基于注意力机制的 Transformer 模型并行计算架构设计研究

苏泽宇
广东工业大学，广东 广州 510520
DOI:10.61369/ASDS.2025070016

摘　　要： 研究针对大规模 Transformer 模型训练所面临的计算效率和资源瓶颈，提出一种基于注意力机制的并行计算体系结构设计方法。在分析 Transformer 模型计算特点和并行要求的基础上，设计分层任务划分策略，优化计算负荷分配；建立混合并行通讯调度框架，降低交互代价，实现计算和存储资源的高效协同管理；构建弹性容错机制，保证分布式训练的稳定性。重点解决自注意机制二次复杂性导致的计算困难，以及大规模模型训练过程中通信和存储瓶颈。从细粒度的并行设计和系统优化两个方面，为百亿参数的 Transformer 模型的高效训练提供切实可行的技术方案。

关 键 词： 注意力机制；Transformer 模型；并行计算；架构设计

Research on Parallel Computing Architecture Design of Transformer Model Based on Attention Mechanism

Su Zeyu
Guangdong University of Technology, Guangzhou, Guangdong 510520

Abstract: This study addresses the computational efficiency and resource bottlenecks faced in the training of large-scale Transformer models and proposes a parallel computing architecture design method based on the attention mechanism. Based on the analysis of the computational characteristics and parallel requirements of the Transformer model, a hierarchical task division strategy is designed to optimize the distribution of computational load. Establish a hybrid parallel communication scheduling framework to reduce interaction costs and achieve efficient collaborative management of computing and storage resources. Build an elastic fault-tolerant mechanism to ensure the stability of distributed training. The focus is on addressing the computational difficulties caused by the secondary complexity of the self-attention mechanism, as well as the communication and storage bottlenecks during the training of large-scale models. From the aspects of fine-grained parallel design and system optimization, it provides practical and feasible technical solutions for the efficient training of Transformer models with hundreds of billions of parameters.

Keywords: attention mechanism; Transformer model; parallel computing; architecture design

引言

近年来，深度学习模型规模不断扩大，特别是以 Transformer 框架为基础，已在自然语言处理和计算机视觉等领域展现出优异的性能，然而其计算复杂度和内存需求呈现二次方型增长，给硬件性能带来了严峻挑战。传统的单设备训练模式受限于存储空间和计算资源，很难支持百亿量级参数下的模型高效训练，分布式并行是突破该瓶颈的重要技术途径。然而，已有的数据并行、模型并行和流水并行方法在通信开销、负载平衡和计算效率等方面仍存在明显不足，特别是在注意力机制下的计算 - 通讯耦合问题更加突出。现有研究主要集中在单一维度上进行优化，缺少对跨域计算特征和异构硬件体系结构的协同分析，导致计算资源利用率低。

一、Transformer 模型计算特性与并行需求分析

Transformer 模型的计算特点是其自注意力机制和多层前向网络（FFN）相结合，其计算复杂度随序列长度呈二次方型增长，而其计算代价与模型维呈二次函数关系。在长序列场景中，注意力矩阵生成和软件计算成为存储资源瓶颈，而矩阵乘法计算

作者简介：苏泽宇（2004-），男，汉族，湖南醴陵人，本科，研究方向：信息与计算科学。

密集特性对硬件计算能力提出了更高的要求。传统的单显卡训练受限于内存存储空间和运算单元吞吐能力，很难有效地处理大规模参数和长序列输入，分布式并行是必然的选择[1]。然而，已有的并行策略面临着明显的挑战：数据并行引入梯度同步代价，模型并行依赖于注意力头部导致通讯延时增大；流水线并行因 Transformer 前馈特性而产生气泡速率问题。特别值得一提的是，自注意力机制具有动态稀疏性，传统的静态切分方法很难适应其非平衡计算负载分布。此外，不同硬件结构（GPU、TPU 等）在矩阵计算和通讯带宽等方面的差异进一步加剧了并行优化的复杂度[2]。为此，需要充分考虑 Transformer 计算图结构、数据相关性和硬件特性等因素，设计细粒度的计算任务划分策略，减少通信开销，提高计算资源利用率。

二、基于注意力机制的 Transformer 模型并行计算架构设计

（一）计算任务划分与分布式执行规划

该模型将注意力计算分为三个核心部分：注意力计算，前馈网络计算，分层规范化，其中注意力计算占了主要的计算开销。为解决自注意机制的二次复杂性，采用序列与注意力头两个维度的联合分割策略，实现 QKV 矩阵在不同计算节点上的分解，降低单个节点的内存占用。在前馈网络中，为了保证矩阵乘法运算的负载平衡，采用了列并联和行并联的分割方法。计算任务划分需要与硬件拓扑相结合，优先考虑高带宽设备之间的通讯需求，避免跨节点间数据传输的“瓶颈”[3]。分布式执行规划采用静态调度和动态调整相结合的方法，根据计算图上的依赖关系对任务执行序列进行优化，以缩短设备闲置时间。针对长时间序列的注意力计算，采用块稀疏化的方法进行注意力计算，只保留局部注意区域和全局重要联系，减少计算和通讯开销。在计算节点之间采用异步梯度融合的方法，降低了节点之间的同步延迟，并利用梯度压缩技术减少了数据传输的开销。

（二）通信拓扑结构与数据交换优化

根据 Transformer 的计算特点，采用混合并行方式，将数据并行、张量并行和流水并行相结合，以满足不同计算阶段通信需求。在并行组中，采用 All-Reduce 的梯度融合方法；在张量并行组中，使用 All-Gather 与 Reduce-Scatter 实现张量片的融合。为了降低节点间的通讯延时，需要采用高带宽的互联设备，例如 NVLink、 InfiniBand 等[4]。在注意力计算中，QKV 矩阵的传输采用了重叠通信和计算相结合的方法，将通信开销隐藏起来。

数据交互优化的重点是降低冗余通信、提高带宽利用率。避免历史序列数据的重复传输，基于键值的缓存共享机制。针对前馈网络的权值梯度同步问题，先局部融合，再全局同步的分层融合策略，减少了通信量。通信协议的选择需要与硬件特征相结合，如 GPU 集群采用 NCCL 优化群集通信，TPU 集群采用专用高速互联技术。

（三）计算与存储资源协同管理

Transformer 模型在训练过程中需要大量的内存和计算资源，需要通过计算和存储协同管理来提高资源利用效率。在内存管理中，采用动态块技术，把大权重矩阵和中间激活值分割成可互换的内存块，并根据需要装载到计算单元中。该算法采用激活值再计算技术，以牺牲部分运算时间为代价，减少内存占用，适合长序列训练场景[5]。存储资源分配需要与计算任务优先级相结合，保证高频访问数据驻留在缓存中，降低数据载入时延。

在计算资源调度方面，采用细粒度的流水策略，通过前向运算、后向传播和参数更新的迭代执行，降低设备闲置时间。考虑到注意力计算并行度高的特点，采用多码流执行方式，使不同注意力磁头能够相互独立地调度，从而提高 GPU SM 处理器的利用率。存储访问模式优化以降低存储空间碎片为目标，通过统一的存储池管理技术对存储空间进行动态分配和释放。

（四）容错与弹性扩展支持

分布式培训系统对硬件故障和计算节点的动态变化要求较高的容错性。使用检查点机制对模型状态进行周期性的存储，并结合增量快照技术降低内存开销。在故障恢复过程中，采用分布式一致性协议保证参数的同步，避免了节点故障造成的训练中断[6]。可伸缩性支持计算节点在训练过程中动态增加或减少，采用参数化服务器结构或者 All-Reduce 组重配置机制以适应资源调度需求的变化。

在大规模训练场景中，为适应节点的加入和退出，通信组需要进行动态调整。在计算任务的再分配中，使用一致性散列算法，保证系统的负载平衡。在容错设计中，还需要考虑梯度融合的鲁棒性，通过冗余计算节点或者梯度检验等机制来防止数据丢失和错误传播。

三、基于注意力机制的 Transformer 模型并行计算架构性能评估

（一）性能评估实验设计

通过对比分析和渐进式优化等方法，对基于注意力机制的 Transformer 并行计算体系结构性能进行全面验证。实验环境为 32个节点，配置8片 A100 GPU，采用 NVLink3.0和 InfiniBand HDR 200 G 互联技术，保证硬件配置能够满足大规模分布式训练的需要。基线对比实验采用 FP32精度的标准数据并行、模型并行和流水并行方案，优化实验组采用 FP16混合精度混合并行框架，结合稀疏注意力和弹性扩张技术[7]。

实验环境包括静态加载和动态加载两种方案。静态场景中固定节点数目（8个节点），并对不同并行策略下的计算资源和吞吐量进行比较；动态场景模拟真实训练环境下资源弹性变动（16→32个节点），对框架扩展稳定性进行评估。以 WMT2020 英德翻译语料和 PG-19长文本数据为测试样本，分别对短、长序列两种场景下的性能进行测试。利用 NVIDIA Nsight 和 PyTorchProfiler 等工具，实现对训练吞吐量、通讯延时、显存量和计算单元利用率的精细监控。实验结果见下表：

表1 基于注意力机制的 Transformer 并行计算架构性能评估数据

评估指标	基线架构 (FP32)	本文架构 (FP16+ 优化)	通信开销占比 (%)	计算利用率 (%)	训练吞吐量 (样本 / 秒)
单节点训练	1.00 (基准)	–	–	78.32	112.45
数据并行 (8 节点)	6.14	7.82	23.67	65.41	684.29
模型并行 (8 节点)	5.87	7.35	31.45	72.68	523.76
混合并行 (8 节点)	6.92	8.74	18.23	84.56	892.47
流水线并行 (4 阶段)	5.21	6.93	12.78	69.34	587.62
稀疏注意力优化	–	9.12	9.45	91.27	1024.83
弹性扩展 (16→32 节点)	–	15.38	14.56	88.92	1843.75

（二）并行计算效率对比分析

在计算利用率和训练吞吐率方面，8个节点配置时，混合并行结构可以达到84.56% 的计算利用率，比单数据并行或者模型并行方案要高，说明该方法可以有效地均衡计算和通信负载。该方法能够有效降低数据冗余，特别适合长序列场景。在弹性扩展实验中，节点数由16增加到32，只降低3.64%，表明该体系结构具有很好的横向扩展能力。与之相比，基线 FP32模型的并行效率一般不高，特别是在模型并行模式下，其计算效率只有72.68%，说明传统的静态分割策略对硬件资源的利用率并不高。

（三）通信开销与系统可扩展性分析

通信开销占比直接影响到分布式训练效率，8个节点时，混合并行结构通信占比18.23%，较单数据并行23.67%、模型并行31.45% 有所下降，表明通信拓扑优化可有效减少节点间的数据传输。基于稀疏注意力优化的通信开销降低到9.45%，表明键值对共享和块计算策略可以有效降低冗余通信。在弹性扩展方案中，通信占比由原来的14.56% 小幅上升到16.92%，仍然好于基线方案，证明了动态分组通信机制对节点规模的变化具有较强的适应性。尽管流水线并行通信占比最小 (12.78%)，但其计算利用率却只有69.34%，表明其气泡率问题仍然制约着系统的整体效率，需要对任务调度策略进行进一步优化。

（四）计算资源利用率与训练吞吐量优化空间

训练吞吐率直接反映了体系结构的实际价值，8个节点下，稀疏注意力优化算法达到1024.83个抽样 / 秒，较混合并行算法提高了14.83%。当弹性扩展到32个节点的时候，吞吐量达到了每秒1843.75个样本，但是增长速度比线性增长的预期要慢，显示出了通信和存储的瓶颈。计算效率和吞吐量呈正相关关系，这两种方法都是通过降低计算单元数量来提高计算效率。然而，流水线并行吞吐率 (587.62个样本 / 秒) 与计算资源利用率偏低，提示其任务划分粒度可能过于粗化，需要进一步探索更加细粒度的流水调度策略。

四、基于注意力机制的 Transformer 模型并行计算架构性能优化策略

（一）重构注意力计算任务划分方式

考虑到 Transformer 模型自身注意力机制的计算特点，基于细粒度任务划分策略的任务分配优化算法。该算法将多注意力计算分解成若干个独立的子任务，并根据计算单元数目动态分配注意力头，以保证各节点的负载平衡。针对长序列处理场景，引入块稀疏注意力机制，将全局注意力计算转化为局部计算和关键连通的混合模式，降低计算复杂性 [8]。在 QKV 矩阵乘算法中，采用行列混合分割策略，结合硬件存储层次，优化数据布局，减少节点间的数据传输。前馈网络采用两层并行策略，即第一层采用列并联，第二层采用行并联，利用重叠运算和通信掩盖部分延时。在任务划分过程中，充分考虑计算节点之间的拓扑关系，将通信密集的计算任务分配给高带宽互联节点，以避免网络拥塞。

（二）优化混合并行通信调度机制

建立多层通讯调度架构，协调数据并行、模型并行以及流水并行之间的数据交互。采用分层梯度聚合策略，先完成节点内的聚合，然后再进行全局的同步，从而降低了数据传输的开销。在模型并行过程中，针对张量切分问题，设计异步全盖瑟和 Reduce-Scatter 组合通讯方式，实现计算和通信的交叉 [9]。针对 GPU 集群，基于硬件特征的通讯协议选择器，通过对 GPU 集群进行 NCCL 优化，并将其切换到专用 XLA 通信后端。建立通信缓冲管理系统，实现高频、小数据通信的打包处理，提高网络带宽利用率。

（三）实施计算存储协同管理方案

设计动态的视频存储分配策略，根据不同的运算需求，灵活地调整存储资源；该算法采用块缓存的方法，将有效计算块保存在内存中，其余的暂存到主存储器。研究激活价值的智能再计算机制，根据计算图分析，选择性地丢弃中间结果，并在反向传播过程中根据需要进行再计算。建立统一的内存池管理系统，通过优化的自适应算法对内存空间进行分配，降低内存碎片。可计算任务感知的预取策略，根据计算过程顺序，将后续计算需要的数据提前加载。针对大规模模型参数，采用参数分片存储策略，将非活跃参数自动卸载到 CPU/NVMe 存储器中，并采用 PCIe 异步传输方式进行快速恢复。

（四）构建弹性容错训练框架

设计一种分布式一致性校验点系统，利用差分快照技术对模型状态进行周期性的存储。实现校验点的最优存储，只保留参数的变化量，减少存储开销。利用心跳机制对计算节点的状态进行实时监测，实现节点的故障检测和恢复。建立灵活的参数化服务器体系结构，支持计算节点在训练过程中的动态增加和减少以及模型分片的自动重分配 [10]。实现了通讯群组的动态重构，在节点加入和退出时，对 All-Reduce 群组拓扑进行自动优化。设计梯度检验和修复机制，利用冗余计算对传输误差梯度进行检测和修正。

五、结语

综上所述，通过系统的并行计算体系结构设计，将计算任务优化、通讯调度优化、资源协同管理和容错增强相结合，有效解决大规模 Transformer 模型训练所面临的计算效率和资源瓶颈问题。未来，应深入探索计算和通信的深度耦合优化问题，探索异构环境中的自适应并行策略，强化面向新型硬件体系结构的专用优化设计，不断提高分布式深度学习系统的性能极限和能效比。

参考文献

[1] 徐晓轶，毛艳芳，吕晓祥 . 基于 Transformer 和关键词信息聚合的电力科研成果命名实体识别 [J]. 计算机应用 ,2024,44(S2):66-71.

[2] 石彬，成苗，张绍兵，曾尚 . 基于模糊核估计和交替 Transformer 的二维码图像去运动模糊算法 [J]. 计算机应用 ,2024,44(S2):234-239.

[3] 钟来民，陆卫忠，傅启明，马洁明，崔志明，吴宏杰 . 基于 Transformer-BiLSTM 特征融合的 DNA 结合蛋白预测方法 [J]. 微电子学与计算机 ,2023,40(12):1-9.

[4] 廖健文，杨盈昀，卢玥 . 基于稀疏 Transformer 的长短时序关联动作识别算法 [J]. 中国传媒大学学报 (自然科学版),2023,30(06):56-63.

[5] 唐雷，许子祥，高广谓 . 基于 Transformer 与注意力聚合的人脸超分辨率 [J]. 计算机与数字工程 ,2023,51(12):2977-2983.

[6] 唐梦瑶，黄江涛 . 基于盒注意力机制和 Transformer 的人脸微表情识别方法 [J]. 人工智能科学与工程 ,2023,(09):57-67.

[7] 熊巍，熊承义，高志荣，陈文旗，郑瑞华，田金文 . 通道注意力嵌入的 Transformer 图像超分辨率重构 [J]. 中国图象图形学报 ,2023,28(12):3744-3757.

[8] 石德硕，李军侠，刘青山 . 自注意力融合调制的弱监督语义分割 [J]. 中国图象图形学报 ,2023,28(12):3758-3771.

[9] 郑晓旭，舒珊珊，文成玉 . 基于注意力多分支卷积和 Transformer 的手写文本识别 [J]. 成都信息工程大学学报 ,2023,38(06):649-655.

[10] 刘华咏，黄聪，金汉均 . 注意力增强的视觉 Transformer 图像检索算法 [J]. 电子测量技术 ,2023,46(23):50-55.

