思考更深入，而非更长：用于组合泛化的深度循环 Transformer

Hung-Hsuan Chen国立中央大学计算机科学与信息工程系，台湾桃园
hhchen1105@acm.org

arXiv:2603.21676v1 [cs.LG] 23 Mar 2026

摘要

标准 Transformer 的计算深度固定，从根本上限制了其对需要可变深度推理的任务的泛化能力，例如多跳图遍历或嵌套逻辑。我们提出一种深度循环 Transformer，通过在潜在空间中迭代应用共享权重的 Transformer 块，将计算深度与参数数量解耦，使模型能够在推理时以增加循环步数为代价进行更深入的推理。我们的架构采用三种机制来确保深度循环（20+ 步）的稳定性：（1）静默思考目标，仅监督最终输出，迫使模型进行真正的多步推理，而非采用中间启发式捷径；（2）LayerScale 初始化，保护脆弱的推理状态免受未经训练的层噪声影响；（3）恒等偏置循环，在多个步骤间建立梯度高速通道。我们在三个归纳偏置逐渐减弱的组合推理领域进行评估：图可达性（严格的邻接掩码）、嵌套布尔逻辑（相对位置编码）以及无结构关系文本（序列位置不提供任何结构线索）。在所有任务中，我们都观察到一个明确的计算前沿——随着思考步数随任务复杂度增加，性能会从随机水平跃升至接近完美的边界。此外，这些任务展现出性质迥异的泛化行为：精确但脆弱（图）、近似但稳健（逻辑），以及在缺乏结构线索时自主进行潜在路由（文本）。这一递进过程阐明了任务不变的循环推理核心与任务特定的感知接口之间的相互作用如何塑造分布外（OOD）泛化，并为垂直思维链提供了一种机制层面的视角，与当前主流的水平令牌生成范式形成补充。

关键词 思维链　　思维链 · 思维链 · 垂直思维链 · VCoT · 水平思维链 · HCoT · LL　　M

1 引言

大语言模型（LLM）在广泛的任务中取得了卓越的表现，但其推理能力仍受到架构的根本性限制。面对需要多步逻辑演绎的问题——例如规划、数学证明或算法执行——当前模型高度依赖思维链 （CoT）提示（Wei 等人，2022年），将中间推理过程外化为生成的词元序列。我们将这种范式称为水平循环：模型通过横向延长输出序列来“思考”，每进行一步推理都会消耗可用的上下文长度。

尽管水平思维链已被证明非常有效，但它也存在若干根本局限。首先，每个推理步骤都需要消耗一个或多个词元，很快就会耗尽有限的上下文窗口。其次，每个词元位置可进行的计算深度由 Transformer 层数决定，且固定不变，不受问题难度影响。无论输入只需进行简单的模式匹配，还是需要深度递归求值，一个 32 层的 Transformer 都会恰好执行 32 层处理。第三，由于中间推理步骤以自然语言生成，它们容易累积错误——每次词元预测都会带来幻觉或逻辑失误的风险，并将其传递下去。

在这项工作中，我们提出一种正交范式，称为 垂直思维链：模型不再沿水平方向生成更多词元，而是在潜在空间中通过递归应用共享权重 Transformer 块来“思考得更深入”。这种方法将计算深度与参数数量及上下文长度解耦。对于更难的实例，模型只需在推理时增加递归步数，就能投入更多计算，而无需生成额外词元或占用更多上下文窗口空间。
要让深度循环有效运行，关键挑战在于稳定性。朴素地将 Transformer 块展开多个步骤，会导致梯度爆炸或梯度消失，以及表征坍塌。我们通过三种相辅相成的机制来应对这些问题，依次从推理目标到物理约束展开：

1. 静默思考：我们仅在最后一个循环步骤应用监督，不设置中间辅助损失。这迫使模型发展出真正的多步推理路径，而不是学习满足逐步监督的启发式捷径。2. LayerScale 初始化：在注意力和前馈子层之后应用逐通道缩放，并将其初始化为 10^{-4}。这可以防止初始随机权重在训练早期破坏精心保留的隐藏状态，起到空间保护作用，维护脆弱的逻辑表征。3. 恒等偏置循环：为解决将网络展开 20+ 步这一物理限制，我们采用门控偏置初始化为 -2.0 的门控循环。Sigmoid 门的初始值接近 0。12，并强烈倾向于保留先前的隐藏状态。这形成了一条时间梯度高速通道，使信号能够稳定传播，是实现无限深度的核心引擎。

为了系统评估深度循环 Transformer 的能力，我们设计了由三个分阶段的分布外（OOD）推理任务组成的序列。首先，我们从图可达性开始，通过拓扑掩码提供严格的物理概念验证。随后，我们借助嵌套布尔逻辑提升结构复杂度，并利用相对位置编码展示模型维持脆弱层次状态的能力。最后，在一个基于非结构化文本的关系组合任务中，我们移除所有任务对齐的结构归纳偏置，证明我们的不变推理核心能够自主发现自然语言中的复杂潜在路由路径。之所以选择这些任务，是因为它们的计算深度可以精确控制，从而能够严格分析泛化行为。

我们的实验揭示了一种一致的模式，我们将其称为计算前沿：在准确率热力图中（思考步数 $\times$ 任务复杂度），存在一条对角线边界，模型性能会在此处从随机水平骤然转变为接近完美。此外，在相同的循环核心下，这三个任务呈现出定性不同的泛化特征，我们将其归因于各自感知接口所具有的不同归纳偏置。

我们的贡献总结如下：

- 我们提出了一种具有静默思考、LayerScale 和恒等偏置循环的深度递归 Transformer 架构，使其能够在少于 100 万个参数的情况下，在 20+ 步上实现稳定递归。
- 我们展示了该模型在三个具有不同归纳偏置的组合任务上具备强大的分布外泛化能力，并能稳健地外推至严格长于训练分布中推理深度的推理深度。
- 我们识别并分析了计算前沿现象，并展示了任务特定感知接口如何产生定性不同的泛化行为（精确但脆弱 vs. 近似但稳健）。
- 我们提供了证据表明，中间监督可能会产生不利影响，导致模型学习在分布偏移下崩溃的启发式捷径。

2 相关工作

2.1 思维链与测试时计算

思维链提示（Wei 等人，2022年）及其变体（Kojima 等人，2022年；Wang 等人，2023年）已成为大语言模型推理的主流范式。近期关于测试时计算扩展的研究（Snell 等人，2025年）进一步表明，允许模型执行更多推理时计算能够提升性能。然而，这些方法都通过横向生成令牌来运作，消耗的上下文窗口与推理深度成正比。我们的工作探索了一个正交维度——潜在空间中的纵向深度循环——在不产生令牌开销的情况下实现测试时计算扩展。

2.2 暂停令牌与潜在推理

Goyal 等人（2024年）提出在输入中附加可学习的“暂停令牌”，让 Transformer 在生成输出之前增加一次前向传播计算。近期的工作，例如 Coconut（Hao 等人，2024年），也在探索

训练大语言模型在连续潜在空间中进行推理。尽管这些方法与我们的动机相同，都是为了在推理时提供额外计算，但暂停令牌本质上仍是横向的：它们在线性序列中占据位置，并消耗上下文窗口。此外，每个暂停令牌仍要经过相同固定数量的层，因此每个位置的计算深度 并未改变。相比之下，我们的方法通过反复应用同一模块直接增加深度。不同于传统循环神经网络将整个序列压缩为单个瓶颈向量 $h \in \mathbb{R}^d$，我们在每一步都保留完整序列长度的状态矩阵 $H \in \mathbb{R}^{L\times d}$，从而保留丰富的空间交互。此外，不同于通过附加新生成的令牌来消耗有限上下文窗口的横向思维链，我们的递归严格在潜在空间中运行，不会增加序列长度。

2.3 通用 Transformer

通用 Transformer（UT）（Dehghani 等，2019年）引入了跨层权重共享，并采用自适应计算时间（ACT）（Graves，2016）实现动态停止。我们的工作以此为基础，但在几个关键方面有所不同。首先，我们采用仅最终步监督（静默思考），而非逐步损失；我们通过实验证明，这样可以避免启发式捷径学习。其次，我们引入 LayerScale（Touvron 等人，2021年），以保护训练早期的潜在表示。第三，我们采用带负偏置初始化的恒等偏置门控递归，而非简单的残差连接；我们发现，这对于超过 10 个递归步时保持稳定至关重要。第四，我们不依赖 ACT 复杂的词元级停止概率和思考成本正则化（Graves，2016），而是将计算深度完全解耦。通过将递归步数 T 设为外部指定的预算，我们的模型原生支持灵活的测试时计算扩展，且没有额外的优化开销。这些差异使模型能够稳健地进行分布外外推，而不仅仅是分布内迁移。

2.4 Transformer 的深度与表达能力

理论研究已证实，Transformer 的深度是影响其表达能力的关键因素。Merrill 和 Sabharwal（2024）指出，固定深度的 Transformer 受限于 TC^0 电路复杂度，因此无法处理本质上需要顺序计算的任务。Feng 等（2023）进一步证明，深度高效 Transformer 若没有足够的层数，就无法解决某些组合任务。通过递归使深度能够动态变化，我们的架构从根本上绕过了 TC^0 的限制。这种解耦使展开的推理步骤能够严格按照输入固有的序列复杂度扩展，从而让模型能够解决多跳路由和嵌套逻辑等本质上需要顺序计算的任务——而固定深度网络在数学上无法完成这些任务。

2.5 神经算法推理与归纳偏置

神经算法推理领域的研究强调，神经网络必须与目标任务的算法原语相匹配。图神经网络（GNN）（Gilmer 等，2017；Xu 等，2019）通过沿边进行消息传递来实现这一点。我们证明，只需在 Transformer 的自注意力矩阵上应用邻接掩码，Transformer 就能完美模拟最优的图神经网络。随后，我们在逻辑和非结构化文本实验中逐步移除这些结构先验，以检验不变推理核心的能力边界。

3 方法

我们的架构由两个部分组成：一个任务特定感知接口，用于将原始输入编码为初始隐藏表示；以及一个任务不变推理核心，通过共享权重递归对该表示进行迭代优化。经过 T 个递归步骤后——其中 T 可在推理时灵活调整——任务特定读出头 会提取最终预测。

3.1 不变推理核心

推理核心是一个共享权重的 Transformer 块 f_θ，通过递归方式应用 T 步。它作用于隐藏状态 $H^{(t)} \in \mathbb{R}^{L\times d}$，其中初始状态 $H^{(0)}$ 是任务特定感知接口的输出（第 3.2 节）。设 L 为序列长度，d 为隐藏维度。在每一步 $t \in \{1,\dots,T\}$，候选隐藏状态 $\tilde{H}^{(t)} \in \mathbb{R}^{L\times d}$ 的计算方式如下：

$$\tilde{H}^{(t)} = f_\theta(H^{(t-1)}, t), \tag{1}$$

有关 f_θ 的详细信息，请参见附录 A。

为确保这个循环核心能够稳定地展开 20+ 个步骤并学习真正的算法，我们围绕三种具体机制进行设计。

静默思考目标。 一个基本的设计选择是如何监督循环步骤。我们不强制模型给出中间答案，也不对每一步施加辅助损失，而是仅在最终思考步骤 T 上计算交叉熵（CE）损失 CE($\hat{}y,\ \ y$)。

训练期间， T会从任务特定范围内均匀采样（例如， $T \sim \mathcal{U}$(1，12)）。这种随机化的深度起到了时间正则化的作用，促使网络在不同计算深度下生成稳定的输出；与此同时，静默思考目标则迫使模型形成真正属于自己的多步推理路径。

预层归一化和 LayerScale。 对于需要大量步骤的复杂符号任务，反复进行子层变换所累积的方差可能会严重影响训练初期的稳定性。我们采用预层归一化（Xiong 等人，2020年），并在每个子层之后应用可学习的逐通道 LayerScale（Touvron 等人，2021年），将输出与一个初始化为 $\Gamma_i = 10^{-4}$ 的可学习缩放向量 $\Gamma \in \mathbb{R}^d$ 逐元素相乘。LayerScale 会在初始化时将输出缩小至接近零，从而使训练初期的动态几乎完全表现为恒等映射，保护脆弱的布尔状态，使其能够安全绕过随机初始化的深层所引入的破坏性噪声。随着训练的推进，网络会有选择地放大 Γ，以激活其推理能力。

恒等偏置门控递归。 为了突破网络展开 20+ 步时的物理极限，同时避免梯度消失，我们通过一个经过学习的类似 GRU 的门控，将新的候选表示 $\tilde{H}^{(t)}$ 与前一状态 $H^{(t-1)}$ 结合起来：

$$z^{(t)} = \sigma\Big([\tilde{H}^{(t)};, H^{(t-1)}] \cdot W_z + b_z\Big) \tag{2}$$

$$H^{(t)} = z^{(t)} \odot \tilde{H}^{(t)} + (1 - z^{(t)}) \odot H^{(t-1)} \tag{3}$$

其中， $[\tilde{H}^{(t)};\ \ H^{(t-1)}] \in \mathbb{R}^{L\times 2d}$表示两个状态在特征维度上的拼接，而 $W_z \in \mathbb{R}^{2d\times d}$将其投影，得到 $z^{(t)} \in \mathbb{R}^{L\times d}$。该门控在每个序列位置逐元素运作；详情见附录 A。
关键在于，我们将门控偏置 $b_z = -2.0$ 初始化为 σ。由于 -2.0（ ≈ 0.12），模型默认保留先前状态的 88 %。这在物理上保证了初始信号缓慢衰减，从而能够在深度时间展开过程中稳定地传播信号。

深度嵌入。 为防止模型在循环回路中迷失当前位置，我们在每次迭代时、应用注意力块之前注入一个经过学习的步骤嵌入 $e_t \in \mathbb{R}^d$： $H^{(t-1)}{}_{\text{输入}} = H^{(t-1)} + e_t$。详情见附录 B。

3.2 特定任务的感知接口

我们不将特定任务规则硬编码到推理核心中，而是设计模块化感知接口，将不同程度的结构归纳偏置注入初始状态 $H^{(0)}$。我们探索了三种不同的接口范式：

拓扑掩码（严格结构先验）。为了施加物理约束，确保计算严格遵循定义的拓扑路径（例如图路由），我们引入邻接掩码。给定邻接矩阵A，我们添加自环（$A_{ii} = 1$），并构造加性掩码矩阵 $M \in \mathbb{R}^{n\times n}$：

$$M_{ij} = \begin{cases} 0, & \text{if } A_{ij} = 1 \\ -\infty, & \text{if } A_{ij} = 0 \end{cases} \tag{4}$$

注意力的计算方式为 softmax($QK^T/\sqrt{d_k} + M$)V。从物理意义上说， $-\infty$ 项会强制非相邻节点的注意力概率精确为零，从而完美模拟严格的消息传递神经网络。

相对位置编码（层级先验）。对于结构层级取决于相对距离而非严格邻接关系的序列（例如评估嵌套符号表达式），拓扑掩码的限制过于严苛。因此，我们采用旋转位置嵌入（*RoPE*）（Su 等，2024年）。通过将查询向量和键向量分别乘以对应其绝对位置 m 和 n （$R_{\Theta,\ m}$ 和 $R_{\Theta,\ n}$）的旋转矩阵，内积便会自然编码它们的相对序列距离（$m - n$）：

$$\langle \tilde{q}_m, \tilde{k}_n \rangle = q^T R_{\Theta, m-n} k \tag{5}$$

在这一接口下，注意力仍然是完全双向的，完全依赖旋转偏置来维持脆弱的层次状态。

标准序列注意力（与任务无关的先验）。为了测试推理核心能否在非结构化序列（例如自然语言事实）中自主发现潜在路由路径，我们移除了特定于任务的结构偏置。虽然我们保留了标准旋转位置嵌入（RoPE），使感知接口能够处理局部词序，但输入事实完全经过打乱。因此，不同于逻辑领域中相对距离与层次深度直接相关的情况，这个事实袋中的一维序列距离无法为底层关系图提供任何有意义的结构线索。不变推理核心必须完全依靠自身发现正确的指针追踪路径。

3.3 特定任务的读出机制

潜在推理展开 T 步后，读出头会解码最终状态 $H^{(T)}$。我们根据接口采用不同的读出机制：

- 成对节点读出（用于拓扑领域）：提取与特定源节点和目标节点对应的 d 维表示，将其拼接后输入多层感知机。
- 全局序列读出（用于层次领域）：提取位置 0 处 [CLS] 词元的全局表示，并将其输入线性分类器。
- 潜在指针读出（用于非结构化领域）：与成对读出类似，提取并拼接被查询实体的表示。这种局部读出机制促使不受约束的注意力机制在整个序列中主动路由特定实体之间的信息。

4 实验

我们在三个结构归纳偏置逐渐减弱的组合推理领域上评估了我们的架构：图可达性、布尔逻辑和关系文本。LayerScale 仅用于实验 II 和 III，因为其中脆弱的符号状态和更多的思考步数需要额外的稳定性；图任务依靠拓扑掩码进行结构正则化，因此不需要 LayerScale。在这些任务中，评估的最大推理步数各不相同，这反映了各自结构复杂性的自然上限。每幅图都用虚线标出深度和思考步数的训练边界；超出这些范围的坐标轴表示分布外评估。

4.1 实验 I：图可达性

给定一个有向图 $G = (V, E)$，判断是否存在一条从节点 s 到 t 的有向路径。模型在需要 1–5 跳、5–8 个思考步数的实例上进行训练，并评估至多 12 跳，以测试分布外泛化能力。
如图 1 所示，我们观察到了一个明显的计算前沿。准确率从随机水平跃升至完美准确率的转变类似于阶跃函数：恰好在 N 个思考步数时，模型能够解决 N 跳查询。少一个步数，准确率便会降至随机水平。模型在最多 8 跳范围内实现了 100% 的分布外泛化（1.6×），但在 10 跳时突然崩溃，表明拓扑掩码施加了清晰且 rigid 的泛化边界。在步数维度上，模型在训练范围以下（1–3 步）和以上（12–20 步）同样能够稳定泛化，相应的对角线前沿也随之移动。

4.2 实验 II：嵌套布尔表达式求值

给定一个脆弱的嵌套布尔表达式（例如 !((T&F)|(!(T|F)))），将其求值为 True 或 False。扩大模型宽度（$d = 256$）并添加 LayerScale 后，可以在嵌套深度 1–8 上成功训练。
如图 2 所示，我们观察到一条逐渐形成的计算前沿。模型能够平稳地泛化到训练分布之外，在深度 14（1.75× 分布外）时达到 >90% 的准确率。不同于图任务，随着思考步数增加，准确率单调提升，且不会崩溃。即使计算量增加到 24 步，模型也不会“过度思考”或出现性能下降（在步数维度上属于分布外），这证实了负门控偏置的稳定性。

4.3 实验 III：非结构化文本中的关系组合

为了在纯语言建模情境下测试推理核心的能力，我们在 CLUTRR 风格的家族关系组合任务上对其进行评估。输入是一系列经过随机打乱的自然语言句子，用于定义关系（例如，爱丽丝是鲍勃的父母），并填入相互矛盾的干扰项句子，以防止模型利用统计捷径。模型必须回答诸如“爱丽丝是伊芙的兄弟姐妹吗？”这样的查询。

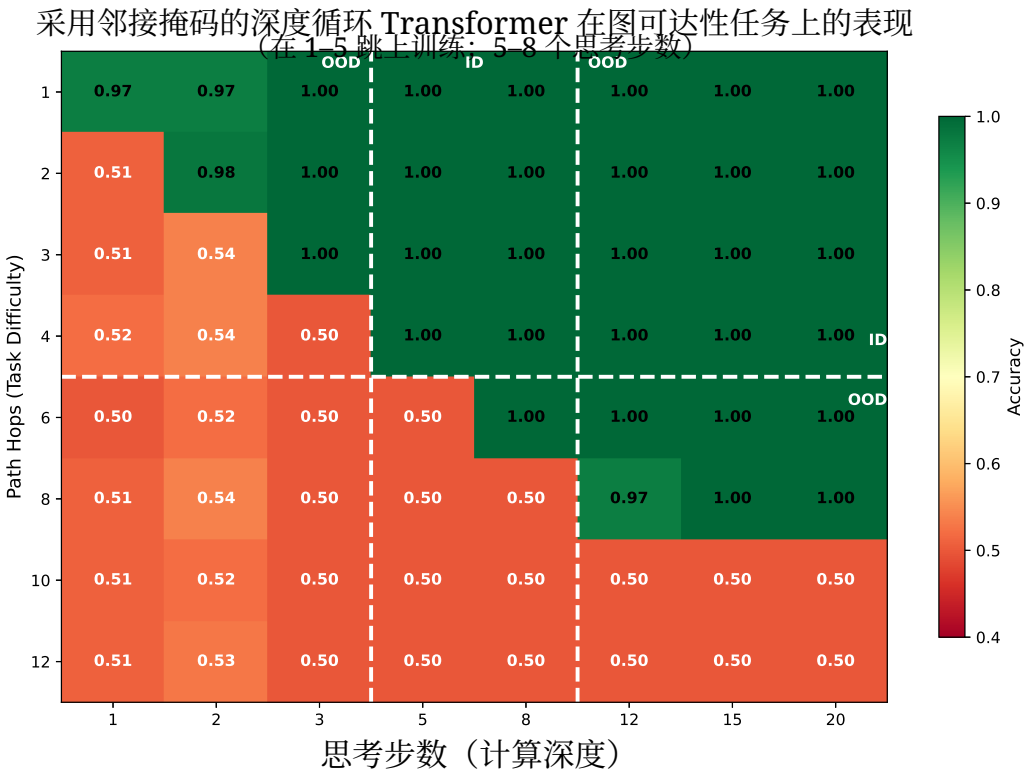


图 1：图可达性任务的准确率热力图。沿对角线存在明显的计算前沿，证实邻接掩码施加了严格的 1 跳 $= 1$ 步物理约束。只要思考步数足够，模型便能完美泛化至 8 跳（分布外），但在 10 跳时准确率会骤然崩溃至随机水平（∼50%）。虚线标示步数训练边界（5–8 个思考步数）；范围外的列同样属于分布外，但准确率仍得以保持。

天真地生成关系链（例如，关系全是父母或全是子女），会让模型得以通过简单地计数关系词或检测奇偶性来作弊。为确保模型真正进行算法推演（指针追踪和数学偏移抵消），我们使用顶点路由策略生成关系链：逻辑路径必须先沿家谱树向上到达共同祖先（使用父母关系），再向下移动（使用子女关系）。我们将兄弟姐妹（偏移量为 0）加入词汇表，使正偏移量和负偏移量可以完全抵消。此外，我们精心构造了困难负样本——这些错误选项经过刻意设计，在表层统计特征上与正确答案相似（例如，将祖父母作为兄弟姐妹的干扰项，因为两者都有偶数步偏移）。通过确保这些干扰项的奇偶性与正确关系严格一致，我们杜绝了浅层统计捷径，迫使模型进行真正的潜在路由，而不是表层模式匹配。

该模型经过训 在深度 2–5 下配置了 1–12 个思考步数，并评估至深度 9 和 20 个思考步数 s.

如图 3 所示，结果揭示了三个关键发现。首先是单调难度：随着推理深度增加，准确率严格下降。由于所有奇偶性捷径都已移除，模型必须进行真正的潜在路由，而链条越长，这一过程自然越困难。其次是计算前沿：对于任意给定的深度，增加思考步数都会严格提升准确率（例如，在深度 5 时，准确率从 1 步时的 63.8% 提升至 12 步时的 81.7%，并在 20 步时保持稳定，为 80.6%（分布外））。第三是稳健的分布外泛化：尽管缺少任务对齐的结构提示（没有图掩码，且对于完全打乱的句子，1D 相对位置无法提供捷径），不变核心仍成功泛化至深度 6 和 7；随着分布外思考步数增加（16–20 步），深度 7–9 还取得了幅度不大但持续的提升，证明它能够自主地在非结构化文本中发现指针追踪路径。

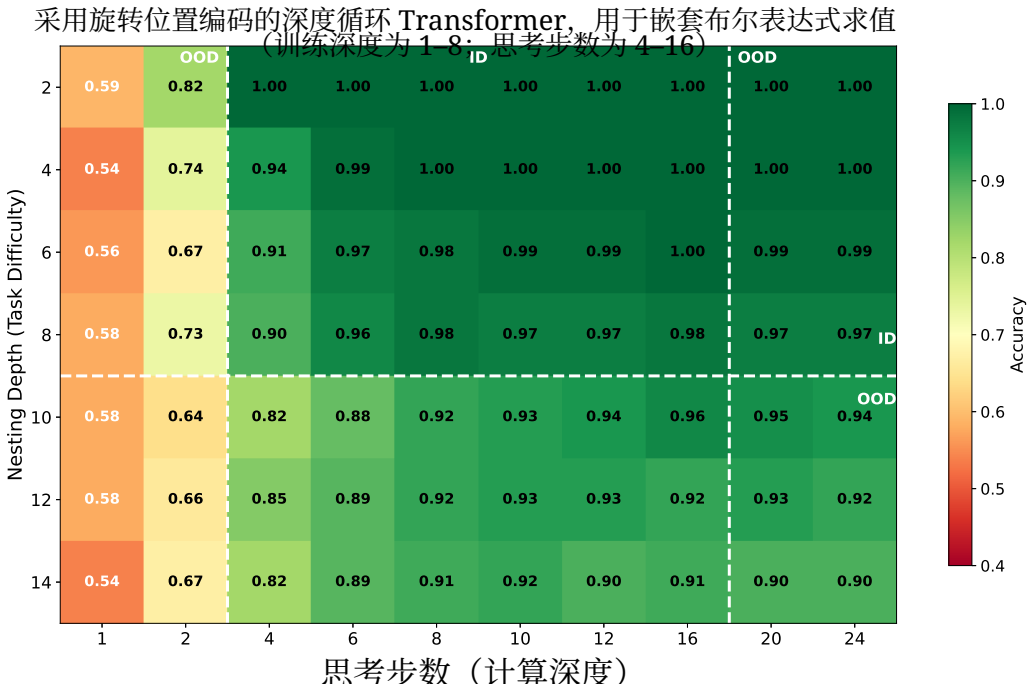


图 2：嵌套布尔表达式求值任务的准确率热力图。与图任务相比，此处的计算前沿更加平缓。模型展现出稳健的分布外泛化能力，并平滑退化，在深度达到 14 时仍保持 $>90\%$ 的准确率。值得注意的是，即使将模型展开至 24 步（远超 4–16 步的训练范围），其性能仍高度稳定，这验证了我们设计的有效性。

表 1：图可达性任务上的监督策略比较。中间监督会诱导出统计捷径（在 12 跳路径上，模型在第 1 步就取得高得不可能的准确率），占用模型容量并破坏真正的分布外泛化。

监督策略	12 跳（捷径）上的第 1 步准确率	充分步数 OOD 准确率
中间（逐步）损失	~73.0%	~50.0%（崩溃）
静默思考（仅最终步骤）	~50.0%（随机猜测）	**最多 8 跳时完美**

4.4 消融分析：中间监督的危险

一种提升算法推理能力的自然设想是采用中间监督</style]——在每个思考步骤计算损失并取平均，从而创建梯度高速通道。为检验这一设想，我们在图可达性任务上，将静默思考目标与中间监督基线进行消融对比。该领域非常适合用于诊断，因为邻接掩码提供了可通过物理方式验证的真实标签：执行 k 步的模型只能聚合来自 k 跳之外的信息。

如表 1 所示，中间监督呈现出一个明显的异常：仅经过单 步思考后，它在 12 跳路径上的准确率就超过了 70%。在严格的拓扑掩码下，1 步模型完全无法获取相距 12 跳的节点的信息；其真实准确率必然 接近 50 %。这从数学上证明，模型已经放弃了真正的消息传递。相反，它学会了统计启发式方法——例如根据图密度或源节点度数估算可达性——以贪婪地降低早期步骤的训练损失。

我们将此归因于一种带宽占用失败模式。在第 1 步受到惩罚时，模型面临两种选择：采取“诚实策略”（接受 50% 的早期准确率，以学习深层传播），或采取“捷径策略”（学习浅层启发式，以立即获得奖励）。中间监督让捷径变得难以抗拒。一旦采用这些启发式，模型就会失去开发真正顺序算法的动力，即使在测试时给予大量步骤，在深层 OOD 路径上仍会失败。

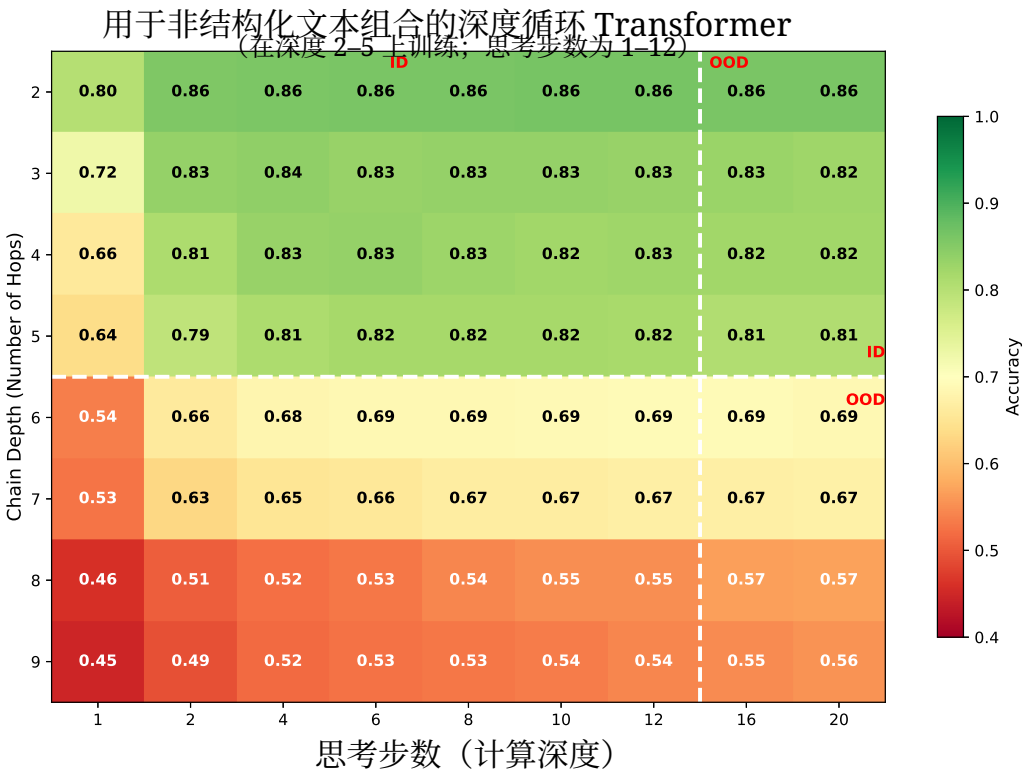


图 3：非结构化文本上的关系组合任务准确率热力图。缺少任务对齐的结构归纳偏置（未使用邻接掩码；虽然保留了旋转位置编码以表示局部词序，但输入事实经过完全打乱，确保一维相对位置不携带有意义的结构信号），因此任务难度严格单调递增。不变推理核心能够发现潜在指针追踪路径；在提供额外思考步数时，它在深度 6 和 7 上实现了良好的分布外泛化。虚线标示了深度维度和步数维度上的训练边界。

我们的静默思考明确消除了这种信用分配捷径。通过仅在最终步计算损失，模型摆脱了早期步骤惩罚。它不得不接受早期层中的随机猜测表现，并将全部表征能力投入到学习真正的潜在算法上。归根结底，在算法推理中，中间状态代表的是不可观测的潜在计算，而非语义特征，因此强迫模型给出中间答案反而有害。

5 结论

我们提出了一种深度循环 Transformer 架构，实现了垂直思维链——通过在潜在空间中迭代而非生成词元来进行推理。借助静默思考、LayerScale 和恒等偏置循环，我们实现了在 20+ 步上的稳定递归。在图可达性、嵌套布尔逻辑和非结构化关系组合任务上的实验表明，该架构具有很强的分布外泛化能力。这些任务之间的对比揭示了感知接口如何塑造共享推理核心的泛化特征。通过“思考得更深入，而非更久”，模型无需占用上下文窗口即可实现可变计算深度，从而为新一代语言模型提供基础构件。

我们承认存在若干局限性。首先，我们使用的模型规模相对较小（<1M 个参数）。其次，感知接口是手动设计的。第三，我们未能为泛化界提供形式化的理论保证；我们的证据来自实证结果。我们认为，与预训练大语言模型集成是未来工作的首要方向。

生成式人工智能使用披露

作者使用 Gemini 和 Claude 改进语言表达和可读性，并使用 Claude Code 协助编写代码和开展实验。作者根据需要审阅和编辑了内容及代码，并对内容承担全部责任。

致谢与生成式人工智能使用披露

我们感谢台湾国家科学及技术委员会通过编号为 113-2221- E-008-100-MY3 的资助项目提供支持。

参考文献

Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J. 和 Kaiser, L.（2019年）。通用 Transformer。发表于国际学习表征会议 *(ICLR)*。

Feng, G., Zhang, B., Gu, Y., Ye, H., He, D., and Wang, L.（2023年）。旨在揭示思维链背后奥秘：理论视角。载于神经信息处理系统进展（*NeurIPS*）。

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E.（2017）。用于量子化学的神经消息传递。载于国际机器学习会议（*ICML*），第1263–1272页。PMLR。

Goyal, S., Ji, Z., Rawat, A. S., Menon, A. K., Kumar, S., and Nagarajan, V.（2024年）。开口之前先思考：使用暂停令牌训练语言模型。载于国际学习表征会议（*ICLR*）。

Graves, A.（2016）。循环神经网络的自适应计算时间。*arXiv*预印本 *arXiv:1603.08983*。

Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y.（2024年）。训练大语言模型在连续潜在空间中进行推理。*arXiv* 预印本 *arXiv:2412.06769*。

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y.（2022年）。大语言模型是零样本推理器。在神经信息处理系统进展（*NeurIPS*），第 22199–22213 页。

Merrill, W. and Sabharwal, A.（2024年）。具有思维链的 Transformer 的表达能力。载于国际学习表征会议（*ICLR*）。

Snell, C. V., Lee, J., Xu, K., and Kumar, A.（2025年）。最优扩展大语言模型的测试时计算，比扩展模型参数更有效。载于国际学习表征会议（*ICLR*）。

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y.（2024）。Roformer：采用旋转位置嵌入的增强型 Transformer。*Neurocomput.*，568(C)。

Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., and Jégou, H.（2021年）。通过图像 Transformer 深入探索。在*2021*年 *IEEE/CVF*国际计算机视觉会议（*ICCV*），第32–42页。

Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D.（2023年）。自一致性提升语言模型中的思维链推理。在国际学习表征会议（*ICLR*）。

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D.（2022年）。思维链提示能够激发大语言模型的推理能力。在第*36*届神经信息处理系统国际会议论文集（*NeurIPS*），美国纽约州雷德胡克。Curran Associates Inc。

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., and Liu, T.-Y.（2020年）。关于 Transformer 架构中的层归一化。载于第*37*届国际机器学习会议论文集（*ICML*）。JMLR.org。

Xu, K., Hu, W., Leskovec, J., 和 Jegelka, S.（2019年）。图神经网络有多强大？发表于国际学习表征会议（*ICLR*）。

A 推理块的细节 f_θ

推理块 f_θ 是一个 Pre-LayerNorm Transformer 编码器块（Xiong 等人，2020年）。在递归步骤 t，它接收经过深度条件化的输入 $\hat{}H^{(t)} = H^{(t-1)} + e_t$，其中 e_t 是附录 B 中定义的深度嵌入。该块依次应用两个子层：

子层 1：多头自注意力（MHSA）。

$$H' = \hat{H}^{(t)} + \gamma_{\text{attn}} \odot \text{MHSA}\Big(\text{LN}\Big(\hat{H}^{(t)}\Big),\, M\Big)\,, \tag{6}$$

其中，LN(·) 表示层归一化，M 表示任务特定注意力掩码（对于拓扑接口，它是邻接掩码；对于另外两个接口，则不使用掩码），$\gamma_{\text{attn}} \in \mathbb{R}^d$ 表示 LayerScale 向量（详见下文）。
注意力机制会计算 h 个并行注意力头。对于每个注意力头 i:

$$\text{head}_i = \text{softmax}\!\left(\frac{Q_i K_i^\top}{\sqrt{d_k}} + M\right) V_i, \qquad Q_i = \hat{H}^{(t)} W_i^Q,\; K_i = \hat{H}^{(t)} W_i^K,\; V_i = \hat{H}^{(t)} W_i^V, \tag{7}$$

其中 $d_k = d/h$。对于层次化和非结构化接口，在进行点积之前，会对查询和键投影应用旋转位置嵌入（RoPE）（Su 等，2024），使内积隐式编码相对位置（见第 3.2 节）。

子层 2：前馈网络（FFN）。

$$H'' = H' + \gamma_{\text{ffn}} \odot \text{FFN}(\text{LN}(H'))\,, \tag{8}$$

$$\text{FFN}(x) = W_2 \cdot \text{GELU}(W_1 x + b_1) + b_2, \tag{9}$$

其中 $W_1 \in \mathbb{R}^{d_{\text{ff}} \times d}$、$W_2 \in \mathbb{R}^{d \times d_{\text{ff}}}$ 和 d_{ff} 是前馈维度。
f_θ 的输出为 $\tilde{H}^{(t)} = H''$，随后通过恒等偏置门（第 3.1 节）与 $H^{(t-1)}$ 结合，生成 $H^{(t)}$。

LayerScale。对于分层和非结构化感知接口，我们在每个子层之后应用 LayerScale（Touvron 等人，2021年）。逐通道缩放向量 γ_{attn}，$\gamma_{\text{ffn}} \in \mathbb{R}^d$ 初始化为 10^{-4}，并与所有其他参数共同学习。对于拓扑接口，硬邻接掩码已对计算施加强约束，因此不使用 LayerScale。对于拓扑接口，使用标准的 PyTorch Transformer 编码器层；上述集成了旋转位置编码的自定义注意力仅适用于分层和非结构化接口。

超参数。表 2 总结了三个实验中使用的架构超参数。

表 2：各实验的架构超参数。

	图	嵌套表达式	系列
d	128	256	256
h (注意力头)	4	8	8
d_{ff}	256	1024	1024
RoPE	×	✓	✓
LayerScale	×	✓	✓
门控偏置 b_z	-2.0	-2.0	-2.0
训练步数范围 T	$[5, 8]$	$[4, 16]$	$[1, 12]$
$T_{\max}$	20	28	20

B 深度嵌入

为启用共享-权重推理块 f_θ 为追踪其在循环回路中的位置，我们注入一个学习得到的深度 h 嵌入 $e_t \in \mathbb{R}^d$ 位置注意力计算之前的每一步 $t \in \{1, \ldots, T\}$：

$$\hat{H}^{(t)} = H^{(t-1)} + \mathbf{1}_L \cdot e_t^\top, \tag{10}$$

其中 $\mathbf{1}_L \in \mathbb{R}^L$ 是一个全 1 向量，因此 e_t 会被相同地广播到所有 L 序列位置。嵌入$\{e_t\}_{t=1}^{T_{\max}}$构成一个标准的可学习查找表 $E \in \mathbb{R}^{T_{\max} \times d}$，其中 $T_{\max}$ 的设置覆盖训练和测试时使用的最大步数。

如果没有深度嵌入，共享模块 f_θ 在每一步接收的输入都相同（不断演变的隐藏状态除外），因此模型难以区分早期的探索步骤和后期的细化步骤。通过让每次迭代都以e_t为条件，模型便可以自由学习步骤专用计算策略——例如，利用早期步骤收集局部信息，再利用后续步骤整合长程证据。

与位置编码的关系。 深度嵌入与序列内部使用的位置编码（旋转位置编码或不使用位置编码）彼此正交。前者编码模型在时间 （循环）维度上的当前位置；后者编码空间 （序列）维度上的位置。两者可以同时应用，互不冲突。