Locally Valid, Globally Biased
A Theoretical Analysis of Token Masking in LLM Decoding

Anonymous ACL submission

Abstract

Large language models are increasingly used in contexts where steering their outputs is essential to ensure validity with specific formats or syntax. *Token logit masking* enforces such constraints efficiently by filtering invalid tokens at each generation step. However, its statistical properties remain poorly understood. We prove that token masking produces samples from the true posterior distribution $p(x|C)$ if and only if $\prod_i Z(x_{<i}) = P(C)$ for all valid sequences, where $Z(x_{<i})$ is the probability mass preserved on valid continuations. This condition is rarely satisfied in practice. Through theoretical analysis and concrete toy example, we show that token masking systematically distorts the distribution: sequences traversing low-Z prefixes are artificially amplified, while high-Z sequences are suppressed. We propose the variance of $\prod_i Z(x_{<i})$ as a practical diagnostic to detect and quantify this distortion without computing the intractable normalization constant $P(C)$.

1 Introduction

Constrained language models are increasingly used for structured generation tasks—code, JSON, agentic function calling—where outputs must satisfy strict syntactic or semantic constraints. Ideally, one would sample from the true *posterior distribution*:

$$p(x \mid C) = \frac{p(x)}{P(C)}, \quad P(C) = \sum_{x \in C} p(x), \quad (1)$$

where C is the set of valid sequences. Computing $P(C)$, however, is infeasible for rich constraints (Rosenfeld et al., 2001).

Two main directions address this challenge. The first emphasizes *efficient enforcement* via grammar-based decoding (Poesia et al., 2022; Willard and Louf, 2023; AI, 2023; Gerganov et al., 2023; Geng et al., 2023; Ugare et al., 2025; Koo et al., 2024; Dong et al., 2025; Park et al., 2025; Tuccio et al., 2025; Chen et al., 2025; Sun et al., 2025) or heuristic logit biasing (Lu et al., 2022; Nakshatri et al., 2025). Among these, *token logit masking*—zeroing logits for tokens that would violate constraints—is now the standard due to its simplicity, output validity, and integration into production systems (Kapoor et al., 2025; OpenAI, 2024).

The second direction focuses on *distributional correctness*, treating constrained generation as probabilistic conditioning (Zhang et al., 2024; Ahmed et al., 2025) or employing sampling-based inference (rejection, importance, or SMC) (Lew et al., 2023; Ye et al., 2025; Lipkin et al., 2025). These methods approximate the true posterior distribution but are computationally expensive. Recent work seeks to unify the two lines of research (Park et al., 2024; Loula et al., 2025), but still relies on iterative procedures, making it costly for on-demand applications. However, the *distributional properties of token masking remain uncharacterized*: while it enforces local validity, it does not ensure global posterior consistency.

We summarize our contributions as follows::

- We prove that token masking distribution concides with the true posterior ditribution $p(x|C)$ if and only if $\prod_i Z(x_{<i}) = P(C)$ for all valid x, where $Z(x_{<i})$ is the preserved probability mass at step i (Theorem 1). This rarely holds in practice.
- We show that token masking systematically distorts probabilities—amplifying sequences traversing prefixes with low Z and suppressing those with high Z.
- We introduce a diagnostic based on the variance of $\prod_i Z(x_{<i})$ across valid sequences, enabling estimation of distortion magnitude without computing $P(C)$.

Rather than proposing new corrections, we establish the theoretical foundation for *why* they are nec-

essary. Our diagnostic framework offers a tractable way to quantify when efficient masking suffices and when posterior correction becomes essential.

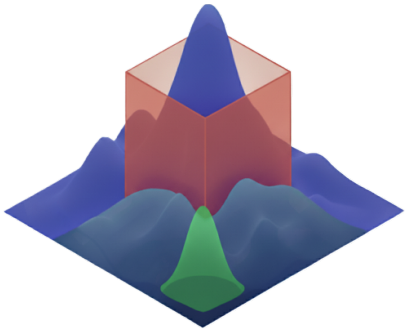

Figure 1: **Token Masking induces distortion posterior ranking**. The blue distribution shows the *unconstrained Language Model probability*. The red volume delineates the region of sequences that the LM considers highly probable but which are strictly invalid according to the constraints ($x \notin C$). The green mass illustrates the artificially *overweight* effect due to masking approach.

2 Problem Setup and Notation

We consider an autoregressive language model that generates sequences $x = x_1 \cdots x_T$ token by token according to the factorized distribution:

$$p(x) = \prod_{i=1}^{T} p(x_i \mid x_{<i}), \tag{2}$$

where each token x_i is drawn from a finite vocabulary V, $x_{<i} = x_1 \cdots x_{i-1}$ denotes the prefix before position i, and $x_{\leq i} = x_1 \cdots x_i$ includes position i. In constrained generation tasks, outputs must satisfy structural or syntactic requirements specified by a constraint set C. We assume a prefix-compatible constraint function $c : V^* \to \{0, 1\}$ where $c(u) = 1$ if prefix u can be extended to some valid sequence in C. A sequence x belongs to C if and only if $c(x_{\leq i}) = 1$ for all $i \leq T$.

Constrained decoding can be naturally formulated as posterior inference (Xefteri et al., 2025). By Bayes' rule, the posterior distribution over sequences given constraints C is:

$$\overbrace{p(x \mid C)}^{\text{posterior}} = \frac{\overbrace{P(C \mid x)}^{\text{likelihood}}\,\overbrace{p(x)}^{\text{prior}}}{\underbrace{\sum_{x'} P(C \mid x')\, p(x')}_{\text{normalization constant}}}, \tag{3}$$

where x' ranges over all possible sequences, and the likelihood $P(C \mid x)$ equals 1 if x satisfies the constraints and 0 otherwise. Considering only valid sequences, this simplifies to:

$$p(x \mid C) = \begin{cases} \frac{p(x)}{P(C)} & \text{if } x \in C \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $P(C) = \sum_{x \in C} p(x)$ is the total probability mass over valid sequences. Computing $P(C)$ is generally intractable for realistic constraints, making exact sampling from $p(x \mid C)$ impractical (Rosenfeld et al., 2001).

To address this challenge, practical methods approximate the true posterior distribution by enforcing constraints at each generation step. The most common approach is *token logit masking*, which removes invalid token continuations during generation to produce only valid outputs. At each step i, it defines a normalized distribution over valid next tokens. The joint probability of a complete sequence under tokenlogit masking is:

$$p_M(x) = \prod_{i=1}^{T} \frac{p(x_i \mid x_{<i})\, c(x_{\leq i})}{Z(x_{<i})}, \tag{5}$$

where the denominator represents the *local preserved mass*:

$$Z(x_{<i}) = \sum_{x_i} p(x_i \mid x_{<i})\, c(x_{\leq i}). \tag{6}$$

The term $Z(x_{<i})$ quantifies the proportion of probability mass the model assigns to valid continuations of prefix $x_{<i}$. A value close to 1 indicates that most next-token candidates proposed by the model are constraint-compatible, while a small value signals that the constraint severely restricts the model's natural generative distribution. This local normalization constant plays a central role in our analysis: as we show in Theorem 1, the fidelity of token masking as an approximation to $p(x \mid C)$ depends critically on how $Z(x_{<i})$ varies across different valid sequences. From this point onward, we refer to token logit masking simply as token masking.

3 Main Theoretical Result

In this section, we present the conditions under which token masking exact approximates the true posterior distribution. Our main result reveals that unbiased sampling requires a strong invariance property that is rarely satisfied in practice.

Theorem 1 (Equivalence Condition)**.** *The masking distribution equals the true posterior distribution,*

$p_M(x) = p(x \mid C)$ *for every valid sequence* $x \in C$, *if and only if:*

$$\prod_{i=1}^{T} Z(x_{<i}) = P(C) \quad \forall x \in C. \tag{7}$$

Proof. For any valid sequence $x = (x_1, \ldots, x_T) \in C$, we have $c(x_{\leq i}) = 1$ for all i. From the token masking formula (Equation 5) :

$$p_M(x) = \prod_{i=1}^{T} \frac{p(x_i \mid x_{<i})\, c(x_{\leq i})}{Z(x_{<i})} \tag{8}$$

$$= \prod_{i=1}^{T} \frac{p(x_i \mid x_{<i}) \cdot 1}{Z(x_{<i})} \tag{9}$$

$$= \frac{\prod_{i=1}^{T} p(x_i \mid x_{<i})}{\prod_{i=1}^{T} Z(x_{<i})} \tag{10}$$

$$= \frac{p(x)}{\prod_{i=1}^{T} Z(x_{<i})}. \tag{11}$$

The true posterior is:

$$p(x \mid C) = \frac{p(x)}{P(C)}. \tag{12}$$

For $p_M(x) = p(x \mid C)$, we require:

$$\frac{p(x)}{\prod_{i=1}^{T} Z(x_{<i})} = \frac{p(x)}{P(C)}, \tag{13}$$

which holds if and only if $\prod_{i=1}^{T} Z(x_{<i}) = P(C)$ for all $x \in C$. □

Theorem 1 reveals a fundamental limitation of token masking: equality between the token-masking distribution and the true posterior holds only if the cumulative product of local normalization constants remains invariant across all valid sequences. This condition demands *path-independence*—every valid sequence x, regardless of which prefixes it traverses, must yield the same product $\prod_{i=1}^{T} Z(x_{<i}) = P(C)$. In practice, this almost never holds. Different valid sequences pass through different prefixes, and at each generation step the local preserved mass $Z(x_{<i}) = \sum_{x_i} p(x_i \mid x_{<i})\, c(x_{\leq i})$ depends on the specific prefix $x_{<i}$, the model's conditional distribution at that point, and which continuations remain admissible under the constraints. Since these factors vary substantially across generation paths, the local normalization constants—and consequently their products—typically differ from one valid sequence to another. Path dependence is therefore the norm rather than the exception, making exact posterior recovery via token masking generally unattainable.

Since language models assign vastly different probability distributions to different contexts, and since constraints affect different continuations differently, the values $Z(x_{<i})$ and $Z(x'_{<i})$ will typically differ even for the same position i. Consequently, their products $\prod_{i=1}^{T} Z(x_{<i})$ and $\prod_{i=1}^{T} Z(x'_{<i})$ will also differ. Path-dependent products are thus the norm, not the exception, making exact posterior inference through token masking generally unattainable.

4 Quantifying the Distortion

The previous section showed that masking matches the true posterior only under a strong and rarely satisfied condition. Here, we formalize the *direction* of the distortion introduced by masking, showing that some sequences become artificially amplified while others are suppressed relative to the posterior distribution and proposing a Variance-based classification to asses distributional distorsion.

4.1 Relative Probability Ratios

For two valid sequences $x^{(1)}, x^{(2)} \in C$, the relative distortion induced by masking is given by:

$$\frac{p_M(x^{(1)})}{p_M(x^{(2)})} = \frac{p(x^{(1)})}{p(x^{(2)})} \cdot \underbrace{\frac{\prod_i Z(x^{(2)}_{<i})}{\prod_i Z(x^{(1)}_{<i})}}_{\text{distortion factor } R(x^{(1)}, x^{(2)})}. \tag{14}$$

The derivation of Eq. (14) is provided in Appendix A.1. This expression highlights how masking reweights the model's sequence probabilities through the distortion factor R, which governs the deviation from the true posterior ranking.

4.2 Variance-Based Classification

We assess distributional distortion via the variance of cumulative log-preserved mass over a sample $\mathcal{S}$ of valid sequences:

$$\mathrm{Var}_{\mathcal{S}}\left[\sum_i \log Z(x_{<i})\right]. \tag{15}$$

This statistic captures how unevenly probability mass is preserved across prefixes:

- *High variance*: A few outlier sequences have extreme $\prod_i Z$ values (e.g., the 00000 path), indicating rare but severe overweighting.

- *Low variance, low mean*: The entire constraint set conflicts with the model—most prefixes have low Z, implying global misalignment and systematic overweighting.

- *Low variance, high mean*: The model aligns well with constraints; Z remains high and masking induces minimal distortion.

Practically, high variance suggests filtering or debiasing isolated outliers, while low variance with low mean indicates a deeper model–constraint mismatch requiring stronger intervention (e.g., fine-tuning).

5 Case Study: A Toy Grammar

To make the effect of masking distortion concrete, we consider the minimal grammar example introduced in (Park et al., 2024) that generates sequences of length 5:

$$\begin{aligned} S &\to 1\,A_1 \mid 00000, \\ A_i &\to 1\,A_{i+1} \mid 0\,A_{i+1}, \quad i \in \{1,2,3\}, \\ A_4 &\to 1 \mid 0. \end{aligned}$$

This grammar produces 16 sequences starting with 1 and 1 exceptional sequence: 00000.

5.1 Model Specification

To isolate the effect of low local preserved mass ($Z \ll 1$) on distortion, we define a language model (LM) with two distinct behavioral branches, summarized in Appendix A.2:

1. *The Forced Path (*00000*):* In this sequence, the grammar forces the model to follow tokens (0) that is considers unlikely by the model (e.g., $p(0 \mid u) \ll 0.5$). As a result, the *local preserved mass* is very low at each step ($Z(u) \ll 1$), since most of the model's probability mass (e.g., $p(1 \mid u)$) is effectively masked out.

2. *The Compatible Path (*$1w_2w_3w_4w_5$*):* After the initial 1, the grammar imposes no restrictions on valid continuations, where each $w_i \in \{0,1\}$. While the model's conditional probabilities remain path-dependent, all continuations are valid, yielding a *maximal local preserved mass* at every step ($Z(u) = 1.0$), independent of the specific conditional probabilities.

The initial probability is set to $p(1) = p(0) = 0.5$. The specific path-dependent conditional probabilities for both branches are provided in Appendix A.2.

5.2 Quantify the Distortion

For sequences starting with 1 (e.g., $x^{(1)} = 10101$): The cumulative Z product is:

$$\prod_i Z(x^{(1)}_{<i}) = 1.0.$$

While for the exceptional sequence ($x^{(2)} = 00000$): The grammar forces 0 at every step, so $Z(u) = p(0|u)$. The cumulative product is:

$$\prod_i Z(x^{(2)}_{<i}) = 1{\times}0.2{\times}0.1{\times}0.05{\times}0.02 = 2{\times}10^{-5}.$$

As detailed in Appendix A.3, the true posterior strongly favors the sequence 10101, with a relative probability factor of 2310 compared to 00000.

Under token masking, however, this ratio drops to only 0.0462. In other words, masking *completely inverts the ranking* implied by the posterior, favoring 00000 over 10101 by roughly $1/0.0462 \approx 21.65$.

Importantly, this does not mean that the model itself prefers 00000. Rather, the inversion arises from the masking approach: sequences that traverse through prefixes with low local preserved mass $Z(u)$ are amplified when the probabilities are normalized at each step. As a result, sequences that the model considers unlikely under its own distribution can become disproportionately likely to be sampled under token masking.

This amplification can be explicitly quantified using the *distortion ratio* R introduced in equation (14), which measures how token masking reweights sequences relative to the true posterior.

6 Conclusion

Token masking matches the true constrained posterior only if $\prod_i Z(x_{<i})$ is identical across all valid sequences—a condition rarely met. Masking systematically *amplifies* sequences with low-Z prefixes, causing a possible reverse of the relative ranking under the true posterior distribution. Local preserved mass Z provides a practical, observable diagnostic measure to test and quantify when token masking poorly approximates the posterior. Future work will explore efficient correction strategies leveraging Z.

7 Limitations

Our analysis applies specifically to *open-weight models* that expose token-level logits, and to *constraint enforcement methods* based on hard token masking. In these settings, the preserved probability mass $Z(x_{<i})$ can be exactly computed at each decoding step since the set of valid tokens is explicitly known.

By contrast, our framework does not directly extend to closed-weight or API-based models, where logits are inaccessible, nor to approaches that apply *soft logit biasing* or heuristic penalties instead of hard masking (Lu et al., 2022; Nakshatri et al., 2025). In such cases, defining or estimating the preserved mass Z becomes ambiguous, as the notion of *valid continuation* is probabilistically blurred.

References

Kareem Ahmed, Kai-Wei Chang, and Guy Van den Broeck. 2025. Controllable generation via locally constrained resampling. In *The Thirteenth International Conference on Learning Representations*.

Guidance AI. 2023. Guidance: A language model programming framework. https://github.com/guidance-ai/guidance. Accessed: October 6, 2025.

Junyi Chen, Shihao Bai, Zaijun Wang, Siyu Wu, Chuheng Du, Hailong Yang, Ruihao Gong, Shengzhong Liu, Fan Wu, and Guihai Chen. 2025. Pre^3: Enabling deterministic pushdown automata for faster structured LLM generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11253–11267, Vienna, Austria. Association for Computational Linguistics.

Yixin Dong, Charlie F. Ruan, Yaxing Cai, Ziyi Xu, Yilong Zhao, Ruihang Lai, and Tianqi Chen. 2025. XGrammar: Flexible and efficient structured generation engine for large language models. In *Eighth Conference on Machine Learning and Systems*.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. 2023. Grammar-constrained decoding for structured NLP tasks without finetuning. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Georgi Gerganov and 1 others. 2023. Llama.cpp: A port of facebook's llama model in c++. https://github.com/ggerganov/llama.cpp. Accessed: October 6, 2025.

Parv Kapoor, Akila Ganlath, Changliu Liu, Sebastian Scherer, and Eunsuk Kang. 2025. Constrained decoding for robotics foundation models. *Preprint*, arXiv:2509.01728.

Terry Koo, Frederick Liu, and Luheng He. 2024. Automata-based constraints for language model decoding. In *First Conference on Language Modeling*.

Alexander K. Lew, Tan Zhi-Xuan, Gabriel Grand, and Vikash Mansinghka. 2023. Sequential monte carlo steering of large language models using probabilistic programs. In *ICML 2023 Workshop: Sampling and Optimization in Discrete Space*.

Ben Lipkin, Benjamin LeBrun, Jacob Hoover Vigly, João Loula, David R. MacIver, Li Du, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Timothy J. O'Donnell, Alexander K. Lew, and Tim Vieira. 2025. Fast controlled generation from language models with adaptive weighted rejection sampling. In *Second Conference on Language Modeling*.

João Loula, Benjamin LeBrun, Li Du, Ben Lipkin, Clemente Pasti, Gabriel Grand, Tianyu Liu, Yahya Emara, Marjorie Freedman, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Alexander K. Lew, Tim Vieira, and Timothy J. O'Donnell. 2025. Syntactic and semantic control of large language models via sequential monte carlo. In *The Thirteenth International Conference on Learning Representations*.

Ximing Lu, Sean Welleck, Peter West, Liwei Jiang, Jungo Kasai, Daniel Khashabi, Ronan Le Bras, Lianhui Qin, Youngjae Yu, Rowan Zellers, Noah A. Smith, and Yejin Choi. 2022. NeuroLogic a*esque decoding: Constrained text generation with lookahead heuristics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 780–799, Seattle, United States. Association for Computational Linguistics.

Nishanth Sridhar Nakshatri, Shamik Roy, Rajarshi Das, Suthee Chaidaroon, Leonid Boytsov, and Rashmi Gangadharaiah. 2025. Constrained decoding with speculative lookaheads. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4681–4700, Albuquerque, New Mexico. Association for Computational Linguistics.

OpenAI. 2024. Introducing structured outputs in the api. https://openai.com/index/introducing-structured-outputs-in-the-api/. [Accessed: 10/09/2025].

Kanghee Park, Jiayu Wang, Taylor Berg-Kirkpatrick, Nadia Polikarpova, and Loris D'Antoni. 2024. Grammar-aligned decoding. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Kanghee Park, Timothy Zhou, and Loris D'Antoni. 2025. Flexible and efficient grammar-constrained decoding. In *Forty-second International Conference on Machine Learning*.

Gabriel Poesia, Alex Polozov, Vu Le, Ashish Tiwari, Gustavo Soares, Christopher Meek, and Sumit Gulwani. 2022. Synchromesh: Reliable code generation from pre-trained language models. In *International Conference on Learning Representations*.

Ronald Rosenfeld, Stanley Chen, and Xiaojin Zhu. 2001. Whole-sentence exponential language models: A vehicle for linguistic-statistical integration. *Computer Speech & Language*, 15(1):55–73.

Xintong Sun, Chi Wei, Minghao Tian, and Shiwen Ni. 2025. Earley-driven dynamic pruning for efficient structured decoding. *Preprint*, arXiv:2506.01151.

Gabriele Tuccio, Luana Bulla, Maria Madonia, Aldo Gangemi, and Misael Mongiovì. 2025. GRAMMAR-LLM: Grammar-constrained natural language generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3412–3422, Vienna, Austria. Association for Computational Linguistics.

Shubham Ugare, Tarun Suresh, Hangoo Kang, Sasa Misailovic, and Gagandeep Singh. 2025. Syncode: LLM generation with grammar augmentation. *Transactions on Machine Learning Research*.

Brandon T. Willard and Rémi Louf. 2023. Efficient guided generation for large language models. *Preprint*, arXiv:2307.09702.

Vicky Xefteri, Tim Vieira, Ryan Cotterell, and Afra Amini. 2025. Syntactic control of language models by posterior inference. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25350–25365, Vienna, Austria. Association for Computational Linguistics.

Haotian Ye, Himanshu Jain, Chong You, Ananda Theertha Suresh, Haowei Lin, James Zou, and Felix Yu. 2025. Efficient and asymptotically unbiased constrained decoding for large language models. In *The 28th International Conference on Artificial Intelligence and Statistics*.

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck, and Nanyun Peng. 2024. Adaptable logical control for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

A Appendix

A.1 Derivation of the Distortion Ratio

We formalize the relationship between the masking distribution $p_M(x)$ and the true posterior $p(x \mid C)$.

For any two valid sequences $x^{(1)}, x^{(2)} \in C$:

True Posterior Ratio:

$$\frac{p(x^{(1)} \mid C)}{p(x^{(2)} \mid C)} = \frac{p(x^{(1)})/P(C)}{p(x^{(2)})/P(C)} = \frac{p(x^{(1)})}{p(x^{(2)})}.$$

The true posterior preserves the relative ranking of the unconstrained model, as the global normalization constant $P(C)$ cancels out.

Masking Distribution Ratio: From Equation (11), the joint probability under masking is $p_M(x) = \frac{p(x)}{\prod_{i=1}^{T} Z(x_{<i})}$. The ratio of probabilities under masking is:

$$\begin{aligned}\frac{p_M(x^{(1)})}{p_M(x^{(2)})} &= \frac{p(x^{(1)})/\prod_i Z(x^{(1)}_{<i})}{p(x^{(2)})/\prod_i Z(x^{(2)}_{<i})} \\ &= \frac{p(x^{(1)})}{p(x^{(2)})} \cdot \frac{\prod_i Z(x^{(2)}_{<i})}{\prod_i Z(x^{(1)}_{<i})}. \qquad (16)\end{aligned}$$

For $p_M(x)$ to equal $p(x \mid C)$, the two ratios must be equal:

$$\frac{p(x^{(1)})}{p(x^{(2)})} = \frac{p(x^{(1)})}{p(x^{(2)})} \cdot \frac{\prod_i Z(x^{(2)}_{<i})}{\prod_i Z(x^{(1)}_{<i})}.$$

This equation holds if and only if (canceling the ratio of unconstrained probabilities $\frac{p(x^{(1)})}{p(x^{(2)})}$)):

$$\frac{\prod_i Z(x^{(2)}_{<i})}{\prod_i Z(x^{(1)}_{<i})} = 1 \quad \leftrightarrow \quad \prod_i Z(x^{(1)}_{<i}) = \prod_i Z(x^{(2)}_{<i}). \qquad (17)$$

This confirms that the equivalence $p_M(x) = p(x \mid C)$ requires the path-independence of the cumulative Z product for all valid sequences.

A.2 Model Specification Details

The conditional probabilities used in the Toy Grammar Case Study (Section 5) are defined to starkly contrast the forced path (low Z) and the compatible path (high Z).

For the 00000 path, the grammar only permits 0 as the continuation token. Therefore, the local preserved mass $Z(u)$ is simply $p(0 \mid u)$. Since the model is defined to strongly prefer 1 (high $p(1 \mid u)$), the $Z(u)$ value is very low, leading to the cumulative distortion.

A.3 Detailed Calculations for Toy Grammar Case Study (Revised)

We calculate the probabilities for the two paths, $x^{(1)} = 10101$ and $x^{(2)} = 00000$, using the specifications from Table 1.

Table 1: Conditional Probabilities for Toy Grammar Branches

Prefix u	$\mathbf{p(0 \mid u)}$	$\mathbf{p(1 \mid u)}$	$\mathbf{Z(u)}$
Path 1: $1\ldots$ **(Compatible Branch,** $Z = 1$**)**			
ϵ	0.5	0.5	1.0
1	0.4	0.6	1.0
10	0.45	0.55	1.0
101	0.35	0.65	1.0
1010	0.4	0.6	1.0
Path 2: 00000 **(Forced Branch,** $Z \ll 1$**)**			
ϵ	0.5	0.5	1.0
0	**0.2**	0.8	**0.2**
00	**0.1**	0.9	**0.1**
000	**0.05**	0.95	**0.05**
0000	**0.02**	0.98	**0.02**

Path 1: $x^{(1)} = 10101$ Model probabilities (path-dependent):

$$\begin{aligned} p(x^{(1)}) &= p(1) \cdot p(0|1) \cdot p(1|10) \cdot p(0|101) \cdot p(1|1010) \\ &= 0.5 \cdot 0.4 \cdot 0.55 \cdot 0.35 \cdot 0.6 \\ &\approx 0.0231 = 2.31 \times 10^{-2}. \end{aligned}$$

Cumulative Z product: $\prod_i Z(x^{(1)}_{<i}) = 1$.

Path 2: $x^{(2)} = 00000$ Model probabilities (forced path):

$$\begin{aligned} p(x^{(2)}) &= p(0) \cdot p(0|0) \cdot p(0|00) \cdot p(0|000) \cdot p(0|0000) \\ &= 0.5 \cdot 0.2 \cdot 0.1 \cdot 0.05 \cdot 0.02 \\ &= 0.00001 = 1 \times 10^{-5}. \end{aligned}$$

Cumulative Z product (low, due to forcing):

$$\prod_i Z(x^{(2)}_{<i}) = 1 \cdot 0.2 \cdot 0.1 \cdot 0.05 \cdot 0.02 = 2 \times 10^{-5}.$$

Masking vs. Posterior Comparison (Revised)

1. *True Posterior Ranking*:

$$\frac{p(x^{(1)} \mid C)}{p(x^{(2)} \mid C)} = \frac{p(x^{(1)})}{p(x^{(2)})} = \frac{2.31 \times 10^{-2}}{1 \times 10^{-5}} = 2310.$$

   The true posterior favors $x^{(1)}$ by a factor of 2310.

2. *Masking Ranking*:

$$\begin{aligned} \frac{p_M(x^{(1)})}{p_M(x^{(2)})} &= \frac{p(x^{(1)})}{p(x^{(2)})} \cdot \frac{\prod_i Z(x^{(2)}_{<i})}{\prod_i Z(x^{(1)}_{<i})} \\ &= 2310 \cdot \frac{2 \times 10^{-5}}{1} \approx 0.0462. \quad (18) \end{aligned}$$

   The masking distribution reverses the ranking, favoring $x^{(2)}$ over $x^{(1)}$ by a factor of $\frac{1}{0.0462} \approx 21.65$.