AI

Red Hat

# 推理入门

## 加速通往高效之路

# 目录

# 简介

优化 AI 模型推理是降低基础架构成本、减少延迟并提高吞吐量的最有效方法之一，尤其当企业组织在生产环境部署大型模型时，这种优化更为关键。

本电子书介绍了推理性能工程和模型优化的基础知识，重点探讨量化、稀疏化和其他有助于降低计算和内存需求的技术，以及虚拟大语言模型（vLLM）等能够提升推理效率的运行时系统。

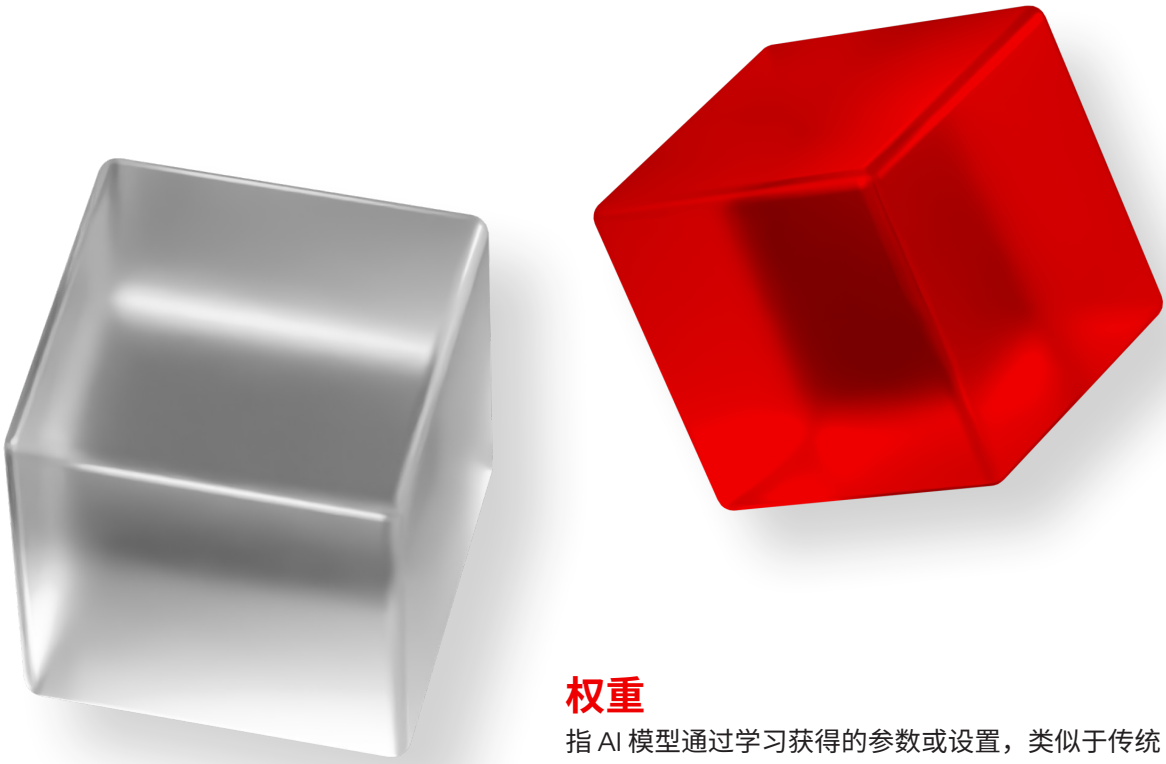
本电子书还概述了使用红帽的开放式方法、经验证的模型存储库以及 LLM Compressor 和红帽® AI 推理服务器等工具的优势。无论您是在图形处理器单元（GPU）、张量处理单元（TPU）还是其他加速器上运行，本指南都将提供切实可行的见解，帮助您构建更智能、更高效的 AI 推理系统。

# 关键术语概览

## 了解模型组件

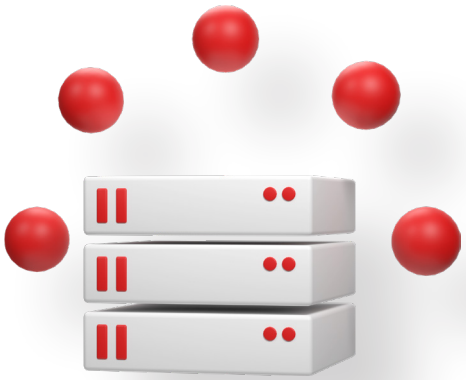
### 激活值

指模型处理信息（输入词元）时生成的临时数据，类似于计算过程中产生的中间结果。通常需要高精度的激活值才能保证结果的准确性。



### 权重

指 AI 模型通过学习获得的参数或设置，类似于传统软件中的配置文件或设置。权重决定了模型分析和预测数据的方式，通常在降低精度后仍能有效运作。



# 量化

量化通过使用较低精度的格式存储 AI 模型参数（权重）和中间数据（激活值），减少每个数值占用的位数，从而缩减 AI 模型的规模并降低资源需求。这项技术有助于高效管理资源，类似于在计算机上压缩文件。如果操作得当，量化**不会显著降低模型性能**。

- **权重量化**可压缩模型参数的存储体积，**提升推理过程中的内存使用效率**。[1]
- **激活值量化**可最大限度降低推理过程中中间输出（临时数据）的内存需求，**提高执行速度和效率**。[2]
- **KV 缓存量化**可缩减缓存键值张量的内存占用量，帮助模型**更高效地处理长提示和并发请求**。[3]

## 16 位、8 位和 4 位量化的精度级别：

- **16 位（FP16/BF16）**是标准精度，可保持准确性，但需要占用大量内存，对超大型模型而言成本高昂。
- **8 位（FP8/INT8）**可将内存使用量减少至 16 位的约一半，在保持模型准确性的同时显著提升效率。
- **4 位（INT4）**可大幅缩减模型规模并降低内存需求，从而能够利用更少的资源实现部署；但是，除非采用先进量化方法进行精细管理，否则可能会导致准确性明显下降。

1 Laboone、Maxime，“**权重量化简介**”，*towards data science*，2023 年 7 月 7 日。
2 “**AWQ：用于 LLM 压缩和加速的激活感知权重量化**”，GitHub，2025 年 8 月 8 日访问。
3 Turganbay、Raushan，“**通过键值缓存量化实现更长文本生成**”，*Hugging Face*，2024 年 5 月 16 日。

# 通过稀疏化减少计算负载

**稀疏化**通过将模型的部分参数有意设置为零来减少计算需求，使系统能够绕过不必要的运算，例如跳过表单上的空白部分。这样一来，无需完全重新训练模型即可提高速度和效率。

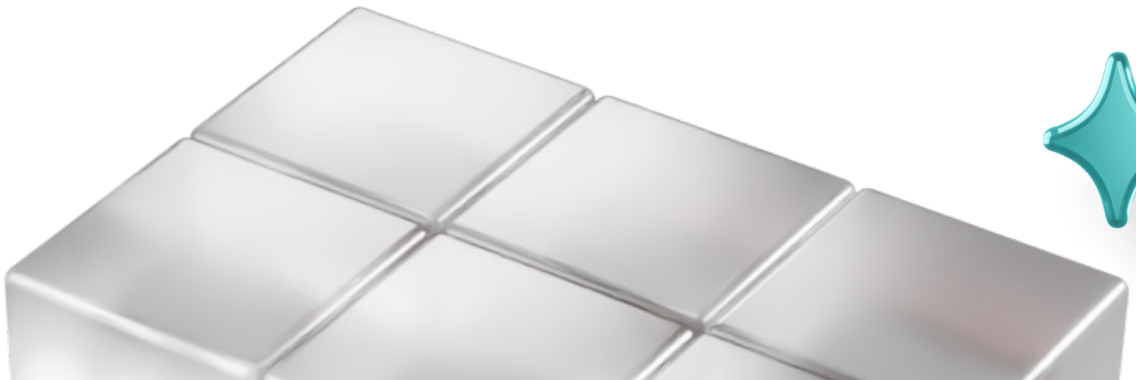
**2:4 稀疏化**是一种结构化方法，它精确地将每 4 个参数中的 2 个设为零，使专用硬件能够快速识别并高效绕过这些非活跃参数块，从而节省计算时间，实现更快的性能。

# 大语言模型的演进

大语言模型（LLM）主要基于 Transformer 架构构建，已从研究实验演变为驱动实际应用的基础工具。这些模型的规模通常达到数百亿甚至数千亿参数，能够实现高水平的推理、创造力和领域特异性。这些能力均通过一个称为“推理”的过程得以实现。

推理是指经过训练的模型处理新输入数据并生成输出的过程，例如预测句子中的下一个词或识别图像中的物体。与需要从大型数据集中学习的训练不同，推理侧重于应用所学知识来做出实时决策。因此，推理必须快速高效，特别是在生产环境中部署模型以支持交互式应用、实时分析或大规模自动化时，这一点尤为重要。

推理模型将文本、图像或音频等输入数据作为词元进行处理，通过多层 Transformer 架构传递这些词元来生成预测结果。词元是输入数据被拆分后形成的离散单元，拆分后再由模型进行处理。在基于文本的模型中，词元可表示单个字符、子词或整个词，具体取决于所用的词元化策略。



这些模型将输入词元传递到深度、多层的 Transformer 架构进行处理，通过应用一系列数学运算来分析上下文、权衡关联性，并确定可能的输出。每一层都在逐步优化模型对输入的理解，最终每次生成一个词元作为预测结果。这种逐步生成词元的方式可产生高度准确且贴合上下文的输出，但也增加了推理工作负载的计算负担，对于层数众多的大型模型尤其如此。

除了基于文本的 LLM，类似架构如今也已成为包括视觉模型和多模态系统在内的众多 AI 领域的基石。视觉模型同样应用这些基于词元的 Transformer 计算原理来处理图像和视频。它并非将文本拆分为词元，而是将像素数据转换为嵌入。这些嵌入可捕获空间模式、边缘、纹理以及视觉元素之间的关系，使模型能够执行图像分类、目标检测、分割和视觉问答等任务。在生产环境中完成部署后，视觉模型可支持自动化检查、医学成像和内容审核等用例。

随着企业组织更广泛地采用 AI，模型架构的规模和复杂性也在持续增长。混合专家模型（MoE）等新方法旨在通过每次推理仅激活模型的部分组件来提升性能，从而减少所需的总计算量。这些创新为打造更强大的模型开辟了道路，同时有助于在性能、成本和能耗之间取得平衡。

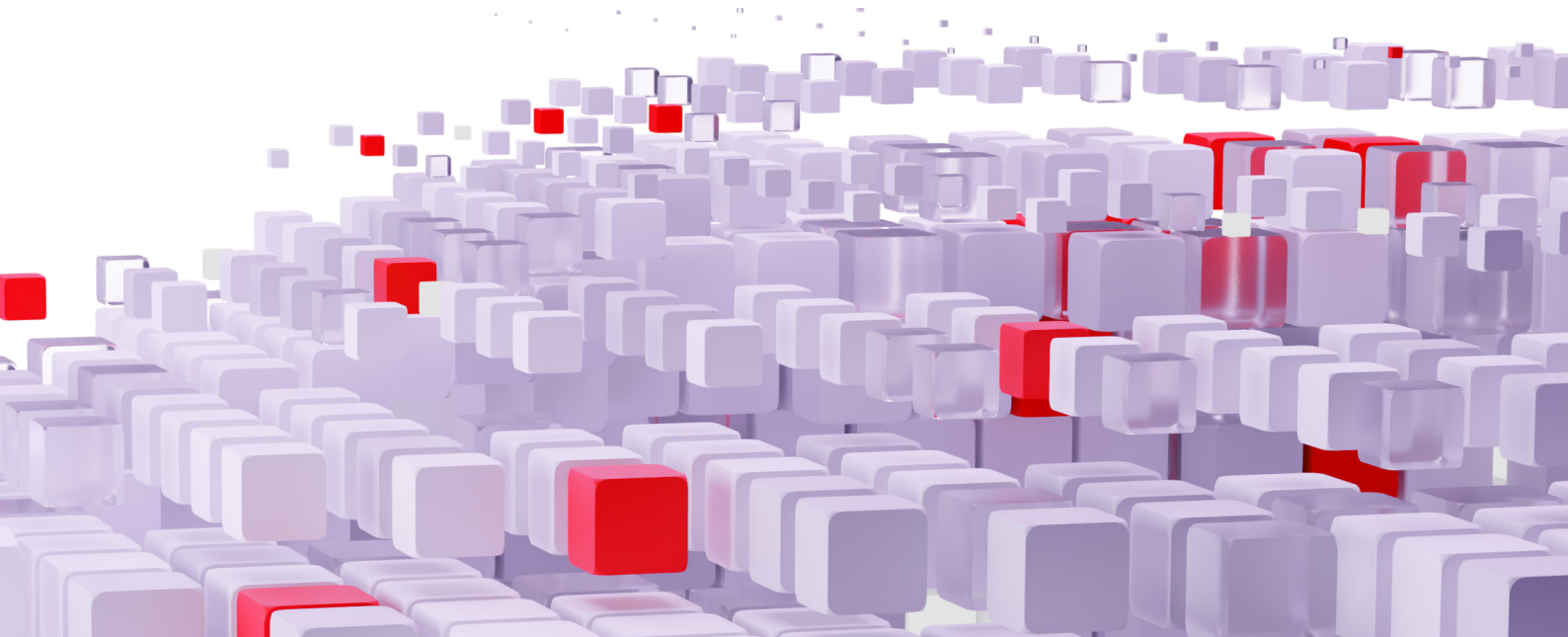
模型无论规模大小，若要在生产环境中具备实用性，都离不开高效的部署、服务与优化。因此，对于希望部署模型的企业组织而言，推理性能工程是核心要务。

# 推理服务面临的挑战

为大型模型提供推理服务面临着诸多挑战。

具有数十亿参数的模型需要大量 GPU 内存来存储其权重和中间状态，如键值（KV）缓存。随着并发请求数量或输入长度的增加，内存限制会成为关键瓶颈，制约模型的吞吐量和响应能力。基本的服务方法常常受限于低效的批处理技术，导致硬件资源利用率不足和延迟增加。

此外，在 Transformer 架构中实施注意力机制可能属于计算密集型任务，尤其是在处理长输入的情况下，这会显著拖慢响应速度。要应对这些挑战，需要实施精细的运行时优化，例如高效内存管理、高级批处理策略，以及分页注意力等经过优化的注意力机制。这些优化措施共同作用，可以提升实际应用中的性能和响应能力。



# 全栈式的**推理性能**优化方法

推理优化是指在将 AI 模型部署到生产环境后，提高其运行效率的过程。在生产环境中运行 LLM 的成本可能会迅速升高，尤其是在处理高词元数量、长提示以及不断增长的使用需求时，更为昂贵。**推理中的成本优化，归根结底就是在不牺牲准确性和用户体验的前提下，降低内存消耗、提高吞吐量，并尽可能减少硬件需求。**

虽然模型训练通常是单次性任务（模型再训练除外），但推理会持续进行，通过实时生成输出来响应用户输入。对于 LLM 和视觉模型而言，推理可能会迅速成为 AI 部署中成本最高、资源消耗最密集的环节，尤其是在跨混合或全球基础架构进行扩展时，这一挑战更为严峻。

要大规模、高效地提供 LLM 服务，需要采用全面的全栈式优化策略，既能解决模型本身的问题，又能解决服务运行时问题。虽然我们主要着眼于通过**量化**和**稀疏化**来优化模型参数，但也可以通过**分块预填充**[4]、**前缀缓存**[5]、**推测解码**[6]以及**解耦式预填充与解码**[7]等技术来优化推理服务过程，从而进一步提升性能。

4 “优化与调优”，*vLLM*，2025 年 8 月 7 日。

5 “什么是自动前缀缓存？”，*vLLM*，2025 年 8 月 8 日访问。

6 “推测解码如何将 vLLM 性能提升高达 2.8 倍”，*vLLM*，2024 年 10 月 17 日。

7 Du、Kuntai，“vLLM 办公时间 - vLLM 中的解耦式预填充和 KV 缓存存储 - 2024 年 11 月 14 日”，*YouTube*，2024 年 11 月 18 日。

# 推理运行时和模型格式概述

由于基本运行时往往会成为性能瓶颈，因此要高效地为大型模型提供服务，需要选择合适的推理运行时。热门运行时方案包括：

- **vLLM**。虚拟大语言模型是一个由 vLLM 社区维护的开源代码库。该代码库有助于 LLM 更高效地大规模执行计算。具体而言，vLLM 是一种推理服务器，可通过更高效地利用 GPU 内存来加快生成式 AI 应用的输出速度。凭借分页注意力（允许在 GPU 内存中高效处理更多词元）等创新技术，vLLM 拥有卓越的吞吐量和低延迟性能，因此在业界得到了广泛采用。

- **Triton**。Triton 常被误认为是独立运行时，但它其实更多的用作各种后端引擎（包括 TensorRT 和 vLLM）的前端应用编程接口（API）。虽然在 NVIDIA GPU 上搭配使用 Triton 与 TensorRT 可能会略微提升性能，但代价是设置复杂性增加，且模型支持有限。客户普遍反映，要通过 Triton 实现性能提升，需要投入的精力远超 vLLM。

- **SGLang**。SGLang 是一个衍生自 vLLM 的较新运行时，针对特定用例进行了优化。它采用了许多与 vLLM 相同的底层组件，但支持的模型架构更少。虽然 SGLang 在为数不多的场景下可能优于 vLLM，但其灵活性有限且社区支持不足，难以吸引企业进行广泛采用。

# 提高模型效率的双重方法

## 1：优化推理运行时（vLLM）

### 运行时的局限性

如上一章所述，**由于基本推理服务方法存在固有局限性，高效运行 LLM 并非易事**。

这些运行时的局限性包括 GPU 内存使用效率低下、批处理能力欠佳以及词元生成速度缓慢等。运行时通常以低效方式存储 KV 缓存等中间计算数据，这会消耗大量 GPU 内存，从而限制处理并发请求的能力。此外，过于简单的批处理策略可能导致 GPU 闲置或利用率不足，从而显著降低吞吐量。同时，基本运行时的注意力机制运行缓慢，在处理长输入序列时会导致延迟大幅增加。

## 为什么选择 vLLM

vLLM 通过提供专门针对推理性能进行优化的先进技术，解决了诸多运行时难题：

- **连续批处理**：vLLM 通过并行处理来自多个传入请求的词元，最大限度地减少 GPU 空闲时间。它不再一次处理一个请求，而是将来自不同序列的词元分组到多个批次中，从而显著提高 GPU 利用率和推理吞吐量。

- **PagedAttention**：vLLM 采用一种名为 PagedAttention 的新型内存管理策略，能够高效处理大规模 KV 缓存。这项技术可动态分配和管理 GPU 内存页面，大幅增加了并发请求的数量，并支持更长的序列，而不会出现内存瓶颈。

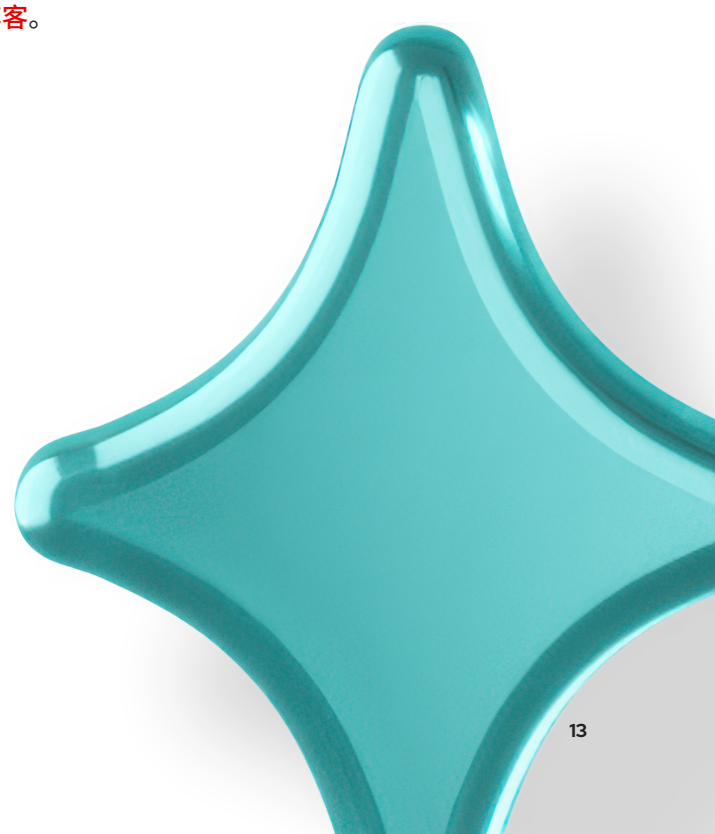
如需深入了解，请阅读这篇关于 vLLM 的技术博客。

## 部署 vLLM 的优势

**全面集成功能**：vLLM 可直接从 Hugging Face 等热门存储库加载模型，并在 Triton 推理服务器等框架内充当高性能后端。它与包括 NVIDIA GPU、AMD GPU 和 Google TPU 在内的广泛硬件平台兼容，进一步简化了企业级部署。

**标准化和厂商中立**：通过使用 vLLM 等广泛采用的运行时，企业组织可获得标准化优势，既能跨各种硬件环境实现可靠的性能，又能避免受制于专有解决方案。

如需深入了解 vLLM 的并行技术，请阅读这篇技术深度解析博客。



# 2：优化 AI 模型

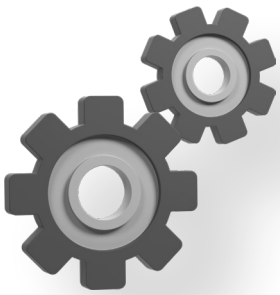
## 优化大语言模型的重要性

在生产环境中，主要挑战之一是管理内存和计算效率。大型模型通常需要占用大量 GPU 内存来存储参数以及 KV 缓存中的上下文，尤其是在处理长提示或多个并发请求时。如果模型未经优化，可能会低效运行，导致运维成本增加。另一个关键问题是延迟：用户期望获得实时响应，而模型的大规模或执行效率低下引发的延迟，可能会对用户体验和下游工作流效率产生负面影响。

## 为什么要压缩模型

压缩模型有助于解决企业组织在大规模部署 AI 时面临的两大核心挑战：成本效益和性能优化。

当模型规模增长至数十亿参数后，在生产环境中部署这些模型并提供服务会耗费大量资源，需要极高的内存和计算能力。模型压缩技术（包括量化和稀疏化）可在基本保持准确性的前提下，略微降低参数的精度和数量，同时显著减少内存占用和计算需求。通过压缩模型，企业组织可以使用更少的 GPU 或其他加速器更高效地运行 AI 工作负载，从而大幅降低运维成本并加快推理速度，这对于需要实时响应的应用至关重要。



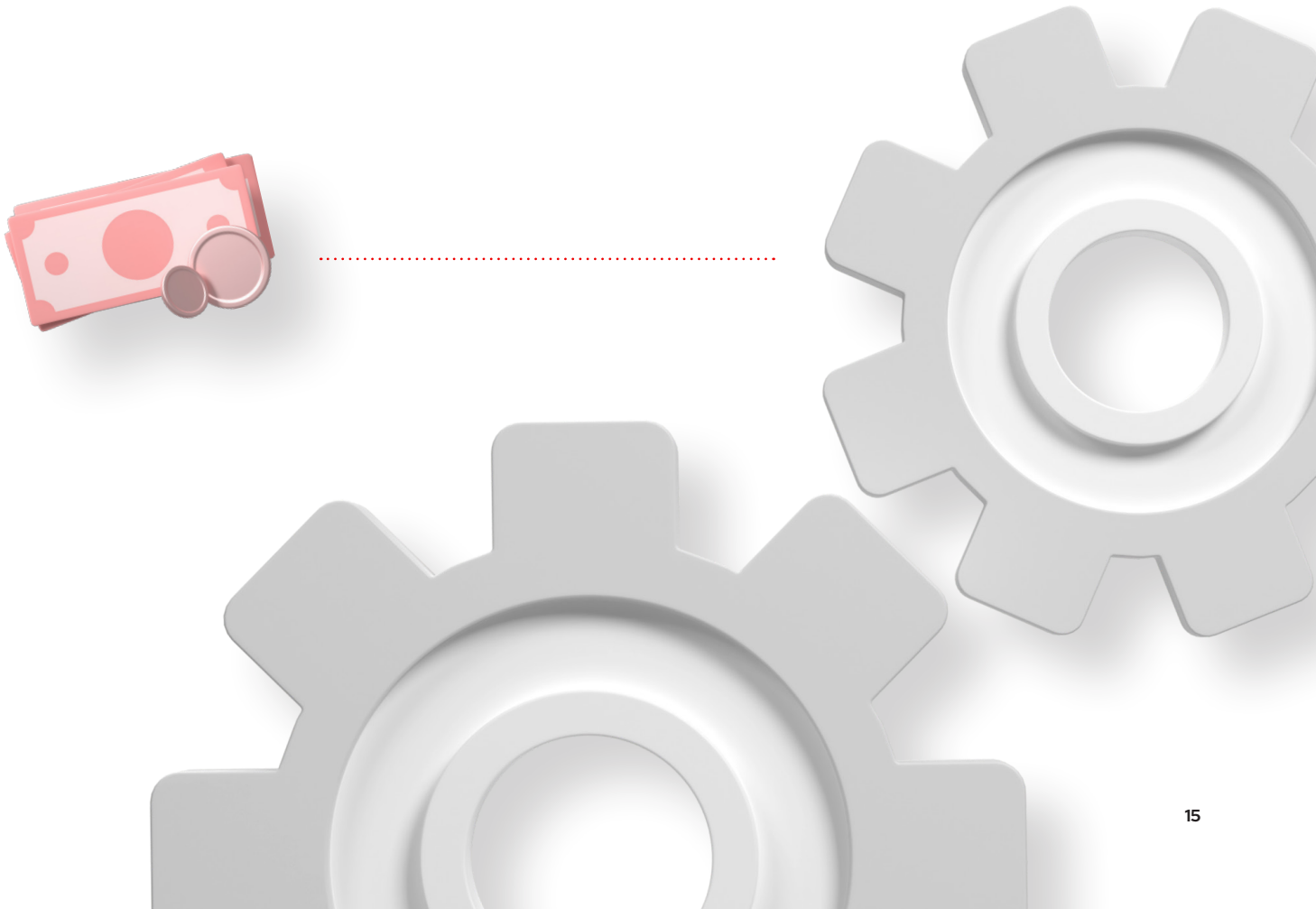
## 如何针对我的推理对模型进行成本优化？

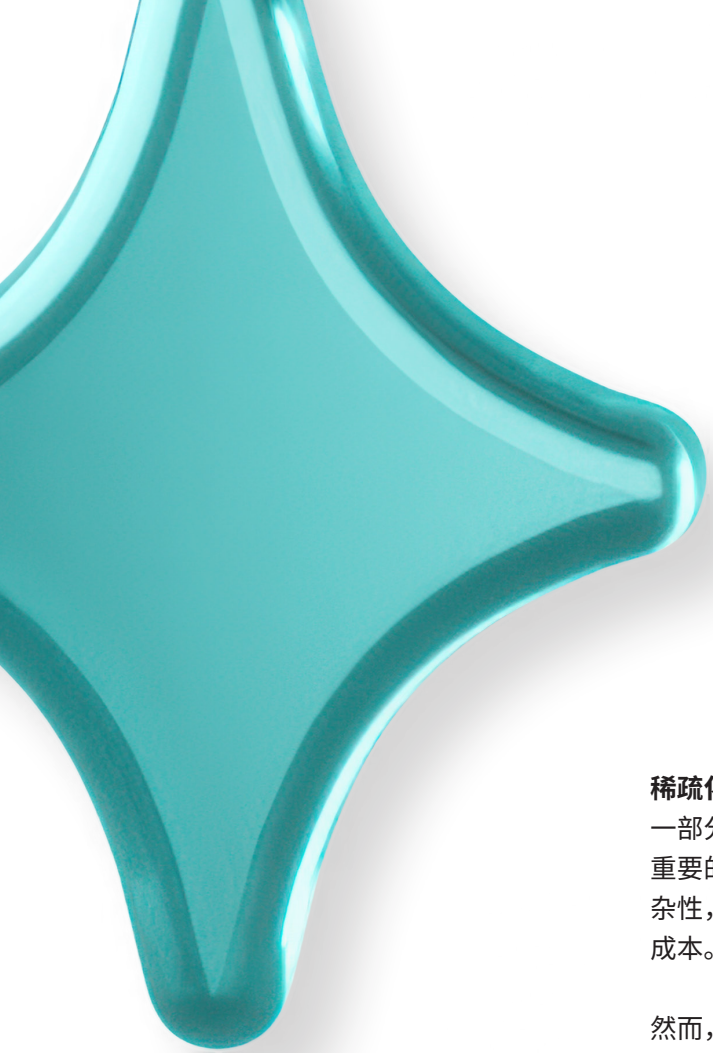
降低这些成本的最有效方法之一是压缩模型。量化和稀疏化等压缩技术可缩减模型规模并降低计算需求，使推理工作负载能够在数量更少或规格更低的 GPU 上运行。

**量化**通过降低模型数值（特别是权重和激活值）的精度来优化模型。通常，模型以 16 位精度（甚至 32 位精度）运行，采用 FP16 或 BF16 等格式。

量化会将这些数值压缩至更低精度的格式，如 8 位（INT8 或 FP8）甚至 4 位整数（INT4）。这一过程可显著减少存储模型参数所需的内存，从而大幅缩减模型规模。例如，一个包含 700 亿参数的 Llama 模型可从大约 140GB 缩减至 40GB。这种缩减不仅可以释放内存以用于额外计算，还能提高吞吐量，尤其适用于内存受限的场景。例如，搭载 48GB VRAM 的 GPU 处理 40GB 模型的速度比处理 140GB 模型的速度要快。

然而，过度量化可能会因精度损失而影响准确性。为缓解这一问题，细粒度量化采用缩放因子来保持模型准确性，通常能将精度损失控制在 1% 以内。量化可通过优化硬件利用率，使计算吞吐量翻倍，从而显著降低延迟和运维成本。





**稀疏化**通过结构化地减少参数来优化模型，本质上是将模型中很大一部分权重设为零。这项技术的工作原理是识别并剔除冗余或不太重要的权重，简化推理过程中的计算。稀疏化可以大幅降低模型复杂性，从而减少内存使用量和计算负载，加快推理速度并降低运维成本。

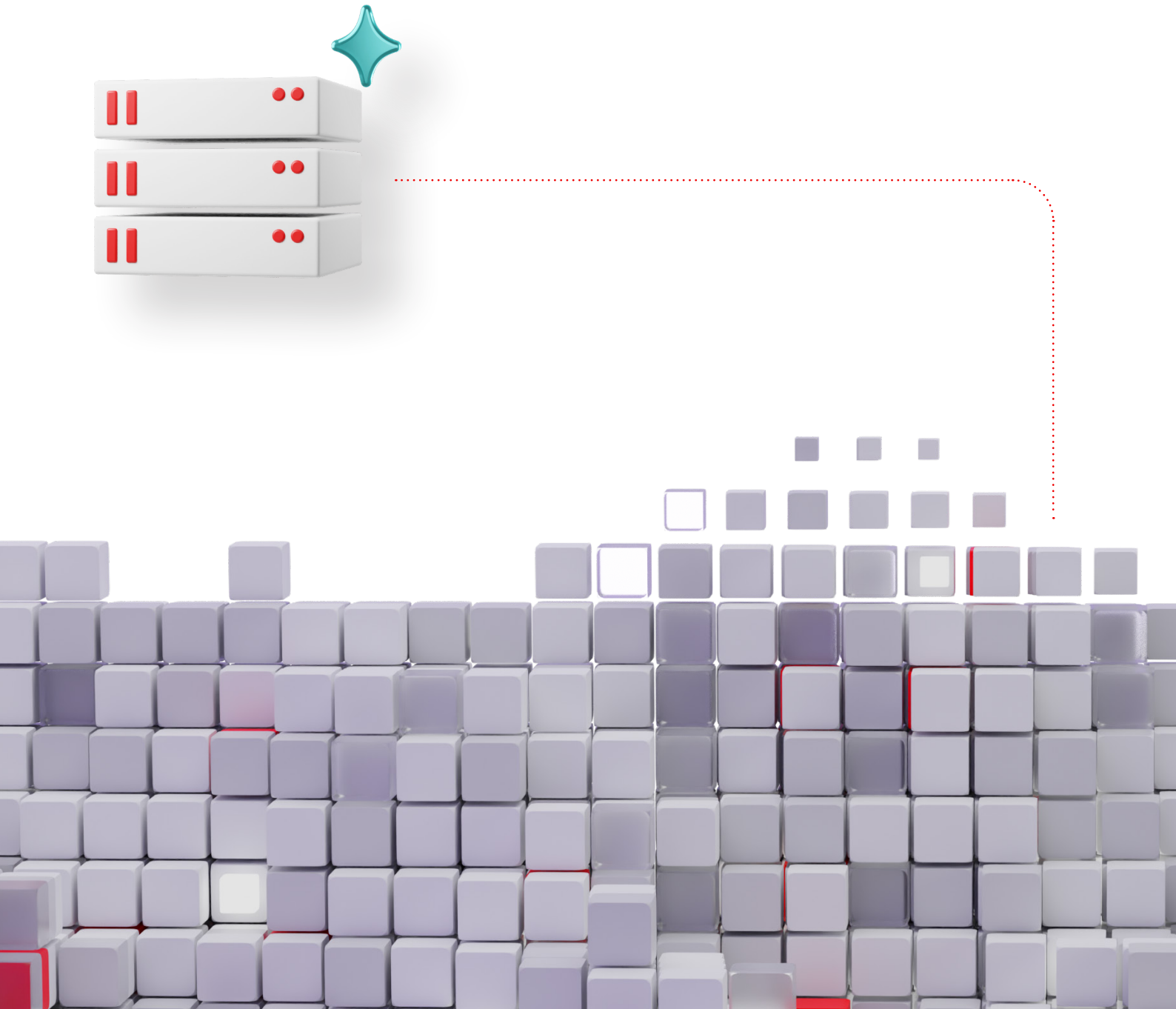
然而，若要有效实现稀疏化，需要对模型进行再训练，这一计算密集型步骤需要投入大量前期资源。稀疏化的效率取决于硬件功能，例如，由 GPU 等现代加速器支持的半结构化稀疏化技术，其特定的权重置零模式可以加快计算速度。这项技术的关键优势在于，如果实施得当，可以大幅降低计算需求。

尽管稀疏化能带来显著效益，尤其是与量化等其他优化方法结合使用时效果更佳，但其优化过程通常更为复杂。因此，建议在超大规模场景或具备专用硬件设置的环境中采用该技术。通过谨慎应用稀疏化，企业组织可以提高推理效率，但考虑到其复杂程度较高，通常建议将量化作为主要优化技术。

通过采用压缩工作流和经验证的运行时，企业组织可以更有效地管理运维成本，支持可扩展性，并为未来 AI 使用量的增长做好准备，同时避免过度投入基础架构资源。

## 准确性会受到影响吗？

虽然量化和稀疏化等模型压缩技术可以降低内存与计算需求，但其设计的核心目标是保持可接受的准确度水平。例如，8 位量化通常可提供接近基线的准确性，同时将内存消耗减半。使用权重舍入和校准等高级量化技术进行优化后，即使是 4 位模型也能保持强劲性能。而诸如 2:4 稀疏化这类结构化稀疏模式，允许硬件加速器跳过冗余运算，且不会降低输出质量。在许多生产场景中，团队能够显著节约资源，而模型性能仅有微乎其微的损失甚至没有下降。测试和验证仍然必不可少，但在大多数应用场景中，实施良好的压缩技术能够在无损准确性的情况下实现高效推理。

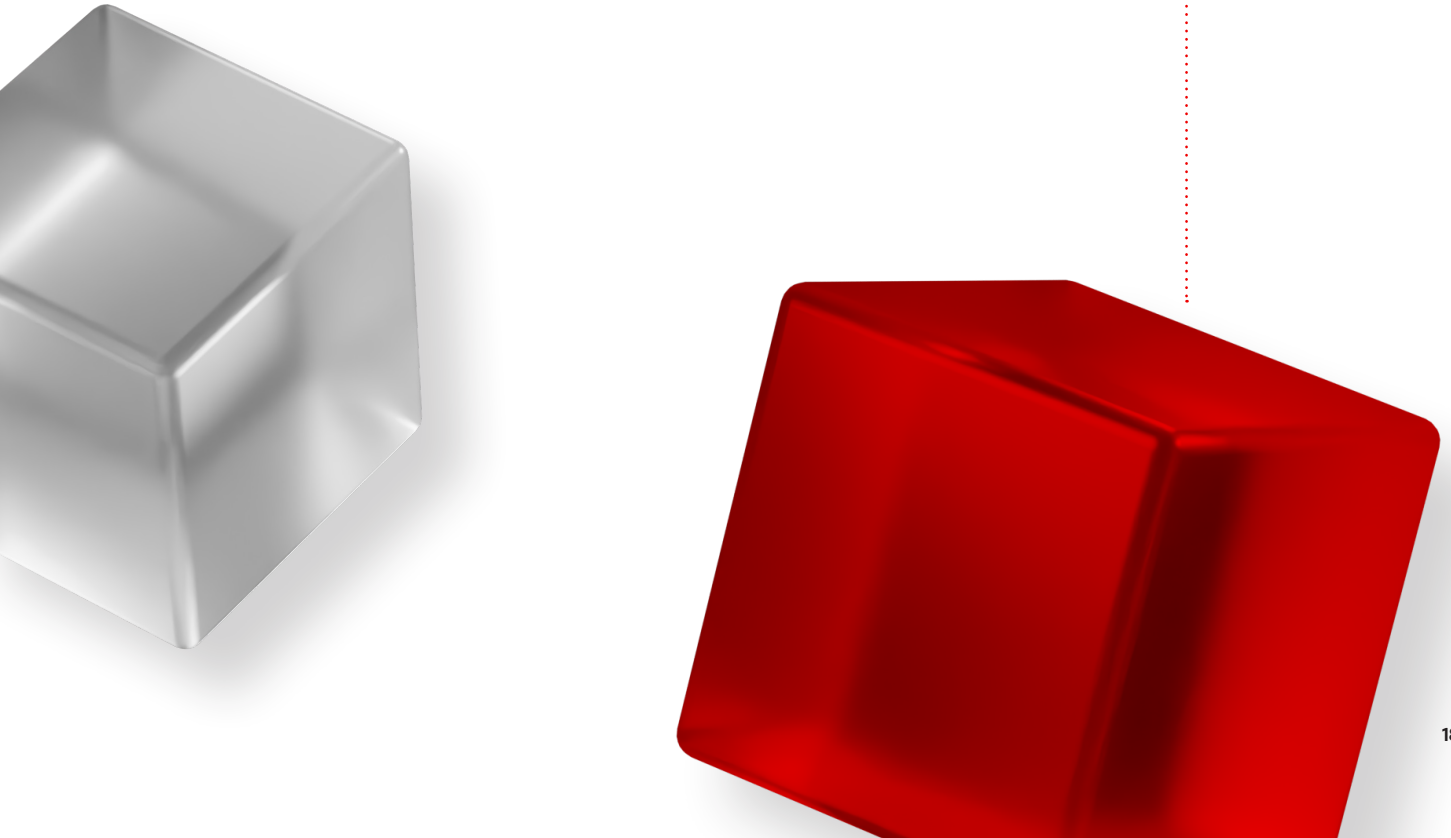


# 红帽 AI

## 什么是红帽 AI?

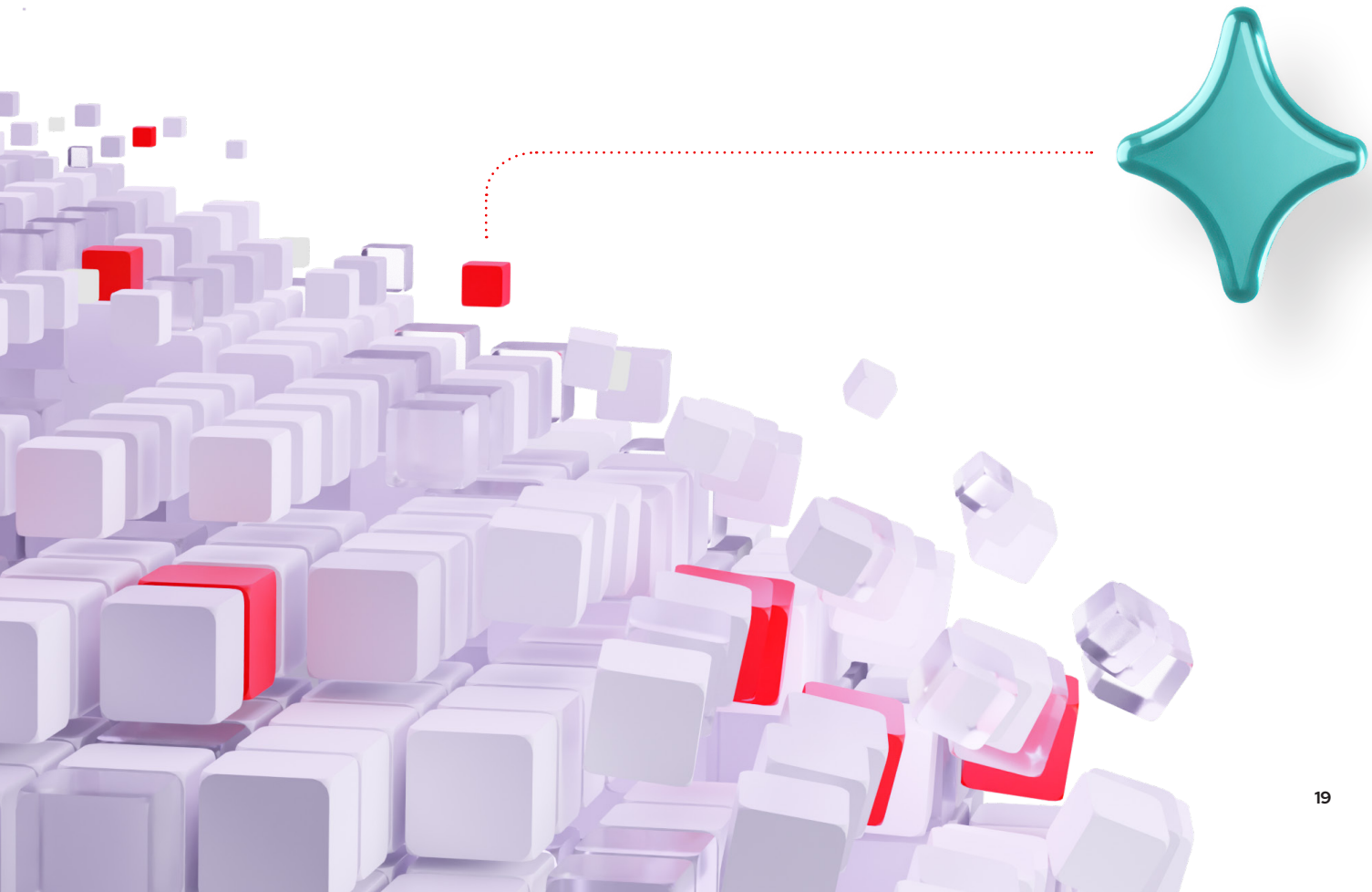
红帽 AI 平台是一个面向混合云环境的 AI 平台，帮助企业组织加速 AI 创新，同时降低开发和交付 AI 解决方案的运维成本。它可简化与私有数据的集成，通过优化的模型和高效的推理来帮助降低成本，并在可扩展、高度灵活的平台上加快交付代理式 AI 工作流。

红帽 AI 使企业组织能够大规模管理和监控预测性 AI 与生成式 AI 模型的整个生命周期，从单服务器部署到高度扩展的分布式平台皆可涵盖。该平台依托开源技术和广泛的合作伙伴生态系统，致力于在各类基础架构上实现卓越的性能、稳定性以及 GPU 支持。



红帽 AI 包括：

- **经过优化和验证的模型。**经过预先评估和性能测试的模型，可减轻测试和微调负担。
- **LLM Compressor。**帮助用户对热门模型进行量化和压缩的工具包，在降低推理资源需求的同时，保持准确性不受影响。
- **模型定制。**用于微调或适配基础模型以满足特定企业需求的工具。
- **高性能推理运行时。**基于 vLLM 且经过优化的运行时，采用先进的批处理和内存管理技术，实现高效、可靠且可扩展的模型服务。
- **LLMOps。**用于在生产环境中简化 LLM 部署、监控和管理的实践与工具。
- **AI 安全防护与评估。**用于评估模型准确性、公平性及稳健性的框架与方法，确保实现可靠且负责任的 AI 部署。
- **灵活且一致的扩展。**提供基础架构支持，确保跨混合云环境扩展 AI 时兼具灵活性与一致性。
- **代理式 AI 加速交付。**专为快速部署先进自主 AI 系统而设计的功能，助力企业组织始终处于 AI 创新的前沿。



# 利用红帽 AI 优化模型

红帽 AI 采用先进的技术帮助企业组织优化 AI 模型，从而实现效率、准确性和成本效益之间的平衡。

红帽 AI 重点关注模型优化的两个主要方面：高效的运行时和压缩模型。通过结合运用这些方法，红帽 AI 产品组合可在降低计算资源需求的同时，提供快速推理性能。具体而言，红帽 AI 推理服务器采用连续批处理和高效利用内存的方法，确保模型每秒处理更多词元，以更少的 GPU 使用量实现更高吞吐量。

红帽 AI LLM Compressor 为本电子书探讨的压缩技术提供了标准化实施方法，可在保持 99% 准确性的前提下实现优化。它帮助用户生成热门模型的优化版本，同时针对 vLLM 等推理运行时进行了调优。如此一来，便可在各类硬件设置上更轻松地运行高性能压缩模型。

红帽 AI 提供全面的验证服务，帮助企业组织自信从容地筛选、部署及扩展优化的模型。鉴于可用的 LLM 种类繁多，企业组织往往难以精准识别在准确性、性能和成本效益方面最契合自身用例的模型。为了应对这些挑战，红帽 AI 使用开源验证工具（如 GuideLLM、Language Model Evaluation Harness 和 vLLM），通过多项评估任务对模型性能进行严格的基准测试。这一验证支持结果复现，并为明智的模型选择提供依据，从而降低复杂性和不确定性。

红帽 AI 还提供**容量指导**，帮助企业组织准确规划 AI 基础架构并优化资源使用，解决硬件利用率不足、计算成本高昂和推理阶段效率低下等常见问题。经过验证的模型、优化的部署设置和量身定制的硬件建议相结合，使企业组织能够提高灵活性、加速部署并实现可预测的性能，同时有效地控制成本。

依托压缩技术和优化的运行时，红帽 AI 使大规模部署 LLM 变得切实可行，帮助团队满足不断增长的需求，同时保持对成本、复杂性和计算资产使用的有效管控。

# 后续步骤

希望降低 LLM 部署与运维的成本和复杂性？详细了解**红帽 AI 推理服务器**，或联系您的红帽代表以开始使用。