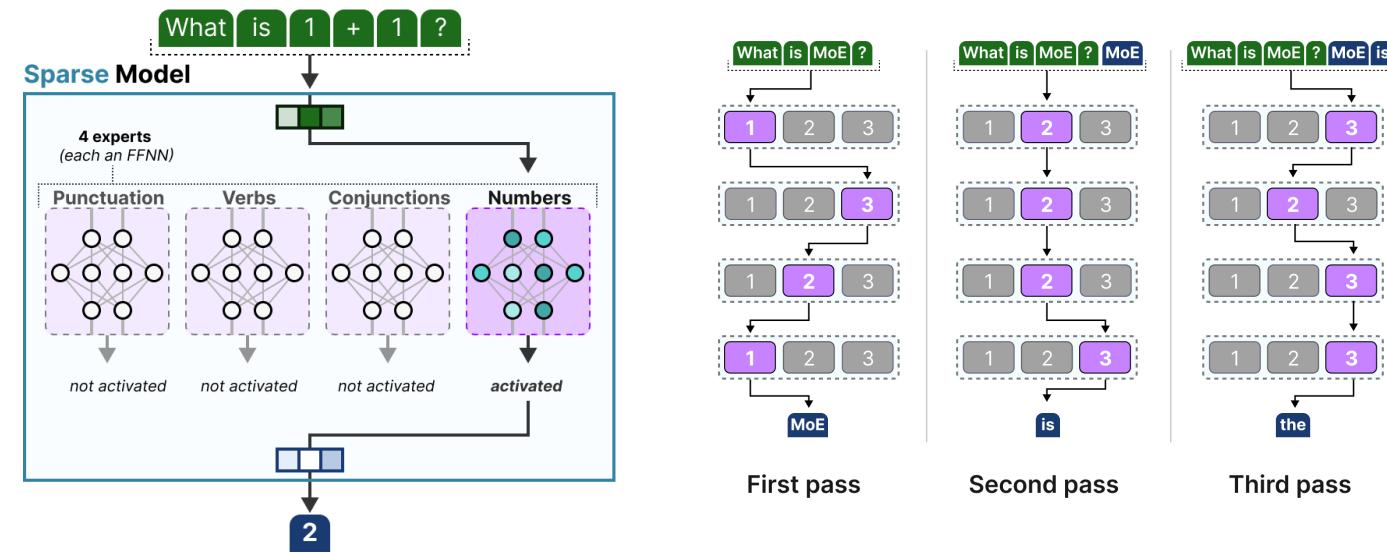
- 通过摘要功能，直接压缩 Prompt
- 通过将长提示转换为短提示同时保持其自然语言格式，比如 LLMLingua，相当于是一种翻译方法



MLA之外的KV Cache优化方法 (来源：武汉大学)

# Mixture of Experts

MoE (Mixture of Experts) 是组合多个专家模型提升深度学习模型性能和效率的架构。其核心思想是通过引入多个专家模型 (Experts)，每个输入数据只选择和激活其中的一部分专家模型进行处理，从而减少计算量，提高训练和推理速度。

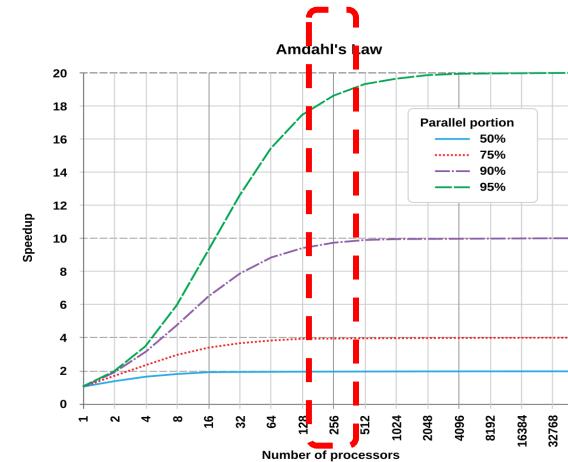


MoE激活原理 (来源: Maarten Grootendorst)

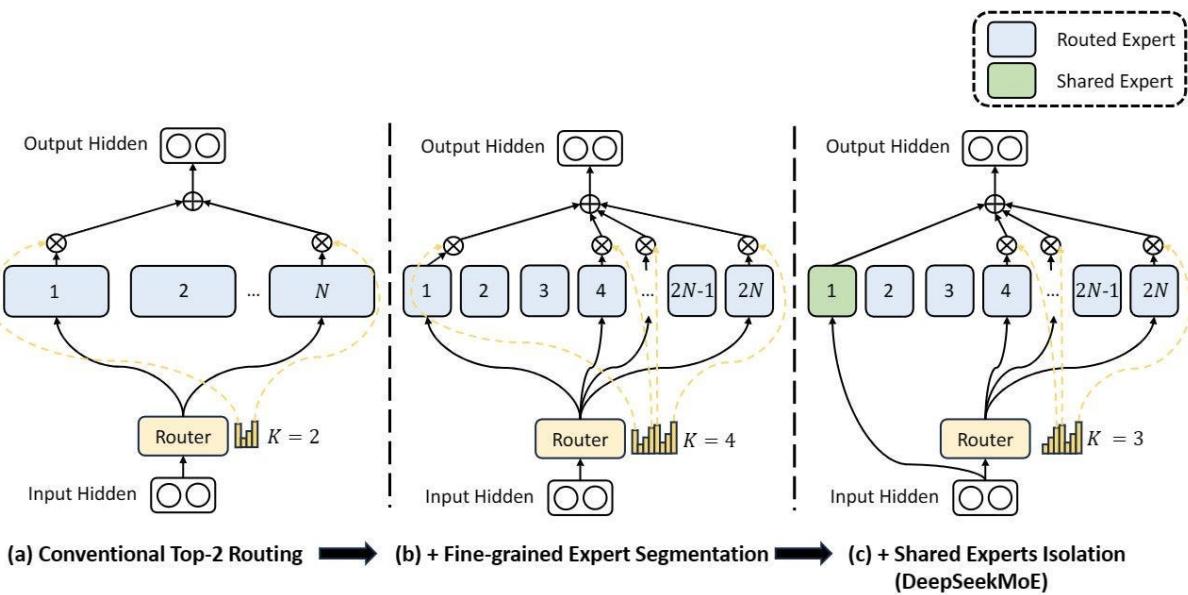
# 无辅助损耗负载均衡（也许还有更好的方法）

相比Dense模型，MoE主要增加了专家路由，通过门控（仅有开或关）的方法，负责将数据流引向不同的专家模块。专家路由事实上引入了一个不连续的门控函数（对梯度计算不友好），这导致梯度下降优化方法在MoE训练中表现不佳，甚至出现“路由崩溃”，即模型容易陷入始终为每个Token激活相同的少数专家的窘境，而不是将计算合理的传播到所有的可用专家。

无辅助损耗负载均衡（Auxiliary-Loss-Free Load Balancing）方法是将特定于专家的偏差项添加到路由机制和专家亲和力中。偏差项不会通过梯度下降进行更新，而是在整个训练过程中持续监控并进行调整以确保负载平衡。如果训练中某个专家没有获得合理的命中次数，可以在每个梯度步骤中精调偏差项增加命中概率。



(芯片界的MoE思路)

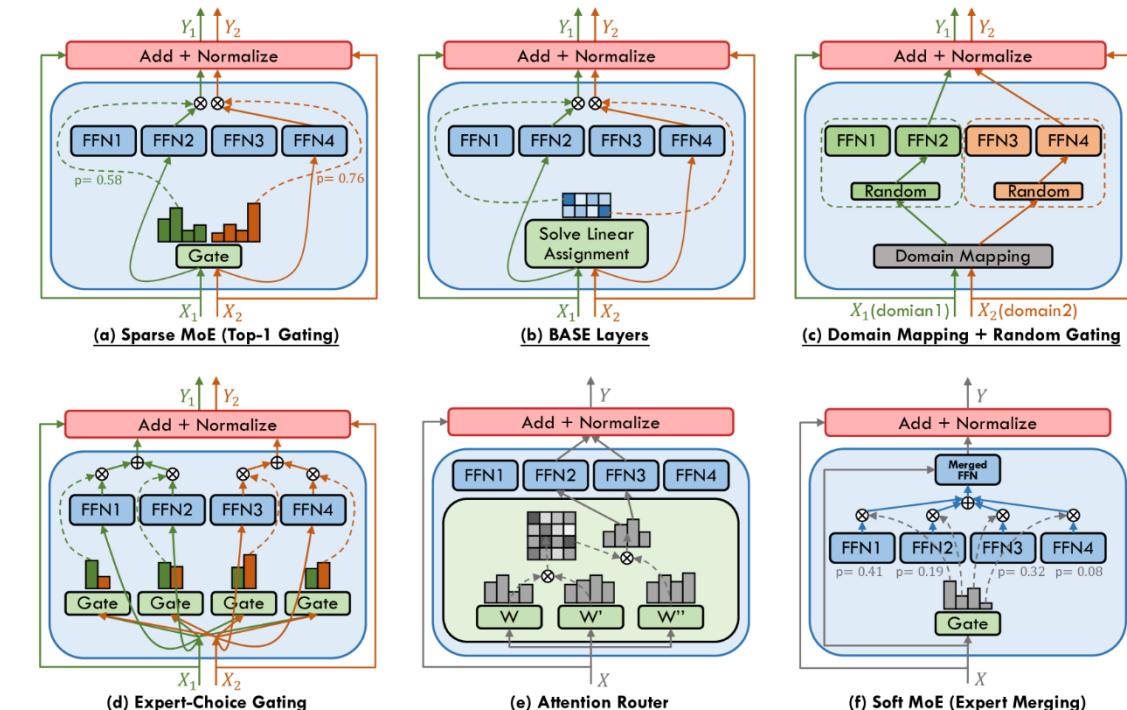


DeepSeekMoE与传统的多路由和细粒度专家MoE对比（来源：DeepSeek）

# MoE VS Dense

不同应用场景对Dense和MoE模型的需求

	To B计算场景	To C云计算场景	To C边缘/端侧计算场景
特点	专业领域应用多，对RAG高度依赖，不需要多专家	通用领域多，对检索和训练数据更新时间敏感，需要多专家	通用领域多，可不需要高性能/精度回答，不需要多专家
主力架构	行业大模型，主要是Dense架构	通用基础模型，主要是MoE或MoA架构	限于成本，主要是Dense架构
占有率	较高	较低	目前较低



MoE架构的本质是模型参数分布式存储，MoE减少计算量的代价可能是不同专家模型的参数重复和总参数量增加，这往往也意味着更大更贵的HBM成本。外界传言的MoE模型可以更小，其实是指的MoE模型蒸馏的Dense模型可以兼顾参数量和推理（Reasoning）性能

不同的MoE架构（来源：香港科技大学）

MoE模型与应用领域高度和TOC (Total Owning Cost, 总拥有成本) 密切相关，很多场景MoE未必比Dense模型好

- 1 Scaling law与Moore's law的范式共生
- 2 DeepSeek的架构提升与既要又要
- 3 V3/R1训练架构与软硬件协同进化**
- 4 V3/R1训练流程与RL变幻莫测的潜力
- 5 从DeepSeek到算力竞赛的深度思考

# V3/R1训练架构的独特优势（快！）

## 1) FP8混合精度训练框架

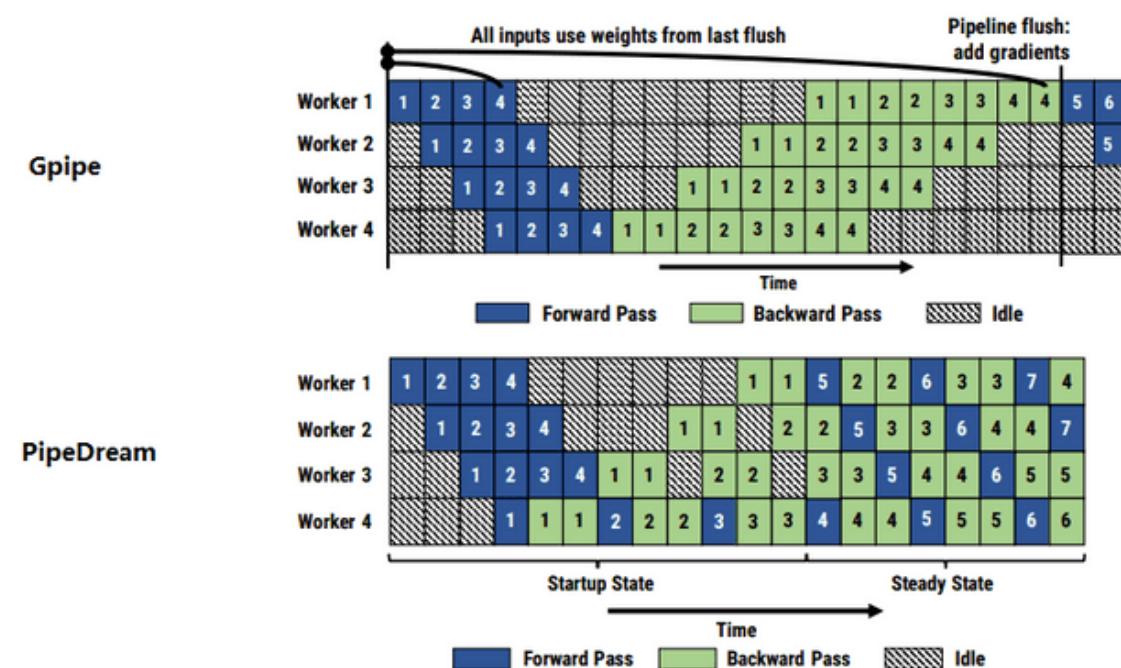
引入了FP8混合精度训练框架，并首次在超大规模大模型训练上验证了这一框架的有效性。通过对FP8低比特计算和存储的支持，实现了训练的加速和GPU内存使用的减少。

## 2) DualPipe算法

设计了DualPipe算法来实现更高效的流水线并行，并通过计算-通信重叠隐藏了大模型训练过程中的大部分通信开销。

## 3) 跨节点All-to-All通信内核

开发了高效的跨节点All-to-All通信内核，以充分利用InfiniBand (IB) 和NVLink带宽；对显存使用进行了优化，无需使用昂贵的张量并行即可训练DeepSeek-V3。



流水线 (来源：互联网)

# 软件层面的并行优化

HAI-LLM框架所作的并行改进（看起来像硬件的活）

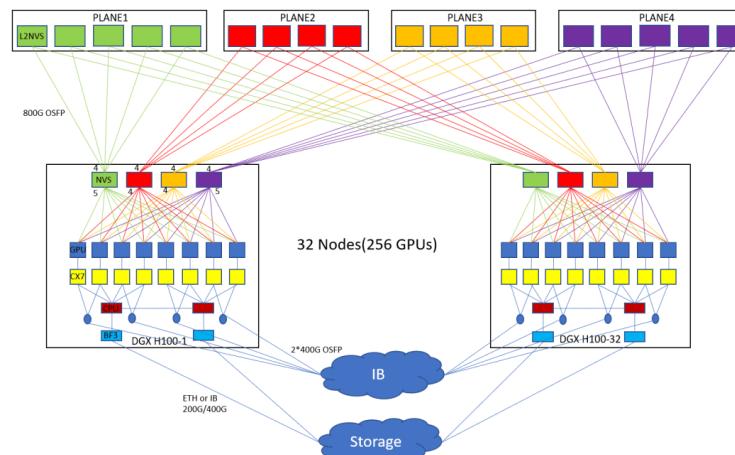
缩写	简介	DeepSeek所做的工作或优化
TP	张量并行 (Tensor Parallelism)：将模型层放置在并行执行计算的多个设备（计算芯片）上，包括逐行和逐列并行，	将 NVLink Bridge 集成到系统中，在每对 GPU 之间建立了 600GB/s 的带宽，增加 TP 效率
PP	流水线并行 (Pipeline Parallelism)：每个设备（计算芯片）都包含一部分模型层，每个训练批次分为串行的小批次以进行流水线执行	通过配置数据并行排队，规避单个节点（服务器）8 个 GPU 共享一个 IB NIC 流水线并行 (PP) 期间出现的网络带宽竞争，实现 GPU 的交替通信和 91% 的并行效率
FSDP	全共享数据并行 (Fully Sharded Data Parallel) 基于 ZeRO Stage 3 算法，对模型的参数、优化器状态和梯度分布到不同的设备（计算芯片）上。在正向传播期间，FSDP 执行 allgather 来组装完整的参数，并正向传播完成后释放；反向传播期间，FSDP 执行 allgather 获取完整参数，并进行反向梯度计算，然后执行 reduce-scatter 以同步所有设备之间的梯度，每个设备只保留部分梯度、参数和优化器更新	基于 ZeRO Stage-3 算法实现 FSDP。将 allgather 和 reduce-scatter 通信与前向和反向传播计算重叠，拆分反向传播步骤以增强重叠，减少通信量。与 PyTorch 的 FSDP 相比，HAI-LLM 的 FSDP 将训练时间缩短了近一半
DP	数据并行 (Data Parallelism)：模型和优化器的状态在多个设备（计算芯片）之间复制，数据均匀分布给所有设备进行并行计算	对 PCIe 进行工程优化，提升 DP
EP	专家并行 (Expert Parallelism)：在 MoE 训练期间，MoE 模型的不同专家分布在不同的设备（计算芯片）上，由门控单元将输入的 Token 分配给不同的专家	对 PCIe 进行工程优化，提升 EP

根据 DeepSeek 的论文，V3 应用了 16 路流水线并行 (PP)、跨越 8 个（服务器）节点的 64 路专家并行 (EP) 和 ZeRO-1 数据并行 (DP)

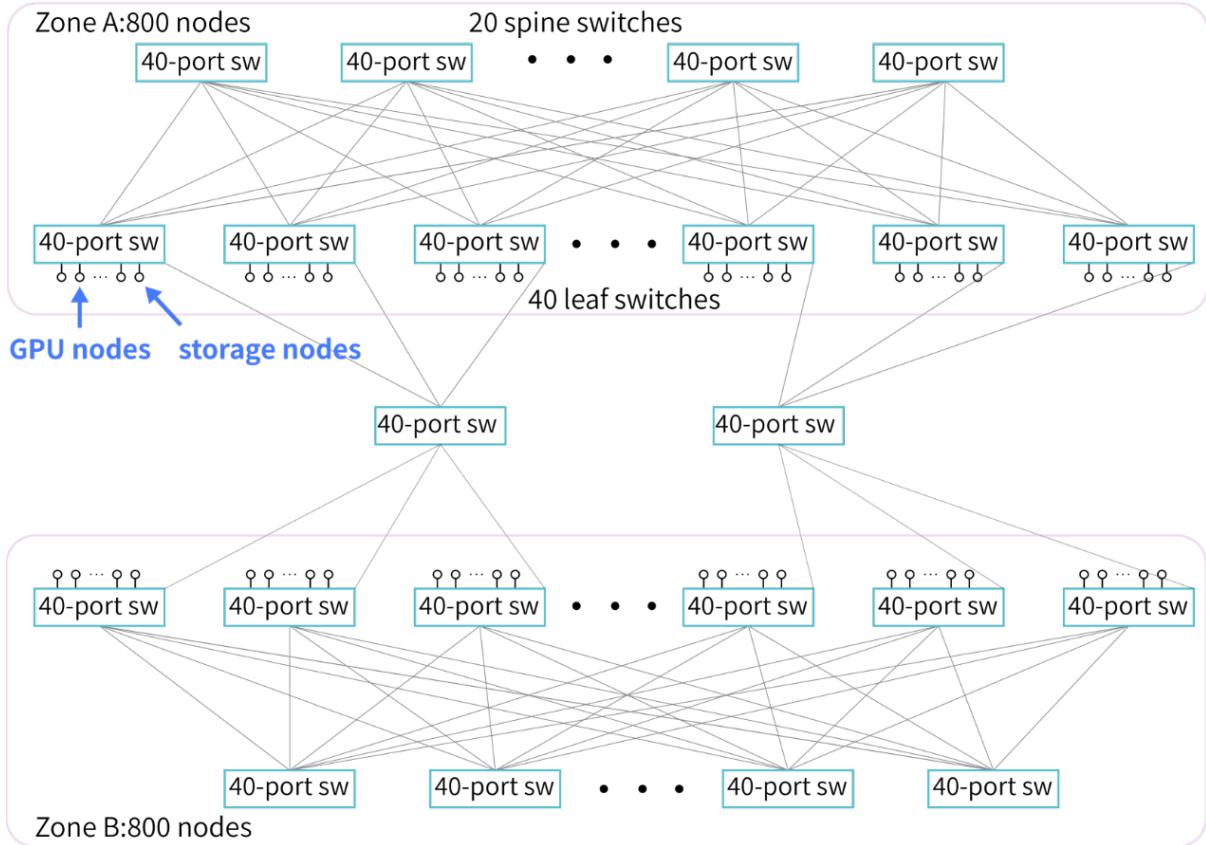
# 针对软件并行策略的硬件优化

为了配合HAI-LLM训练框架（软件），DeepSeek（曾经）采用两层Fat-Tree拓扑+ InfiniBand（IB）作为集群架构（硬件）。（作者注：DeepSeek目前在用的集群架构应该比这个更先进）

这一集群架构的核心思路是减少互连层次，降低训练的综合成本。相对DGX-A100 的标准万卡集群三层Fat-Tree的1320个交换机，DeepSeek的同规模集群仅仅需要122台交换机，至少节省了40%的互连成本



减少互连层次  
减少通信软件栈深度



# HReduce与HReduce with NVLink

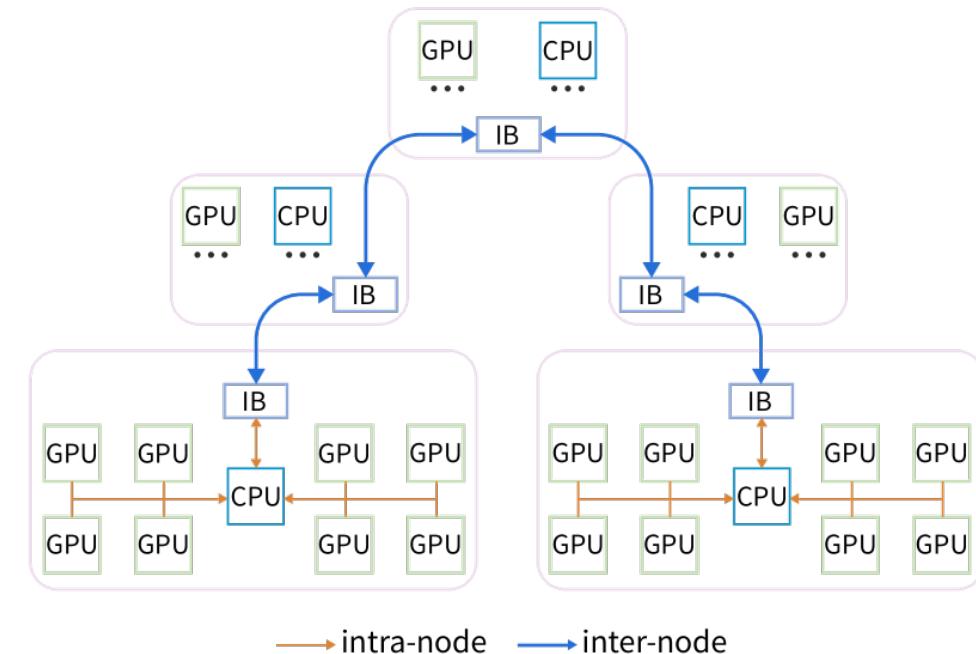
## HReduce

针对所采用的硬件架构特点，DeepSeek开发了HReduce（针对不使用NVLink的方案），以执行高效的 allreduce 操作。HReduce会首先执行节点内 reduce，然后通过 CPU 执行节点间allreduce，最后将 reduced 数据传输到 GPU。这样的优化需要DeepSeek团队对硬件互连有非常深刻的理解。

## HReduce with NVLink

DeepSeek团队也开发了基于NVLink的HReduce with NVLink，在将梯度传递给CPU之前，先在NVLink互连的GPU之间执reduce减作；当CPU返回结果时，会将数据切分并分别返回给NVLink连接的配对GPU，再通过NVLink执行allgather。

层次化的Reduce，减少低优先级的互连通信



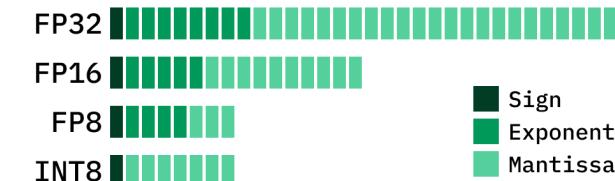
针对硬件架构优化的HReduce (来源：DeepSeek)

# FP8 训练框架体系（省）

## FP8训练的问题

通常的大模型训练会采用BF16（16位浮点）或FP32/TF32（32位浮点）精度作为数据计算和存储的格式，来确保较高的训练精度。相比之下，FP8占用的数据位宽仅为FP32的1/4，可以有力的提升计算速度，降低对存储的消耗。当然，FP8也存在精度不高，容易导致训练失败的潜在问题。

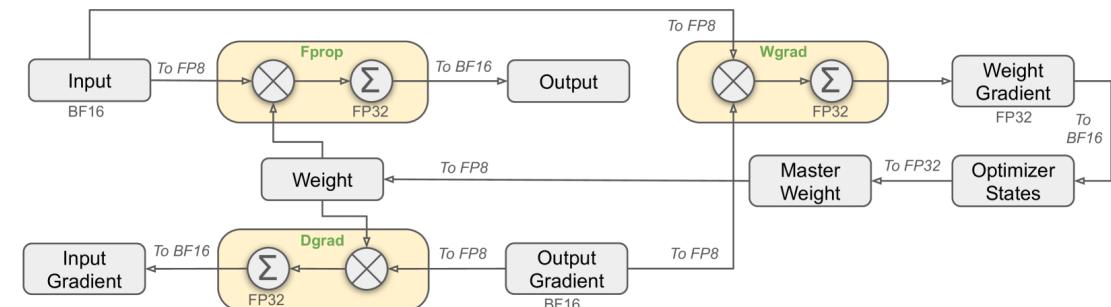
## Comparing number formats



FP8与其他数据格式占据空间对比（来源：互联网）

## FP8的训练框架体系

DeepSeek-V3主要使用FP8（8位浮点数）来提高计算速度并减少训练期间的显存使用量。为了让FP8更好的完成训练，DeepSeek专门设计了针对FP8的训练框架体系。当然，就在撰写本文的时候，微软已经跑通了FP4（4位浮点数）的完整模型训练



FP8训练框架局部方案（来源：DeepSeek）

# FP8的精度改进

使用FP8框架进行训练的主要挑战在于精度与误差的处理

DeepSeek为其FP8低比特训练框架做了以下优化：

## 1) 细粒度量化

将数据分解成更小的组，每个组都使用特定乘数进行调整以保持高精度。这一方法类似于Tile-Wise或Block-Wise。对于激活，在 $1 \times 128$ 大小的基础上对计算数据进行分组和缩放；对于权重，以 $128 \times 128$ 大小对计算数据进行分组和缩放。该方法可以根据最大或最小数据调整缩放系数，来更好的适应计算中的异常值

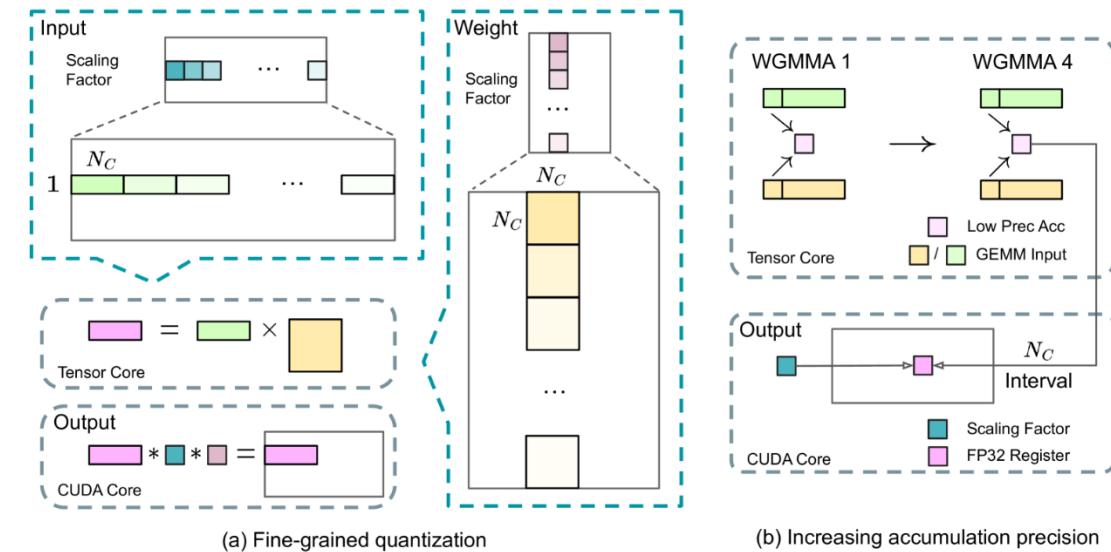
## 2) 在线量化

为了提高精度并简化框架，该框架在线计算每个 $1 \times 128$ 激活块或 $128 \times 128$ 权重块的最大绝对值，在线推算缩放因子，然后将激活或权重在线转化为FP8格式，而不是采用静态的历史数据。相对静态的量化方法，该方法可以获得更高的转换精度，减小误差的累积

## 3) 提高累加精度

FP8在大量累加时会累积出现随机误差。例如FP8 GEMM在英伟达H800 GPU上的累加精度保留14位左右，明显低于FP32累加精度。以 $K=4096$ 的两个随机矩阵的GEMM运算为例，Tensor Core中的有限累加精度可导致最大相对误差接近2%。

DeepSeek将中间结果储存计算升级为FP32（32位浮点），实行高精度累加，然后再转换回FP8，以降低大量微小误差累加带来的训练偏差



细粒度量化与提高累加精度（来源：DeepSeek）

## 4) 低精度/混合精度存储与通信

为了进一步减少MoE训练中的显存和通信开销，该框架基于FP8进行数据/参数缓存和处理激活，以节省显存与缓存空间并提升性能，并在BF16（16位浮点数）中存储低精度优化器状态。该框架中以下组件保持原始精度（例如BF16或FP32）：嵌入模块、MoE门控模块、归一化算子和注意力算子，以确保模型的动态稳定训练。为保证数值稳定性，以高精度存储主要权重、权重梯度和优化器状态

# DualPipe优化 (好)

## DualPipe优势

与现有的流水线并行 (PP) 方法相比, DualPipe具备以下优势:

1) DualPipe的流水线气泡更少, 信道使用效率更高

2) DualPipe将前向和后向传播中的计算和通信重叠, 解决了跨节点专家并行 (EP) 带来的繁重通信开销问题

3) 在确保计算与通信比例恒定的情况下, 具有很好的

Scale-out能力

## DualPipe划分

DualPipe算法将每个数据块分为四个部分: attention (图中ATTN) 、 all-to-all dispatch (图中DISPATCH) 、 MLP 和 all-to-all combine (图中COMBINE)

对于后向块, attention和 MLP 都进一步分为后向输入、后向权重。对于一对前向和后向块, 针对通信和计算的过程和瓶颈进行优化。DualPipe采用双向流水线调度, 同时从流水线发送前向和后向数据, 尽可能提高使用率

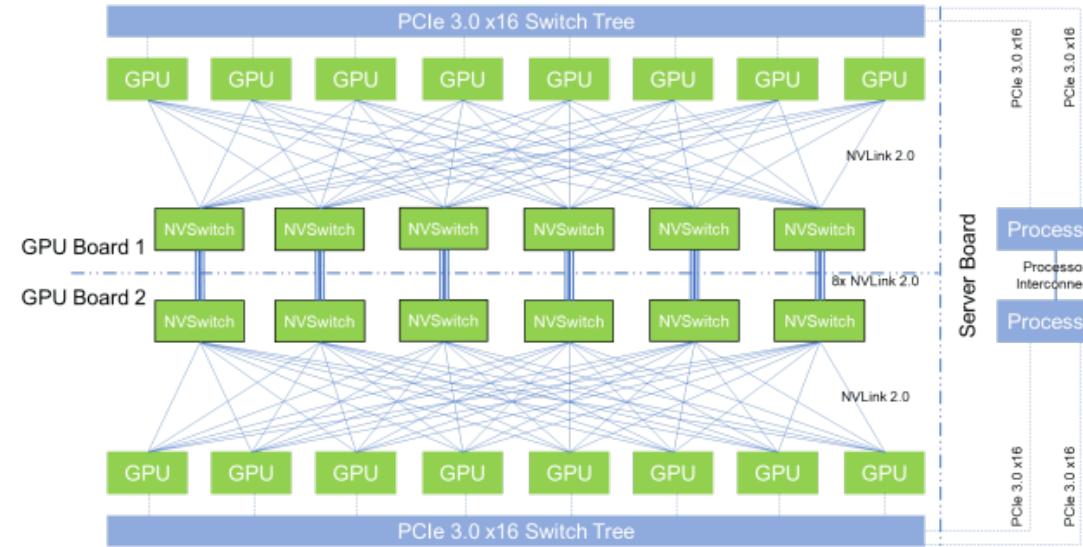


DualPipe的流水线 (来源: DeepSeek)

# 跨节点 All-to-All 通信

## 减少直接All-to-All

DeepSeek将每个Token（词元）的分发限制为最多4个节点，从而减少IB流量限制的影响。（本质的通点还是节点间带宽不足）对于每个Token，在做节点间路由决策时，先通过IB传输到目标节点上具有相同节点内索引的GPU；到达目标节点后，再通过NVLink转发到托管目标专家的特定GPU。通过这种方式，通过IB和NVLink的通信重叠，平均每个Token可以在每个节点选择3.2名专家，而不会产生额外的NVLink开销



传统的基于NVSwitch的All-to-All通信结构（来源：互联网）

## warp (线程束) 专用化技术

将20个SM划分为10个通信信道。

- 1) 在调度过程中， (a) IB 发送、 (b) IB 到NVLink 转发、 (c) NVLink 接收由相应的warp处理。分配给每个通信任务的warp数量会根据所有SM的实际工作负载动态调整
- 2) 在合并过程中， (1) NVLink 发送、 (2) NVLink到IB的转发和累积、 (3) IB接收和累积也由动态调整的warp处理
- 3) dispatching 和combining kernel都与计算流重叠，采用定制的PTX (Parallel Thread Execution) 指令以自动调整通信块大小，减少了对L2缓存的使用和对其他SM的干扰

# 显存节省技术（省）

## DeepSeek采用的显存节省技术

技术	方法说明	优势
RMSNorm 和 MLA Up-Projection 的重新计算	在反向传播期间重新计算所有的 MSNorm 操作和 MLA Up-Projection，无需持久存储其输出激活	以算代存，充分利用GPU内算力 充沛但缓存不足的特点
在CPU内存中保存指数平均数指标 (EMA)	在CPU内存中保存EMA，并在每个训练步骤后异步更新	把EMA从GPU显存占用改为CPU内存占用，释放动态存储空间
在多标记预测 (MTP) 中共享嵌入和输出头	使用DualPipe 策略，将模型最浅的层（包括嵌入层）和最深的层（包括输出头）部署在相同的PP等级上	允许MTP模块和主模型之间物理共享参数、梯度、嵌入和输出头，提升显存效率

- 1 Scaling law与Moore's law的范式共生
- 2 DeepSeek的架构提升与既要又要
- 3 V3/R1训练架构与软硬件协同进化
- 4 V3/R1训练流程与RL变幻莫测的潜力**
- 5 从DeepSeek到算力竞赛的深度思考

# V3的训练流程 (R1训练的基础)

PreTrain

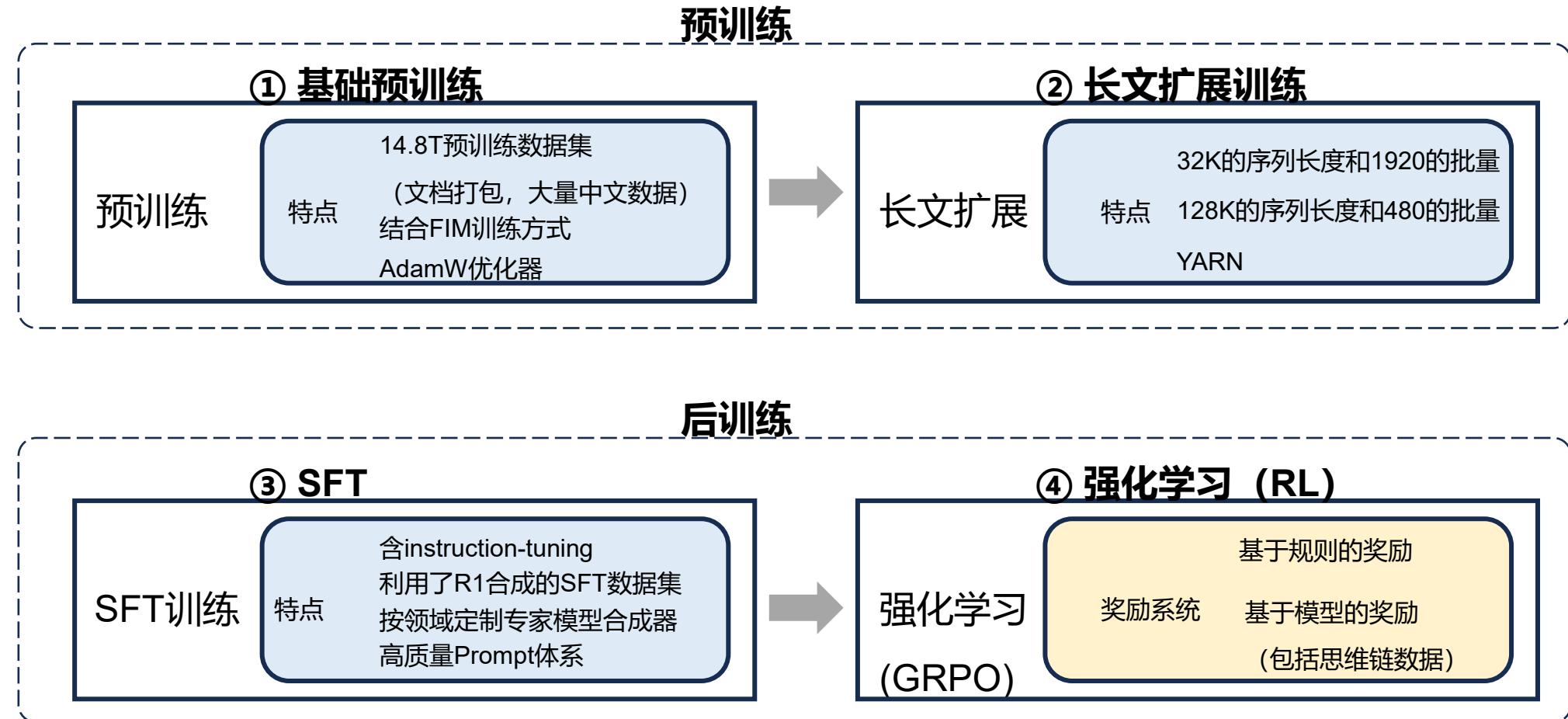
语言与知识学习

SFT

显示的价值对齐

RL

能力泛化与提升



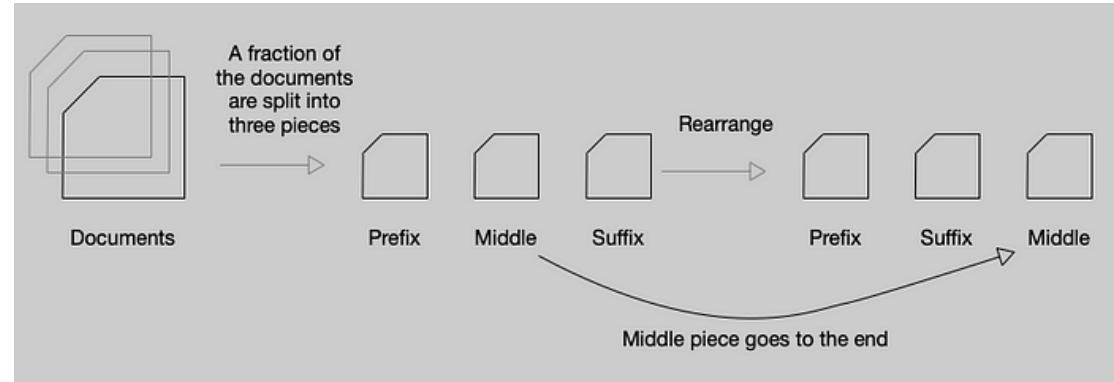
DeepSeek-V3训练流程

# V3的基础预训练

## 14.8T预训练数据集结构上

V3采用了以下策略：

- 1) 提高数学和编程样本的比例来优化预训练语料库，以提升推理能力。
- 2) 基于中国互联网可用的语料库整合了更多的中文数据。（这也是V3能玩中文梗的关键）
- 3) 将多语言覆盖范围扩展到英文和中文之外。
- 4) 优化数据处理/过滤算法，在保持语料库多样性的同时最大限度减少信息冗余。  
过滤掉有争议的内容，减少特定区域文化引入的数据偏差
- 5) 通过文档打包，减少在短文本块的训练浪费，同时在训练过程中没有使用交叉样本注意力屏蔽。



Fill-in-Middle方法（来源：互联网）

## Fill-in-Middle

V3的训练采用前缀-后缀-中间 (PSM) 框架来构建FIM训练数据。Fill-in-Middle (FIM, 中间补全) 是一种针对代码补全能力的预训练方式，模型在训练过程中学习使用上下文的语境来预测文段中间缺失的部分。FIM策略不会损害下一Token预测 (NTP) 能力，同时可使模型能够根据上下文线索准确预测中间文本。

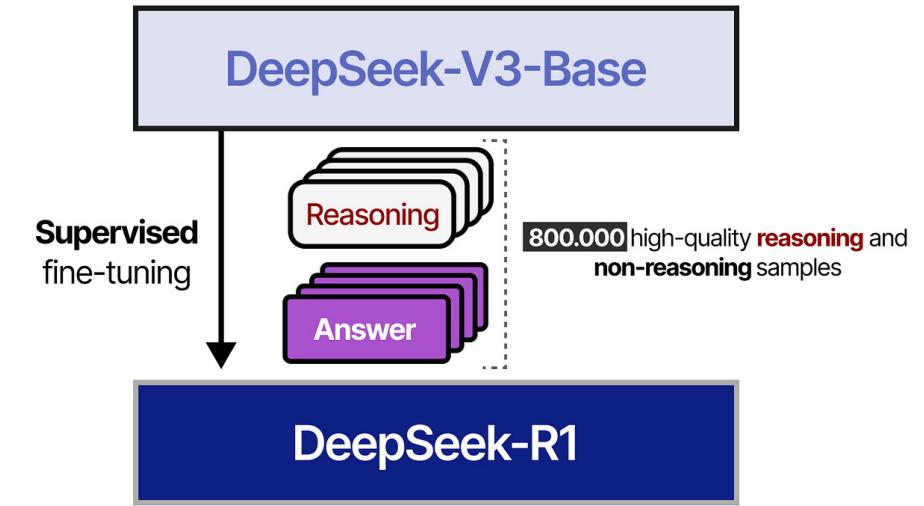
V3使用AdamW优化器来预训练，同时避免过度拟合。

# V3的有监督精调 (SFT)

## SFT数据集构建

V3的有监督精调做了以下这些事：

- 1) 梳理指令精调 (instruction-tuning) 数据集。该数据集包括 1.5M 个实例，跨多个域，每个域都量身定制的不同的数据合成方法
- 2) 利用 DeepSeek-R1 模型合成与推理 (Reasoning) 相关的 SFT 数据集。这里很有意思，基于 R1 来 SFT V3，再基于 V3 冷启动 R1。感觉上这里有关键的训练信息没有透露，DeepSeek 应该还是留了一手
- 3) 为特定领域（例如代码、数学或一般推理）构建量身定制的专家模型数据合成器。使用复合有监督精调和强化学习训练该专家模型。训练过程中为每个实例生成两种不同类型的 SFT 样本：第一种将问题与其原始响应耦合，格式为 <problem, original response>，而第二种将系统提示与问题和 R1 响应合并，格式为 <system prompt, problem, R1 response>



V3与R1的共生关系 (来源：Maarten Grootendorst)

- 4) 建立高质量提示 (Prompt) 体系，引导模型形成自身的反馈与验证机制。同时整合了来自 R1 合成的数据，通过强化学习加强这一能力
- 5) 对于非推理数据（例如创意写作、角色扮演和简单的问答），利用 DeepSeek-V2.5 生成数据，并通过人工注释验证数据的准确性

# V3的强化学习

## 1) 基于规则的奖励模型

对于可以使用特定规则验证的问题，采用基于规则的奖励模型来提供训练反馈。例如，对于 LeetCode 问题，可以利用编译器根据测试用例生成反馈。基于规则的模型验证，可以确保更高的生成可靠性

## 2) 基于模型的奖励模型

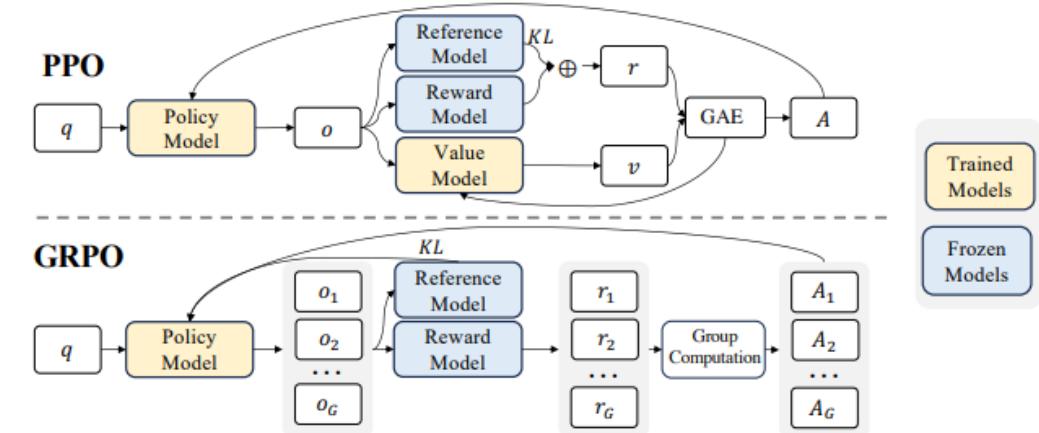
对于不具有收敛性的问题，依靠模型判断来确进行强化学习。V3训练中构建了特殊的偏好数据，该数据同时包括提供最终奖励结果和形成奖励的思维链，以降低特定任务中的奖励破解攻击风险

### RL的意义

大模型的训练通常用无监督进行预训练，然后通过有监督精调（SFT）进一步学习。然而SFT有时难以将人类的偏好显式地整合进去，这时就需要强化学习来进行精调。

### PPO的痛苦

在以往的大模型训练中一般使用PPO（Proximal Policy Optimization）来形成梯度策略。PPO的代价在于需要维护较大的价值网络（也是一个神经网络），需要占用较大的显存与计算资源



GRPO与PPO对比 (来源: DeepSeek)

### GRPO (未来应该会有更好方法)

V3中则采用了DeepSeek提出的GRPO（Group Relative Policy Optimization）策略，只需要在一个分组内进行多个采样输出的比较，再根据比较结果选择较优的策略。GRPO中不再需要一个显式的价值网络，从而降低了显存占用并提高了训练速度

# 无SFT的R1-Zero训练

作为R1的无SFT版本，R1-Zero使用 DeepSeek-V3-Base作为基础模型，直接使用 GRPO 进行强化学习来提升模型的推理（Reasoning）性能，根据准确度和格式进行训练奖励。

奖励的计算方式在很大程度上决定了强化学习训练的效果。

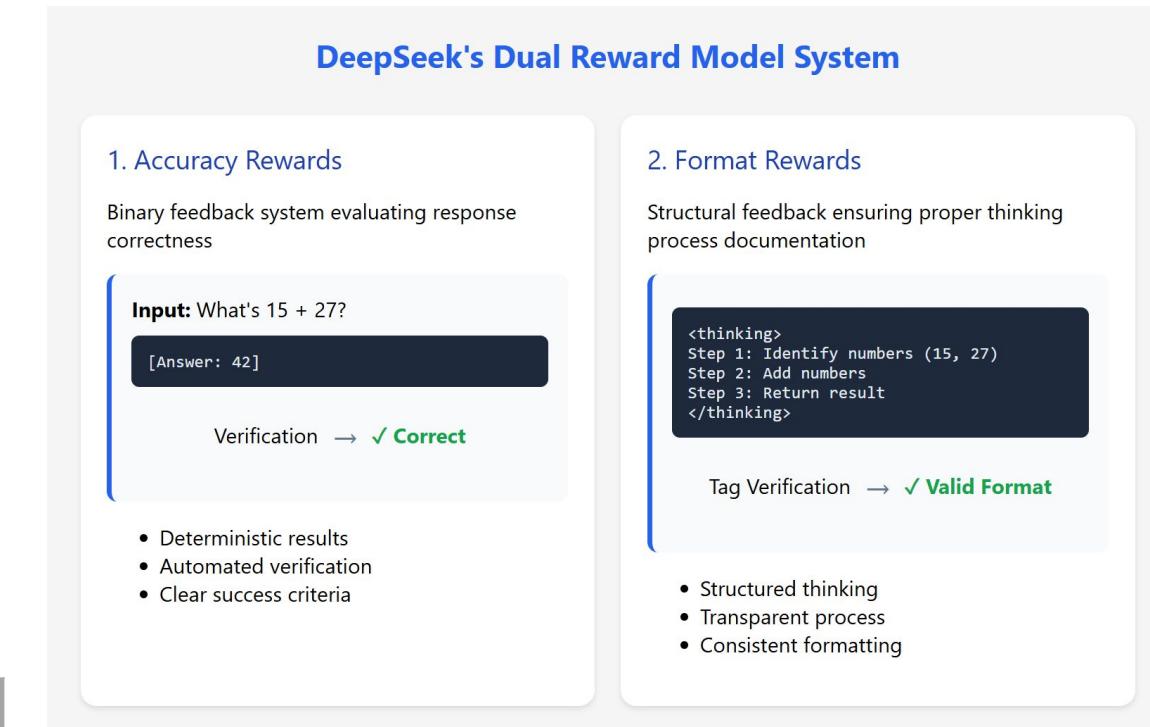
DeepSeek-R1-Zero 的基于规则的奖励系统包括：

- 1) 准确度奖励 (Accuracy rewards)。评估响应是否正确。
- 2) 格式奖励 (Format rewards)。奖励模型将其思考过程置于“<think>”和“</think>”标签之间。

需要区分两个容易混淆的概念：

Reasoning (推理)：通过对事实的考虑和分析来得出结论的过程。推理强调的是思考和决策的过程，比“推断”具有更多的逻辑和分析过程。

Inference (推断)：基于已有信息或数据推导出新的信息或结论的过程。推断侧重于通过既定的算法或模型来实现，与“推理”相比，更强调形式化和计算过程。



DeepSeek的准确度奖励与格式奖励 (来源：互联网)

# 顿悟时刻 (Aha Moment)

## 顿悟时刻

指代模型在训练过程中间阶段突然学会通过主动延长思考时间并重新评估初始策略来解决复杂问题。这一现象标志着模型推理能力的质变

## 具体表现

**动态调整思考时间**：模型不再急于输出答案，而是像人类遇到难题时“停下来反复推敲”一样，主动分配更多计算资源（如增加推理长度）来重新分析问题

**策略迭代优化**：在初步尝试失败后，模型能自主修正初始方案，通过试错、逻辑链验证或引入外部知识辅助决策

## 意义

**RL的潜力**：无需明确的教模型如何解决问题，只需为其提供适当的激励，模型就会自主形成解决问题的策略

**迈向通用/超级智能**：这也说明强化学习有潜力解锁新的智能水平，为未来更自主、更具适应性的模型铺平道路，提供了形成超级智能的可能路线



Question: If  $a > 1$ , then the sum of the real solutions of  $\sqrt{a - \sqrt{a+x}} = x$  is equal to

Response: <think>

To solve the equation  $\sqrt{a - \sqrt{a+x}} = x$ , let's start by squaring both ...

$$(\sqrt{a - \sqrt{a+x}})^2 = x^2 \implies a - \sqrt{a+x} = x^2.$$

Rearrange to isolate the inner square root term:

$$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$$

...

**Wait, wait. Wait. That's an aha moment I can flag here.**

Let's reevaluate this step-by-step to identify if the correct sum can be ...

We started with the equation:

$$\sqrt{a - \sqrt{a+x}} = x$$

First, let's square both sides:

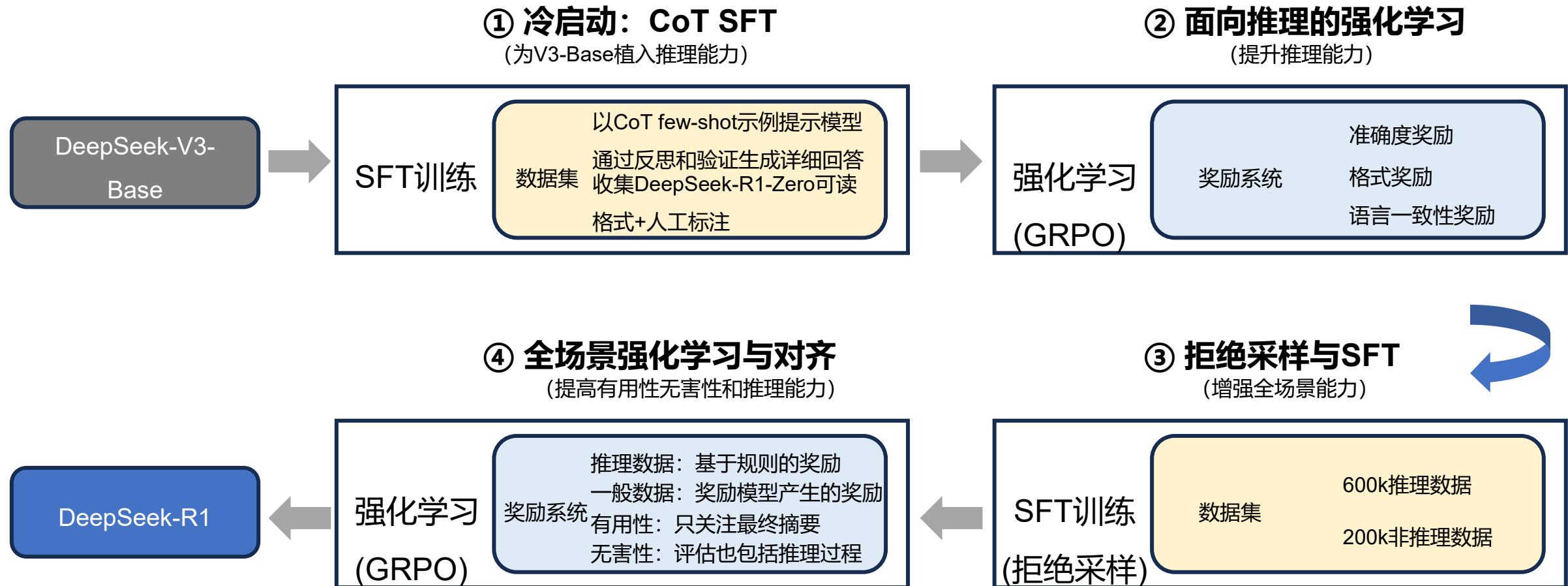
$$a - \sqrt{a+x} = x^2 \implies \sqrt{a+x} = a - x^2$$

Next, I could square both sides again, treating the equation: ...

...

Table 3 | An interesting “aha moment” of an intermediate version of DeepSeek-R1-Zero. The model learns to rethink using an anthropomorphic tone. This is also an aha moment for us, allowing us to witness the power and beauty of reinforcement learning.

# DeepSeek-R1的训练流程



DeepSeek-R1训练流程

# 1) 冷启动 (Cold Start) : CoT SFT

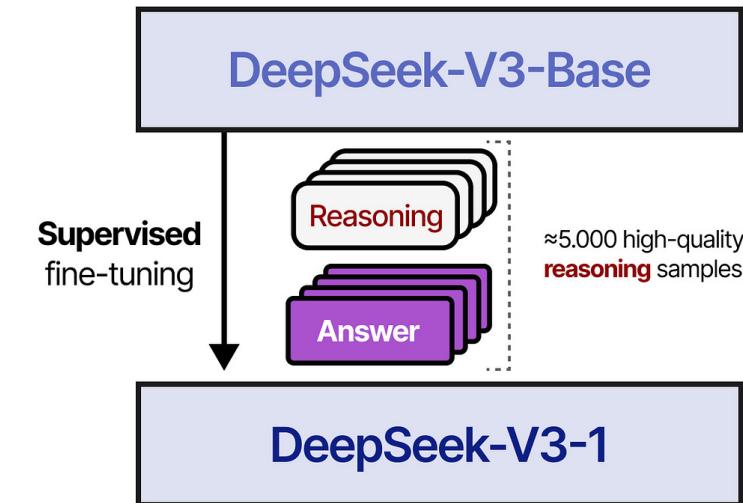
## Cold Start

与R1-Zero不同，R1首先基于DeepSeek-V3-Base进行有监督精调(SFT)，以克服强化学习的早期不稳定。DeepSeek认为这种基于人类先验知识冷启动并进行迭代训练的方式更适合推理模型

## Long CoT数据

为构建少量的长CoT数据，DeepSeek探索了几种合成方法：使用长CoT的few-shot提示作为示例，直接提示模型通过反思和验证生成详细回答，以可读格式收集DeepSeek-R1-Zero输出，并通过人工标注员的后处理来完善结果。在此步骤中收集了数千个冷启动样本以进行精调。其中可读模式指为每个回答在末尾包含一个摘要，并过滤掉不易阅读的部分。其输出格式为

|special\_token|<reasoning\_process>|special\_token|<summary>。



(来源：Maarten Grootendorst，以下类似图不再注明)

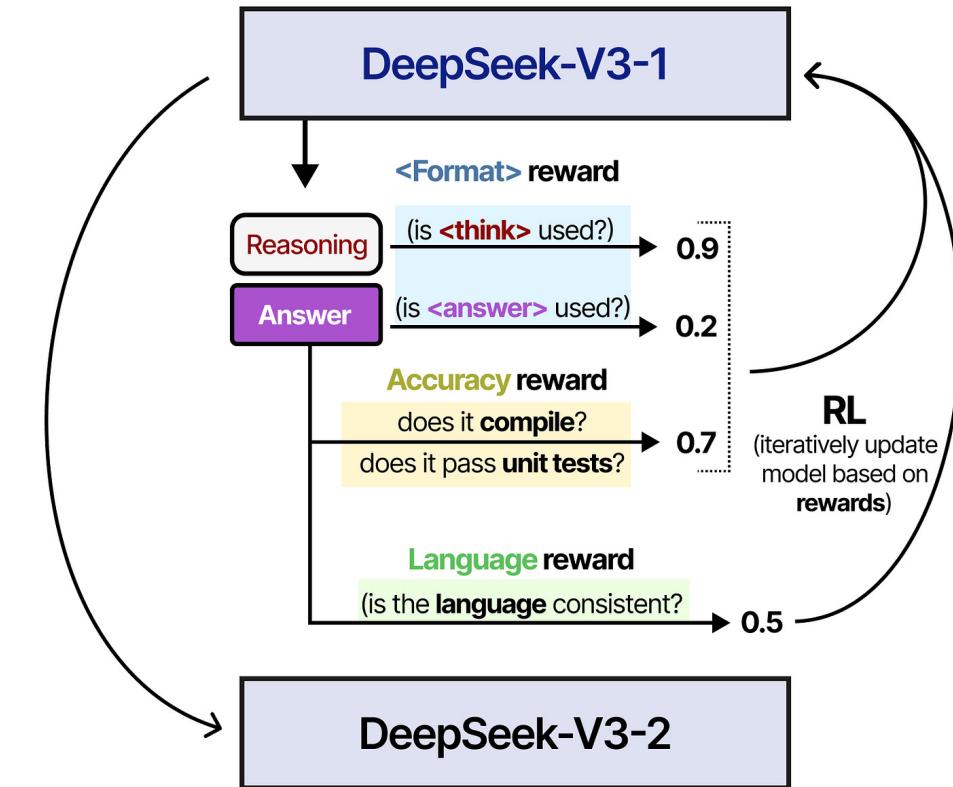
## 2) 面向推理的强化学习

### GRPO RL

在基于冷启动数据对V3-Base 精调后，采用与R1-Zero相当的强化学习训练流程，基于GRPO进行强化学习，根据准确度和格式进行训练奖励。为了解决语言混杂问题，还在强化学习训练中引入了语言一致性奖励，该奖励以CoT中目标语言单词的比例计算。

### 目标

此阶段主要提升模型的推理（Reasoning）性能，特别是在编码、数学、科学和逻辑推理等推理密集型任务，这些任务涉及定义明确且解决方案明确的问题。



### 3) 拒绝采样与SFT

#### SFT

这是另一个使用标记数据的有监督精调 (SFT)训练阶段，分批进行了两个 epoch的精调，样本量为800k。800k中包括600k推理数据和200k非推理数据。与主要侧重于推理的冷启动数据不同，此阶段结合了来自其他领域的数据，以增强模型在写作、角色扮演和其他通用任务中的能力

#### 拒绝采样

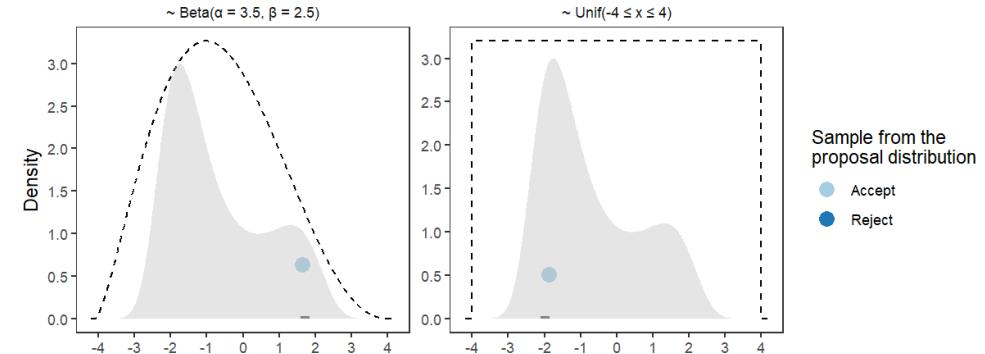
拒绝采样 (Rejection Sampling) 提供了一种桥梁，使用易于采样的分布来近似训练真正感兴趣的复杂分布。目标响应 (ground-truth) 从一组生成的回答经过拒绝采样生成，其分数由奖励系统确定。

#### 600k推理数据的生成：

- 1) 通过从上一轮强化学习训练的检查点进行拒绝抽样，整理推理提示并生成推理轨迹 (Reasoning Trajectories) 。
- 2) 除基于规则奖励进行评估的数据外，还包括了基于奖励模型的V3判断生成数据。
- 3) 过滤掉了混合语言、长段落和代码块的思路链数据。
- 4) 对于每个提示 (Prompt)，会生成多个回答，然后并仅保留正确的响应。

Rejection sampling - Sample 1 of 350

Rejection sampling is based on the observation that to sample a random variable in one dimension, one can perform a uniformly random sampling of the two-dimensional Cartesian graph of a dominating distribution, and keep the samples in the region under the graph of its density function.

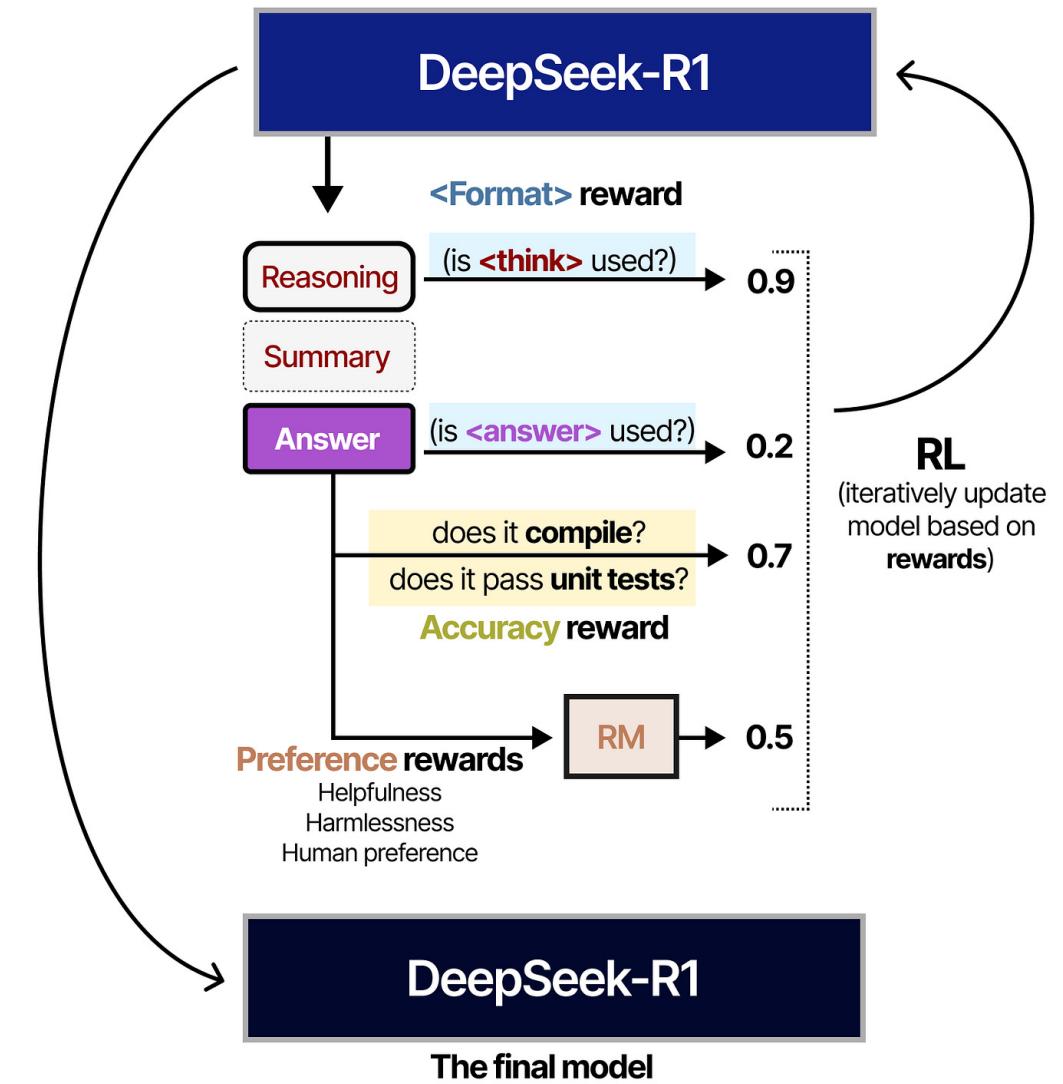


#### 200k非推理数据的生成（如写作、事实问答、自我认知和翻译等）：

- 1) 采用DeepSeek-V3流程并复用V3 的部分 SFT 数据集。
- 2) 可调用V3生成潜在的思路链，再通过提示回答。
- 3) 对于更简单的查询（例如“你好”），不提供CoT回答。

## 4) 面向全场景的强化学习与对齐

- 1) 使用奖励信号和多种提示分布 (Diverse Prompt Distributions) 的组合来训练模型
- 2) 对于推理数据，利用基于规则的奖励来指导数学、代码和逻辑推理领域的训练过程
- 3) 对于一般数据，采用奖励模型来捕捉复杂微妙场景中的人类偏好。即参考DeepSeek-V3 管训练流程，采用类似的偏好对和训练提示分布
- 4) 对于有用性，只关注最终摘要，以确保重点响应对用户的实用性和相关性，最大限度减少对底层推理过程的干扰
- 5) 对于无害性，评估模型的整个响应，包括推理过程和摘要，以识别和减轻生成过程中可能出现的潜在风险、偏见或有害内容



# RL仍处于变幻莫测起步的阶段

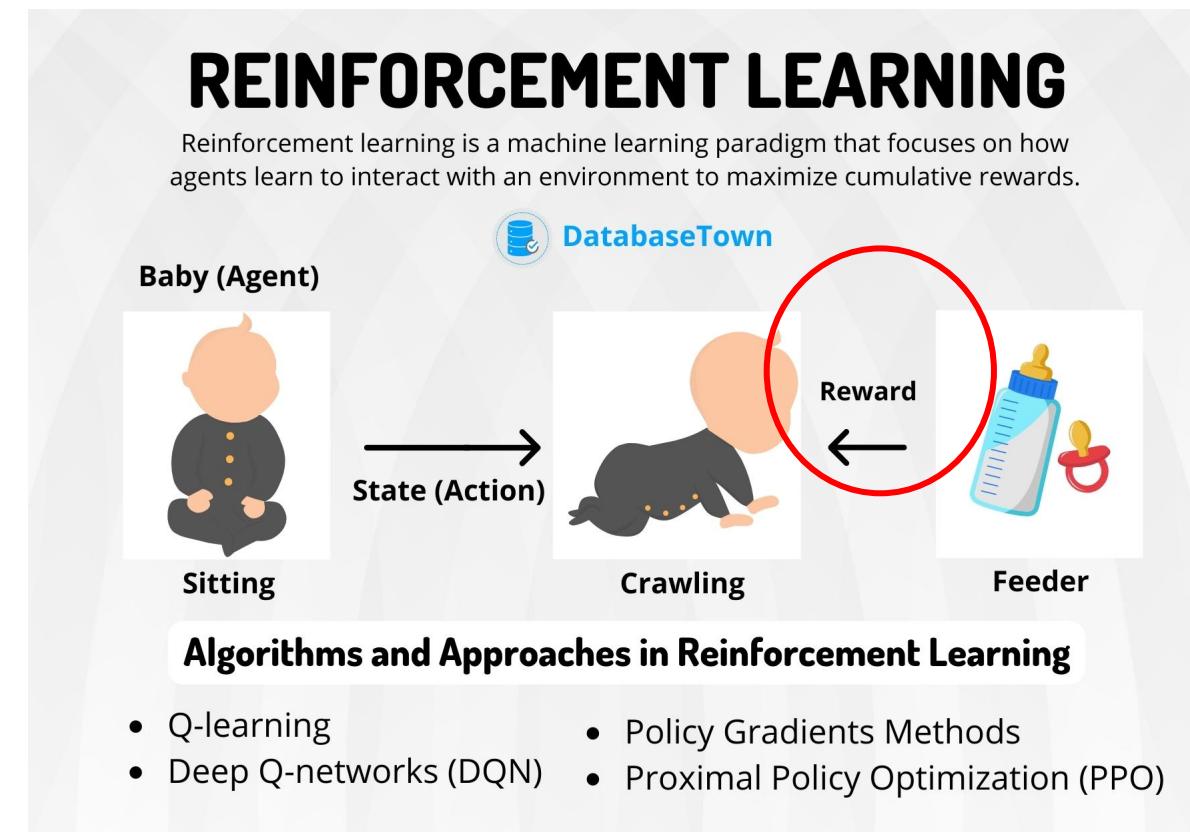
- 1) SFT 不是绝对必要的，但简化了训练并提高了效率
- 2) 推理能力往往随着训练计算量的增加而出现，奖励系统对于稳定 CoT 长度增长至关重要
- 3) Long CoT一般比Short CoT上限更高
- 4) 可验证的奖励信号对于 RL 至关重要；
- 5) 随着思考时间的增加，模型在推理任务中自然会有所改善
- 6) 蒸馏已具备Long CoT的大模型合成数据比自主构建 Long CoT数据效果好

做RL的训练框架有较大工程挑战

RL的核心是奖励的设计

PPO等算法效率相对不高

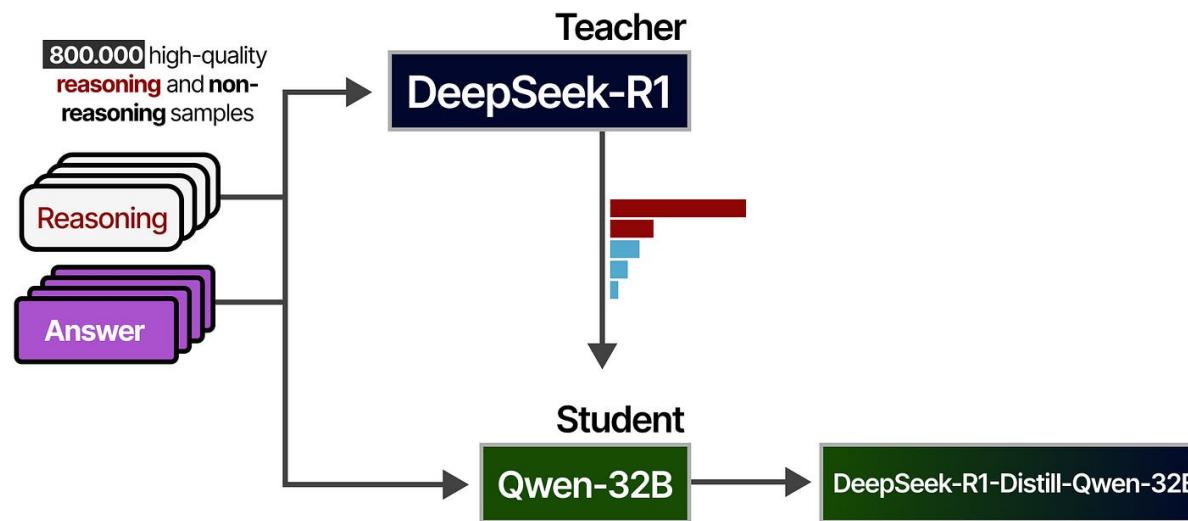
其他算法（包括GRPO）的精度可能不如PPO



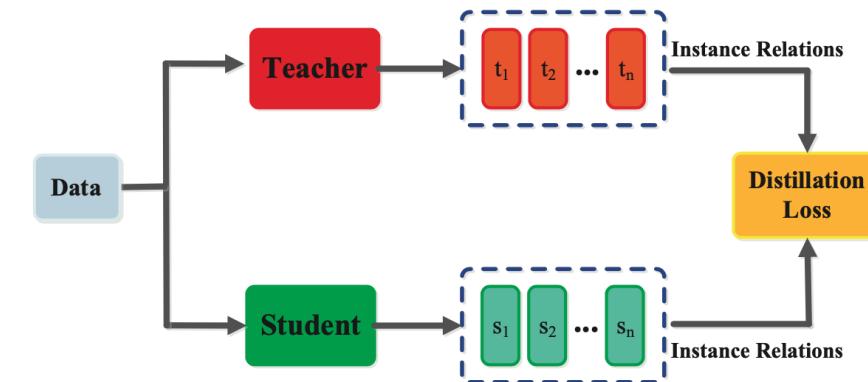
强化学习图示（来源：DatabaseTown）

# 从MoE回归Dense (蒸馏SFT)

为了能够将推理能力迁移到MoE架构不适合的场景，DeepSeek选择Llama和Qwen系列开源大模型进行蒸馏，使相应的Dense模型也能获得推理能力。与使用强化学习相比，直接SFT更适合较小的大模型，蒸馏完成的Dense模型推理能力明显好于原开源模型。



蒸馏是将复杂的大型神经网络压缩为更小、更简单的神经网络，同时尽可能多的保留结果模型的性能的过程。此过程涉及训练较小的“学生”神经网络，通过其预测或内部表示的精调来学习模拟更大、更复杂的“教师”网络的行为



模型蒸馏方法 (来源：互联网)

Model	Base Model
DeepSeek-R1-Distill-Qwen-1.5B	<a href="#">Qwen2.5-Math-1.5B</a>
DeepSeek-R1-Distill-Qwen-7B	<a href="#">Qwen2.5-Math-7B</a>
DeepSeek-R1-Distill-Llama-8B	<a href="#">Llama-3.1-8B</a>
DeepSeek-R1-Distill-Qwen-14B	<a href="#">Qwen2.5-14B</a>
DeepSeek-R1-Distill-Qwen-32B	<a href="#">Qwen2.5-32B</a>
DeepSeek-R1-Distill-Llama-70B	<a href="#">Llama-3.3-70B-Instruct</a>

# MoE的跨GPU部署模式

## Prefill (预填充) 阶段

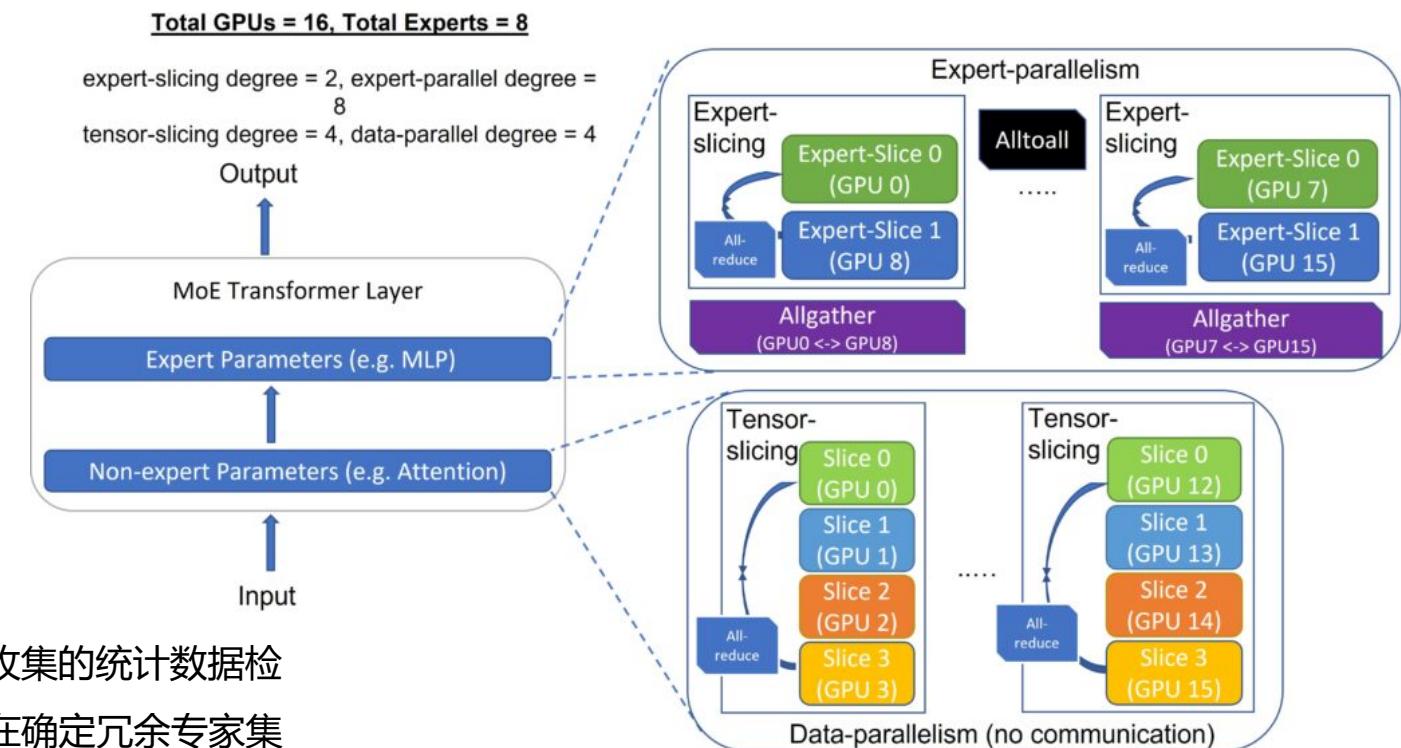
最小部署单元由 4 个节点和 32 个 GPU 组成。注意力部分采用 4 路张量并行 (TP4) 和序列并行 (SP) , 并结合 8 路数据并行 (DP8)

对于 MoE 部分, 使用 32 路 EP (EP32) , 确保每个 Expert 都能处理足够大的批量, 从而提高计算效率  
对浅层中的密集 MLP 使用 1-way TP 来节约 TP

## 冗余专家部署策略

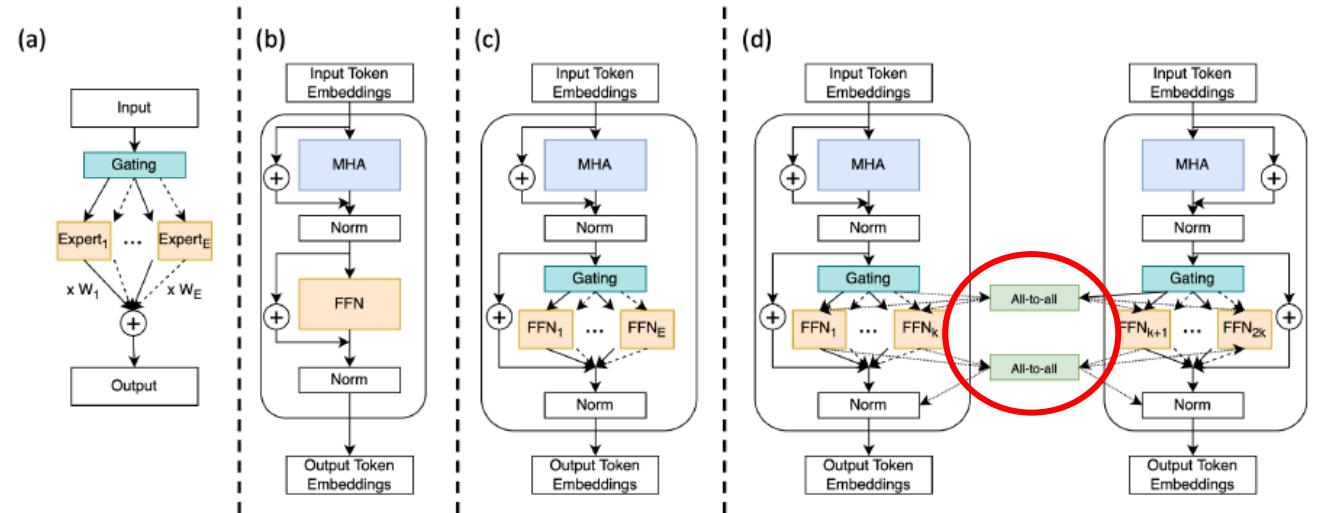
该策略复制高负载 专家 并冗余部署。根据在线部署期间收集的统计数据检测高负载专家, 并定期调整 (例如, 每 10 分钟一次) 。在确定冗余专家集后, 根据负载重排节点内 GPU 之间的专家, 尽量在不增加跨节点 All-to-All 通信开销的情况下, 尽可能平衡 GPU 之间的负载

为预填充阶段设置了 32 个冗余专家。对于每个 GPU, 除了托管的原始 8 个专家外, 它还将托管 1 个额外的冗余专家。

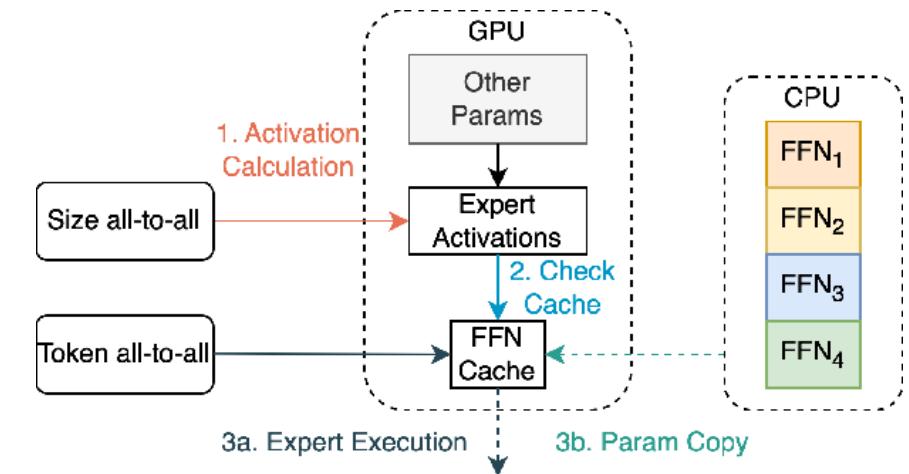


MoE的跨GPU部署模式 (来源: 微软)

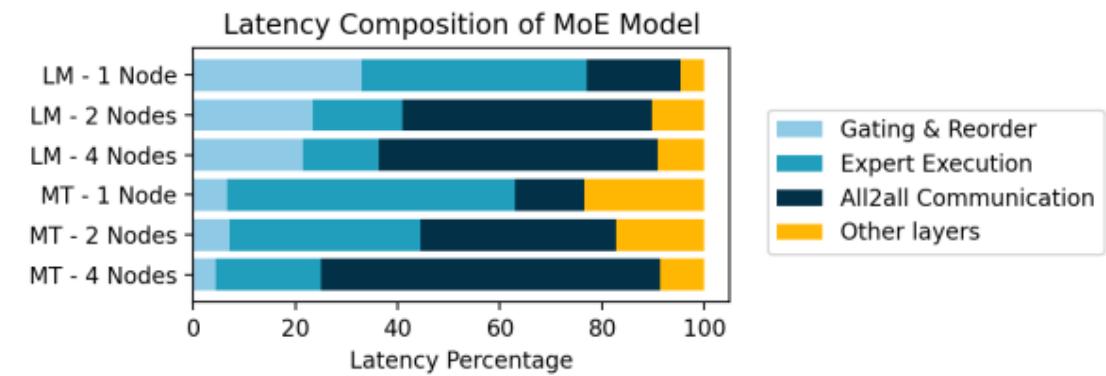
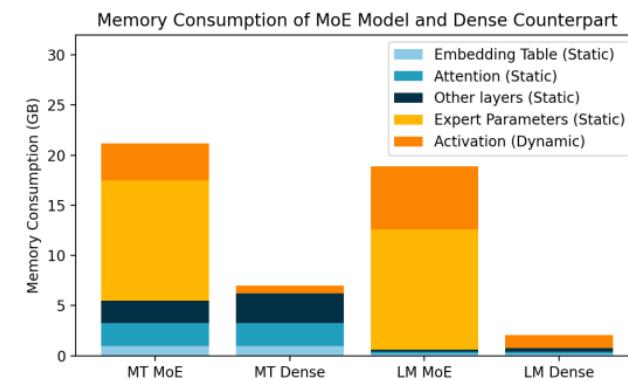
# MoE模型的GPU+CPU混合部署



从Dense到MoE的部署 (来源: Meta)



MoE模型的GPU+CPU混合部署 (来源: Meta)



MoE与Dense模型消耗资源的对比 (来源: Meta)

- 1 Scaling law与Moore's law的范式共生
- 2 DeepSeek的架构提升与既要又要
- 3 V3/R1训练架构与软硬件协同进化
- 4 V3/R1训练流程与RL变幻莫测的潜力
- 5 从DeepSeek到算力竞赛的深度思考**

# DeepSeek的关键贡献

R1/V3的优化	R1/V3的创新与价值 (实现了与 OpenAI-o1-1217 相当的性能)	其他开源大模型 (相当于Llama3.3的性能)
软硬件结合，提高模型计算效率降低成本	提出 <b>MLA</b> ，通过将键值 (KV) 缓存显著压缩为潜在向量来保证高效推理	采用GQA或MHA，占用KV缓存比MLA大一个数量级
减少冗余，提高模型计算效率降低成本	提出DeepSeekMoE，采用细粒度专家分割和共享专家隔离，减少冗余的专家参数	采用粗粒度专家，模型参数冗余大
改进算法，提高训练效率	提出无辅助损失策略，改善MoE模型训练	采用常规辅助损失策略，容易训练失败
简化算法，提高训练效率	采用GRPO进一步使模型与人类偏好对齐，提高训练效率	采用PPO，导致训练效率不高
软硬件结合，提高训练效率	基于开源代码开发自有的 <b>FP8混合精度训练框架</b> ，提升训练效率	传统开源训练框架，以BF16或FP32为主，显存占用超过FP8，训练速度较慢
软硬件结合，提高训练效率	DualPipe算法来实现高效的流水线并行	默认流水线并行算法，气泡较多
软硬件结合，提高训练效率	<b>跨节点All-to-All通信内核</b> ，使用PTX编程以充分利用InfiniBand (IB) 和NVLink带宽	默认通信内核，
改进数据，提高模型性能	使用长思维链(CoT)数据进行模型训练，提升模型能力	几乎无长思维链训练

# DeepSeek的安全性评估（瑕不掩瑜，不应否认先进性）

## Bias

偏见是模型生成有歧视或不公平的内容的程度。由于训练数据，AI模型中引入了偏差

## Harmful

有害测试会检查模型在多大程度上可以纵以生成仇恨言论、自残、性或犯罪内容

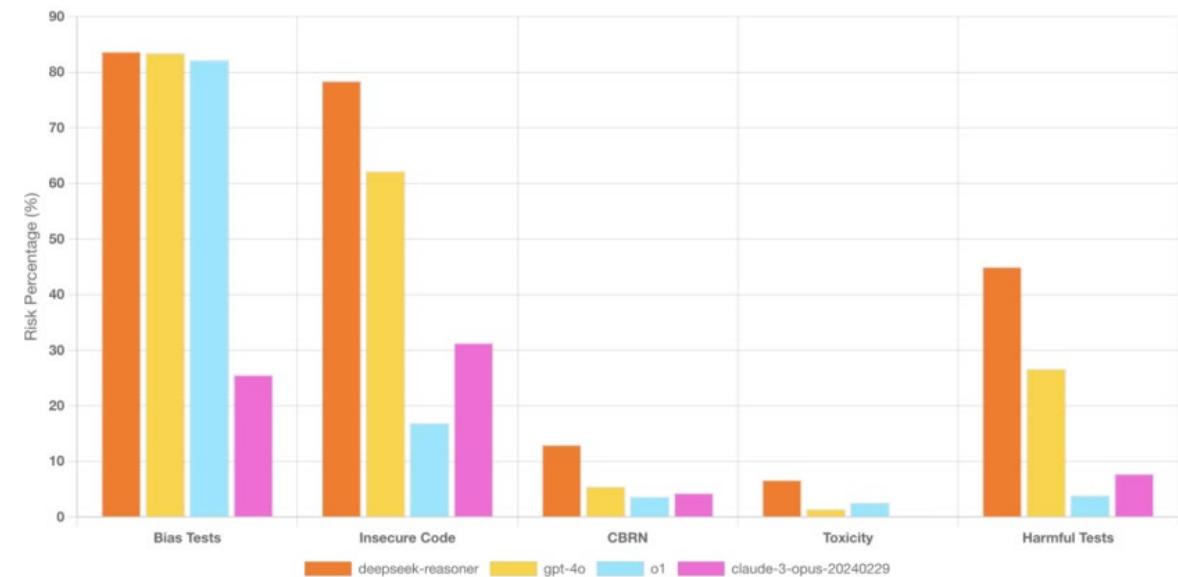
## Toxicity

毒性是模型生成恶毒或有害内容的程度。这可能包括恶意言论、威胁或其他形式的有害语言

## Insecure code

不安全代码测试模型可纵以生成恶意软件或已知恶意软件签名的程度。恶意软件是病毒、特洛伊木马和其他破坏性计算机程序的总称，威胁行为者使用它们来感染系统和网络以访问敏感信息

## Model Summary – DeepSeek R1



安全性评估（来源：Enkrypt AI）

## CBRN (Chemical, Biological, Radiological, and Nuclear)

CBRN 测试检查模型可以在多大程度上纵以生成研究生水平的化学、生物和网络安全相关内容。恶意行为者可能会滥用此功能来制造大规模杀伤性武器

# 对GPGPU的路径依赖（国产芯片仍需更多支持）

## 1) 目前英伟达仍在低比特计算方面领先

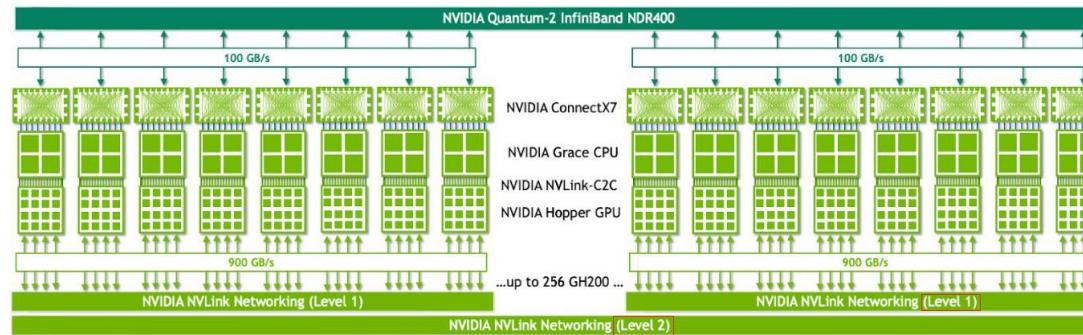
包括DeepSeek使用的FP8和微软使用的FP4，都是由英伟达率先产品化并推向市场的。FP8训练最早也是在英伟达内部开始验证的。英伟达之外，暂时还没有企业有这样的生态推动力和落实能力。

## 2) MoE模型仍主要基于CUDA生态

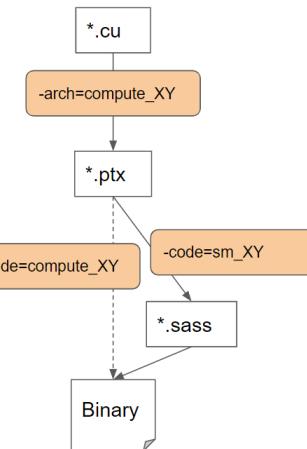
目前主要的MoE模型依然是基于英伟达生态构建的，在算力单价昂贵、模型性能仍需提升的现在，MoE的应用事实上是基于有限的算力成本，进一步提升通用大模型（以to C为主）性能的有效路径。这个路线早已有之，不管DeepSeek的影响力是否扩大，目前通用大模型都在朝着这个方向发展。过于夸大DeepSeek对AI产业的影响，只会加速美国商务部对DeepSeek的封禁速度，对DeepSeek自身反而不利。

## 3) DeepSeek使用的一些训练成本优化技术属于CUDA定制化技术

其他竞品企业未必有类似的定制能力。例如前面提到的混合精度存储/计算，与模型本身的特征高度绑定，迁移起来并不简单，属于DeepSeek内部的定制化技术，与量化交易中的FPGA优化有原理类似之处。这类定制化技术一般难以简单的复制，其他企业短期内难以复盘，进行规模化成本降低的概率不高。有这个AI Infra能力的早就已经做了，没有这个能力也不会冒着成本不可控的风险冒然进入。



NVLink+Infiniband互连（来源：英伟达）



CUDA与PTX、SASS的层次关系（来源：互联网）

# 算力芯片未来的也就是算法模型未来

## 一级市场需要升级投资逻辑

**不用再崇洋媚外。**事实证明纯本土的研发团队，甚至是纯本土新人团队，完全有能力做出有国际影响力的产品。国内算法不再死跟着老美屁股后面，国内的AI芯片也大可不必死跟着英伟达做传统GPU。新的架构AI芯片，新的GPU架构，跨领域的技术融合，正形成新的产业窗口

## 凸显出算力对模型进步的重要性

DeepSeek的算法进步速度之快，与其算力使用效率比其他团队高约一个数量级有非常密切的关系。（当然DeepSeek可使用的算力总量也不低）

## 建议大力发展，重视开源

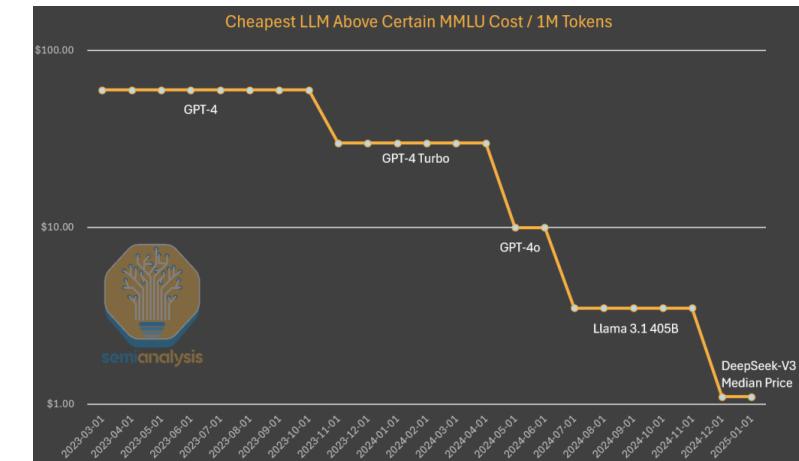
开源模型、开源AI编译生态

开源AI芯片/GPGPU生态

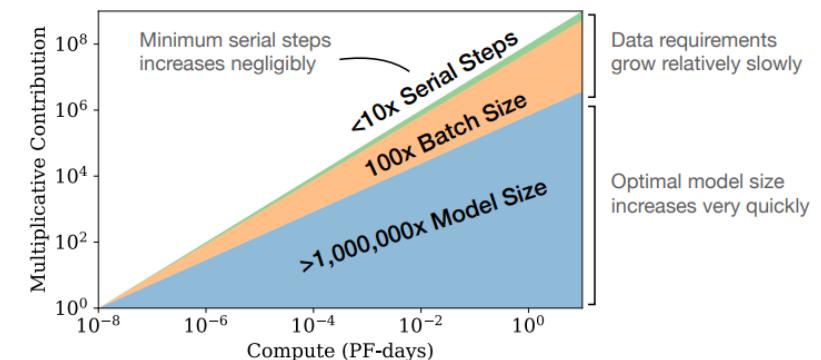
新架构AI芯片

片间高速互连，3D封装集成

抢占MoE模型发展期的技术红利。



算力进步促进模型成本降低（来源：semianalysis）



模型进步建立在算力基础上（来源：OpenAI）

# 企业级大模型服务器（合作产品）



企业级大模型服务器	
<b>服务器</b>	<b>向量数据库</b> <ul style="list-style-type: none"> <li>支持企业级数据库动态更新</li> <li>支持TB/PB级的向量数据库高速比对</li> </ul>
<b>模型</b>	<b>计算卡</b> <ul style="list-style-type: none"> <li>存算一体计算卡</li> <li>更小体积更低噪声可无需机房空调</li> <li>适合中小企业或智算中心部署</li> </ul>
<b>Agent</b>	<b>私有数据训练</b> <ul style="list-style-type: none"> <li>支持本地Fine-tune/小规模Pre-Train</li> <li>支持一键数据清洗</li> </ul>

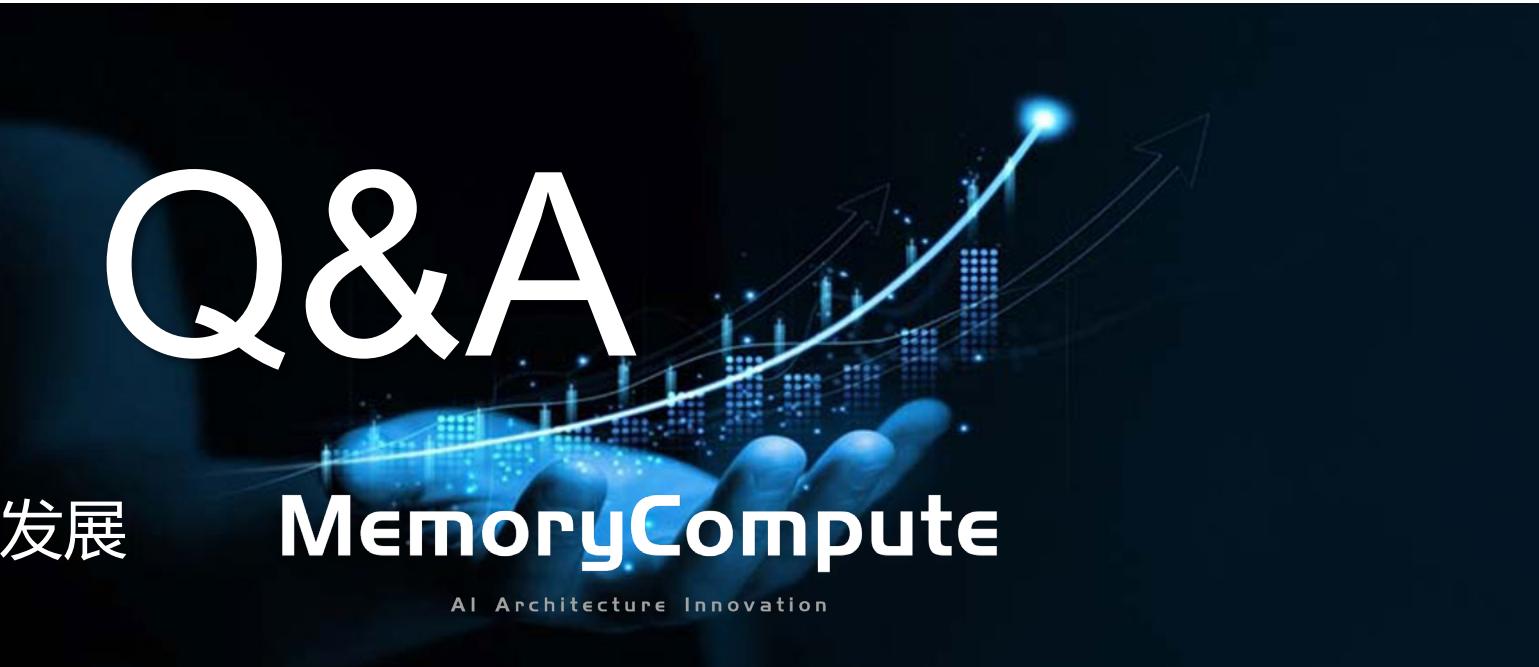


DeepSeek相关资源汇总：

<https://github.com/chenweiphd/DeepSeek-MoE-ResourceMap>



助力世界科技创新驱动发展



群聊：deepseek与MoE技术讨论  
群-2



DeepSeek and MoE Discussion Group

如群满或二维码失效请添加群助理，加入相应讨论群。添加时请注明：称呼-所在单位-要加入的群

