arXiv:2611.31990v1 [cs.AI] 24 Nov 2026

边缘规则能看到多少依赖关系？扩散语言模型并行解码中的总相关

Dovud Asadov*
新乌兹别克斯坦大学
d.asadov@newuu.uz

Rustambek Urokov*新乌兹别克斯坦大学
r.urokov@newuu.uz

Qobiljon Toshnazarov新乌兹别克斯坦大学
q.toshnazarov@newuu.uz

Abduaziz Kayumov
新乌兹 beki 别克斯坦大学 iversi
ty a.kayumov@newuu.uz

Mardon Hazratov
新乌兹别克斯坦大学
m.hazratov@newuu.uz

Husanboy Mansuraliyev
New Uz beki stan Un iversi
谢谢 h.mansuraliyev@newuu.uz

摘要

掩码扩散语言模型之所以速度快，是因为每一步会提交多个词元。每个词元都从各自的边缘分布中采样，因此词元之间的依赖关系会丢失，而这部分损失的大小就是已提交集合的条件总相关 C。大多数解码器依据逐位置边缘分布选择集合，这可以给 C 设定上界，却无法确定其实际值。我们探究它们能多准确地预测这一数值。我们针对两个系列中的四个掩码扩散语言模型，对固定候选集的 C 进行估计。在四个模型中的三个上，表现最佳的仅依赖边缘分布的预测器误差约为 1 比特（0.74 至 1.17），比仅知道上界的基线低 1.8 至 3.7 倍；而其大部分准确性其实来自上界本身。在一个子集上，我们的估计对集合的排序与并行提交的精确 KL 代价相近，但在散文文本以及文本其余部分仍处于掩码状态时，其中位数比精确代价低 0.4 至 1.1 比特。Dream-instruct 是个例外，主要体现在散文文本上，其条件分布的自洽性最低。在六种常见的边缘规则中，在所有 24 组配对比较里，汇总整个集合的结果都优于依据最弱位置进行门控。测量需要经过验证的读出方式：Dream 和 LLaDA 将某个位置的预测放在不同的 logit 索引处，而使用错误索引会使不一致性诊断指标膨胀 14 至 $41\times$。我们给出了三项检查，用于验证所采用的约定。我们的结果关注的是对固定集合的 C 进行预测，而非依赖感知解码器能带来什么收益。

1 引言

掩码扩散语言模型（dLLM） [1, 2]从完全掩码的序列开始，并在少量去噪步骤中逐步填充序列。它的速度得益于在同一步骤中提交多个位置。因此，决定哪些位置应一起提交，便成为解码器作出的主要决策。

同时提交是有代价的。假设两个掩码位置应当组成“纽约”或“洛杉矶”，两者的概率各为二分之一。那么，模型单独对每个位置进行预测时，结果就像抛一枚公平硬币：第一个位置为“纽约”中的“纽”或“洛杉矶”中的“洛”，第二个位置为“约”或“杉矶”。并行
提交会独立地对这两个位置进行采样，因此有一半的时间会写成“纽杉矶”或“洛约”。两个边缘分布都没有错。丢失的是两个位置之间的依赖关系。
这种损失有一个精确的衡量方式。在一步中确定集合 S，会将 S上的联合分布替换为其边缘分布的乘积，而这种替换的 Kullback–Leibler 代价就是条件分布

*贡献相同。

第40届神经信息处理系统会议（NeurIPS 2026年）。专题研讨会：扩散语言模型：基础、效率与推理（DiffuLM）。

总相关

$$\mathrm{C}(S \mid c) \;=\; \sum_{i \in S} H(Y_i \mid c) \;-\; H(Y_S \mid c) \;\geq\; 0, \tag{1}$$

其中 c 是已解码上下文。C 是在给定 c 的条件下， S 各位置共享的信息量。当且仅当它们条件独立时，该值恰好为零；在上面的示例中为 1 比特，并且它为任何因子化提交 [9, 16] 的 KL散度提供下界。

大多数解码器不会测量 C。它们会根据每个位置的边缘分布分别计算规则，然后选择 S：词元置信度 [4, 5]、前二差值、预测熵或跨步稳定性[6, 17]。在该示例中，边缘规则看到的是两枚公平硬币；如果各位置相互独立且 C 为零，它看到的仍然是同样的两枚公平硬币。因此，边缘规则无法确定C。它只能为 C 提供上界（§2）。这一点已为人所知：文献 [18] 对此进行了陈述，并据此推动了若干解码器的发展 [20, 25]。

But 解码器从未遇到最坏情况。它只会遇到训练模型产生的联合分布。Th e ope我们提出的问题，也是本文要回答的问题是：

> 对于真实的扩散大语言模型，从边缘分布中可以恢复多少 C 的变化？读取这些边缘分布的规则又会遗漏多少？

要回答这个问题，需要将文献中尚未结合的两项工作整合起来：在真实检查点上按候选集测量 C，以及对该检查点进行可信的读出。

读取边缘分布并非易事。Dream [2] 保留了其自回归骨干网络的偏移，因此索引 i 处的输出预测的是 $i{+}1$处的词元；而 LLaDA [1]则在原位预测每个被掩码的词元。因此，两种使用最广泛的开放检查点需要采用相反的约定，却没有任何类型检查能发现这种差异。若流水线在错误的索引处读取 Dream，仍会返回数值，而且看起来也合情合理（§3.2）。

贡献。

- **边缘分布对 C 的预测误差约为 1 比特**。我们在来自两个系列的四个检查点上，按候选集估计 C，并拟合了能力最强的纯边缘分布预测器。在四个检查点中的三个上，其误差为 0.74 至 1.17 比特，比只知道上界的基线低 1.8 至 3.7 倍。大部分准确性来自上界本身，因为真实候选集只使用了约一半的上界（§4）。
- **规则如何汇聚信息，比它读取哪种统计量更为重要**。我们通过配对检验，将六种已部署的规则与测得的 C 进行比较。在全部 24 次比较中，对集合取均值都优于根据其最弱位置进行门控；而前二差值在合成任务上的优势并未延续到这里（§5）。
- **当模型不一致时，该估计意味着什么**。扩散大语言模型的条件分布不一定与某个单一联合分布一致。对于这种情况，我们给出了一个关于该估计所度量内容的精确恒等式，并将其与并行提交的 KL 成本进行核对（§6）。
- **一种读出协议**。三项检查测量检查点的 logit位置约定，而不是直接假定该约定；一项消融实验展示了错误约定会如何影响上述统计量（§3.2）。

作为参考点，我们还精确给出了最坏情况（§2）；这是一个简短的刻画，建立在 [16, 定理 1] 的因子分解界和 [9,§4] 的分析之上。

2 设置：总相关以及边缘分布所能证明的内容

并行提交。 在每一步中，掩码扩散大语言模型根据当前序列 x_t 预测每个掩码位置，且各预测彼此独立： $p_\theta(x_{t-1} \mid x_t) = \prod_i p_\theta(x^i_{t-1} \mid x_t)$[12, 13]。解码器选择一组 S 掩码位置，并从各自的边缘分布中分别采样。借助$|S| = 1$，这就是任意顺序的自回归，不会有任何信息损失。借助 $|S| > 1$，提交会损失公式（1）中的 C$(S \mid c)$。我们将位置 i 处的词元记为 Y_i，将模型能够看到的全部内容记为 $k = |S|$，并将 c 用于表示一切模型可以看到的信息。本节假设模型的条件分布 合成为一个联合分布：存在一个关于 Y_S 的单一联合分布，而这些条件分布就是它的条件分布。§6 讨论它们不构成联合分布的情况。

上界 W。边缘分布确定了公式（1）的第一项，即边缘熵之和 B。由于联合熵非负，

$$0 \;\le\; \mathrm{C}(S \mid c) \;\le\; \sum_{i \in S} H(Y_i \mid c) \;=:\; B(S \mid c). \tag{2}$$

更紧的上界利用了这样一个事实：联合分布的不确定性绝不会低于其最不确定坐标的不确定性，$H(Y_S \mid c) \;\ge \max_i H(Y_i \mid c)$。令 $\mathrm{C}_{\max}$ 为所有具有给定边缘分布的联合分布中最大的 C。于是

$$\mathrm{C} \;\le\; \mathrm{C}_{\max} \;\le\; W \;:=\; B - \max_i H(Y_i \mid c) \;\le\; (1 - 1/k)\, B. \tag{3}$$

我们称之为 W 该上界。它是边缘熵所允许的最大依赖关系，也是边缘规则所能认证的全部内容。特别是，当且仅当 S 至多有一个位置存在不确定性时，边缘分布才能认证 $\mathrm{C} = 0$。

在 $[0, \mathrm{C}_{\max}]$ 范围内，边缘分布无法确定任何东西：每个值都能由某个联合分布实现；对于相同的边缘分布，$\mathrm{C}_{\max} = W$（附录 C）。

相同的边缘分布，不同的 C。 一个清晰的例子是 n 枚公平硬币与 n 枚恰好以 $n/2$ 个正面朝上的条件为条件的硬币之间的对比。两种情况下每枚硬币都是公平的，因此边缘分布完全相同，且 $B = n$比特。但前者的 C 为 $= 0$，后者的 C 为 $= n - \log_2 \binom{n}{n/2}$，在 n=4时即为 1。415 比特。任何边缘规则都必须给这两者相同的评分 [22]。

六种边缘规则。我们为解码器使用的六种规则评分。每种规则读取每个位置上的一个统计量，并将其汇总到候选集上，得到一个安全分数，分数越高表示越适合提交。这些统计量包括置信度（最大概率）、前二差值（最大概率减去第二大概率）以及负熵。每种统计量有两种汇总方式：取候选集中最弱位置的值（最小置信度、最小概率差、负最大熵），或在整个候选集上汇总（平均置信度、平均概率差、负熵和，即 $-B$）。我们将第一类称为最小值规则，第二类称为汇总规则。

两种机制。候选集始终是由 k 个相邻位置组成的块，除非另有说明，为 k=4。我们在两种机制下测量候选集。在后缀机制中，从该块到文本末尾的所有内容都会被遮蔽，这正是继续生成提示词的解码器所看到的情况。在填充机制中，只有该块被遮蔽，且两侧都有上下文。后缀块的不确定性要高得多：其中位数 B 为 11.7 至 17.3 比特，而填充机制下为 3.5 至 4.3 比特（表 11）。

3 在真实模型上进行测量

3.1 估计量

公式（1）中的边缘熵来自一次前向传播。联合熵 $H(Y_S \mid c)$ 则不是，因此我们利用链式法则构造它；该法则对位置的任意顺序 π 都成立：$H(Y_S \mid c) = \sum_t H(Y_{\pi(t)} \mid Y_{\pi(<t)},\ c)$。在实践中，我们随机抽取一个顺序 π，并逐个位置揭示该块。在步骤 t，给定上下文和截至目前已揭示的词元，我们读取模型对位置 $\pi(t)$ 的分布，记录其精确的熵，从中采样一个词元，并将该词元写入。记录的熵相加后得到联合熵的一个估计量，

$$\hat{H}_\pi(Y_S \mid c) \;=\; \sum_{t=1}^{|S|} H_\theta\big(Y_{\pi(t)} \mid y_{\pi(<t)}, c\big), \qquad y_{\pi(t)} \sim p_\theta\big(\,\cdot \mid y_{\pi(<t)}, c\big), \tag{4}$$

然后我们对 M 个相互独立的顺序取平均：

$$\widehat{\mathrm{C}}_M(S \mid c) \;=\; \underbrace{\sum_{i \in S} H_\theta(Y_i \mid c)}_{B(S \mid c)} \;-\; \frac{1}{M} \sum_{m=1}^{M} \hat{H}_{\pi_m}(Y_S \mid c). \tag{5}$$

如果这些条件分布复合成一个联合分布，那么在任意单一顺序下，$\hat{}\, H_\pi$ 都是 $H(Y_S \mid c)$的无偏估计量，而 $\widehat{\ }$取均值只会降低方差。如果它们无法复合，C 仍然有明确定义，但其含义会发生变化。我们将这一情况推迟到第 6 节讨论。

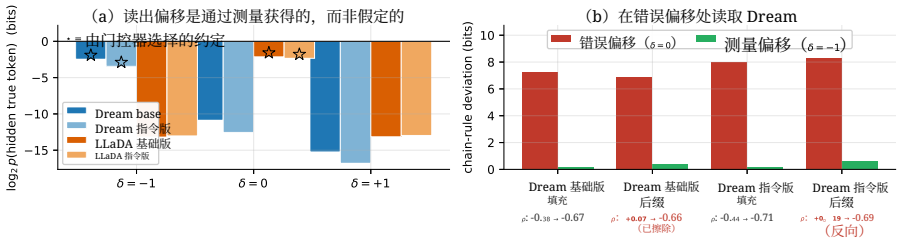

图 1：**(a)** 偏移门：在索引 $p + \delta$ 处读取隐藏的真实词元的对数概率。星号标记门控器选择的约定：Dream 和 LLaDA 选择相反的约定。**(b)** 偏移消融实验，仅改变读出。在两个 Dream 检查点上，链式法则偏差从 14 膨胀至 $41\times$。每组数据下方的 ρ 表示置信度与解码误差在错误偏移和测量偏移处的相关性。

我们在全文中使用 $M{=}64$。对于每个块，我们报告标准误 $\mathrm{sd}_m(\hat{H}_{\pi_m})/\sqrt{M}$，并在 C 参与方差分解的任何地方对其进行$\hat{}$校正。该估计量仅通过模型的条件分布与模型交互，因此相同的代码既可运行于检查点，也可运行于精确预言机。在 C 具有闭式形式的合成任务上（附录 C），它恢复出的解析值与真实值相差不超过 0.0087 比特。

3.2 验证读出

我们报告的一切都建立在一个约定之上：给定一个掩码序列和一个位置 p，返回模型分布中对应于该位置 p词元的分布。在生成任何数值之前，三个门控器会在每个检查点上检验这一约定。

门1：哪个逻辑值索引承载位置 p? 我们对一个块进行掩码，并读取索引 $p + \delta$处模型为原本确实位于该处的词元赋予的对数概率，以评估 $\delta \in \{-1, 0, +1\}$。真实词元是隐藏的，因此模型无法通过复制取胜，只有正确的 δ 才能获得较高分数。对于 $k{=}1$ 和 $k{=}4$，获胜者必须相同，并且至少领先次优项一个比特。在未被告知的情况下，该门控器通过6.9至11.0比特的差值恢复了每个模型文档所记录的约定（图1a、表3）。

门控2：前向传播是否可复现? 两次完全相同的调用必须返回完全相同的逻辑值。此外，估计量结合的两种批次形状必须在每个最大值索引上保持一致，并且 B的均方根误差不得超过0.074比特，这只是我们报告的最小误差的十分之一。四个检查点在第二块GPU上均通过了该测试，而主运行采用了较弱的形式（附录B）。我们还会在每个块上检查一个结构不变量：按顺序解码的第一个位置必须与并行提交来自相同的分布，因此其最大值索引绝不能与并行提交不一致。

门槛3：它会写入吗? 我们在校准后的偏移量处，对完全掩码的序列进行贪心解码，并读取文本。在错误的偏移量处，我们测试的三个检查点都出现了解码坍缩：一个词元填满了 32 个散文位置中的 21 到 32 个，而在校准后的偏移量处，每次最多只填入 3 个（附录 F）。

错误的偏移量会造成什么影响。为了了解门控器能防止什么，我们将 Dream 的偏移量从测得的 -1 改为朴素设定的 0，并重新运行。语料库、块、揭示顺序和代码路径均保持不变。三个量发生了变化（图 1b）。

不一致性诊断指标被放大。链式法则偏差 [29] 固定一个块的赋值，并测量其链式法则对数概率随揭示顺序变化的幅度。当条件分布复合为一个联合分布时，该值恰好为零。在错误的偏移量处，该值为 6.9 到 8.3 比特，而在正确的偏移量处为 0.18 到 0.59，比值为 14 到 41。读者会因此得出结论：Dream 的条件分布与任何联合分布都相去甚远。
一个核心相关性消失了。一个常见的结果变量是解码误差：并行提交与顺序参考解码不一致的频率。在后缀机制中，使用正确偏移量时，它与置信度的秩相关为 -0.66 和 -0.69。使用错误偏移量时，在 Dream-base 上为 $+0.07$ $[-0.06, 0.19]$，在 Dream-instruct 上为 $+0.19$ $[0.07, 0.31]$。错误偏移量会抹去一个检查点上的相关性，并在另一个检查点上将其反转。读者会得出结论：置信度与解码误差毫无关系。

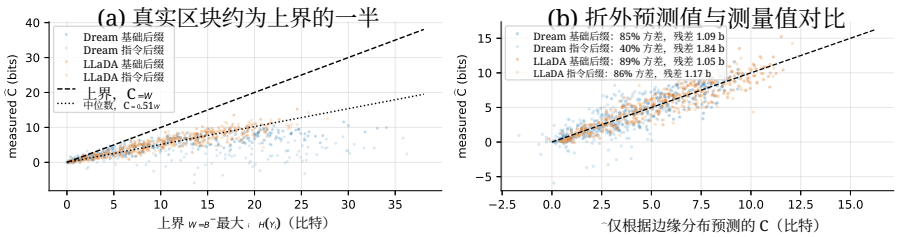

图 2：**仅限后缀机制**。填充设置见表 1。**(a)** 测得的 $\hat{C}$ 与 $B - H(Y)$ 的上界 $\max W = B - H(Y)$ 对比。虚线表示 $C = W$，点线表示 Dream-instruct 之外部分的中位数比例。**(b)**仅基于边缘分布的最佳预测与测得的 $\hat{C}$ 对比，采用折外评估。

$\hat{C}$变得不可能。在填充情形下，使用错误偏移量时，C 的中位数为 -4.3 和 -4.8 比特，而使用正确偏移量时为 $+0.78$ 和 $+0.05$。没有任何联合分布的 C 为 < 0，因此，强负估计不可能是一个发现。在这里，估计量揭示了另外两个统计量所掩盖的误差。

4 边缘分布能解释多少C

协议。 我们将估计量 (5) 应用于 Dream-Coder 7B [3]和 LLaDA 8B [1], 的基础版和指令微调检查点，每个检查点和机制包含 247 至 516 个块（附录 A）。在八种$\hat{C}$设置中，我们接着考察能否仅根据块的边缘分布预测 C。我们针对由各位置的前8个概率和熵构成的五组特征，遍历k-NN 和岭回归，共 40 种候选方案，并保留表现最佳者。所有得分均采用折外评估，且同一文档的所有块均归入同一折。

两个定义有助于理解这些数字。预测器的残差 是其折外均方根误差（以比特计），并已扣除估计量自身的方差（0.7% 至 7.5% 的 Var[C]）。参考 并不是一个拟合出的模型。它是一个参考基准，已知每个块的边缘熵，因此其上界为 W，但对耦合一无所知：它将 C 视为在 $[0, W]$ 上均匀分布，并猜测 $W/2$。上界并非误差，而参考基准将其转化为可与残差比较的误差。

Dream-instruct 的表现与另外三个检查点不同。以下范围不包括它，$\hat{C}$我们将在本节末尾再讨论它。在少数块中，C 落在 $[0, W]$之外。我们保留这些块，并在 §6 中讨论。

$\hat{C}$边缘分布可以将C 预测到约一个比特的精度。 仅使用边缘分布的最佳预测器，残差为 0.74 至 1.17 比特（表1、图2b）。作为参考，所有八种设置下的中位数界 W 为 1.3 至 10.8 比特。残差比参考基准低 1.8 至 3.7 倍。若参考基准已知 $C_{\max}$ 而非W，范围会更窄，因此这些比值算是偏宽松的。

其中大部分来自上界本身。图 2a 展示了真实区块在 $[0, W]$ 中的位置。它们并未遍布整个范围。 W 大于一比特的区块，其 W 的中位数介于 0.44 至 0.56 之间。因此，单参数预测器 $f \cdot W$ 已经表现出色。在 LLaDA-base 上，它与最佳预测器的表现相当（分别为 0.74 和 1.07 比特，对比 0.74 和 1.05 比特）。完整的边际剖面在 Dream-base 的后缀机制中最有帮助，可将残差从 1.81 比特降至 1.09 比特。六条规则在 W:$\hat{}$ 之外几乎没有额外贡献，因为 W 六条规则中任意一条与 C 的秩相关都介于 -0.28 和 $+0.23$ 之间（表 10）。

Dream-instruct 在散文领域是个例外。 八种设置中有六种结果高度一致。两种 Dream-instruct 设置则不然（1.62 和 1.84 比特），而超出部分集中在散文上。在同一个合并预测器下，它在散文上的残差分别为 2.23 和 2.58 比特，高于其他所有检查点在散文上的残差；在其训练领域——代码上，残差则分别为 0.92 和 1.28 比特（表 9）。散文也是其$\hat{}$链式法则偏差最大的领域，而且在这里，并行提交的 KL 成本比 C 高出数个比特（§6）。因此，对于 Dream-instruct 的散文，C 并不能衡量提交的成本，拟合效果不佳也在预料之中。这并非普遍的指令微调效应，因为 LLaDA-instruct 与基础检查点的表现一致。

		测量值 $\widehat{\mathrm{C}}$		解释方差百分比		残差（比特）			上界
模型／机制	n	mean	s.d.	B	最佳	最佳	$f \cdot W$	参考	W
Dream 基础模型／填充	494	1.70	2.25	65	75	1.13	1.41	2.04	2.0
Dream 基础模型／后缀	247	4.90	2.79	81	85	1.09	1.81	4.06	10.8
Dream 指令版 / 填充	494	0.54	1.89	3	26	1.62	2.03	2.53	1.3
Dream 指令版 / 后缀	247	2.54	2.37	15	40	1.84	3.59	4.01	6.7
LLaDA 基础版 / 填充	516	1.78	2.38	86	90	0.74	0.74	1.52	1.7
LLaDA 基础版 / 后缀	258	5。47	3。24	88	89	1.05	1.07	3.48	10.3
LLaDA 指令版 / 填充	516	1.67	2.28	78	85	0.87	0.95	1.60	1.6
LLaDA 指令版 / 后缀	258	5.03	3.13	81	86	1.17	1.34	3.44	9.5

表1：**边缘分布对 $\widehat{\mathrm{C}}$ 的预测效果**。 测得的 $\widehat{\mathrm{C}}$：以比特为单位的均值和经噪声校正的标准差。解释方差百分比： R^2 仅凭自身以及40个预测变量中最佳者（§4）的折外 B。残差：该预测器、单参数 $f \cdot W$以及一个了解边缘熵的参考方法的经噪声校正折外误差，其中 C 在 $[0, W]$ 上均匀分布。 W：中位数界。

这一结果在 k=8 时仍成立。 在两个基础检查点上使用八位置块重复测量，得到的残差为 1.60 至 2.59 比特。这比参考值低 3.4 至 5.0 倍，而在 k=4 时则低 1.8 至 3.7 倍（表7）。

5 比较六种边缘规则

现在，我们根据测得的 C，对§2中的六种规则进行评分。评分采用斯皮尔曼秩相关系数 ρ，衡量某条规则的安全分数与各块中 C 之间的相关性。好的规则会在 C 较低时将块判定为安全，因此忠实规则的 ρ 应为负值（表2）。

每条规则都能追踪 C，主要是通过确定性。 在 8 种设置中，6 条规则产生的全部 48 个相关性均为负值，且经文档聚类自助法检验后均不包含零。这一点在预料之中。由于 C $\leq W$，任何能追踪确定性的规则也能追踪上界；而对 W 进行条件化会消除大部分相关性（§4）。真正有参考价值的是比较各规则之间的表现。

规则如何汇总，比它读取什么统计量更重要。 在 8 种设置中的每一种里，汇总规则都胜过读取相同统计量的最小值规则：平均置信度胜过最小置信度，平均概率差胜过最小概率差，熵和胜过最大熵。在 $|\rho|$ 中，提升幅度为 0。066 至 0。235，且配对自助法在全部 24 项比较中都排除了零（表 10）。像阈值规则那样，以置信度最低的位置作为门槛 [4, 5], ，是较弱的汇总方式。除 Dream-instruct 外，汇总规则达到 $\rho = -0.83$ 至 -0.94 。

来自合成任务的结论无法迁移。 在转向真实检查点之前，我们在 ParallelBench 的合成列表操作任务（附录 C）中，从 355 个候选集上对同样的六条规则进行了评分，其中 C 是完全已知的。在这些任务中，前二差值的表现远优于置信度：在配对自助法下， 0.394 $[0.296, 0.494]$ 中的 ρ 个案例中前二差值排名 C 的效果更好（表4）。在真实检查点上，这一优势不复存在。在 min 版本中，置信度在八种设置中的七种表现更好，其中六种具有显著性，优势最高达 0.189 $[0.121, 0.264]$。在汇总规则中，两个统计量都没有占据主导地位。平均置信度在五种设置中显著更好，平均概率差在两个 Dream- 指令微调设置中更好，另有一种设置打成平手。同样，熵和在三种设置中显著优于平均置信度，在两种设置中则显著逊于平均置信度（表 10）。

这说明了什么，又没有说明什么。 这些结论针对的是对 C 的预测。聚合规则是否能带来更好的解码器，是另一个问题，需要通过端到端比较来判断。我们也不使用通常的结果变量——解码误差——作为目标。它只是 C 的一个较弱替代指标，相关性仅为 $+0.23$ 到 $+0.64$；在后缀机制中，它还取决于参考解码的揭示顺序（附录 E）。

模型／机制	置信度（最小值）	置信度（均值）	边际值（最小值）	边际值（均值）	负熵（最大值）	负熵（总和）
Dream 基础模型／填充	-0.72^{*}	-0.83^{*}	-0.69^{*}	-0.83^{*}	-0.77^{*}	-0.83^{*}
Dream 基础模型／后缀	-0.79^{*}	-0.89^{*}	-0.65^{*}	-0.86^{*}	-0.78^{*}	-0.88^{*}
Dream 指令版／填充	-0.23^{*}	-0.33^{*}	-0.28^{*}	-0.37^{*}	-0.17^{*}	-0.23^{*}
Dream 指令版／后缀	-0.46^{*}	-0.54^{*}	-0.45^{*}	-0.58^{*}	-0.36^{*}	-0.45^{*}
LLaDA 基础版 / 填充	-0.73^{*}	-0.86^{*}	-0.69^{*}	-0.85^{*}	-0.79^{*}	-0.89^{*}
LLaDA 基础版 / 后缀	-0.78^{*}	-0.89^{*}	-0.60^{*}	-0.83^{*}	-0.84^{*}	-0.94^{*}
LLaDA 指令版 / 填充	-0.75^{*}	-0.85^{*}	-0.71^{*}	-0.84^{*}	-0.80^{*}	-0.87^{*}
LLaDA 指令版 / 后缀	-0.76^{*}	-0.89^{*}	-0.61^{*}	-0.85^{*}	-0.81^{*}	-0.90^{*}

表2：**每条边缘规则相对于测量值** C。斯皮尔曼 ρ 各规则安全分数与 C 之间的相关系数，因此忠实规则的相关系数为负。*：文档聚类自助法区间不包含零。规则之间的配对比较见表 10。

6 C 所测量和未测量的内容

公式（1）至（3），以及将 C 解读为并行提交的 KL 成本，都假设条件分布复合后构成一个联合分布。扩散大语言模型的任意顺序条件分布未必 [30]。估计量不需要这一假设：C 始终 B 等于取平均的链式法则熵的负值。

只有在一致模型中，这三个量才相等。令 q_π 为模型在按顺序 π 揭示 S 时生成的 S 联合分布，令 $\bar{q} = \mathbb{E}_\pi q_\pi$ 为顺序服从均匀分布的采样器所产生的联合分布，令 p_i 为并行边缘分布。有三个数值值得关注。第一个是估计量返回的结果，$\mathbb{E}\,\mathrm{C}_M = B - \mathbb{E}_\pi H(q_\pi)$，因为无论条件分布如何，每个 H_π 都是 $H(q_\pi)$ 的无偏估计量。第二个是并行提交相对于固定顺序采样器的代价，对各种顺序取平均后为 $\mathbb{E}_\pi \,\mathrm{KL}(q_\pi \,\|\, \prod_i p_i)$。第三个是它相对于随机顺序采样器的代价，即 $\mathrm{KL}(\bar{q} \,\|\, \prod_i p_i)$。后两个量由一个精确恒等式联系起来，

$$\mathbb{E}_\pi \,\mathrm{KL}(q_\pi \,\|\, \textstyle\prod_i p_i) \;=\; \mathrm{KL}(\bar{q} \,\|\, \prod_i p_i) \;+\; I(\pi; Y_S), \tag{6}$$

其中 $I(\pi; Y_S) \le \log_2 k!$ 表示揭示顺序所携带的关于结果的信息量。当条件分布复合时，这三个数值都等于 C，且 $I(\pi; Y_S) = 0$。若它们不复合，则分为两种情况。

当每个揭示顺序都保留并行边缘分布时，$\mathbb{E}\mathrm{C}_M$ 就等于每种顺序的 KL 成本，即公式 (6) 的左侧。它仍然是一种成本，并且比相对于随机顺序采样器的成本高出顺序信息。两个公平比特即可说明这一差距。假设一种顺序使第二个比特以 0.9 的概率与第一个比特一致，而另一种顺序使其以 0.9 的概率不一致。每种顺序的成本都是 0.531 比特，而两者的混合分布是均匀分布，成本为零。

当揭示顺序不保留并行边缘分布时，C 根本不是一种散度，而且可能为负。

C 距离提交成本有多远。我们在块的一个子集上直接测量了公式 (6) 的两个项（附录 F，表 12）。在随机块上，LLaDA-base 和 Dream-base 的散文以及它们的后缀机制中，每种顺序的 KL 成本的中位数比 C 高 0.4 到 1.1 比特。C 对块的排序仍然与 KL 成本一致（斯皮尔曼相关系数为 0.95 和 0.90，汇总各项设置）。因此，在这些检查点上，C 按提交成本对块进行排序，但低估了其大小；第 4 节的残差是相对于 C 的残差，而不是相对于 KL 成本的残差。在 Dream-instruct 散文中，这一差距为 4.4 和 5.6 比特，此时 C 并不能衡量该成本。

不可能的取值。不存在联合分布能使 $\mathrm{C} < 0$ 或 $\mathrm{C} > W$，因此 C 离开 $[0, W]$ 的块是出现问题的直接迹象。在超过两个标准误的范围内，0% 至 25% 的块低于零，另有 0.4% 至 9.5% 高于 W（表 9）。其中大多数偏差都很小。在 Dream-instruct 之外，这些块中有 79% 的偏差在 0.2 比特以内，其中 86% 出现在代码上。在这些情况下，标准误过小，因而不可信（附录 F）。超过 0.2 比特的实质性案例，在 Dream-instruct 之外占块的 0.4% 至 5.9%，在其中则占 8.1% 和 14.6%。我们报告这些比例，而不是进行截断。排除低于零的块后，Dream-instruct 的残差从 1.62/1.84 比特变为 1.19/1.57 比特，而其他检查点基本保持不变。

7 相关工作

并行提交的代价。 并行提交的代价已被分析为 C [9, 16], 中的 KL下限，即因子化步骤丢弃的依赖关系 [10], ，也通过推理调度方案 [14], 中的步骤数、收敛速率 [15], 、生成顺序 [23], 以及高斯场模型 [11] 进行了分析 [21]。关于 C 的数据集级估计可追溯至非自回归翻译 [16]。ParallelBench [9] 提供了带有解析 C 的合成任务供我们复用，并发现置信度规则在此处不如随机选择，即 C > 0。我们在真实检查点上估计每个候选集的 C，并考察边缘分布能够恢复其中多少。

依赖感知解码器。 许多近期的解码器都会考虑提交之间的依赖关系。它们通过注意力 [7, 8, 26, 25],依赖图 [24],学习到的互信息[19] 或学习到的预测器 [20], 来估计这种依赖关系 [27], ，也可能通过平均场步骤 [33], 条件独立性检验 [34], 辅助自回归模型 [28] 学习策略 [18]或协调遍来协调提交。我们衡量的是边缘分布对固定集合依赖关系的预测效果，而非此类解码器能带来什么收益。

不一致的条件分布。 人们早已知道，掩码语言模型的条件分布未必能组合成任何联合分布 [30], ，这会影响采样 [31] 和去噪顺序 [32]。我们的诊断指标——链式法则偏差——源自 [29]。

8 局限性

$\hat{}$C 并不总是总相关。 若不存在单一的联合分布，就没有唯一的量配得上$\hat{}$总相关这一名称；C 可能低估一次提交的 KL 成本，或无法对其进行衡量（§6）。

残差的上界。 我们的预测器读取每个位置的前8个概率和熵。这涵盖了我们评分的所有单步规则（无论采用何种聚合方式），但不涵盖完整的边缘分布，也不涵盖 [6, 17], 的跨步稳定性规则，后者会读取多个解码步。更好的预测器只会降低残差，因此我们将其表述为上界，而非不可能达到的值。在我们报告的相同折外评分上从40个候选项中选择获胜者，会使残差产生向下偏差；但在内层折上进行选择、在外层折上进行评分，最多只会使其变化0.035比特（附录 D）。在 LLaDA-base 上将 M 加倍至256，最多只会使其变化0.034比特（附录 D）。

范围。 k=4 个块，并以 k=8 作为检查，采用贪心参考解码；每篇文档使用一个128词元窗口，涵盖英语散文和 Python（附录 A）。来自两个系列的四个检查点均在不使用聊天格式化的原始文本上运行，这使输入具有可比性，但并非指令模型的实际部署方式。这些数字描述的是固定连续块上的一个解码步，而不是端到端生成质量。部署的解码器会选择自己的集合，可能提交更大的集合，也可能进行采样而不是取最大值索引。在一种 GPU 类型上测得的数字，在另一种 GPU 上未必能逐比特复现（附录 F）。

9 结论

读取逐位置边缘分布的规则无法确定并行提交中的依赖关系，只能用 W 给出其上界。在真实检查点上，这个上界包含了大部分信息。在来自两个系列的四个检查点上$\hat{}$，除 Dream-instruct 外，边缘分布对估计值 C 的预测残差为 0.74 至 1.17 比特；与忽略耦合的参考值相比，该预测值低 1.8 至 3.7 倍，这主要是因为真实块稳定地使用了其上界的一定比例。规则如何对集合进行汇总，比它读取哪种$\hat{}$统计量更重要。需要注意两点。C 对块的排序与并行提交的 KL 成本相符，但在散文和后缀机制中低于该成本；而在 Dream-instruct 散文中，它并不能衡量该成本。此外，我们评估的是固定集合，因此依赖感知解码器能带来哪些收益，还需要通过匹配计算量的解码研究来验证。

两种广泛使用的开放式扩散大语言模型采用了相反的 logit 位置约定。用错约定会让这两类模型生成的贪心散文都崩溃，但从不解码文本的流水线却察觉不到。成本最低的防范措施，是将读出结果与模型无法看到的信息进行核对，并报告核对结果。

参考

[1] Shen Nie、Fengqi Zhu、Zebin You、Xiaolu Zhang 等。《大语言扩散模型》。arXiv:2502.09992，2025年。[2] Jiacheng Ye、Zhihui Xie、Lin Zheng、Jiahui Gao 等。《Dream 7B：扩散大语言模型》。arXiv:2508.15487，2025年。[3] Zhihui Xie、Jiacheng Ye、Lin Zheng、Jiahui Gao 等。《Dream-Coder 7B：面向代码的开放式扩散语言模型》。arXiv:2509.01142，2025年。（检查点：Dream-Coder-v0 基础版 / 指令微调 7B。）[4] Chengyue Wu、Hao Zhang、Shuchen Xue、Zhijian Liu 等。《Fast-dLLM：通过启用 KV 缓存和并行解码实现扩散大语言模型的免训练加速》。arXiv:2505.22618，2025年。[5] Tianyi Wu、Xiaoxi Sun、Yanhua Jiao、Yulin Li 等。《Dynamic-dLLM：用于扩散大语言模型免训练加速的动态缓存预算与自适应并行解码》。arXiv:2606.26120，2026年。[6] Amr Mohamed、Guokan Shang 和 Michalis Vazirgiannis。《少即是多：扩散语言模型的互稳定性采样》。arXiv:2606.16908，2026年。[7] Bumjun Kim、Dongjae Jeon、Moongyu Jeon 和 Albert No。《DAPD：通过注意力实现依赖感知并行解码，面向扩散大语言模型》。发表于 ICML，2026年。arXiv:2603.12996。[8] Lipeng Wan、Junjie Ma、Jianhui Gu、Zeyang Liu 等。《GeoBlock：从扩散语言模型的依赖几何推断块粒度》。arXiv:2603.26675，2026年。[9] Wonjun Kang、Kevin Galim、Seunghyuk Oh、Minjae Lee 等。《ParallelBench：理解扩散大语言模型并行解码的权衡》。发表于 ICLR，2026年。arXiv:2510.04767。[10] Anji Liu、Oliver Broadrick、Mathias Niepert 和 Guy Van den Broeck。《离散Copula扩散》。arXiv:2410.01949，2024年。[11] Shaorong Zhang、Longxuan Yu、Rob Brekelmans、Luhan Tang 等。《掩码扩散模型中的生成顺序与并行解码：信息论视角》。arXiv:2602.00286，2026年。[12] Subham Sekhar Sahoo、Marianne Arriola、Yair Schiff、Aaron Gokaslan 等。《简单有效的掩码扩散语言模型》。发表于 NeurIPS，2024年。arXiv:2406.07524。[13] Jiaxin Shi、Kehang Han、Zhe Wang、Arnaud Doucet 等。《离散数据的简化与广义掩码扩散》。发表于 NeurIPS，2024年。arXiv:2406.04329。[14] Guhao Feng、Yihan Geng、Jian Guan、Wei Wu 等。《扩散语言模型的理论收益与局限性》。arXiv:2502.09622，2025年。[15] Sitan Chen、Kevin Cong 和 Jerry Li。《掩码扩散模型的最优推理调度方案》。arXiv:2511.04647，2025年。[16] Fei Huang、Tianhua Tao、Hao Zhou、Lei Li 等。《论非自回归Transformer的学习》。发表于 ICML，2022年。arXiv:2206.05975。[17] Yue Wu 和 Jian Huang。《扩散语言模型的稳定性加权解码》。arXiv:2604 17068，2026年。[18] Lin Yao。《不要单独确定：扩散语言模型中的联合词元确定》。arXiv:2607.04469，2026年。[19] Jai Sharma、Yifan Wang 和 Bryan Li。《掩码离散序列模型中成对互信息的神经估计》。arXiv:2605.20187，2026年。[20] Liran Ringel、Ameen Ali 和 Yaniv Romano。《离散扩散语言模型中的依赖引导并行解码》。arXiv:2604.02560，2026年。[21] Linjun Li。《受掩码扩散启发的高斯随机场模型中的单步最低方差选择：总相关与平方根碰撞阈值》。arXiv:2607.17522，2026年。[22] Nazia Riasat。《边缘分布匹配而结构失效时：生成模型中的协方差保真度》。MathAI 2026年一篇论文的扩展版本。arXiv:2603.17041。[23] Yunxiao Zhao 和 Changxiao Cai。《扩散语言模型对内在依赖的适应》。arXiv:2602.20126，2026年。[24] Lizhuo Luo、Zhuoran Shi、Jiajun Luo、Zhi Wang 等。《DAWN：扩散大语言模型的依赖感知快速推理》。arXiv:2602.06953，2026年。[25] Xueyu Zhou、Yangrong Hu 和 Jian Huang。《DOS：掩码扩散语言模型的面向依赖的采样器》。arXiv:2603.15340，2026年。[26] Heqiang Qi、Wei Huang、Mingyuan Bai 和 Xiangming Meng。《面向掩码扩散语言模型的簇级注意力引导并行解码》。arXiv:2605.29607，2026年。

[27] Tamim Zoabi、Ameen Ali、Liran Ringel 和 Lior Wolf。《离散扩散语言模型的平均场并行解码》。arXiv:2606.15805，2026年。[28] Bohang Sun、Max Zhu、Francesco Caso、Jindong Gu 等。《路径至关重要：为扩散语言模型学习词元提交策略》。arXiv:2605.24697，2026年。[29] Lin Yao。《无序语言模型中的解码：链式法则偏差与均匀扩散》。arXiv:2606.00997，2026年。[30] Tom Young、Yunan Chen 和 Yang You。《掩码语言模型中的不一致性》。arXiv:2301 00068，2022年。[31] Suvadip Sana、Sami Wolf、Neer Mehta、Alina Shah 等。《掩码语言模型上 Glauber 动力学的混合时间》。arXiv:2605.16378，2026年。[32] Jeonseong Kim。《路径依赖去噪：从非保守场视角看扩散语言模型中的顺序坍缩》。arXiv:2605.09303，2026年。[33] Iskander Azangulov、Teodora Pandeva、Niranjani Prasad、Javier Zazo 和 Sushrut Karmalkar。《通过条件独立性检验从掩码扩散模型进行并行采样》。载于 ICLR，2026年。arXiv:2510.21961。[34] Michael Hersche、Nicolas Menet、Ronan Tanios 和 Abbas Rahimi。《扩散语言模型中的局部一致并行解码》。arXiv:2603.20216，2026年。

A 实验细节

语料库。共 285 份文档：英语散文部分包括来自 wikitext-103-raw-v1 的 121 篇 WikiText-103 文章，每篇至少 600 个字符；Python 部分则包括全部 164 道 OpenAI HumanEval 题目。每份文档在对应检查点自身分词器下的前 128 个词元构成一个窗口；较短的文档会被剔除。因此，Dream 使用全部 121 份散文文档和 126 道代码题目（共 247 个窗口），而 LLaDA 使用 121 份散文文档和 137 道代码题目（共 258 个窗口）。

两种机制。 一个块是窗口内连续的 k 个位置。后缀 会遮蔽从该块到窗口末尾的所有内容，就像解码器继续处理提示词时所看到的那样，每个窗口包含一个块，起点位于窗口中点。填充 只遮蔽该块，每个窗口包含两个块，其起点从窗口后四分之一处开始随机选取，间隔为 $4k$。每个检查点因此有 247 或 258 个后缀块，以及 494 或 516 个填充块。

$\widehat{\ }$估计量与诊断。 C 的 $M{=}64$ 个揭示顺序、链式法则偏差的 8 个顺序、顺序敏感性控制的 8 个参考顺序，以及每个位置概率最高的 8 项（用于特征分析）。参考解码采用贪心方式。

方差校正。 该估计量有噪声，因此 $\mathrm{Var}[\widehat{\mathrm{C}}] = \mathrm{Var}[\mathrm{C}] + \mathbb{E}[\mathrm{se}^2]$, ，任何折外平方误差中也会出现相同的偏移。计算 R^2 或残差之前，我们会从两者中减去测得的 $\mathbb{E}[\mathrm{se}^2]$。

预测变量。 在种子 0 下采用五个按文档分组的折，因此不会有文档同时被分到训练集和测试集。特征经过标准化。五组特征分别为最小置信度、 B、逐位置置信度和熵、完整剖面（逐位置的前8个概率和熵，按置信度对位置排序），以及包含这两个标量的完整剖面。在全部五组特征上进行 k-NN 与 $k \in \{5, 10, 20, 40\}$ 的遍历（当 $k < n/6$ 时使用），以及带有 $\lambda \in \{0.1, 1, 10, 100\}$ 和未正则化截距的岭回归，从而得到 40 个候选项。

计算。 一张 A100 80GB。两个系列均在 bfloat16 下运行于 transformers 4.46.2 中，这是 Dream 的远程代码所要求的。对每个 Dream 检查点进行一次完整测量运行需要 4,545 次批次前向调用，对每个 LLaDA 检查点则需要 4,762 次。第 2 节和附录 C 中的所有内容均为精确计算，并在 CPU 上运行。

可用性。 代码、配置、每块的结果和验证探针见 https: // github.com/DovudAsadov/dllm-total-correlation,tag paper-numbers.

B 读出门

表3报告了每个检查点的门1：各候选偏移处隐藏的真实词元的对数概率、门控器选定的偏移，以及它相对于次优项的差值。每个检查点都选中了其文档中说明的约定。

	读出偏移		在 $p{+}\delta$ 读取 $\log_2 p$(隐藏的真实词元)			差距（比特）			
模型	测得	已记录	$\delta{=}{-}1$	$\delta{=}0$	$\delta{=}{+}1$	$k{=}4$	$k{=}1$	翻转	门控器
Dream-Coder-7B（基础版）	-1	-1（自回归偏移）	-2.47	-10.87	-15.24	8.4	6.9	0	✓
Dream-Coder-7B（指令微调）	-1	-1（自回归偏移）	-3.44	-12.55	-16.80	9.1	8.4	0	✓
LLaDA-8B（基础版）	0	0（原位）	-13.19	-2.13	-13.15	11.0	16.0	0	✓
LLaDA-8B（指令微调）	0	0（原位）	-13.03	-2.34	-12.97	10.6	14.7	0	✓

表3：**门1：读出偏移是经过测量的，而非假定的**。在索引 $p{+}\delta$ 处读取经过掩码处理的 $k{=}4$ 块中隐藏的真实词元的对数概率。差距：在 $k{=}4$ 处且包含一个隐藏词元时，获胜的 δ 相对于次优项的差值。翻转：两次完全相同的前向调用之间最大值索引发生变化。

主运行和发布版本中的门控2。 主运行要求两次相同的调用返回完全相同的逻辑值，并要求单独对序列进行评分或在批次中进行评分时，最大值索引均不发生变化。这两点

在全部四个检查点上都成立：两次相同的调用差异恰好为零，批次组成不会改变任何最大值索引，尽管它会使对数概率最多变化 0.56（在 Dream-base、Dream-instruct、LLaDA-base 和 LLaDA-instruct 上分别为 0.162、0.556、0.281 和 0.224）。发布版本的门控器更加严格：估计量所结合的两种批次形状必须在每个最大值索引上都一致，并且在 B 上的均方根差异不超过 0.074 比特。在第二块 GPU 上，全部四个检查点均通过，两种形状完全一致。单独对序列进行评分（测量过程中从未这样做）时，两个 Dream 检查点各有 64 个最大值索引中的 1 个发生翻转（附录 F）。

主运行和发布版本中的门槛3。 主运行只要求解码结果非空，并且每个检查点都在校准后的位置写出了流畅文本。例如，Dream-base 解码以“Theatre . In 2002 , he appeared in”开头。发布版本的门控器还要求错误的位置产生§3.2所述的崩溃，并且任何错误都会导致失败。

C 精确结果：最坏情况与合成审计

详细讨论最坏情况。 具有固定边缘剖面的联合分布集合（运输多面体）是凸且紧的。它始终包含乘积测度，因此 $\mathsf{C} = 0$ 是可实现的。在另一端，最小熵耦合给出 $\mathsf{C}_{\max} = B - \min_P H(P)$，于是公式（3）由$H(P) \geq \max_i H(Y_i) \geq B/k$推出。当所有边缘分布相等时，使每个位置都相等的联合分布给出 $\mathsf{C} = W = (k-1)H(Y_1)$，因此该上界可以达到。由于 C 在连通集上连续，因此 0 到 $\mathsf{C}_{\max}$ 之间的每个值都能实现。熵 H 是凹函数，因此其在该多面体上的最小值在某个顶点处取得；对于较小的 k，我们通过精确的顶点枚举计算 $\mathsf{C}_{\max}$： k个公平比特恰好跨越 $[0, k-1]$ 比特。例如，四个置信度均为 0。9 的二元位置携带 $B = 1.876$ 比特，并允许 $\mathsf{C}_{\max} = 1.407$ 比特。

精确审计。 为了评估 C 已知精确值的规则，我们使用 ParallelBench [9] 的合成列表操作族及其解析 C。枚举形式与闭式形式的 C 相差 1.6×10^{-14}比特。表4给出了针对该 C 进行评估的六条边缘规则。在这355个候选集上，六条规则归并为四种不同排序。每条规则也都取得了完美的安全集 AUROC，但这只是因为在该族中 $\mathsf{C} = 0$当且仅当 $B = 0$成立，因此任何确定性检测器都能取得完美分数。因此，我们将该排序视为一个有待在真实检查点上检验的假设，而§5发现它在那里并不成立。

Rule	斯皮尔曼(·, C) [95% CI]	肯德尔	AUROC$_{\text{safe}}$	gap
置信度（最小值）、负熵（最大值）	-0.203 $[-0.345, -0.055]$	-0.104	1.0000	0
置信度（均值）	-0.201 $[-0.344, -0.054]$	-0.103	1.0000	0
差值（最小值/均值）	-0.596 $[-0.683, -0.499]$	-0.539	1.0000	0
负熵（总和）	-0.335 $[-0.471, -0.185]$	-0.198	1.0000	0

表4：**精确审计**。针对来自 REPLACE_RANDOM 和 SHUFFLE 的355个候选集，比较六条边缘规则与解析 C，每行对应一种不同排序。当且仅当 $B = 0$ 时，$\mathsf{C} = 0$，因此安全集 AUROC 达到完美。差距：§2中匹配边际对上的得分差异，该匹配边际对的 C 相差1。42比特。

D 残差的稳健性

我们针对三种选择检验§4中的残差：估计器预算 M、块大小 k以及预测器的选择方式。这些因素均未改变结论。

估计器预算。表 5 跟踪每个基础检查点的48个后缀状态块中，估计量在 M 上的收敛情况。随着 $1/\sqrt{M}$，标准误下降。表6在主运行的块上，于 M=256重复该分析，残差最多变化0。034比特。

块大小。表7给出了§4以范围形式引用的各设置下的数值，针对两个基础检查点上的八位置块。

选择预测器而不受赢家诅咒影响。表 8 重复了第 §4 节的分析，并将模型和特征集选择嵌套在其自身的折中。残差最多上升 0.035 比特。

M	标准误（比特）	标准误$\times\sqrt{M}$	$\lvert\widehat{\mathsf{C}}_M - \widehat{\mathsf{C}}_{256}\rvert$ 均值	max	每块前向计算
Dream 基础版，后缀机制，$n = 48$块，$k = 4$					
4	2.821	5.64	2.00	10.65	16
8	1.856	5.25	1.34	4.05	32
16	1.323	5.29	1.05	3.82	64
32	0.882	4.99	0.67	1.67	128
64	0.671	5.36	0.39	1.45	256
128	0.479	5.42	0.22	0.57	512
256	0.348	5.56	0	0	1024
LLaDA 基础版，后缀机制，$n = 48$块，$k = 4$					
4	1.551	3.10	1.25	7.63	16
8	1.121	3.17	0.98	4.46	32
16	0.783	3.13	0.50	2.09	64
32	0.586	3.31	0.44	1.49	128
64	0.432	3.45	0.31	1.15	256
128	0.302	3.41	0.14	0.64	512
256	0.208	3.33	0	0	1024

表 5：**估计量收敛于** $1/\sqrt{M}$。在每个检查点的 48 个固定块上，$\widehat{\mathsf{C}}_M$ 的逐块标准误及其与 $\widehat{\mathsf{C}}_{256}$ 的偏差，以及前向计算成本。每个较小的 M 都会复用同一组 256 个顺序中的前 M 个，因此该偏差低估了独立运行的误差，低估幅度为 $\sqrt{256/(256-M)}$。主要运行使用 M=64。

		估计量标准误（比特）		残差（比特）			
模型／机制	n	M=64	M=256	M=64	M=256	$\lvert\Delta\rvert$	秩一致性
LLaDA 基础版／后缀	258	0.335	0.168	1.055	1.088	0.034	0.991
LLaDA 基础版 / 填充	516	0.072	0.040	0.742	0.732	0.010	0.988

表6：**估计器预算并非主要影响因素**。相同的块采用 M=64 和 M=256 揭示顺序。秩一致性：两个估计值在各块上的斯皮尔曼相关。两次运行的批量大小也不同，分别为 8 和 4 个块，因此这同时检验了预算和批次形状。

		$\widehat{\mathrm{C}}$	% 方差	残差（比特）		上界	参考
模型／机制	n	mean	解释。	k=8	k=4	W	（比特）
Dream 基础模型／填充	250	8.36	90	2.19	1.13	14.3	7.37
Dream 基础模型／后缀	247	15.97	83	2.59	1.09	40.3	12.09
LLaDA 基础版／填充	250	9.31	96	1.60	0.74	14.5	6.12
LLaDA 基础版／后缀	250	18.29	92	2.14	1.05	36.2	10.64

Table 7: **每种设置下的** k=8 **复现**。采用表1中的协议，在 tw 上使用八位置块 o base 检查点，紧邻 k=4 残差。W 以及如表1所示的参考。

		残差（比特）		
模型／机制	n	报告值	嵌套	Δ
Dream 基础模型／填充	494	1.129	1.132	+0.003
Dream 基础模型／后缀	247	1.090	1.095	+0.005
Dream 指令版／填充	494	1.623	1.633	+0.010
Dream 指令版／后缀	247	1.840	1.845	+0.005
LLaDA 基础版／填充	516	0.742	0.743	+0.000
LLaDA 基础版 / 后缀	258	1.055	1.072	+0.017
LLaDA 指令版 / 填充	516	0.873	0.908	+0.035
LLaDA 指令版 / 后缀	258	1.174	1.172	−0.002

表 8：**选择预测器的代价**。表1 的残差（报告值）与采用相同流程、在 5 个外层文档分组折中嵌套 5 个内层折进行选择的结果之比较（嵌套）。

E 进一步结果：领域、配对比较和对照

按领域划分的残差。 表 9 按领域划分残差，并报告 C 超出 $[0, W]$ 的块。在每个检查点上，散文都比代码更难预测，而 Dream-instruct 的散文则表现得尤为突出。

	残差（比特）		链式法则偏差		不可能（占块的百分比）		
模型／机制	散文	代码	散文	code	$\hat{\mathrm{C}} < 0$	$\hat{\mathrm{C}} > W$	替代
Dream 基础模型／填充	1.35 [1.17, 1.54]	1.01 [0.74, 1.27]	0.42	0.06	9.5	7.9	5.9
Dream 基础模型／后缀	1.33 [1.12, 1.54]	1.01 [0.86, 1.15]	0.76	0.28	0.0	1.6	1.2
Dream 指令版／填充	2.23 [1.95, 2.49]	0.92 [0.54, 1.35]	0.59	0.03	25.1	8.9	14.6
Dream 指令版 / 后缀	2.58 [2.21, 2.92]	1.28 [1.06, 1.47]	1.17	0.30	4.9	8.1	8.1
LLaDA 基础版 / 填充	1.00 [0.84, 1.16]	0.54 [0.44, 0.65]	0.28	0.04	8.5	4.3	1.4
LLaDA 基础版 / 后缀	1.18 [0.97, 1.38]	0.93 [0.83, 1.05]	0.47	0.21	0.0	0.4	0.4
LLaDA 指令版 / 填充	1.16 [0.98, 1.35]	0.58 [0.47, 0.68]	0.27	0.04	7.6	9.5	2.3
LLaDA 指令版 / 后缀	1.45 [1.22, 1.68]	1.00 [0.83, 1.17]	0.67	0.20	0.0	0.8	0.4

表 9：**残差和定义不明确的区块所在之处。** 残差：合并岭回归预测器的噪声校正残差，在每个领域上评分，并给出按文档聚类的95%区间。链式法则偏差：中位数，见 §3.2。不可能：超出 $[0, W]$ 两个标准误以上的块所占百分比；替代 超出 0.2 比特的块。

配对比较。表 10 给出了§5 中各规则之间的配对比较，以及 W §4 中给出的偏相关。

	边际值–置信度	边际值–置信度	负熵–置信度	汇总–最小	偏 ρ给定 W	
模型／机制	（最小值）	（均值）	（总和与均值）	（上限）	在六条规则中	
Dream 基础模型／填充	$+0.033^{*}$	$+0.004$	-0.000	-0.048	-0.14	$+0.04$
Dream 基础模型／后缀	$+0.140^{*}$	$+0.037^{*}$	$+0.017$	-0.063	-0.28	$+0.21$
Dream 指令版／填充	-0.045^{*}	-0.036^{*}	$+0.101^{*}$	-0.044	-0.21	$+0.28$
Dream 指令版／后缀	$+0.007$	-0.032^{*}	$+0.094^{*}$	-0.034	-0.38	$+0.30$
LLaDA 基础版 / 填充	$+0.041^{*}$	$+0.009^{*}$	-0.027^{*}	-0.074	$+0.11$	$+0.23$
LLaDA 基础版 / 后缀	$+0.189^{*}$	$+0.063^{*}$	-0.042^{*}	-0.068	-0.04	$+0.08$
LLaDA 指令版 / 填充	$+0.041^{*}$	$+0.010^{*}$	-0.017^{*}	-0.057	-0.05	$+0.06$
LLaDA 指令版 / 后缀	$+0.144^{*}$	$+0.046^{*}$	-0.012	-0.072	-0.18	$+0.22$

表 10：**配对比较，以及规则在确定性之外所能看到的内容。**前三列：对文档-$\hat{\mathrm{C}}$进行配对的聚类自助法，比较 ρ（第一，C）与 $-\rho$（第二，C）；当第二条规则表现更好时为正，当区间不包含零时为 *。合并与最小值的比较：三个合并减最小值区间的上端点中的最大值；当合并在三个区间中均胜出时为负。最后两列：六条规则的偏斯皮尔曼相关系数范围，其中给定 W 后的 $\hat{\mathrm{C}}$。

对照。表 11 给出了各设置中诊断指标的中位数。

模型／机制	中位数置信度	B	$\hat{\mathrm{C}}$	链式法则偏差	条件敏感度	成对距离
Dream 基础模型／填充	0.538	4.2	0.78	0.18	0.080	0.00
Dream 基础模型／后缀	0.159	17.3	5.06	0.42	0.358	0.29
Dream 指令版／填充	0.575	4.3	0.05	0.20	0.068	0.00
Dream 指令版／后缀	0.262	11.7	2.24	0.59	0.311	0.31
LLaDA 基础版／填充	0.565	3.6	0.68	0.11	0.050	0.00
LLaDA 基础版／后缀	0.177	16.6	5.50	0.28	0.366	0.21
LLaDA 指令版 / 填充	0.597	3.5	0.55	0.10	0.049	0.00
LLaDA 指令版 / 后缀	0.205	14.8	5.03	0.37	0.352	0.21

表 11：**对照。** 各块的中位数。链式法则偏差：同 §3.2。条件敏感度：$p(Y_j \mid c)$ 与 $p(Y_j \mid y_i, c)$ 之间的全变差距离。成对距离：在 8 种揭示顺序下得到的贪心分配之间的分歧。

$\widehat{\text{C}}$解码误差是 C 的一个较弱替代指标。并行提交错误通常通过并行提交与顺序参考解码之间的不一致来衡量。在后缀机制中，该变量定义得并不明确：在 8 种不同揭示顺序下解码的贪心参考，其不同赋值的中位数为 2 个，两两不一致度为 0.21 至 0.31。在填充情形中，它与顺序无关。

其他对照。每个检查点都会对已填充的内容作出反应，在后缀机制中的中位数全变差距离为 0。31 至 0。37，而在填充中为0。05 至 0。08。§3.2 的结构不变量在全部 3，030 个块上均成立。

F 在第二块 GPU 上进行验证

我们在 NVIDIA GB10 上使用 scripts/verify/probe.py 重新运行了估计量，选取主运行中部分块，以测量（6）的两项（表 12），并测试主运行未测试的门控器。§6 中引用的差距使用随机块（每个单元 6 至 13 个块），斯皮尔曼相关系数按设置汇总，而其中的中位数顺序信息最多为 0。98 比特。

	所有探测的块					仅随机块		
模型／机制／领域	n	$\widehat{\text{C}}$	KL散度	$I(\pi;Y_S)$	KL散度$-I$	n	KL$-\widehat{\text{C}}$	I
Dream 基础模型／填充／散文	12	$+0.55$	$+1.24$	0.06	$+1.03$	10	$+0.52$	0.06
Dream 基础模型／填充／代码	28	$+0.00$	$+0.11$	0.01	-0.02	6	-0.04	0.00
Dream 基础模型／后缀／散文	9	$+7.60$	$+9.14$	0.98	$+7.75$	9	$+1.01$	0.98
Dream 基础模型／后缀／代码	11	$+1.91$	$+1.35$	0.23	$+1.13$	7	$+0.41$	0.38
Dream 指令版／填充／散文	18	$+0.14$	$+5.87$	0.88	$+5.05$	12	$+4.45$	0.97
Dream 指令版 / 填充 / 代码	22	-0.00	$+0.01$	0.01	-0.00	4	$+1.47$	0.16
Dream 指令版 / 后缀 / 散文	11	-0.72	$+8.44$	1.10	$+6.95$	6	$+5.56$	0.84
Dream 指令版 / 后缀 / 代码	29	$+1.18$	$+1.42$	0.14	$+1.28$	10	$+0.18$	0.16
LLaDA 基础版 / 填充 / 散文	15	$+1.99$	$+2.40$	0.18	$+2.06$	13	$+0.47$	0.28
LLaDA 基础版 / 填充 / 代码	25	-0.00	-0.00	0.00	-0.02	3	-0.31	0.01
LLaDA 基础版 / 后缀 / 散文	11	$+6.77$	$+7.91$	0.37	$+7.77$	10	$+1.09$	0.39
LLaDA 基础版 / 后缀 / 代码	6	$+3.67$	$+3.88$	0.15	$+3.96$	6	$+1.15$	0.15

表 12：**在子集上测得的 (6) 两项**。在 NVIDIA GB10 上，每个模型和机制最多取 24 个标记块和 16 个随机块，结果以比特为单位的中位数（附录 F）。$\widehat{\text{C}}$ 取自 B 估计量自身的一次前向传播。*KL* 为 $\mathbb{E}_\pi \text{KL}(q_\pi \,\|\, \prod_i p_i)$，即 I 顺序信息；KL$-I$ 则是相对于随机顺序采样器的成本。在条件分布可组合时，$\widehat{\text{C}}$、KL 和$-I$ 一致，且 $I = 0$。

在人类评估（HumanEval）窗口中，偏移门按 12.6 至 15.9 比特的优势选出文档所述约定，而散文窗口中该优势为 6.9 至 11.0 比特。带有熵容差的门控2（scripts/verify/gate2.py；两种机制下各取 16 个散文和代码窗口）在全部四个检查点上均通过。仅对序列评分——测量中从未这样做——会使 Dream-base 在某个位置的熵最多变化 11.3 比特。将 B 取自估计量自身的首次前向传播，而非由八个样本组成的批次，会使 197 个不可能性标记中有 2 个发生变化；主运行标记为超出 $[0, W]$的 101 个块中，有 80 个再次超出该范围（Dream-instruct 中 53 个有 39 个超出）。在近乎确定性的块上，逐块标准误会低估误差：采样路径几乎完全一致，因此标准误接近于零，却漏掉了罕见分支。即使是期望值不可能为负的 KL 成本，也能看出这一点。在下方标记为低于零的近乎确定性块中，KL 版本有 38 个中的 16 个低于负两个标准误，最多低 0.23 比特。

跨GPU可复现性。在相同的块上，B 在 A100 与 GB10 之间的差异平均为每个设置 0。04 至 0。19 比特，而在单个 Dream 块上最高可达 3。5 比特。