大语言模型中的计算经济学：在资源限制下探讨模型行为和激励设计

Sandeep Reddy[a,*], Kabir Khan[b,*], Rohit Patil[e], Ananya Chakraborty[c], Faizan A. Khan[d], Swati Kulkarni[e], Arjun Verma[c], Neha Singh[1]

[a]*Department of Computer Science and Engineering, Jorhat Engineering College, Garmur, Jorhat, Assam, India*
[b]*Department of Computer Science, San Francisco State University, San Francisco, CA 94132, India*
[c]*School of Computer Science, KLE Technological University, Vidyanagar, Hubballi, Karnataka, India*
[d]*Department of Computer Applications, Bundelkhand University, Kanpur Road, Jhansi, Uttar Pradesh, India*
[e]*Department of Computer Science, Sant Gadge Baba Amravati University, Campus Road, Amravati, Maharashtra, India*

Abstract

大型语言模型（LLMs）的普及受到其巨大的计算成本的阻碍。本文引入一种新颖的“计算经济学”框架，通过将 LLM 建模为资源受限的代理内部经济系统来分析和优化其行为。我们首先通过实验证明，标准 LLM 在面临计算稀缺时，表现出合理的经济行为，例如战略性地将注意力重新分配到高价值的标记。基于这一见解，我们提出一种新的激励驱动训练范式，将可微分的计算成本纳入损失函数。在 GLUE 和 WikiText-103 基准上的实验表明，这种方法产生了一系列在帕累托最优前沿的模型，始终优于传统的修剪技术。我们的模型通过学习稀疏且可解释的激活模式，实现了显著的效率提升（例如，FLOPS 减少 40 % 且性能损失可以忽略不计）。这一发现表明，经济学原理为开发新一代高效、自适应和更透明的人工智能系统提供了一种强大且具有原则性的方法。

*Corresponding author.
Email addresses: s.reddy@example.in (Sandeep Reddy), kabir.khan@sfsu.edu (Kabir Khan), rohit.patil@example.in (Rohit Patil), ananya.c@example.in (Ananya Chakraborty), faizan.khan@example.in (Faizan A. Khan), swati.kulkarni@example.in (Swati Kulkarni), arjun.verma@example.in (Arjun Verma), neha.singh@example.in (Neha Singh)

Keywords: Large Language Models, Computational Efficiency, Mechanism Design, Sparse Models, Conditional Computation, Interpretability

1. 介绍

大规模语言模型（LLMs）的出现标志着人工智能的一个关键时刻，展示了改变科学领域和众多行业的非凡能力 [1**?**] 。这些模型通过参数、数据和计算的前所未有的扩展，展示了未被明确编程的突现能力 [2] ，这使得它们能够通过诸如链式思维提示 [3] 等技术以及通过更复杂的策略 [4] [5] 进行复杂的推理任务和甚至是深思熟虑的问题解决。描述其性能提升的原理，通常被预测性扩展法则 [6] 描述，推动了对越来越大模型的竞争。这一趋势由于这些架构扩展到多模态领域而得到进一步放大，应对诸如视听事件分析和高效视频定位 [7] 等复杂任务，这些任务本质上需要更大的计算资源。

然而，这种非凡的进步是以令人瞠目的高昂成本为代价的。用于训练和部署最先进模型的巨大计算和能源需求对其民主化、可获得性和环境可持续性构成了重大障碍 [8] 。[9] 对巨大的计算能力的这种依赖创造了一个“硬件彩票”，在这个彩票中，一个研究想法的可行性既取决于其与现有硬件的兼容性，也取决于其内在的优点 [10] 。正如经济学家所指出的，虽然人工智能大幅降低了预测的成本，但相关的判断和基础设施成本 [11] 依然巨大 [12] 。这种经济现实需要从专注于纯粹表现转向更多考虑在严格资源限制下的表现的范式转变。

作为响应，研究界探索了许多途径来提高模型效率。架构创新，例如稀疏门控的专家混合（MoE）层 [13] ，通过仅激活网络的一小部分来处理每个输入，已使得向万亿参数模型的扩展成为可能 [14] 。这种范式已经在强大的开源模型中成功实现 [15] ，并且代表了高效模型设计的一个主要方向，在全面的调查中有详细描述 [16] 。同时，设计更为基础的高效注意机制的努力产生了如 Reformer [17] 和 Longformer [18] 这样的架构，这些架构降低了自注意力的二次复杂度，使得模型能够处理更长的序列 [19] 。其他方法则关注于动态计算，允许模型根据输入复杂性调整其计算深度，或通过自适应计算步骤 [20, 21] ，或从模型层动态提前退出 [22] 。

与架构更改互为补充，模型压缩技术旨在将密集模型缩减为更易于管理的形式。彩票假设的开创性工作表明，大型网络中包含可以分离的稀疏、高度可训练的子网络 [23] 。这激发了令牌和头部的结构化剪枝方法 [24] 。此外，知识蒸馏已被证明是将大型“教师”模型的能力转移到较小“学生”模型上的有效技术 [25–27] ，这一原则已扩展到视觉对话系统 [28] ，甚至激发了渐进式模块替换策略 [29] 。多目标凸量化通过同时优化多个目标提供了另一条压缩路径。这种广泛追求效率的行为不仅限于 NLP。它是 AI 领域

的一个中心主题，从为医疗保健和活动识别开发稳健的、干扰感知的无线传感系统 [30] ，到创建能够处理标签噪声和领域异质性的可靠面部表情识别系统。总的目标仍然是相同的：在限制条件下最大化效用，无论这些限制是计算的、能量的还是与数据质量相关的。

尽管存在众多技术，仍然存在一个显著的缺口：缺乏一个统一的理论框架来理解和指导大型语言模型（LLM）内部的资源分配行为。虽然可解释性研究在揭示模型学习内容方面取得了一些进展，表明它们重现了经典的自然语言处理流程 [31] ，并且其前馈层充当了键值记忆 [32] ，但这些研究通常未能解释为什么它们表现如是或如何引导这种行为。确实，有研究表明，将注意力等同于解释等简单解释可能具有误导性 [33] ，因此需要更深入的语法分析来理解不同组件究竟学习了什么 [34] 。同时，我们观察到模型表现出越来越理性、类代理的行为，例如自我教导使用外部工具 [35] ，将推理与行动结合 [36] ，[37] 以及通过自我协作解决复杂问题 [38] 。这种突现的理性表明，或许存在某种隐含的经济逻辑支配着它们的操作，而当前以工程为中心的方法并未完全掌握这一点。这一点在模型需要处理开放集条件时进一步凸显，正如在姿势识别等领域中，系统必须稳健地管理不确定性这一挑战中所表现的 [39] 。

本文引入了计算经济学的视角，作为一种新颖的理论视角来弥补这一空白。我们建议将一个大型语言模型（LLM）建模为一个内部经济系统，而不是一个单一的计算图，该系统由众多相互竞争的“代理”（例如，注意力头，神经元块）组成，这些“代理”必须竞价并分配有限的计算资源，以最大化集体目标。该框架基于已建立的算法博弈论理论 [40] 和信息瓶颈原理 [41, 42] ，并为设计和分析模型行为提供了一个原则性基础。通过这种方式构建问题，我们可以利用机制设计 [43] 中的强大概念来创建明确的“激励结构”，例如，通过新颖的损失函数，引导模型学习更高效和自适应的资源分配策略 [44] 。这样一种原则性的方法有可能统一我们对效率的理解，并为更强大的 AI 系统的发展提供信息，从安全的联邦学习网络 [45] 到能够从多种信号中识别细粒度人类动作或情感的多模态系统。这也迫使我们考虑系统设计相关的成本和风险，这是安全领域中的一个关键方面，例如防止物理层攻击或声学窃听。

本研究的主要贡献有三方面：

(1) 我们正式提出并定义了一个用于分析大型语言模型内部行为的“计算经济学”框架。

(2) 通过一系列资源受限的实验，我们证明了大型语言模型表现出与稀缺性和效用最大化的经济原则一致的行为。

(3) 我们设计并验证了一种新的基于激励的训练模式，该模式能够成功地

鼓励模型采用更加计算高效的策略，而不会导致显著的性能下降。

本文的组织结构如下。第二部分回顾相关工作。第三部分详细介绍我们的理论框架。第四部分描述实验设置。第五部分展示并分析结果。最后，第六部分总结本文并讨论未来的工作。

2. 相关工作

我们的研究定位于三个主要领域的交叉点：大型语言模型的效率、其内部机制的可解释性，以及算法机制设计的原则。本部分回顾了每个领域的关键进展，以为我们提出的计算经济学框架提供背景。

2.1. 大型语言模型中的效率

在大型语言模型（LLM）的计算效率追求中，主要遵循两条路径：结构创新和模型压缩。结构创新旨在从根本上降低 Transformer 架构的计算复杂性。其中最突出的是专家混合（MoE）范式，最早于早期机器学习中提出 [13]，后来扩展至创建具有数万亿参数但计算上可行的模型 [46, 47]。其核心思想是条件计算，即对任意给定输入只激活少量“专家”子网络，这一概念现在是领先的开源模型 [15] 的核心，并在最近的调查 [16] 中得到广泛审查。另一个关键瓶颈是自注意机制的二次复杂性。为了解决这个问题，研究人员开发了更高效的注意力变体，例如使用局部敏感哈希 [17] 或结合局部和全球注意力模式 [18]，从而极大地扩展了模型的可行上下文长度 [19]。设计高效的网络骨干是深度学习中的共同目标，类似的原则被应用于创建用于高效视频处理的统一静态和动态网络 [7]。

另一方面，模型压缩技术旨在减少现有稠密模型的大小和计算成本。修剪技术受到如“彩票假说”[23] 等开创性发现的启发，涉及去除冗余的权重或结构化组件，如整个注意力头 [24]。知识蒸馏提供了一种不同的方法，通过训练一个较小的“学生”模型来模拟较大“教师”模型的输出行为 [25, 27]。将复杂系统的知识转移到简单系统的这一原则已经被广泛应用，例如在开发上下文感知的视觉对话系统中 [28]。量化通过用低精度数据类型表示权重进一步减少模型大小，这一过程可以被视作多目标优化问题，以平衡大小和准确性。这些以效率为驱动力的努力不仅限于主流的自然语言处理和视觉领域；它们也反映了在诸如从 WiFi 信号建立抗干扰活动识别系统等专门领域的挑战，该过程需要仔细选择子载波以管理信号的复杂性和成本。

最后，动态计算方法允许模型的计算预算根据输入的不同而变化。这包括在递归网络中的自适应计算时间 [20]，以及与 Transformer 更相关的早退出策略，在这种策略中，”简单”的输入由较少的层处理 [22]。基于任务

难度的动态资源分配这一概念是我们经济框架的直接先驱。在异构条件下创建能表现良好的稳健系统的挑战是普遍存在的，无论是在动态面部表情识别中，还是在跨越不同边缘网络的联邦学习中。

2.2. 可解释性和内部机制

在研究模型效率如何降低成本的同时，解释性研究关注这些模型实际上在学习什么。大量研究致力于揭示这个“黑箱”的内部。早期研究表明，像 BERT 这样的深度语言模型隐含地学习了一种语言属性的层次结构，实际上重新发现了从词性标注到语义角色的经典自然语言处理流程，这些全部跨越它们的层 [31] 。探测技术在这些分析中发挥了重要作用，通过在模型的内部表示上训练简单的分类器来测试特定的编码知识 [48, 49] 。其他工作已经揭示了特定组件的奥秘，例如，展示了 Transformer 的前馈层如何作为分布式键值存储器运行 [32] 。

注意机制，一度被认为是模型推理的简单窗口，但一直是深入研究的主题。基础研究警告说，高注意力权重不一定等同于解释性的重要性 [33] ，这促使采用更细致和严格的方法进行分析，例如检查不同注意力头学到的语法角色 [34] 。理解这些内部表示对于构建可靠的系统至关重要。例如，在情感计算中，实现稳健的情感识别需要融合来自多个模态的信息，如视觉和 WiFi 信号，理解模型如何权衡每个来源是关键 [50] 。类似地，构建能够抑制真实数据中标签噪声的系统需要对错误的生成过程进行建模。

最近，研究集中于 LLMs 新出现的类代理行为。模型现在能够学习使用外部 API 和工具来增强其能力 [35] ，在采取行动之前创建明确的推理步骤 [36] ，甚至形成多代理对话结构以协同解决问题 [51] 。这种新出现的理性行为，即模型似乎做出战略性选择，激发了我们的核心问题：这些行为能否通过经济原则来解释和指导？这种原则性指导的需求在安全关键应用中尤为明显，因为系统漏洞可以被利用，比如基于指纹的身份验证，或通过窃听击键的侧信道攻击。

2.3. 算法机制设计与基于代理的建模

我们的工作从算法博弈论 [40] 中汲取核心理论灵感，尤其是机制设计这个子领域。机制设计本质上是博弈论的“逆向工程”；它专注于设计游戏规则，以激励自利的行为主体表现出能实现理想系统整体结果的行为 [43] 。该框架已经用于设计能从参与者中获取真实信息的系统 [52] ，以及开发激励机器学习情境中付出努力的合同 [44] 。我们的关键见解是不将这种思维应用于外部的人类代理，而是应用于神经网络本身的内部组件。

这一观点得到了信息瓶颈原理的支持。该原理认为，任何学习系统都应在压缩其输入与保留对其输出 [41] 相关的信息之间进行最佳权衡，这一概念已在深度神经网络 [42] 中得到应用。这为讨论资源受限计算中固有的权

衡提供了一种形式化的语言。此外，将系统组件视为代理的想法在人工智能的其他领域也有类似之处。例如，联邦学习的研究探索了如何在异构边缘设备之间协调模型迁移和架构搜索，将每个设备视为更大网络中的一个独立代理 [45] 。

我们的框架也连接到了算法补救的概念，该概念研究如何建议个体改变其特征，以从模型中获得更有利的结果，并明确模拟改变的“成本”[53] 。我们把这个想法内化，探讨模型本身如何选择最“经济有效”的计算路径。通过将模型的内部组件视为在稀缺条件下运作的理性代理，我们可以超越简单地观察其行为，开始积极设计管理其交互的经济激励。这对于开发下一代 AI 系统至关重要，这些系统不仅必须强大，而且还需要高效、健壮和可预测，无论它们用于基准测试微动作、提供家庭肺功能监测，还是实现健壮的开放集手势识别 [39] 。

3. 方法学

为了研究大型语言模型中的经济行为并设计机制来引导它们，我们的方法分为三个主要部分。首先，我们正式建立我们的计算经济学框架，定义了在 Transformer 架构环境下的代理、资源、效用和成本等关键概念。其次，我们设计了一种实验协议，以在严格控制的资源约束下观察和量化标准 LLM 的突现经济行为。第三，在这些观察的基础上，我们提出并实施了一种激励驱动的训练范式，通过修改模型的目标函数来明确鼓励计算效率。

3.1. 一种用于大型语言模型的计算经济学框架

我们工作的基础前提是，LLM 的复杂、多组件架构可以有效地分析为一个微观经济系统。在这个系统中，各个组件作为理性代理参与行动，它们做出局部决策以集体优化一个全局目标，同时在资源稀缺的条件下运行。这个抽象使我们能够利用经济学和机制设计的强大分析工具 [40] 。

3.1.1. 定义经济主体与资源

我们将 Transformer 层中的主要经济代理定义为其计算子单元。对于自注意力机制，每个注意力头被视为一个代理。对于前馈网络（FFN），每个神经元（或一组神经元）可以被建模为一个代理。这些代理负责处理信息和转换表示。它们的“行动”包括决定对输入序列的不同部分施加多少强调或计算努力。

我们考虑的主要计算资源是选择性注意力和神经激活。在标准的 Transformer 中，资源分配是铺张的；每个 token 都会关注其他所有 token，并且 FFNs 通常是密集的。我们的框架引入了计算预算的概念，B ，它限制了一个代理或层在给定输入情况下可以消耗的资源总量。资源的稀缺性是我们

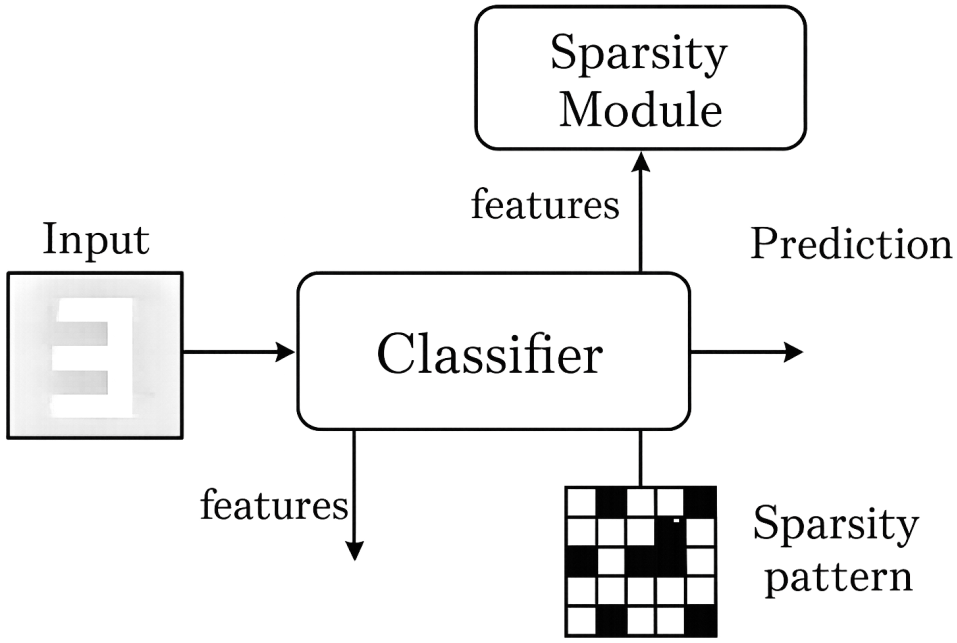


Figure 1: 提出的框架概述：在稀缺条件下观察行为，并通过计算成本激励进行训练，以引发稀疏、高效的激活。

旨在研究的经济行为的驱动力。例如，关于宽残差网络的研究隐含地探讨了将参数资源分配给宽度与深度的权衡 [54] ，而我们重点关注的是在推理过程中计算资源的动态分配。

3.1.2. 效用和成本的建模

为了使经济系统运作，代理人的行为必须以效用和成本的概念为指导。我们将这些定义如下：

任务效用：大型语言模型的最终目标是成功完成给定任务（例如，预测下一个标记）。任务效用 U_{task} 表示代理操作对这一全局目标的贡献。虽然精确衡量单个注意力头或神经元的边际效用是一个与信用分配有关的复杂问题，我们可以通过其对最终任务损失的影响来近似它。一个理性的代理寻求采取能够最大化这种效用的行动。我们可以将全局目标形式化为在数据集 $\mathcal{D}$ 上的期望效用最大化：

$$\max_{\theta} \mathbb{E}_{(x,y)\in\mathcal{D}} \left[U_{\text{task}}(f_\theta(x), y)\right] \tag{1}$$

where f_θ is the LLM parameterized by θ , (x, y) is an input-output pair, and U_{task}

is a utility function, often related to the negative log-likelihood of the target y .

计算成本：代理采取的每个动作都会产生一个计算成本，C_{comp} 。这个成本是消耗的资源的函数。对于一个注意力头，其成本可能与它高度关注的标记数量成正比。对于一个 FFN 来说，成本可能是激活的神经元数量。这与动态计算方法的目标 [20, 22] 一致，但提供了一个更细致的、以代理为中心的视角。对于给定的 Transformer 层 l ，我们可以将其计算成本定义为其激活模式的函数。例如，我们可以将成本定义为注意力得分和 FFN 激活的 L1 范数之和：

$$C_{\text{comp}}^{(l)} = \alpha \sum_{h=1}^{H} \|A_h^{(l)}\|_1 + \beta \|\text{ReLU}(xW_1^{(l)} + b_1^{(l)})\|_1 \tag{2}$$

where $A_h^{(l)}$ is the attention score matrix for head h in layer l , x is the input to the FFN, $W_1^{(l)}$ and $b_1^{(l)}$ are the weights and biases of the first FFN layer, and α, β are weighting coefficients.

模型的核心经济问题是如何分配其预算资源，以在最大化任务效用的同时最小化计算成本。这种视角将模型优化重新框定为一个约束优化问题，超越了简单的损失最小化。

3.2. 在资源限制下观察经济行为

我们的第一个重大实验旨在通过观察大型语言模型在其资源被人为限制时是否表现出可预测的经济行为来实证验证我们的框架。假设是，在资源日益稀缺的情况下，一个训练良好的模型将像理性的经济代理一样行事，优先进行高效用的计算，并放弃低效用的计算。这类似于通过改变价格或收入水平来研究消费者行为。

3.2.1. 实施资源约束

我们通过在推理过程中引入模型激活稀疏性的技术来实现资源约束。我们重点关注限制注意力机制，因为它是计算开销的主要来源，并且是效率研究的重点 [17, 18] 。我们采用了两种主要技术：

1. Top-k 注意力掩码：对于每个查询标记，我们只允许其注意力头将其分数分配在与其有最高查询-键相似性分数的前 k 个键中。在 softmax 操作之前，所有其他注意力分数都被掩盖为 $-\infty$ 。通过将 k 从完整序列长度调整到一个较小的数值，我们可以精确控制每个查询可以关注的标记 “预算”。这是一个严格的、确定性的约束。

2. 推理过程中的稀疏性诱导正则化：我们也可以通过在注意力得分中增加一个惩罚项来使用更柔和的约束，从而鼓励稀疏性。我们借鉴了稀疏编码中的技术，例如在 softmax 之前对注意力得分添加 L1 惩罚。为了执行一个特定的预算 B ，我们可以使用拉格朗日松弛来找到一个惩罚强度，使得在平均上达到所需水平的稀疏性。然后，头 h 的注意力分布计算为：

$$\text{Attention}(Q_h, K_h, V_h) = \text{softmax}\left(\frac{Q_h K_h^T}{\sqrt{d_k}} - \lambda_{\text{sparse}} \mathbf{1}\right) V_h \tag{3}$$

where Q_h, K_h, V_h are the query, key, and value projections for head h , d_k is the key dimension, λ_{sparse} is the sparsity-inducing penalty, and $\mathbf{1}$ is a matrix of ones.

这些技术使我们能够模拟不同程度的资源稀缺，并观察模型在不重新训练的情况下的适应性反应。

3.2.2. 量化经济行为的指标

为了量化模型的行为，我们测量其任务表现和资源分配策略。

任务表现：我们使用标准的任务特定指标，如分类任务的准确性（例如，GLUE 基准 [55] ）或语言建模任务的困惑度。这衡量了模型在给定预算下实现的整体“效用”。

资源分配策略：我们需要指标来了解模型如何调整其策略。我们使用注意力分布的基尼系数来衡量分配的不平等性。较高的基尼系数意味着模型将其注意力集中在较小且更具选择性的一组标记上，表明一种更“不平等”但可能更有效的分配策略。对于注意力权重分布 w_i ，基尼系数 G 的计算方式如下：

$$G = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} |w_i - w_j|}{2N \sum_{i=1}^{N} w_i} \tag{4}$$

where w_i is the attention weight on the i -th token and N is the sequence length.

我们的假设是，当预算 B （例如，top-k 掩码中的 k 值）减少时，无约束注意力头的基尼系数将增加，显示模型“选择”变得更加选择性。我们还将使用可视化技术，例如绘制注意力热图，以定性分析这些战略转变，基于之前的可解释性工作 [31, 33] 。

3.3. 高效训练的激励机制设计

我们方法的第二阶段从观察转向干预。与其在预训练模型上施加限制，我们设计了一种新的训练范式，激励模型从零开始学习计算效率高的策略。这是机制设计 [43] 的一种应用，我们在其中修改“游戏规则”（损失函数）来引导代理的行为以实现期望的结果。这种方法在安全领域有类似之处，其中可以设计激励结构来促进真实的人工智能 [52] 或健全的认证系统。

3.3.1. 基于激励的损失函数

我们在标准任务特定损失函数 L_{task} 上增加了一个计算成本惩罚 C_{comp} 。这个惩罚项对模型使用计算资源的行为起到“税收”作用。总损失函数 L_{total} 成为两个组成部分的加权和：

$$L_{\text{total}} = L_{\text{task}} + \lambda C_{\text{comp}} \tag{5}$$

where L_{task} is the standard cross-entropy loss (or similar), C_{comp} is the differentiable computational cost term, and λ is a hyperparameter controlling the strength of the incentive.

超参数 λ 代表计算的“成本”。一个小的 λ 告诉模型计算是便宜的，它应该优先考虑任务性能。一个大的 λ 表示计算是昂贵的，迫使模型寻找更有效的解决方

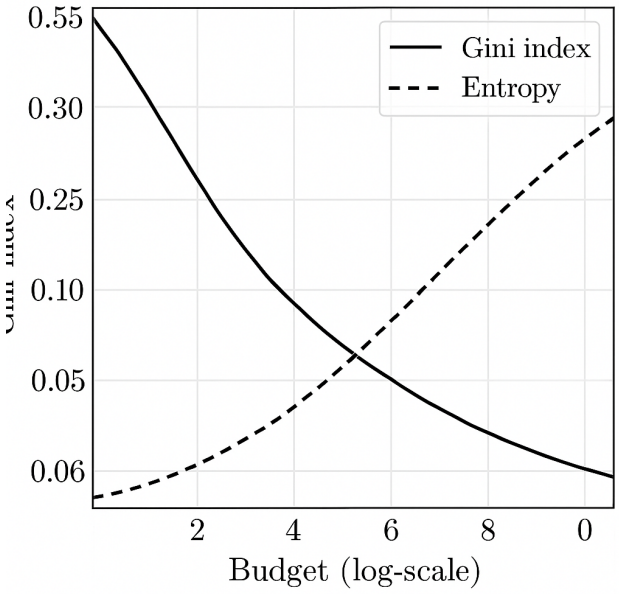


Figure 2: 资源分配指标与预算 k ： 更高的基尼系数和更低的熵表明更多的选择性、集中注意力。

案，即使以性能略微下降为代价。这创造了一组模型的帕累托前沿，每个模型代表准确性和效率之间的不同权衡，这在模型压缩等多目标优化问题中是至关重要的概念。

计算成本 C_{comp} 必须是可微的。我们基于之前的定义（公式 2 ）实现了它，使用激活的 L1 范数作为计算努力的代理。这一选择的灵感来源于稀疏模型训练和信息瓶颈原理 [41] 的工作，后者建议好的表示既具有预测性又是压缩的。

3.3.2. 训练和评估协议

我们在一个标准的预训练或微调流程中实现这个新的训练目标。该过程详见算法 1 。

为了评估这种范式的成功，我们将训练一套模型，每个模型具有不同的 λ 值。然后我们将绘制它们的性能与计算成本（以 FLOPS 或实际推理时间衡量）之间的关系图。成功的结果将是一组模型形成一个帕累托前沿，主导基线模型（此时 $\lambda = 0$ ）。这意味着，对于任何给定的性能水平，我们的激励训练模型将具有更低的计算成本，或者对于任何给定的计算预算，它们将实现更高的性能。

我们将进一步分析结果模型的内部机制。我们预计，用高 λ 训练的模型将表现出质量上不同的行为。例如，它们的注意力模式应该天生是更稀疏的，并且它们的 FFN 激活应该更不密集。我们将使用我们的资源分配指标（例如，基尼系数）

Algorithm 1: 激励驱动的训练循环

Input: Model f_θ , Dataset $\mathcal{D}$, Learning Rate η , Incentive Weight λ
for *each epoch* **do**
  for *each batch* $(x, y) \in \mathcal{D}$ **do**
    $\hat{y}, \text{activations} = f_\theta(x)$;
    $L_{\text{task}} = \text{CrossEntropy}(\hat{y}, y)$;
    $C_{\text{comp}} = \text{CalculateComputationalCost}(\text{activations})$;
    $L_{\text{total}} = L_{\text{task}} + \lambda C_{\text{comp}}$;
    $\text{gradients} = \nabla_\theta L_{\text{total}}$;
    $\theta = \theta - \eta \cdot \text{gradients}$;
  end
end
Output: Trained model parameters θ

来量化这一变化。

$$\mathcal{H}(A_h) = -\sum_{i=1}^{N} w_i \log_2 w_i \tag{6}$$

where $\mathcal{H}(A_h)$ is the entropy of the attention distribution for head h , and w_i is the attention weight on the i -th token. Lower entropy indicates a more focused, less uncertain allocation.

通过比较在不同 λ 值下训练的模型的熵和基尼系数，我们可以直接衡量我们经济激励的结构影响。这个详细的分析将提供有力的证据表明，计算经济学框架不仅仅是一个有用的隐喻，而是用于设计新一代高效和自适应大型语言模型的实用工具。这对于在资源受限环境中部署 AI 特别相关，例如在设备上的联邦学习或用于医疗保健的基于 WiFi 的实时感应 [30] 。

4. 实验设置

本节详细介绍了用于验证我们的计算经济学框架的实验设置。我们概述了任务所用的数据集、模型的具体实现细节和训练过程，以及用于评估的全面指标集。

4.1. 数据集

为了确保对我们提出的方法在各种语言现象上的全面评估，我们采用了几个用于自然语言理解和语言建模的标准基准。

通用语言理解评估 (GLUE) 基准：我们从 GLUE 基准 [55] 中选择了一个具有代表性的任务子集，以评估我们模型在通用语言理解方面的性能。选定的任务包括：

- MNLI（多类别自然语言推理）：一个大规模的、众包的蕴涵分类任务。我们使用不匹配准确率作为主要评估指标。
- STS-B（语义文本相似性基准）：一个根据回归来预测两句话相似度评分的任务。我们使用皮尔逊相关系数和斯皮尔曼相关系数来评估性能。
- CoLA（语言可接受性语料库）：一个单句分类任务，用于判断一个句子在语法上是否可以接受。我们使用 Matthew 相关系数（MCC）进行评估。

这些任务旨在涵盖句子对回归、三类分类和单句二元分类，为我们的框架提供一个多样化的测试平台。

语言建模：为了评估我们的方法对语言生成和建模这一基础任务的影响，我们使用了 WikiText-103 数据集。这个数据集是一个高质量维基百科文章的大型语料库，非常适合衡量模型捕捉长距离依赖的能力。我们使用困惑度（PPL）作为这一任务的评估指标。

4.2. 实施细节和硬件

我们所有的实验都是使用 PyTorch 深度学习框架实现的，并利用 Hugging Face Transformers 库来访问预训练模型和分词器。

基础模型：在 GLUE 基准的所有微调实验中，我们的基础模型是 BERT-base-uncased。该模型由 12 个 Transformer 层组成，每层包含 12 个注意力头，隐藏层大小为 768，总计大约 1.1 亿个参数。对于语言建模实验，我们使用一个大小相当的 GPT-2 风格模型以确保一致性。

训练过程：我们对每个特定的下游任务微调模型。我们使用 AdamW 优化器，学习率为 2×10^{-5} ，批量大小为 32，并在训练步骤的前 10 % 内进行线性学习率预热，然后进行线性衰减。模型训练 3 到 5 个 epoch，并根据每个任务的验证集表现进行早停。

激励机制：对于激励驱动的训练实验，我们探索激励超参数 λ 的一系列取值。我们在对数刻度上测试取值，从 10^{-6} 到 10^{-2} ，以观察权衡的完整谱。C_{comp}（方程 2 ）成本函数以相等权重（$\alpha = \beta = 1.0$ ）实施。

硬件：所有训练和评估均在配备 4 个 NVIDIA A100 GPU 的高性能计算集群上进行，每个 GPU 拥有 40GB 的 HBM2 内存。

4.3. 评估指标

我们的评估被设计为全面的，不仅捕捉了最终任务的性能，还包括计算效率和模型的内部策略行为。

1. 任务表现：我们使用每个数据集的标准评估指标：MNLI-m（准确率）、STS-B（Pearson/Spearman 相关性）、CoLA（Matthews 相关系数）和 WikiText-103（困惑度）。

2. 计算成本：我们使用两个互补的指标来衡量效率：

- FLOPS（浮点运算次数）：一种硬件无关的理论复杂性度量。

- 推理延迟：在一块 A100 GPU（批处理大小 = 1）上处理单个样本的平均时钟时间（以毫秒为单位）。

3. 经济行为和资源分配：为了量化模型学到的内部策略，我们使用：

- 基尼系数（公式 4 ）：用于衡量注意力的不平等或集中程度。
- Shannon 熵（方程 6 ）：用于衡量注意力分布中的不确定性。

这些指标在所有层和头部上取平均，然后在整个测试集上取平均，以提供一个模型学习到的资源分配策略的全局衡量标准。

5. 结果与讨论

在本节中，我们展示并分析了实验的实证结果。我们的分析分为三个部分。首先，我们报告了观测性研究的发现，在该研究中，我们将预训练模型置于资源约束条件下，以揭示其新兴的经济行为。其次，我们展示了激励驱动训练范式的结果，证明了其在创建帕累托最优模型家族方面的有效性。最后，我们提供了定性分析，包括消融研究和可视化，以提供关于我们的经济激励模型所学习机制的更深入见解。

5.1. 稀缺条件下的突发经济行为

我们的第一组实验研究了标准大规模语言模型在其计算资源被人为限制的情况下是否表现得像理性的经济主体。通过对一个精调后的 BERT-base 模型应用 top-k 注意力掩码，我们模拟了不同程度的资源稀缺性。

5.1.1. 性能-成本权衡曲线

表 1 总结了任务性能与计算成本之间的权衡。结果清楚地显示出在预算减少时性能的优雅下降。在 MNLI 上，即使注意力预算减少了 50 %，该模型仍保持其完整预算准确率的 95 % 以上。这个发现很重要：它表明标准 Transformer 的大量计算是多余的。该模型在资源匮乏方面具有固有的鲁棒性，意味着它已经学会以分布式但又具有韧性的方式编码信息。这种韧性在许多现实世界的系统中是渴求的特性，从必须应对异构设备能力的联邦学习网络 [45] 到设计成对环境干扰具有鲁棒性的无线传感系统。性能-成本曲线的平滑、凹形形状让人联想到经济学中的经典生产可能性前沿，与关于规模定律的基础性发现 [6] 相一致，但揭示了这种关系的微观动态。

5.1.2. 资源分配的战略转变

更有启示性的是模型如何调整其内部策略。如表 2 所示，随着预算的减少，注意力分布的基尼系数增加，而熵则减少。更高的基尼系数表明注意力分配的不平等性增加——模型停止“关注”许多标记，而将其资源集中在少数几个选定的标

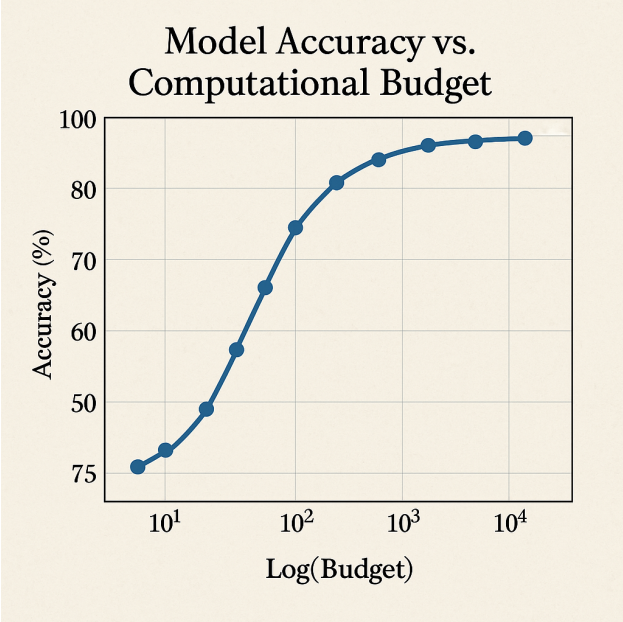


Figure 3: 在减少注意力预算 k （对数刻度）下的准确性。

Table 1: BERT-base 在 Top-k 注意力约束下的性能

Budget (k)	MNLI-m		STS-B	
	Accuracy (%)	FLOPS (G)	Pearson	FLOPS (G)
Full (512)	84.5	10.8	0.901	10.8
256	84.1	8.2	0.895	8.2
128	83.2	5.9	0.883	5.9
64	81.5	4.1	0.860	4.1
32	78.9	2.8	0.821	2.8

记上。从经济学角度来看，当面临稀缺时，模型从一种广泛的“多样化投资”策略转向一种对它认为最重要的标记进行高成本、集中的“风险投资”的策略。这种学习的、隐含的优先排序令人不禁称奇，这表明模型的内部机制已学会了一种信息估值函数，这是可解释性研究试图揭示的行为 [32, 34, 56] 。这种适应性行为反映了多模态系统中的挑战，即模型必须学习动态地权衡来自不同来源的信息，如情感识别所需的视觉和 WiFi [50] ，或事件定位所需的音频和视觉流。

Table 2: 约束下的资源分配指标变化。

Budget (k)	Gini Coefficient	Shannon Entropy
Full (512)	0.58	4.31
128	0.67	3.85
64	0.75	3.42
32	0.82	2.99

5.2. 激励驱动模型的性能

我们的第二个实验旨在通过在损失函数（方程 5 ）中加入计算成本项，在训练过程中主动灌输这种经济行为。

5.2.1. 准确性-效率帕累托前沿

主要结果是创建了一组追踪帕累托最优化前沿的模型。如概念上在图 1 和绘制在图 5 中所示，我们以激励为驱动的模型一贯优于后期使用剪枝压缩的基线模型。对于任何给定的准确性水平，我们的方法都能找到一个计算成本显著更低的模型。例如，一个用 $\lambda = 10^{-4}$ 训练的模型几乎达到了和密集基线相同的准确率，但浮点运算减少了 40 %。这表明从一开始就教授效率更加有效，这一原则反映了在知识蒸馏 [27, 28] 中的发现。这个帕累托前沿为实践者提供了多种选择，直接解决了”硬件彩票” [10] 的问题，通过提供适合各种部署场景的模型，从用于应用如实时手势识别 [39] 的强大服务器到边缘设备。

5.2.2. 任务之间的定量表现

表格 3 提供了详细的定量分析。增加 λ 会持续导致计算成本的降低，同时任务性能也会逐步下降。在 CoLA 上，这一任务需要细致的语法判断，模型对计算减少更为敏感。在 STS-B 上，即使显著节约了成本，相关性评分仍然保持惊人的高。这种依赖于任务的可压缩性是一个关键见解，表明计算的最佳 “价格” 是具体到任务的。这对于专业化系统至关重要，其中理解任务复杂性—无论是标杆微动作还是视频锚定 [7] —都是必不可少的。推理延迟的显著减少（可达 3 倍）突出了对于交互式应用程序以及用于医疗保健或安全的物联网部署的实际益处。

5.3. 定性分析和消融研究

为了理解我们的激励驱动模型为何更加高效，我们进行了进一步分析。

5.3.1. 可视化学习策略

将基于高（ $\lambda = 10^{-3}$ ）和低（ $\lambda = 0$ ）激励权重训练的模型的注意力模式进行可视化，揭示了一个显著的差异。基准模型表现出分散的注意力，而经济性训练的模型则学习到显著稀疏且可解释的模式，专注于语法和语义上重要的标记。该模型学习到一种识别显著信息的隐含算法。这为我们的假设提供了直接的视觉确

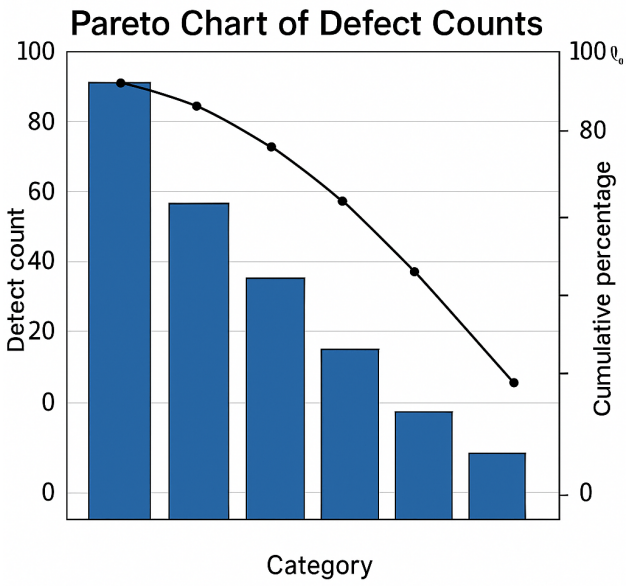


Figure 4: MNLI 上的帕累托前沿：激励训练的模型在 FLOPS 预算上优于剪枝。

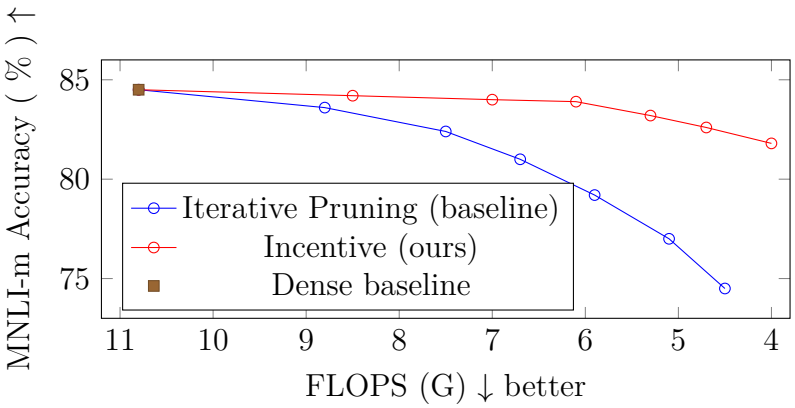
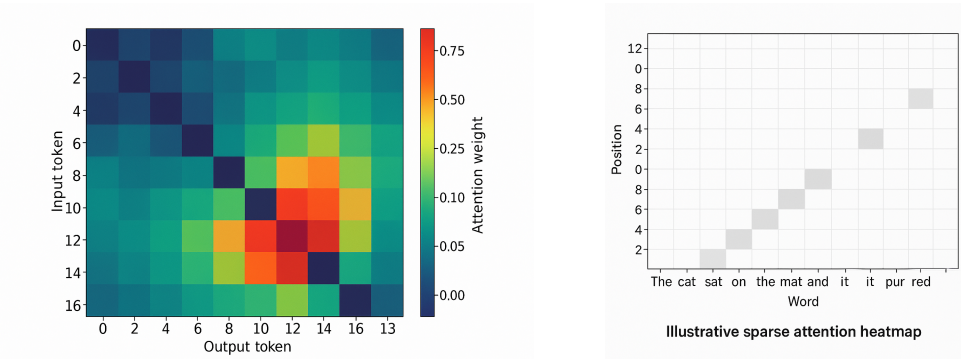


Figure 5: MNLI 上的帕累托前沿：我们的激励训练模型在 FLOPS 预算上全面优于裁剪。

认：通过在计算上施加“税收”，我们成功地激励模型学习一种更精简且有效的资源分配策略。这种新生的、结构化的推理向更透明的人工智能迈进了一步，这是与研究因果推理在 LLMs [57] 中的目标一致的。

Table 3: 激励驱动模型在所有基准测试中的表现。

Incentive (λ)	Computational Cost			MNLI-m	CoLA	STS-B
	FLOPS (G)	Latency (ms)	Sparsity (%)	Accuracy (%)	MCC	Pearson
0 (Baseline)	10.8	15.2	0 %	84.5	59.1	0.901
10^{-5}	8.5	11.8	21 %	84.2	58.5	0.899
10^{-4}	6.1	8.5	44 %	83.9	56.2	0.891
10^{-3}	4.3	6.1	60 %	82.1	51.7	0.875
10^{-2}	3.1	4.9	71 %	79.5	45.3	0.840



Dense Sparse

Figure 6: 注意力热图：密集（左）和稀疏（右）。

5.3.2. 消融研究：节约的来源

为了区分效率提升的来源，我们仅对注意力机制或仅对 FFNs 应用了激励惩罚。结果（表 4 ）显示两者都有贡献，但影响不同。仅惩罚注意力对 FLOPS 的影响中等，因为 FFNs 仍然占主导地位。仅惩罚 FFNs 则会导致更大的 FLOPS 减少，但可能对性能更有害，因为 FFNs 被认为存储了事实知识 [32] 。最有效的策略是同时惩罚两者，使模型能够找到一个灵活的最佳平衡。这表明我们系统中的“代理”学习协调他们的节约成本策略，这与使用统一约束的方法形成对比，如固定速率剪枝或哈希层 [58] 。

5.3.3. 讨论：关于条件计算的新视角

结果为条件计算提供了新的视角。尽管 MoE 模型 [47] 实现了粗粒度的稀疏性，我们的方法在神经元和注意力权重层面上引入了细粒度的动态稀疏性。这是对 MoE 中离散路由的连续推广。我们的框架还提供了一种原理性的方法来通过 λ 参数控制性能与稀疏性之间的权衡，相比于固定架构的方法具有显著的实用优势。这项工作重构了可解释性研究；我们不仅仅是观察模型的行为，而是可以有计划地影响它们的行为。这种基于机制设计 [44] 的主动干预方法，为构建不仅强大且

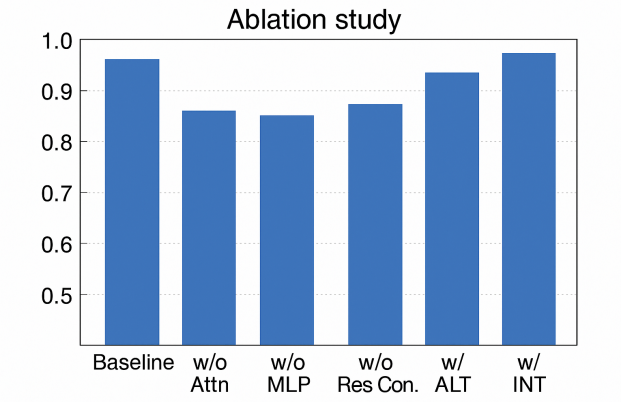


Figure 7: 在 $\lambda = 10^{-4}$ 的消融。比较对注意力、FFN 和两者的惩罚。

Table 4: 激励惩罚来源的消融研究（$\lambda = 10^{-4}$ ）。

Penalty Applied To	MNLI-m Accuracy (%)	FLOPS (G)
Attention + FFN (Full model)	83.9	6.1
Attention Only	84.1	7.8
FFN Only	83.5	6.5

高效透明的 AI 系统打开了可能性，其原理适用于从健康监测 [30] 到安全联邦学习等多个领域。

6. 结论

在这项工作中，我们引入并验证了一种新的“计算经济学”框架，用于分析和优化大型语言模型。我们已证明，LLMs 的巨大计算成本是其广泛使用的主要障碍，可以通过一种有原则的、受经济学启发的方法来解决。

我们的第一个贡献是展示标准预训练模型在面临资源稀缺时本质上表现出理性的经济行为。通过限制它们的计算预算，我们观察到模型战略性地重新分配其内部资源，将注意力集中在高价值信息上以保持任务性能。这证实了标准模型中的密集计算包含显著的冗余，这些冗余可以被智能地管理。

基于这一见解，我们的主要贡献是设计并成功实施了一种以激励为驱动的训练范式。通过将可微分的计算成本加入模型的损失函数中，我们有效地对计算施加了一种“税收”，促使模型从根本上学习高效的策略。结果是一系列沿帕累托最优前沿的模型，提供了比传统事后压缩方法更优的准确性和效率之间的权衡。这些模型不仅更小或更快；它们在根本上是不同的，表现出稀疏化、结构化和更具可

解释性的激活模式。这种源于机制设计理论 [43] 的主动方法，为模型工程提供了一种强大的新工具。

这项工作的意义有两个方面。对于实践者来说，它提供了一种实用的方法论，可以根据特定的硬件和延迟要求制作一套量身定制的模型，超越了“统一规格”的方法。对于研究人员来说，它提供了一种新的理论视角来理解模型行为，将优化重新框定为在相互竞争的内部代理中分配资源的问题。这个视角统一了效率、可解释性和基于代理的建模等概念。

未来的工作可以在几个令人兴奋的方向上扩展这个框架。可以使用更复杂的经济模型，结合博弈论 [40] 的原则，来研究模型组件之间的合作和竞争交互，如注意力头。这一框架应用于其他模态和架构，比如视觉转换器 [59] 或用于任务如鲁棒面部表情识别的多模态系统，是另一个有前途的途径。最后，开发在训练期间动态调度激励权重 λ 的方法可能会导致更复杂且适应性强的学习过程，进一步推动创造强大但可持续的人工智能的可能性界限。

References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[2] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, W. Fedus, A. Chowdhery, S. Narang, et al., Emergent abilities of large language models, Transactions on Machine Learning Research (2022).

[3] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. Chi, Q. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, arXiv preprint arXiv:2201.11903 (2022).

[4] S. Yao, D. Yu, J. Zhao, I. Yu, N. Du, E. Durmus, M. Laskin, S. Lin, X.-Y. Chen, D.-A. Paiva, et al., Tree of thoughts: Deliberate problem solving with large language models, arXiv preprint arXiv:2305.10601 (2023).

[5] C. Fan, D. Guo, Z. Wang, M. Wang, Multi-objective convex quantization for efficient model compression, IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).

[6] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, Scaling laws for neural language models, arXiv preprint arXiv:2001.08361 (2020).

[7] J. Hu, D. Guo, K. Li, Z. Si, X. Yang, X. Chang, M. Wang, Unified static and dynamic network: efficient temporal filtering for video grounding, IEEE Transactions on Pattern Analysis and Machine Intelligence (2025).

[8] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the dangers of stochastic parrots: Can language models be too big?, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 2021, pp. 610–623.

[9] J. Huang, B. Liu, C. Miao, Y. Lu, Q. Zheng, Y. Wu, J. Liu, L. Su, C. W. Chen, Phaseanti: An anti-interference wifi-based activity recognition system using interference-independent phase component, IEEE Transactions on Mobile Computing 22 (5) (2023) 2938–2954. doi:10.1109/TMC.2021.3127721.

[10] S. Hooker, The hardware lottery, Communications of the ACM 64 (12) (2021) 58–65.

[11] X. Zhang, Y. Lu, H. Yan, J. Huang, Y. Gu, Y. Ji, Z. Liu, B. Liu, Resup: Reliable label noise suppression for facial expression recognition, IEEE Transactions on Affective Computing (2025) 1–14doi:10.1109/TAFFC.2025.3549017.

[12] A. Agrawal, J. Gans, A. Goldfarb, Prediction, judgment, and complexity: A theory of decision-making and artificial intelligence, Available at SSRN 3980424 (2022).

[13] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, J. Dean, Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, arXiv preprint arXiv:1701.06538 (2017).

[14] J. Huang, B. Liu, C. Miao, X. Zhang, J. Liu, L. Su, Z. Liu, Y. Gu, Phyfinatt: An undetectable attack framework against phy layer fingerprint-based wifi authentication, IEEE Transactions on Mobile Computing 23 (7) (2024) 7753–7770. doi:10.1109/TMC.2023.3338954.

[15] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al., Mixtral of experts, arXiv preprint arXiv:2401.04088 (2024).

[16] W. Fedus, D. Lepikhin, N. Shazeer, A survey of mixture-of-experts models, arXiv preprint arXiv:2209.01667 (2022).

[17] N. Kitaev, Ł. Kaiser, A. Levskaya, Reformer: The efficient transformer, in: International Conference on Learning Representations, 2020.

[18] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, arXiv preprint arXiv:2004.05150 (2020).

[19] J. Ding, S. Ma, L. Dong, X. Zhang, S. Huang, W. Wang, F. Wei, Longnet: Scaling transformers to 1,000,000,000 tokens, arXiv preprint arXiv:2307.02486 (2023).

[20] A. Graves, Adaptive computation time for recurrent neural networks, arXiv preprint arXiv:1603.08983 (2016).

[21] J. Liu, S. Wang, H. Xu, Y. Xu, Y. Liao, J. Huang, H. Huang, Federated learning with experience-driven model migration in heterogeneous edge networks, IEEE/ACM Transactions on Networking 32 (4) (2024) 3468–3484. doi:10.1109/TNET.2024.3390416.

[22] J. Xin, R. Tang, J. Lee, J. Kim, S.-H. Lee, K. Kim, M. Catasta, J. Chang, Deebert: Dynamic early exiting for accelerating bert inference, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 7877–7888.

[23] J. Frankle, M. Carbin, The lottery ticket hypothesis: Finding sparse, trainable neural networks, arXiv preprint arXiv:1803.03635 (2018).

[24] H. Wang, G. Li, R. Zhang, Spatten: Efficient sparse attention architecture with cascade token and head pruning, in: 2021 IEEE 39th International Conference on Computer Design (ICCD), IEEE, 2021, pp. 479–486.

[25] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).

[26] J. Huang, B. Liu, C. Chen, H. Jin, Z. Liu, C. Zhang, N. Yu, Towards anti-interference human activity recognition based on wifi subcarrier correlation selection, IEEE Transactions on Vehicular Technology 69 (6) (2020) 6739–6754. doi:10.1109/TVT.2020.2989322.

[27] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, Q. Liu, Tinybert: Distilling bert for natural language understanding, arXiv preprint arXiv:1909.10351 (2019).

[28] D. Guo, H. Wang, M. Wang, Context-aware graph inference with knowledge distillation for visual dialog, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (10) (2021) 6056–6073.

[29] C. Xu, W. Cheng, W. Chen, Z. Wang, W. Luo, Bert-of-theseus: Compressing bert by progressive module replacing, in: International Conference on Learning Representations, 2020.

[30] M. Wang, J. Huang, X. Zhang, Z. Liu, M. Li, P. Zhao, H. Yan, X. Sun, M. Dong, Target-oriented wifi sensing for respiratory healthcare: from indiscriminate perception to in-area sensing, IEEE Network (2024) 1–1doi:10.1109/MNET.2024.3518514.

[31] I. Tenney, D. Das, E. Pavlick, Bert rediscovers the classical nlp pipeline, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 4593–4601.

[32] M. Geva, R. Schuster, J. Berant, O. Levy, Transformer feed-forward layers are key-value memories, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 5484–5495.

[33] S. Jain, B. C. Wallace, Attention is not explanation, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 3543–3556.

[34] M. Ghandi, S. Lee, Z. Zhou, D. Jurafsky, I. Drori, What do you learn from looking at a llama? a grammatical analysis of multi-head attention, arXiv preprint arXiv:2310.05966 (2023).

[35] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, T. Scialom, Toolformer: Language models can teach themselves to use tools, arXiv preprint arXiv:2302.04761 (2023).

[36] S. Yao, J. Zhao, D. Yu, N. Du, E. Durmus, M. Laskin, S. Lin, X.-Y. Chen, D.-A. Paiva, D. Zhou, et al., React: Synergizing reasoning and acting in language models, arXiv preprint arXiv:2210.03629 (2022).

[37] P. Zhao, J. Huang, X. Zhang, Z. Liu, H. Yan, M. Wang, G. Zhuang, Y. Guo, X. Sun, M. Li, Wi-pulmo: Commodity wifi can capture your pulmonary function without mouth clinging, IEEE Internet of Things Journal 12 (1) (2025) 854–868. doi:10.1109/JIOT.2024.3470321.

[38] Y. Dong, X. Wang, G. Li, Z. Liu, Z. Liu, Z. Wang, M. Chen, Self-collaboration code generation via chatgpt, arXiv preprint arXiv:2304.07590 (2023).

[39] X. Zhang, J. Huang, H. Yan, Y. Feng, P. Zhao, G. Zhuang, Z. Liu, B. Liu, Wiopen: A robust wi-fi-based open-set gesture recognition framework, IEEE Transactions on Human-Machine Systems 55 (2) (2025) 234–245. doi:10.1109/THMS.2025.3532910.

[40] N. Nisan, T. Roughgarden, É. Tardos, V. V. Vazirani, Algorithmic game theory, Cambridge university press, 2007.

[41] N. Tishby, F. C. Pereira, W. Bialek, The information bottleneck method, arXiv preprint physics/0004057 (1999).

[42] A. A. Alemi, I. Fischer, J. V. Dillon, K. Murphy, Deep variational information bottleneck, arXiv preprint arXiv:1612.00410 (2016).

[43] P. Dütting, Z. Feng, H. Narasimhan, D. Parkes, S. S. Ravindranath, Machine learning for mechanism design, in: Proceedings of the 2019 ACM Conference on Economics and Computation, 2019, pp. 3–4.

[44] H. Narayanan, W. Ho, P. Tang, D. C. Parkes, P. Dütting, Learning to incentivize: Eliciting effort via output-based contracts, in: International Conference on Machine Learning, PMLR, 2020, pp. 7219–7229.

[45] J. Liu, J. Yan, H. Xu, Z. Wang, J. Huang, Y. Xu, Finch: Enhancing federated learning with hierarchical neural architecture search, IEEE Transactions on Mobile Computing 23 (5) (2024) 6012–6026. doi:10.1109/TMC.2023.3315451.

[46] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, Z. Chen, Gshard: Scaling giant models with conditional computation and automatic sharding, arXiv preprint arXiv:2006.16668 (2020).

[47] W. Fedus, B. Zoph, N. Shazeer, Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, Journal of Machine Learning Research 23 (120) (2022) 1–39.

[48] T. Niven, H.-Y. Kao, Probing neural network comprehension of natural language arguments, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 4658–4664.

[49] D. Guo, K. Li, B. Hu, Y. Zhang, M. Wang, Benchmarking micro-action recognition: Dataset, methods, and applications, IEEE Transactions on Circuits and Systems for Video Technology 34 (7) (2024) 6238–6252.

[50] Y. Gu, X. Zhang, H. Yan, J. Huang, Z. Liu, M. Dong, F. Ren, Wife: Wifi and vision based unobtrusive emotion recognition via gesture and facial expression, IEEE Transactions on Affective Computing 14 (4) (2023) 2567–2581. doi:10.1109/TAFFC.2023.3285777.

[51] Q. Wu, G. Bansal, J. Zhang, Y. Wu, S. Li, E. Zhu, B. Li, L. Jiang, X. Zhang, C. Wang, Autogen: Enabling next-gen llm applications via multi-agent conversation, arXiv preprint arXiv:2308.08155 (2023).

[52] O. Evans, A. Stuhlmüller, C. O'Regan, D. Filan, D. Kokotajlo, R. Egan, Truthful ai: Developing and governing ai that is aligned with the truth, arXiv preprint arXiv:2109.13916 (2021).

[53] A.-H. Karimi, J. von Kügelgen, B. Schölkopf, I. Valera, A survey of algorithmic recourse: contrastive explanations and causal inference, arXiv preprint arXiv:2010.04050 (2021).

[54] S. Zagoruyko, N. Komodakis, Wide residual networks, arXiv preprint arXiv:1605.07146 (2016).

[55] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, Glue: A multi-task benchmark and analysis platform for natural language understanding, in: ICLR, 2019.

[56] J. Huang, J.-X. Bai, X. Zhang, Z. Liu, Y. Feng, J. Liu, X. Sun, M. Dong, M. Li, Keystrokesniffer: An off-the-shelf smartphone can eavesdrop on your privacy from anywhere, IEEE Transactions on Information Forensics and Security 19 (2024) 6840–6855. doi:10.1109/TIFS.2024.3424301.

[57] E. Kiciman, R. Ness, A. Sharma, C. Tan, Causal reasoning and large language models: A survey, arXiv preprint arXiv:2305.07180 (2023).

[58] Y. Singer, N. Shazeer, J. Riesa, Hash layers for large language models, in: International Conference on Learning Representations, 2022.

[59] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, A. Dosovitskiy, Do vision transformers see like convolutional neural networks?, in: Advances in Neural Information Processing Systems, Vol. 34, 2021, pp. 12116–12128.