学习跳过变压器的中间层

Tim Lawson* **Laurence Aitchison**
School of Engineering Mathematics and Technology
University of Bristol
Bristol, UK

Abstract

条件计算是一种使 Transformer 更高效的流行策略。现有方法通常针对个别模块（例如，专家混合层）或独立地跳过层。然而，可解释性研究已经表明，Transformer 的中间层表现出更大的冗余性，并且早期层将信息聚合到标记位置。根据这些见解，我们提出了一种新型架构，该架构动态跳过从中间向外开始的不同数量的层。特别是，一个学习到的门控机制基于输入决定是否绕过一组对称的核心块，并且一个门控注意力机制防止后续标记关注被跳过的标记位置。残差归一化使用“三明治”或“周边层归一化”的方案进行控制，并通过自适应正则化损失来控制门控稀疏性。我们的目标是减少“简单”标记的计算需求并可能促进一种新兴的多级表示层次结构，但在我们研究的规模下，与具有较少层数的密集基线相比，我们的方法在验证交叉熵和估计 FLOPs 之间的权衡方面并未实现改进。我们在 `https://github.com/tim-lawson/skip-middle` 发布代码。

1 介绍

我们希望让 Transformers 更高效。这可以通过提高各个模块的效率来实现；例如，已经提出了许多关于注意力机制变体的建议 (Dong et al., 2024)。另一种方法是在推理过程中减少激活的参数数量。条件计算方法将模型的能力（由其总参数数决定）与其推理成本（由给定输入使用的活跃参数子集决定）分离 Bengio et al. 2013, 2016。一个突出的例子是用混合专家（MoE）层替换前馈网络（FFN）模块 (Shazeer et al., 2017)。这些方法减少了计算和内存需求，同时使模型组件能够在多个设备上并行化 (Eigen et al., 2014; Lepikhin et al., 2020; Fedus et al., 2022; Dai et al., 2024)。

减少活跃参数的一种方法是根据输入标记有条件地应用 Transformer 的各个组件。然后，我们可以动态地将较少的计算资源分配给那些“更容易”处理的标记。早期退出方法，在这种方法中深度网络可以在不同层进行预测，在视觉 (Teerapittayanon et al., 2016) 和语言应用 (Elbayad et al., 2020; Xin et al., 2020) 中有着悠久的历史。该方法已被用于动态跳过超过特定深度的 Transformer 层 (Elhoushi et al., 2024; Fan et al., 2024)。其他方法则跳过中间组件 (Wang et al., 2018)，如单独的模块 (Csordás et al., 2021; Peroni and Bertsimas, 2024) 或整个层 (Zeng et al., 2023; Raposo et al., 2024)。

*Correspondence to `tim.lawson@bristol.ac.uk`.

Preprint. Under review.

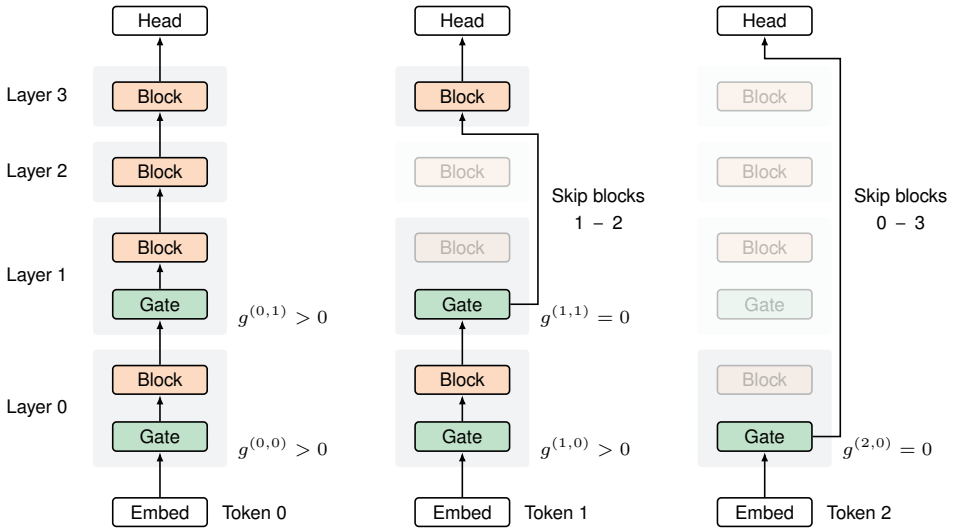

图 1: 我们提出的架构示例（具有四层或块）。我们在模型的前半部分为每个令牌位置和块计算一个标量门值。如果第 ℓ 块中的门为零，我们将跳过该令牌在 ℓ 和 $L-\ell$ 之间的 Transformer 块，并防止其他令牌在其相应自注意力模块的位置上进行关注。

我们主张跳过 Transformer 的中间层更有意义。多位研究人员已经证明中间层表现出更大的冗余性：例如，Lad et al. (2024) 和 González et al. (2025) 发现，在移除或交换预训练模型的层时，影响中心层的操作对性能的影响较小。这种冗余已被用于结构化剪枝 (Fan et al., 2019; Gromov et al., 2024; Men et al., 2024)。

可解释性研究也表明深度网络中的早期、中期和晚期层具有不同的功能。在语言模型中，早期层模块将基于令牌的表示转换为更自然的语义特征 (Elhage et al., 2022; Gurnee et al., 2023)。此外，Kaplan et al. (2024) 显示，对于多令牌单词，早期 Transformer 层的注意力机制会将信息聚合到该词最后一个令牌的残差向量中。相反，晚期层模块将语义特征转换为输出令牌：在网络输出附近，中间状态可以被解码以提取令牌预测 (nostalgebraist, 2020; Belrose et al., 2023)。通过稀疏字典学习 (Lawson et al., 2024) 来看，最早和最晚层的内部激活也与中间层有明显的区别。

令牌与语义特征之间的转换类似于字节级架构 (Xue et al., 2022; Slagle, 2024) 的发展，我们期望能够隐式地学习分词。例如，Pagnoni et al. (2024); Neitemeier et al. (2024); Kallini et al. (2024) 引发了一种由字节级和令牌级表示组成的两层层级结构。鉴于 Kaplan et al. (2024)，我们可以预期这种层级结构可以有利地扩展到跨越多个令牌的层次 (Ho et al., 2024; Videau et al., 2025)。

受这些见解的指导，我们提出了一种跳过机制，根据输入标记从中间向外跳过可变数量的 Transformer 块。这样，我们可以通过对跳过更可能冗余的中间层来为 '较简单' 的输入分配较少的计算资源。层级越中央，处理的标记就越少，从而允许多级表示层次结构出现。不幸的是，在我们能够调查的规模上，这种架构并没有改善语言建模性能与所需计算资源之间的权衡，根据我们在推理时能从门控值的稀疏性中获得最大益处所估计的 FLOPs 来衡量。

2 模型架构

一个标准的仅解码器 Transformer 有 L 层或块，每一层都包含自注意力和 FFN 模块。我们将第 ℓ 层在标记位置 i 处的输入记为 $\boldsymbol{h}^{(i,\ell)} \in \mathbb{R}^d$，其中 d 是模型维度，并将所有标记位置 $i \in 1..N$ 处的输入记为 $\mathrm{H}^{(\ell)} \in \mathbb{R}^{N \times D}$。

高层次上，标准的 Transformer 由（省略层归一化）：

$$\begin{aligned}
\boldsymbol{h}^{(i,0)} &= \mathrm{Embed}(i) \\
\boldsymbol{a}^{(i,\ell)} &= \boldsymbol{h}^{(i,\ell-1)} + \mathrm{Attn}(\mathrm{H}^{(\ell-1)}) \\
\boldsymbol{h}^{(i,\ell)} &= \boldsymbol{a}^{(i,\ell)} + \mathrm{FFN}(\boldsymbol{a}^{(i,\ell)}) \\
\boldsymbol{y}^{(i)} &= \mathrm{Head}(\boldsymbol{h}^{(i,L)}).
\end{aligned}$$

我们建议将此架构修改如下：

$$\begin{aligned}
\boldsymbol{a}^{(i,\ell)} &= \boldsymbol{h}^{(i,\ell-1)} + g^{(i,\ell)}\, \mathrm{GatedAttn}(\mathrm{H}^{(\ell-1)}, \boldsymbol{g}^{(\ell)}) \\
\boldsymbol{h}^{(i,\ell)} &= \boldsymbol{a}^{(i,\ell)} + g^{(i,\ell)}\, \mathrm{FFN}(\boldsymbol{a}^{(i,\ell)}).
\end{aligned}$$

这些修改是高亮显示。如果一个标记位置 i 的门控值 $g^{(i,\ell)} \in \mathbb{R}$ 为零，那么我们就无需计算注意力和前馈模块对应的输出。因此，门控机制充当了一种路由器的角色（图 1）。

2.1 门控机制

对于模型前半部分的每个块 $\ell < L/2$，我们引入一个线性层，该层输出一个标量值 **s 常掩码** $s^{(i,\ell)} \geq 0$，该值在模型前半部分累积：

$$s^{(i,\ell)} = \mathrm{ReLU}\left(\boldsymbol{w}^{(\ell)} \cdot \boldsymbol{h}^{(i,\ell)} + b^{(\ell)}\right), \quad S^{(i,\ell)} = \sum_{\ell' \leq \ell} s^{(i,\ell')}. \tag{1}$$

当累积的软掩码值 $S^{(i,\ell)} \geq 1$ 时，我们在标记位置 i 跳过 Transformer 块 $[\ell, L-\ell)$ 对残差向量的处理。因此，对于模型后半部分的每个块 $\ell \geq L/2$，我们使用其对应块的累积软掩码值 $S^{(i,L-\ell-1)}$。

相应的标量值 **g 吃了** $g^{(i,\ell)} \in [0,1]$ 是：

$$g^{(i,\ell)} = \begin{cases} 1 - \mathrm{clamp}\left(S^{(i,\ell)}, 0, 1\right) & \text{if } \ell < L/2 \\ 1 - \mathrm{clamp}\left(S^{(i,L-\ell-1)}, 0, 1\right) & \text{if } \ell \geq L/2 \end{cases} \tag{2}$$

门值的稀疏性 $g^{(i,\ell)}$ 决定了活跃参数数量的减少：对于单个标记 i，它是门值为零的块数乘以 Transformer 块中的参数数量 N_B。在多个标记的情况下，减少量是 $2N_B \sum_{\ell < L/2} z_\ell$，其中 z_ℓ 是标记在被块 ℓ 处理之前门值的稀疏性。

门控机制引入了参数 $(d+1)L/2$、$\boldsymbol{w}^{(\ell)}$ 和 $b^{(\ell)}$，即每个块 $\ell < L/2$ 在模型前半部分都有一个 $d+1$。如果所有这些参数都为零，则所有的门控值等于一，我们恰好恢复了等效的密集型 Transformer（其构成了第 3 节中的基线）。

2.2 门控注意力

我们还防止其他标记在注意力模块中关注门控标记。我们引入的 GatedAttn 模块修改了注意力机制，使得当某个标记的门值为零时，后续标记不会关注该门控标记：

$$\boldsymbol{o}_i = \frac{\sum_{j<i} g_j \exp(\boldsymbol{q}_i^{\mathrm{T}} \boldsymbol{k}_j) \boldsymbol{v}_j}{\sum_{j<i} g_j \exp(\boldsymbol{q}_i^{\mathrm{T}} \boldsymbol{k}_j)} \tag{3}$$

这相当于向预 softmax 注意力对数几率添加 $\ln g_j$，并且可以在 FlexAttention 框架 (Dong et al., 2024) 内直接实现为分数修改。实际上，我们在 ln 之前将 g_j 的下限设置为 $\epsilon = 1 \times 10^{-6}$，以防止出现无穷大。

我们的注意力机制类似于 Lin et al. (2024) 提出的“遗忘注意力”，不同之处在于我们计算一个适用于每个注意力头的单一门控值，而他们为每个注意力头计算一个门控值。我们需要一个单一门控值来决定是否阻止整个 Transformer 块处理该标记。

2.3 层规范化

现代的变压器通常使用‘预层归一化’方案（预-LN），其中层归一化操作应用于注意力和 FFN 模块的残差输入：

$$\boldsymbol{y} = \boldsymbol{x} + \mathrm{Module}(\mathrm{Norm}(\boldsymbol{x})).$$

采用这种方案，残差激活向量的范数随着深度的增长而增加，后期模块生成具有更大范数的输出 (Lawson et al., 2024; Csordás et al., 2024a; Kim et al., 2025)。

我们提出的门控机制有效地在 Transformer 块的相反对之间引入跳跃连接（章节 2.1）。类似于 Csordás et al. (2024a)，他们考虑使用具有单个共享块的通用 Transformer (UT)，我们希望后续模块接受早期和后期块的输出。我们通过使用 Ding et al. (2021) 提出的“三明治”LN 方案来解决这个问题，该方案由 Kim et al. (2025) 称为“peri-layernorm”，其中层归一化操作应用于注意力模块和 FFN 模块的残差输入和和输出：

$$\boldsymbol{y} = \boldsymbol{x} + \mathrm{Norm}(\mathrm{Module}(\mathrm{Norm}(\boldsymbol{x}))).$$

该方案与 Csordás et al. (2024a) 的“peri-layernorm”不同，后者在残差连接“周围（但不在残差连接上）”应用层归一化操作。

2.4 控制稀疏性

我们的门控架构将前向传递过程中激活的参数数量减少到门值恰好为零的比例，即平均门稀疏度（第 2.1 节）。仅使用标准交叉熵损失时，优化倾向于激活更多参数以提高性能，因此我们需要控制门值的稀疏度。

我们通过引入基于门值的均值和方差的正则化损失来实现这一点，自适应系数根据均值和方差与逐层目标之间的偏差按比例更新。均值项激励较小的门值；方差项激励非均匀分布，使得某些（但不是所有）门值为零。

我们用以下表示一个批次中令牌位置上门值的均值和方差：

$$\overline{g}_\ell = \frac{1}{N}\sum_{i=1}^{N} g^{(i,\ell)}, \quad s_\ell^2 = \frac{1}{N}\sum_{i=1}^{N}\left(g^{(i,\ell)} - \overline{g}_\ell\right)^2. \tag{4}$$

第 ℓ 层的目标对于门值在令牌位置上的总体均值和方差分别是 μ_ℓ^* 和 σ_ℓ^{2*}。除非特别说明，我们选择均值目标 μ_ℓ^* 为初始目标 μ_0^* 与最终目标 $\mu_{L/2}^*$ 之间的线性间隔值。我们选择方差目标 σ_ℓ^{2*} 作为具有 $p = \mu_\ell^*$ 的伯努利分布的方差，即 $\sigma_\ell^{2*} = \mu_\ell^*(1 - \mu_\ell^*)$。

我们将第 ℓ 层门值的均值和方差的自适应系数分别记为 α_ℓ 和 β_ℓ，并将这些系数初始化为零。正则化损失函数如下：

$$\mathcal{L} = \frac{1}{L}\sum_{\ell=1}^{L}\left(\alpha_\ell\,\overline{g}_\ell + \beta_\ell\,s_\ell^2\right). \tag{5}$$

名称	损失	更新规则
`sparsity`	$\frac{1}{L}\sum_{\ell=1}^{L}\alpha_\ell\,\overline{g}_\ell$	-
`sparsity_variance`		-
`adaptive`	$\frac{1}{L}\sum_{\ell=1}^{L}\left(\alpha_\ell\,\overline{g}_\ell+\beta_\ell\,s_\ell^2\right)$	$\alpha_{i+1}=\alpha_i+\gamma\,\mathrm{sign}(\overline{g}_\ell-\mu_\ell^*)$
`proportional`		$\alpha_{i+1}=\alpha_i+\gamma\,(\overline{g}_\ell-\mu_\ell^*)$
`sparsity_variance_l2`	$\frac{1}{L}\sum_{\ell=1}^{L}\left(\alpha_\ell\,\|\overline{g}_\ell-\mu_\ell^*\|_2^2\;+\;\beta_\ell\,\left\|s_\ell^2-\sigma_\ell^{2*}\right\|_2^2\right)$	

表 1: 控制门值稀疏性的替代技术。回忆一下，$\overline{g}_\ell$ 和 s_ℓ^2 是一批中代币位置上的门值的均值和方差，而 μ_ℓ^* 和 σ_ℓ^{2*} 分别是第 ℓ 层上针对门值的均值和方差的目标。对于具有自适应系数的技术，α_ℓ 和 β_ℓ 由相同的算法进行更新。

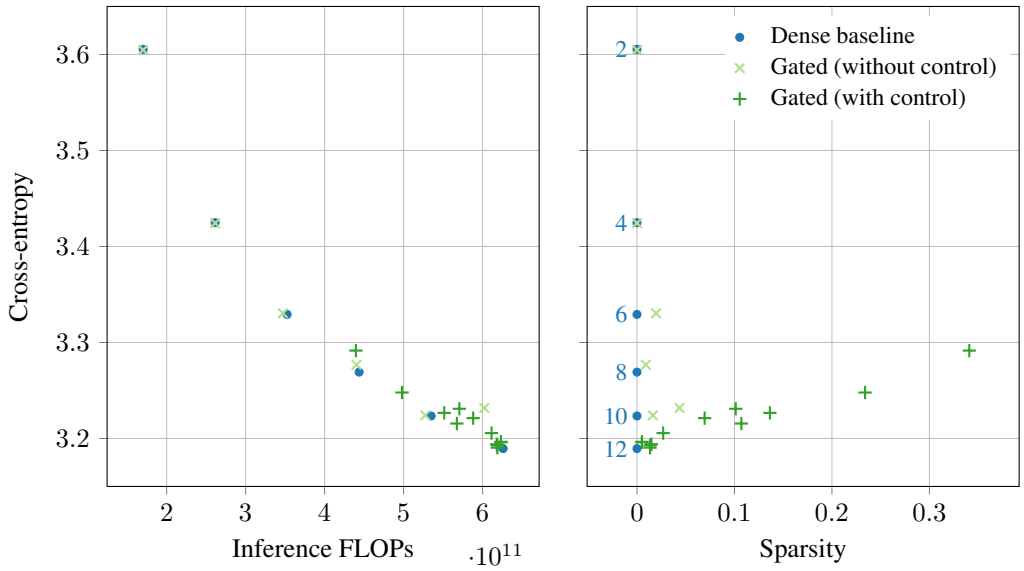

图 2: 我们的带门控的 Transformer 架构与具有 2 到 12 层（标记）的基础模型之间的性能比较。所有带有控制的门控模型都是 12 层架构的变体。我们在 FineWeb 验证集上的 1 亿个令牌上测量了交叉熵。单次前向传递的估计 FLOPs（左侧）假设从验证集（右侧）上的门值最终稀疏性中获得了最大的计算收益。

在每个训练步骤之后，我们按照以下规则更新每个系数：

$$\alpha_{\ell,i+1}=\begin{cases}\alpha_{\ell,i}+\gamma\,(\overline{g}_\ell-\mu_\ell^*) & \text{if } (\overline{g}_\ell-\mu_\ell^*)>\delta\\ \alpha_{\ell,i} & \text{otherwise}\end{cases}\tag{6}$$

因此，更新量与目标值的差异成正比。我们根据小规模实验中的观察结果选择更新倍数 $\gamma=1\times10^{-3}$ 和容差 $\delta=1\times10^{-2}$。我们探索了替代控制机制（表 1），但这些在实证上表现更差。

3 结果

我们评估了在 FineWeb 数据集上预训练的门控 Transformer 架构的性能，以验证交叉熵为标准 (Penedo et al., 2024)。作为基线模型，我们在没有门控机制的情况下训练了等效的密集型

参数	值
模型	
dim	768
n_layers	12
n_heads	12
n_kv_heads	12
vocab_size	50 257
ffn_dim_multiplier	4
multiple_of	256
norm_eps	1×10^{-5}
rope_theta	10 000
use_scaled_rope	False
max_seq_len	1024
initializer_range	0.02

参数	值
数据	
batch_size	512
device_batch_size	32
优化器	
lr	0.001
beta1	0.8
beta2	0.95
eps	1×10^{-10}
weight_decay	0
调度器	
warmup_steps	0.1
start_factor	0.1

表 2: 默认超参数。Transformer 模型架构基于 Llama 3(Grattafiori et al., 2024)，其维度与 GPT-2 小型号 (Radford et al., 2019) 相似；我们使用了 AdamW 优化器 (Loshchilov and Hutter, 2019)，并采用线性预热和余弦衰减。

模型，并且层数范围从 2 到 12 层不等。我们通过估计单次前向传递（批量大小为 1）所需的浮点运算次数（FLOPs）来衡量每个模型的计算需求假设，表明由于门控值的稀疏性，我们能够实现最大可能的好处。门控模型和密集型模型的实际计算需求相似。除非另有说明，否则我们使用了表 2 中的超参数。

图 2 表明，在不控制门的稀疏性时，平均稀疏性趋向于零，并且带门模型的表现类似于密集基线（右侧）。当初始目标平均门 μ_0^* 固定为 1 时，当我们从 1 减少最终目标 $\mu_{L/2}^*$ 到 0 时，稀疏性增加并且估计的 FLOPs 减少（左侧）。然而，所提出的架构并没有在较少层数的情况下改进密集基线的交叉熵。

3.1 实验详情

我们在 FineWeb 数据集的随机采样子集中训练了所有模型，该子集大约有 10B 个标记 (Penedo et al., 2024)，使用 TikToken 库通过 GPT-2 分词器进行预分词 (Radford et al., 2019; Ouyang et al., 2022)。验证集包含约 1 亿个标记。我们使用了一个全局批量大小为 512 个序列（524 288 个标记），并通过数据并行和梯度累积在 4 个 NVIDIA A100 或 GH200 GPU 上实现了每设备的批量大小为 32 个序列。

我们基于 Llama 3 的参考实现构建了底层的 Transformer 模型 (Grattafiori et al., 2024)。特别是，我们使用了：分组查询注意力（GQA；Ainslie et al. 2023）；旋转位置嵌入（RoPE；Su et al. 2024）；带 Swish 激活的门控线性单元 FFN（SwiGLU；Shazeer 2020）；以及根均方（RMSNorm）层归一化 (Zhang and Sennrich, 2019)。相对于 Llama 3 的关键区别在于我们使用了三明治-LN 方案 (Ding et al., 2021; Kim et al., 2025)，而不是预 LN。我们将 RMSNorm 参数初始化为一，并从平均值为零，标准差为 0.02 的正态分布中采样所有其他参数。

训练代码库基于 'nanoGPT speedrun' 仓库 (Karpathy, 2025; Jordan, 2025)。我们使用了 AdamW 优化器，并对所有模型参数 (Kingma and Ba, 2017; Loshchilov and Hutter, 2019) 使用单一的学

习率，以及一个两阶段学习率调度器，在前 10%的训练步骤中进行线性预热，从最大学习率的 10%开始，并在剩余步骤中采用余弦衰减。最后，我们在 PyTorch 中以自动混合精度执行前向传递 `bfloat16`（除了手动将注意力对数转换为浮点 `32`）。

4 相关工作

条件计算通过仅激活给定输入的参数子集，将模型的总参数数量与其推理成本解耦 (Bengio et al., 2013; Eigen et al., 2014; Bengio et al., 2016)。该原理的一个重要应用是使用专家混合层，它用一组较大的“专家”子网络替换 FFN 模块，其中只有少数几个由路由器选择来处理每个输入 (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022; Dai et al., 2024)。虽然像 Zhang et al. (2022); Csordás et al. (2024b); Jin et al. (2024) 这样的相关方法可能有效，但它们操作的是单个模块（例如，FFNs 或注意力机制），并且通常使用强制对每个标记固定计算预算的路由策略 (cf. Wang et al., 2024)。我们的方法通过将条件计算应用于整个 Transformer 块，并根据每个标记的处理需求动态分配可变数量的块来有所不同，这种策略也与模块级技术兼容。

另一种提高效率的方法是动态改变网络深度。早期的退出方法允许模型在中间层生成预测，停止对“简单”输入的计算 (Teerapittayanon et al., 2016; Elbayad et al., 2020; Xin et al., 2020)。这种方法的近期变体可以动态跳过某个深度之后的所有层 (Elhoushi et al., 2024; Fan et al., 2024)。相比之下，我们的方法是基于实证发现 Transformer 的中间层表现出更大的冗余 (Lad et al., 2024; González et al., 2025)。这些发现已被结构化剪枝利用，在训练后静态删除层 (Fan et al., 2019; Gromov et al., 2024; Men et al., 2024)。我们的工作与众不同之处在于它针对更冗余的中间层进行跳过，并且在推理过程中动态执行。

几种方法探讨了跳过整个 Transformer 块的可能性。例如, Csordás et al. (2021) 提出的复制门调节一个块的贡献度，但仍需要计算该块输出的全部内容。然而，我们的门控机制确保可以完全避免跳过块的计算。深度混合（MoD）模型 (Raposo et al., 2024) 在每个块中仅处理前 k 个标记，强制执行一个固定序列计算预算。我们的方法不同之处在于它单独确定每个标记的计算深度，使其能够适应特定于标记的复杂性，而不是整个序列范围内的预算。像 SkipNet(Wang et al., 2018) 这样的方法也允许跳过层，但并不是专门针对在 Transformer 的中间层观察到的独特冗余模式设计的。

我们的门控注意力机制防止了对被屏蔽标记的注意，这在功能上类似于 Lin et al. (2024) 提出的“遗忘注意力”。然而，他们的方法为每个注意力头计算单独的门控，而我们的方法则为每个标记使用单一的门控来决定是否跳过整个 Transformer 块，这是我们目标实现块级计算节省的关键。

最后，我们的工作与分层变换器有关，这些变换器在多个预定义的层级（例如字节和令牌级别；Pagnoni et al. 2024; Neitemeier et al. 2024; Kallini et al. 2024）处理序列。我们从中向外跳过的方法提供了一种更灵活、自发形成的层次结构的可能性，在这种结构中，深层处理一个动态确定的更为复杂的表示子集。

5 结论

我们介绍了一种新型的 Transformer 架构，该架构根据可解释性研究动态跳过不同数量的中间层，这些研究表明这些层次是最冗余的。该机制使用了一个学习到的门控单元，基于输入标记绕过了中心块的一个对称跨度，目标是减少简单标记的计算量并允许表示层次结构的出现。我们的实验表明，在小规模下，与训练具有较少层数的密集基线模型相比，这种架构在验证性能和估计推理 FLOPs 之间的权衡上没有带来改进。这种架构先验的优势可能只有在

显著更大的模型规模中才会显现出来，在这些更大规模的模型中，中间层的冗余性更加明显且门控机制的相对开销更小。尽管如此，我们仍然认为利用模型内部见解来设计更高效和结构化的架构这一原则仍然是未来研究的一个有价值的方向。

参考文献

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, December 2023. URL https://openreview.net/forum?id=hmOwOZWzYE.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting Latent Predictions from Transformers with the Tuned Lens, November 2023. URL http://arxiv.org/abs/2303.08112.

Emmanuel Bengio, Pierre-Luc Bacon, Joelle Pineau, and Doina Precup. Conditional Computation in Neural Networks for faster models, January 2016. URL http://arxiv.org/abs/1511.06297.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation, August 2013. URL http://arxiv.org/abs/1308.3432.

Róbert Csordás, Kazuki Irie, and Jürgen Schmidhuber. The Neural Data Router: Adaptive Control Flow in Transformers Improves Systematic Generalization. In *International Conference on Learning Representations*, October 2021. URL https://openreview.net/forum?id=KBQP4A_J1K.

Róbert Csordás, Kazuki Irie, Jürgen Schmidhuber, Christopher Potts, and Christopher D. Manning. MoEUT: Mixture-of-Experts Universal Transformers. *Advances in Neural Information Processing Systems*, 37:28589–28614, December 2024a.

Róbert Csordás, Piotr Piękos, Kazuki Irie, and Jürgen Schmidhuber. SwitchHead: Accelerating Transformers with Mixture-of-Experts Attention. *Advances in Neural Information Processing Systems*, 37:74411–74438, December 2024b.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models, January 2024. URL http://arxiv.org/abs/2401.06066.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. CogView: Mastering Text-to-Image Generation via Transformers, November 2021. URL http://arxiv.org/abs/2105.13290.

Juechu Dong, Boyuan Feng, Driss Guessous, Yanbo Liang, and Horace He. Flex Attention: A Programming Model for Generating Optimized Attention Kernels, December 2024. URL http://arxiv.org/abs/2412.05496.

David Eigen, Marc'Aurelio Ranzato, and Ilya Sutskever. Learning Factored Representations in a Deep Mixture of Experts, March 2014. URL http://arxiv.org/abs/1312.4314.

Maha Elbayad, Jiatao Gu, Edouard Grave, and Michael Auli. Depth-Adaptive Transformer, February 2020. URL http://arxiv.org/abs/1910.10073.

Nelson Elhage, Tristan Hume, Catherine Olsson, Neel Nanda, Tom Henighan, Scott Johnston, Sheer ElShowk, Nicholas Joseph, Nova DasSarma, and Ben Mann. Softmax linear units, 2022. URL https://transformer-circuits.pub/2022/solu/index.html.

Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed A. Aly, Beidi Chen, and Carole-Jean Wu. LayerSkip: Enabling Early Exit Inference and Self-Speculative Decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12622–12642, 2024. doi: 10.18653/v1/2024.acl-long.681.

Angela Fan, Edouard Grave, and Armand Joulin. Reducing Transformer Depth on Demand with Structured Dropout, September 2019. URL http://arxiv.org/abs/1909.11556.

Siqi Fan, Xin Jiang, Xiang Li, Xuying Meng, Peng Han, Shuo Shang, Aixin Sun, Yequan Wang, and Zhongyuan Wang. Not All Layers of LLMs Are Necessary During Inference, July 2024. URL http://arxiv.org/abs/2403.02181.

William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. ISSN 1533-7928. URL http://jmlr.org/papers/v23/21-0998.html.

Ramón Calvo González, Daniele Paliotta, Matteo Pagliardini, Martin Jaggi, and François Fleuret. Leveraging the true depth of LLMs, February 2025. URL http://arxiv.org/abs/2502.02790.

Aaron Grattafiori, Abhimanyu Dubey, et al. The Llama 3 Herd of Models, November 2024. URL http://arxiv.org/abs/2407.21783.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Dan Roberts. The Unreasonable Ineffectiveness of the Deeper Layers. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=ngmEcEer8a.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding Neurons in a Haystack: Case Studies with Sparse Probing, June 2023. URL http://arxiv.org/abs/2305.01610.

Namgyu Ho, Sangmin Bae, Taehyeon Kim, hyunjik.jo, Yireun Kim, Tal Schuster, Adam Fisch, James Thorne, and Se-Young Yun. Block Transformer: Global-to-Local Language Modeling for Fast Inference. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, November 2024. URL https://openreview.net/forum?id=6osgTNnAZQ.

Peng Jin, Bo Zhu, Li Yuan, and Shuicheng Yan. MoH: Multi-Head Attention as Mixture-of-Head Attention, October 2024. URL http://arxiv.org/abs/2410.11842.

Keller Jordan. KellerJordan/modded-nanogpt, May 2025. URL https://github.com/KellerJordan/modded-nanogpt.

Julie Kallini, Shikhar Murty, Christopher D. Manning, Christopher Potts, and Róbert Csordás. MrT5: Dynamic Token Merging for Efficient Byte-level Language Models. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=VYWBMq1L7H.

Guy Kaplan, Matanel Oren, Yuval Reif, and Roy Schwartz. From Tokens to Words: On the Inner Lexicon of LLMs. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=328vch6tRs.

Andrej Karpathy. karpathy/nanoGPT, May 2025. URL https://github.com/karpathy/nanoGPT.

Jeonghoon Kim, Byeongchan Lee, Cheonbok Park, Yeontaek Oh, Beomjun Kim, Taehwan Yoo, Seongjin Shin, Dongyoon Han, Jinwoo Shin, and Kang Min Yoo. Peri-LN: Revisiting Normalization Layer in the Transformer Architecture. In *Forty-second International Conference on Machine Learning*, June 2025. URL https://openreview.net/forum?id=ci1S6wmXfO&noteId=r07RHYqMC5.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017. URL http://arxiv.org/abs/1412.6980.

Vedang Lad, Wes Gurnee, and Max Tegmark. The Remarkable Robustness of LLMs: Stages of Inference?, June 2024. URL http://arxiv.org/abs/2406.19384.

Tim Lawson, Lucy Farnik, Conor Houghton, and Laurence Aitchison. Residual Stream Analysis with Multi-Layer SAEs. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=XAjfjizaKs.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding, June 2020. URL http://arxiv.org/abs/2006.16668.

Zhixuan Lin, Evgenii Nikishin, Xu He, and Aaron Courville. Forgetting Transformer: Softmax Attention with a Forget Gate. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=q2Lnyegkr8.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, January 2019. URL http://arxiv.org/abs/1711.05101.

Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. ShortGPT: Layers in Large Language Models are More Redundant Than You Expect, October 2024. URL http://arxiv.org/abs/2403.03853.

Pit Neitemeier, Björn Deiseroth, Constantin Eichenberg, and Lukas Balles. Hierarchical Autoregressive Transformers: Combining Byte- and Word-Level Processing for Robust, Adaptable Language Models. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=tU074jg2vS.

nostalgebraist. Interpreting GPT: the logit lens, August 2020. URL https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, March 2022. URL http://arxiv.org/abs/2203.02155.

Artidoro Pagnoni, Ram Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason Weston, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Ari Holtzman, and Srinivasan Iyer. Byte Latent Transformer: Patches Scale Better Than Tokens, December 2024. URL http://arxiv.org/abs/2412.09871.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale, October 2024. URL http://arxiv.org/abs/2406.17557.

Matthew Peroni and Dimitris Bertsimas. Skip Transformers: Efficient Inference through Skip-Routing. October 2024. URL https://openreview.net/forum?id=gdMJlwTcSQ.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-Depths: Dynamically allocating compute in transformer-based language models, April 2024. URL http://arxiv.org/abs/2404.02258.

Noam Shazeer. GLU Variants Improve Transformer, February 2020. URL http://arxiv.org/abs/2002.05202.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer, January 2017. URL http://arxiv.org/abs/1701.06538.

Kevin Slagle. SpaceByte: Towards Deleting Tokenization from Large Language Modeling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, November 2024. URL https://openreview.net/forum?id=KEe4IUp20I.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomput.*, 568(C), February 2024. ISSN 0925-2312. doi: 10.1016/j.neucom.2023.127063.

Surat Teerapittayanon, Bradley McDanel, and H.T. Kung. BranchyNet: Fast Inference via Early Exiting from Deep Neural Networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2464–2469, December 2016. doi: 10.1109/ICPR.2016.7900006.

Mathurin Videau, Badr Youbi Idrissi, Alessandro Leite, Marc Schoenauer, Olivier Teytaud, and David Lopez-Paz. From Bytes to Ideas: Language Modeling with Autoregressive U-Nets, 2025. URL https://arxiv.org/abs/2506.14761.

Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E. Gonzalez. SkipNet: Learning Dynamic Routing in Convolutional Networks. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, pages 420–436, Berlin, Heidelberg, September 2018. Springer-Verlag. ISBN 978-3-030-01260-1. doi: 10.1007/978-3-030-01261-8_25.

Ziteng Wang, Jun Zhu, and Jianfei Chen. ReMoE: Fully Differentiable Mixture-of-Experts with ReLU Routing. In *The Thirteenth International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=4D0f16Vwc3.

Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. DeeBERT: Dynamic Early Exiting for Accelerating BERT Inference. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2246–2251, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.204.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a Token-Free Future with Pre-trained Byte-to-Byte Models. *Transactions of the Association for Computational Linguistics*, 10:291–306, March 2022. ISSN 2307-387X. doi: 10.1162/tacl_a_00461.

Dewen Zeng, Nan Du, Tao Wang, Yuanzhong Xu, Tao Lei, Zhifeng Chen, and Claire Cui. Learning to Skip for Language Modeling, November 2023. URL http://arxiv.org/abs/2311.15436.

Biao Zhang and Rico Sennrich. Root Mean Square Layer Normalization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Xiaofeng Zhang, Yikang Shen, Zeyu Huang, Jie Zhou, Wenge Rong, and Zhang Xiong. Mixture of Attention Heads: Selecting Attention Heads Per Token. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4150–4162, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.278.