Beyond Semantics: Optimizing Surface Formatting for Robust Retrieval-Augmented Generation

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) is essential for extending Large Language Models (LLMs) to knowledge-intensive tasks. While prior research has primarily focused on retrieval quality and prompting strategies, the influence of how the retrieved documents are framed, i.e., context format, remains underexplored. We demonstrate that semantically identical inputs can yield drastically different behaviors based solely on superficial formatting choices. Through mechanistic analysis, we reveal the underlying factors that govern performance differences, showing that suboptimal formats can disrupt information grounding. Moreover, we introduce Contextual Normalization, a lightweight framework that calibrates the input surface formats to the model's internal dynamics. By optimizing the proposed metric, it adaptively selects the most effective format without requiring architectural changes. Extensive experiments demonstrate that the method consistently enhances robustness and accuracy, particularly in challenging long-context scenarios. These findings underscore that reliable RAG depends not only on retrieving the right content, but also on how that content is presented, offering both new empirical evidence and practical techniques.

1 Introduction

Retrieval-Augmented Generation has emerged as a foundational paradigm for enabling large language models to scale to knowledge-intensive tasks by conditioning generation on external documents retrieved from large corpora (Lewis et al., 2020; Borgeaud et al., 2022). In a standard pipeline, a retriever first identifies potentially relevant texts for a given query, and these documents are then concatenated into a prompt for the LLM. With the advent of long-context LLMs that can process tens of thousands of tokens (Xiao et al., 2024; Xu et al., 2024), the opportunities for complex understanding over vast information spaces have been unlocked, making RAG increasingly central to real-world applications such as open-domain QA and scientific literature analysis.

While long-context extensions enable RAG systems to scale to much larger evidence pools, they also introduce new challenges. Recent study (Leng et al., 2024) highlights these limitations by systematically varying context length, from 2K up to 128K tokens across dozens of models, and documenting consistent failure modes when contexts become too long or unwieldy. With the number of retrieved chunks increasing, LLMs face amplified retrieval noise, redundancy across overlapping documents, and dilution of truly relevant evidence. These issues often make it harder for LLMs to distinguish signal from distraction, leading to unstable reasoning and degraded accuracy. Moreover, positional biases (Liu et al., 2024a; Zhang et al., 2024b) further interact with these challenges: LLMs tend to over-attend to the beginning or end of a prompt, leaving evidence buried in the middle underutilized. Together, these factors expose a fundamental brittleness in RAG systems that limits its reliability in real-world deployments.

To mitigate these limitations, a growing line of research has explored strategies to improve RAG performance. One representative approach is inference-time restructuring: prompt optimization (Liu et al., 2024b) selects the best permutation of retrieved chunks, while recent partitioning methods (Zhang et al., 2024a) mitigate "lost-in-the-middle" phenomena by splitting contexts and aggregating answers. While effective, these strategies often incur significant computational overhead due to complex aggregation logic. A second direction focuses on training-centric solutions: methods utilizing synthetic supervision (An et al., 2024) or specialized position-agnostic datasets (He et al., 2024) aim to teach models to ignore positional bias. Although these approaches establish

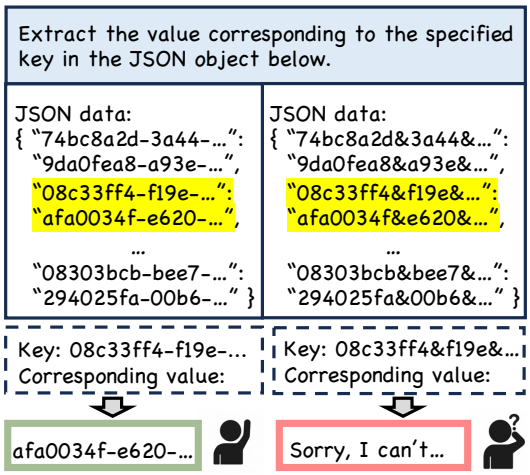


Figure 1: Illustration of Surface Brittleness: different formats yield substantial performance differences.

a strong upper bound for robustness, they face scalability issues and high resource costs. More granular approaches (Zhang et al., 2024b; Hsieh et al., 2024b) target the attention mechanism directly. For instance, Hsieh et al. (2024b) diagnose positional attention bias and propose calibrating attention weights to rectify it. This aligns with the broader trend of leveraging inference-time compute to boost performance (Vladika and Matthes, 2025; Welleck et al., 2024).

In this work, we uncover a third, overlooked critical factor: how the information is encoded. We argue that semantic equivalence does not guarantee attention equivalence during inference. As illustrated in Figure 1, we identify a phenomenon of "Surface Brittleness": purely cosmetic changes to the format of retrieved data, such as replacing a standard hyphen delimiter with an ampersand, can trigger a catastrophic collapse in model performance, even when the semantic content remains identical. This suggests that the bottleneck in long-context RAG is not only the reasoning capacity, but also a perceptual barrier caused by the internal biases. Optimizing the "connective tissue" of the context offers a lightweight, complementary intervention that can be integrated into existing pipelines with minimal overhead.

If the surface format of context can be altered while preserving semantics, could model performance also be systematically improved? To address this question, we propose **Contextual Normalization (C-NORM)**, a lightweight, training-free framework that optimizes the surface texture of retrieved contexts to maximize the model performance. C-NORM is grounded in mechanistic interpretability, which employs the proposed **Attention Balance Score (ABS)** to calibrate the context format, ensuring that the retrieved information is presented in a structure that aligns with the model's pre-trained biases. More precisely, C-NORM acts as a compatibility layer: it automatically selects the optimal formatting strategies that prevent attention sinks and promote uniform information processing.

The contributions of this work can be summarized as follows:

- We identify the Surface Brittleness in RAG systems. We demonstrate that context formatting is a decisive control variable. Through controlled experiments, we show that standardizing "connective tissue" (delimiters) is as critical as ordering, with suboptimal formats causing models to effectively ignore valid evidence.
- We provide mechanistic explanations grounded in tokenization efficiency and attention allocation, showing that format sensitivity stems from how specific tokens disrupt or facilitate the attention flows.
- We introduce a model-agnostic, inference-time calibration method C-NORM. By optimizing the Attention Balance Score, C-NORM aligns the context format to the model's internal dynamics without requiring parameter updates or expensive per-query searching.
- We conduct extensive experiments under both controlled and real-world settings, demonstrating that C-NORM consistently improves the RAG performance across diverse models. Gains are especially pronounced in challenging long-context scenarios, underscoring its practical value for reliable RAG systems.

2 The Hidden Brittleness of Context Formatting

The performance of RAG systems in long-context scenarios is profoundly influenced by the effective integration of retrieved information (Borgeaud et al., 2022; Karpukhin et al., 2020). While previous work (Asai et al., 2023) has primarily focused on the quantity and relevance of retrieved documents, we posit that the internal format of this information, more specifically, how context content in each chunk is structured, acts as a hidden control variable that can drastically alter the model's ability

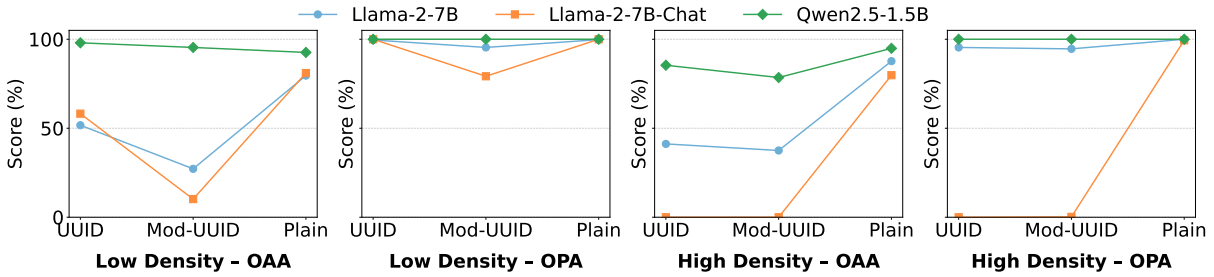


Figure 2: Model performance varies significantly with delimiter choice despite semantic identity. This divergence proves that formatting is not a neutral container but a factor that can blind LLMs before reasoning begins.

to ground information. To investigate this hypothesis, we design a set of experiments centered on the key-value extraction task (Liu et al., 2024a). The goal of the task is to retrieve the value of a specific key from a long JSON object. More details are provided in Appendix B. We manipulate the surface form of the context while keeping semantic content identical. We introduce three distinct formats to probe different aspects of model sensitivity:

- UUID: We use standard 128-bit Universally Unique Identifiers (e.g., `a1b2-c3d4...`). This represents the high-entropy, rigid structures often found in database retrievals or JSON blobs.

- Plain Text: We flatten the data, removing structured identifiers entirely. This tests the model's ability to rely solely on natural language patterns without structural crutches.

- Modified UUID: Crucially, we introduce a format that is semantically identical to the standard UUID but structurally perturbed: we replace the standard hyphen delimiter (`-`) with an ampersand (`&`). If an LLM is robust, this trivial character swap should yield negligible performance differences.

Evaluation Protocol. We design a controlled experiment with 500 samples to evaluate the performance of LLaMA-2-7B, LLaMA-2-7B-Chat (Touvron et al., 2023), and Qwen2.5-1.5B (Yang et al., 2024). The experiments are conducted with two context configurations: low-density (40 32-char chunks) and high-density (10 128-char chunks). For each setting, we record two key metrics: the Overall Averaged Accuracy (**OAA**) across all permuted positions of the gold chunk, which measures robustness of RAG systems, and the Optimal Positioned Accuracy (**OPA**), which captures the model's best-case performance under ideal position of gold key to disentangle syntactic legibility from attention capacity.

The "Hyphen-to-Ampersand" Collapse. As shown in Figure 2, our results reveal a startling fragility in different LLMs. The most significant finding is the extreme sensitivity to arbitrary delimiters. For LLaMA-2-7B-Chat, simply switching the delimiter from a hyphen (UUID) to an ampersand (Modified UUID) causes a catastrophic performance collapse: OAA plummets from 0.810 to 0.102. In some high-density structured cases, the model even refused to answer entirely. Despite the task being semantically identical, the model effectively loses the ability to "see" the information. This suggests that certain tokens trigger failures, rendering the context opaque to the model.

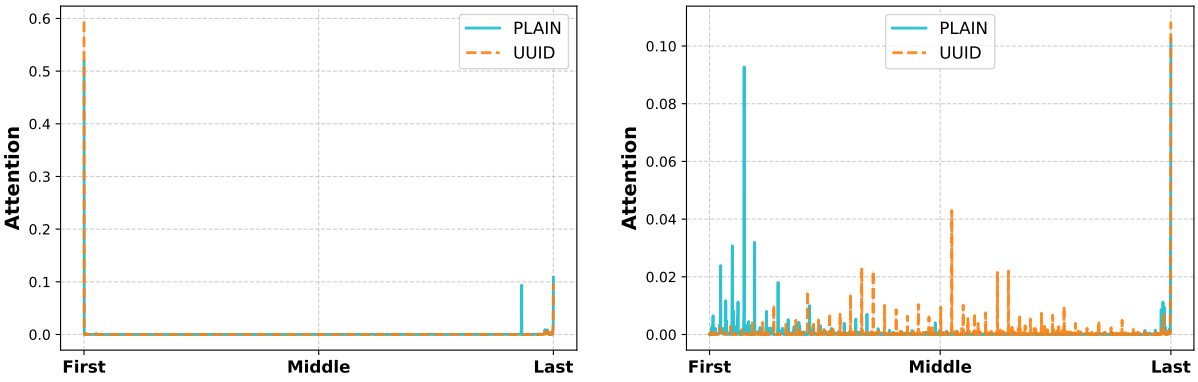
Inconsistent Inductive Biases. There is no universal best format. While the LLaMA family consistently favors unstructured Plain Text, Qwen2.5-1.5B exhibits a shifting preference. In low-density settings, Qwen benefits from the rigid structure of UUIDs. This inconsistency highlights that optimal formatting is a function of model behavior which depends heavily on its internal dynamics.

Context Window $\neq$ Effective Attention. Even models with massive context windows are not immune. Qwen2.5-1.5B, despite its 128k context capability, also suffers degradation when the format clashes with its internal biases. This implies that the LLM performances on key-value extraction are brittle, which can be deactivated simply by choosing the "wrong" delimiter.

These findings demonstrate that formatting is not merely a stylistic choice but a fundamental constraint on model performance. The divergence in model behaviors, where one collapses and another shifts preferences, requires a deeper explanation.

3 Mechanism Analysis: Attention Collapse vs. Balanced Allocation

To understand why context formats affect long-context performance and robustness, we delve into the internal dynamics of different LLMs. We focus specifically on the distribution of attention, which



(a) LLaMA-2-7B (b) Qwen2.5-1.5B

Figure 3: Attention attributions under low-density settings. The x-axis denotes the relative position of input tokens.

governs how the model allocates focus across positions. Meanwhile, we also investigated whether the sheer number of tokens (tokenization length) drives this effect. As detailed in Appendix A, while token count correlates with performance in some models (Qwen2.5-1.5B), it fails to explain the sensitivity in others (LLaMA-2-7B), making attention allocation the more universal explanatory mechanism.

3.1 Attention Attribution

To understand how context format shapes grounding, we observe last-layer attention distributions in both LLaMA-2-7B and Qwen2.5-1.5B. We aim to identify why different context formats lead to different performance patterns across models. Specifically, we construct 20 key-value pairs, place the target key at varying positions, and measure how attention from the final token is allocated across the sequence under both UUID and Plain Text formats.

Figure 3 presents the attention weights from the final token to all preceding tokens. For Qwen2.5-1.5B, the Plain Text format yields sharp attention peaks at the beginning and end of the sequence, while the UUID format produces a more uniform distribution, with increased emphasis on middle positions. On the contrary, in LLaMA-2-7B, UUID contexts concentrate attention at the sequence boundaries, whereas plain-text contexts lead to stronger coverage of the middle portion. This contrast in allocation explains the opposite performance trends observed in Figure 2: formats that encourage more balanced attention across the sequence tend to achieve higher robustness and overall accuracy in long-context retrieval.

On the Role of Training Data. To further probe why different context formats lead to distinct attention allocation patterns, we attempt to trace the effect back to the training data. With Stanford-Alpaca-7B (Taori et al., 2023), we sort tokens in its fine-tuning corpus by frequency of occurrence, and then reconstruct QA contexts where original tokens are replaced with either the most frequent or least frequent tokens. This design tests whether exposure frequency in fine-tuning data influences how attention is distributed across contexts. However, the results (detailed in Appendix C) do not show a clear relationship between token frequency and LLM performance or attention allocation. This negative result reinforces that the grounding mechanism behind context-format sensitivity is more complex than simple token frequency statistics, likely driven by deeper dynamics acquired during both pretraining and fine-tuning.

4 Contextual Normalization: Inference-Time Format Calibration

Inspired by the above findings, we propose Contextual Normalization, C-NORM, a lightweight procedure that standardizes retrieved passages into a format that better supports grounding in long contexts. As shown in Figure 4, the method operates in three stages: (i) candidate formatting of contexts, (ii) attention-guided scoring to select a format, and (iii) application of the chosen format for all contexts in RAG. This procedure is model-aware yet training-free, requiring only a forward pass with attention outputs.

4.1 Candidate Formatting

Given a query $q \in Q$ and a set of retrieved passages $\mathcal{D} = \{d_1, \ldots, d_m\}$, we generate format variants of each passage using sentence-level restructuring. Specifically, with a delimiter $f \in \{\texttt{none}, \texttt{-}, \texttt{\_}, \texttt{:}, \texttt{.}, \sim, \texttt{+}, \texttt{/}, \texttt{\&}, \ldots\}$ and the predefined

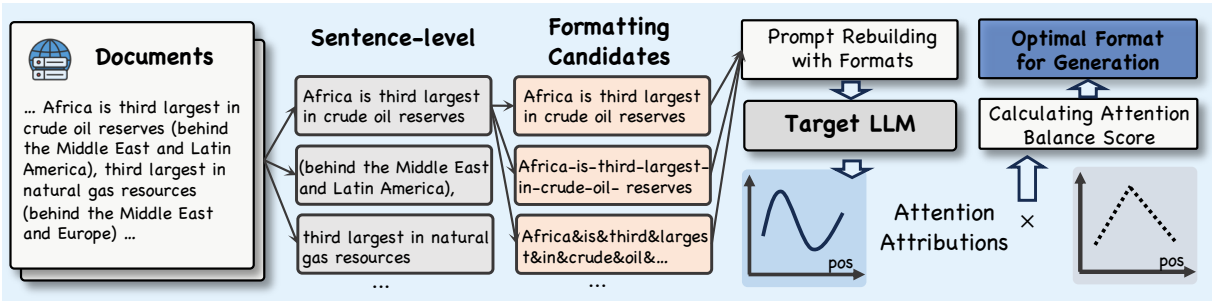


Figure 4: Overview of the proposed C-NORM pipeline.

ratio $p \in [0, 1]$, a fraction p of sentences in d_i are reformatted by replacing whitespace with f. This procedure preserves semantic content while varying structural cues in a controlled manner, creating candidate contexts $\tilde{d}_i^{(f,p)}$. The formatted documents are then assembled as contexts into prompts for finishing the task.

4.2 Attention-Guided Scoring

To assess which format best supports grounding, we propose an **Attention Balance Score (ABS)** from the LLM's internal attention distributions. For each candidate format f, we sample a subset of prompts S with $|S| \ll |Q|$, and extract the last-layer attention vector $a \in \mathbb{R}^T$ corresponding to the final token. We then compute:

$$\mathrm{ABS}(a) = 1 - 2 \cdot |\mu - 0.5|,$$

where

$$\mu = \sum_{t=1}^{T} \left(\frac{t-1}{T-1}\right) \cdot \frac{a_t}{\sum_j a_j}.$$

This score peaks when attention mass is balanced across the sequence, avoiding pathological focus on only the beginning or end of the input. The final delimiter $f^\star$ is chosen by maximizing the average ABS across S sampled prompts:

$$f^\star = \arg\max_f \frac{1}{S} \sum_{s=1}^{S} \mathrm{ABS}(a_s^{(f)}).$$

4.3 Inference-Time Deployment

At inference time, all sentences in the retrieved documents are reformatted with the selected configuration $(f^\star, p)$ before constructing the final prompt. Here, $f^\star$ denotes the delimiter format that has been automatically chosen during the calibration stage, and p specifies the proportion of sentences in which this format is applied. The reformatting step produces a normalized context representation that reduces spurious variability in how evidence is presented to the model. Importantly, the operation is performed at the sentence level, ensuring that semantic content remains intact while surface patterns are harmonized.

To summarize, C-NORM provides a lightweight and training-free mechanism for adapting context structure to the inductive biases of each model. By aligning the input format with the model's internal dynamics, it reduces mismatches between surface structure and processing preferences, thereby systematically mitigating the brittleness.

5 Experiments

To validate the effectiveness of C-NORM in enhancing the robustness and generalization of LLMs under long-context RAG, we design two complementary evaluation settings: a controlled QA test based on NQ-Open and the real-world task from LongBench-v2 to assess generalizability across diverse input types. We show that C-NORM consistently improves LLM's performance.

5.1 Controlled Long-Context RAG Settings

In this case, we propose a controlled test using a permuted version of NQ-Open to evaluate both the robustness to order variation and long-context capacity of LLMs. First, we randomly sample 500 questions from NQ-Open (Liu et al., 2024a). For each question, one gold (relevant) document is identified and mixed with 9 distractors, each containing about 100-300 tokens. We then construct 10 input permutations by placing the gold document at each possible position while shuffling the remaining distractors. The ratio p in C-NORM is fixed at $p = 0.5$ with 8 samples used for selecting the best delimiters. We report the OAA and OPA metrics introduced in Section 2. All results are averaged over three random seeds to reduce variance.

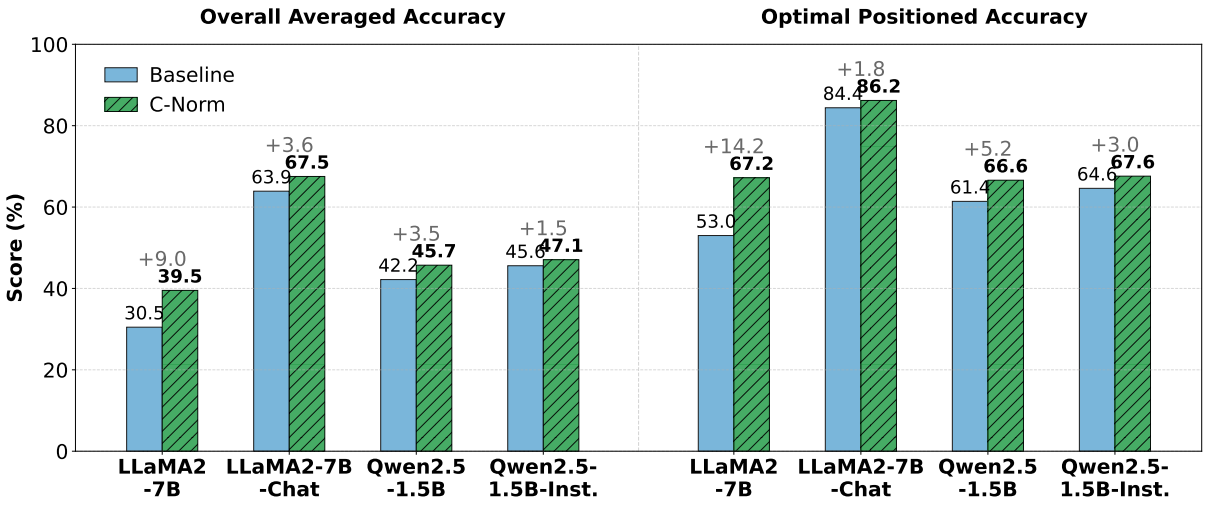


Figure 5: Results on the controlled long-context RAG setting using NQ-Open. We report Overall Averaged Accuracy (OAA) to measure robustness against context order permutations, and Optimal Positioned Accuracy (OPA) to assess capacity under the best placement of the gold document. Baseline denotes the original model, while C-NORM indicates results with contextual normalization.

Models. We adopt several LLMs for evaluation, including LLaMA-2-7B (pretrained context length 4K) (Touvron et al., 2023), LLaMA-2-7B-Chat (4K), Qwen2.5-1.5B (128K) (Yang et al., 2024), and Qwen2.5-1.5B-Instruct (128K). These two model families are specifically selected with their completely different formatting sensitivities demonstrated in Figure 2. By evaluating across such distinct LLMs, we aim to demonstrate the universality of C-NORM in adapting to diverse internal mechanisms. We utilize unaligned QA prompts for base models and official chat templates for instruction-tuned variants, with full prompt specifications provided in the Appendix D. All generations are performed with temperature fixed at 0 to ensure deterministic outputs and eliminate randomness from sampling.

Experimental Results. In the controlled long-context RAG evaluation on NQ-Open, as shown in Figure 5, C-NORM consistently improves both robustness (OAA) and comprehensive capacity (OPA) across all evaluated LLMs. The gains are especially pronounced for LLaMA-2-7B, showing that format adaptation can compensate for the LLM's limited understanding ability. It highlights that long-context performance is not only determined by LLM scale or pretraining context window, but also by how the context is presented. Interestingly, the most effective formats are often not the ones most interpretable to humans. For instance, delimiter-heavy or structurally altered representations outperform plain natural text. This underscores the importance of optimizing the input format for alignment with the model's internal dynamics rather than assuming that human-friendly representations are optimal. By automatically selecting a context format that maximizes balanced attention, C-NORM enables models to reason more reliably across arbitrary evidence positions, offering a practical path toward more robust long-context RAG systems.

5.2 Real-World RAG Settings

To evaluate the real-world utility of C-NORM, we adopt LongBench-v2 (Bai et al., 2024), a benchmark targeting long-context performance across diverse question types. It contains 503 multiple-choice questions drawn from six categories, including single-document and multi-document QA, long in-context learning, dialogue history understanding, codebase comprehension, and structured data understanding, which covers both textual and semi-structured formats. Each question is paired with a long context ranging from 8K to over 2M words, with most falling under 128K.

We evaluate model performance using overall accuracy, complemented by breakdowns across difficulty levels (Easy and Hard) and context lengths (Short, Medium, and Long). For this setting, we adopt LLaMA-2-7B-Chat and Qwen2.5-1.5B-Instruct, leveraging the official task templates provided by LongBench to ensure comparability with prior work. Moreover, we evaluate C-NORM against (1) Baseline, which uses original document formatting, and (2) Min-Token. The latter

Table 1: Evaluation on LongBench-v2. We report overall accuracy along with breakdowns across difficulty (Easy vs. Hard) and context length (Short, Medium, Long).

Model	Method	Overall	Easy	Hard	Short	Medium	Long
LLaMA-2-7B-Chat	Baseline	26.4	25.0	27.3	26.7	24.7	29.6
	Min-Token	26.6	25.0	27.7	27.8	22.8	32.4
	C-NORM	**27.6**	**27.1**	**28.0**	**28.3**	**25.6**	**32.6**
Qwen2.5-1.5B-Instruct	Baseline	23.7	26.6	21.9	27.8	23.7	16.7
	Min-Token	25.6	**27.6**	24.4	29.4	27.0	16.7
	C-NORM	**26.2**	27.1	**25.7**	**29.4**	**27.4**	**18.5**

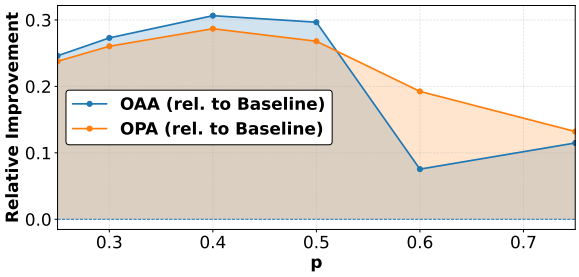


Figure 6: Impact of formatting density p on LLaMA-2-7B performance.

is derived from our analysis in Section 3, which suggests that higher tokenization efficiency correlates with better performance. We explicitly do not compare against other training-free interventions as they operate on orthogonal mechanisms (attention weights and inference logic, respectively) and are complementary to the input-level normalization provided by C-NORM.

Experimental Results. As presented in Table 1, the results on LongBench-v2 demonstrate that C-NORM consistently achieves the highest overall accuracy across both models. Notably, the method excels in the Hard and Long subsets, which represent the most challenging scenarios for RAG. For instance, Qwen2.5-1.5B-Instruct sees a substantial boost (25.7% vs. 21.9%) in the Hard category and LLaMA-2-7B-Chat gains in the Long category (32.6% vs. 29.6%) over the baseline. While Min-Token generally outperforms the baseline, validating our hypothesis in Section 3 that tokenization efficiency is a factor, it lacks stability. For LLaMA-2-7B-Chat, Min-Token actually degrades performance in the Medium setting compared to the baseline (22.8% vs. 24.7%), whereas C-NORM maintains robustness (25.6%). This confirms that simply minimizing sequence length is a useful but insufficient heuristic. Reliability requires the attention-aware calibration provided by C-NORM.

5.3 Discussions

While the experiments above demonstrate the effectiveness of C-NORM, several design choices warrant further analysis. In particular, the choice of delimiters, formatting density p, and the computational trade-offs involved in selection can influence practical deployment.

Delimiter Choices. We first examine the effect of delimiter choices in C-NORM. A wider set of candidate delimiters consistently improves performance, as it increases the chance of identifying a format that better aligns with LLM's internal processing. Interestingly, the best-performing delimiter is not always human-interpretable or intuitive. For instance, in our controlled settings, the selected delimiters vary across models: LLaMA-2-7B preferred ".", LLaMA-2-7B-Chat favored ":", Qwen2.5-1.5B chose "-", Qwen2.5-1.5B-Instruct selected "&". Moreover, we observe that the optimal delimiter can also vary across different context settings and lengths, which makes manual selection impractical. These findings underscore two important insights: (1) delimiters that yield high Attention Balance Scores (ABS) can substantially enhance robustness, confirming the effectiveness of C-NORM; and (2) optimal delimiter preferences are both model-specific and context-dependent, highlighting the necessity of automatic selection via ABS rather than relying on human intuition.

Formatting Density (p). We further investigate the impact of the formatting ratio p, which controls the density of delimiter replacement. As evidenced in Figure 6 by our controlled experiments with LLaMA-2-7B, increasing p from the Baseline ($p = 0$) yields steady gains. Pushing the density further causes performance to recede while remaining the positive improvement. This implies the trade-off between structural guidance and semantic integrity, where p acts as a regularization hyperpa-

rameter that balances the benefits of structural normalization against the risks of over-tokenization.

Efficiency and Sensitivity Analysis. Finally, we examine the sensitivity of C-NORM to the sample size (S) used for best format selection and the resulting computational overhead. By varying S from 1 to 10, we observe that the chosen delimiter and downstream performance remain largely stable, indicating rapid convergence. In practice, setting $S = 6$ is sufficient to robustly identify optimal formats. Consequently, the computational cost, defined by $|S| \times |F|$ forward passes, is negligible and is incurred only once per model-task pair. Crucially, at inference time, C-NORM imposes nearly zero latency penalty. While modifying delimiters might theoretically introduce slight token variations, the sentence-level operation controlled by the density p ensures the negligible increase in token count relative to the total input length.

In summary, our analysis highlights the effectiveness of C-NORM in adapting to diverse models and context settings. The method consistently identifies beneficial delimiters and achieves robust improvements with only a handful of samples.

6 Related Work

Retrieval Augmented Generation (RAG) has been widely adopted to improve model performance on the tasks that requires intensive knowledge resources (Borgeaud et al., 2022; Lewis et al., 2020; Karpukhin et al., 2020). Traditional RAG pipelines usually manage short context windows, typically involving tasks with concise and immediately relevant contexts (Lewis et al., 2020). While effective for short and well-contained queries, the systems face substantial limitations when scaling to more complex or open-ended tasks (Jeong et al., 2024). Many real-world questions require integrating dispersed evidence from multiple documents or reasoning over lengthy documents such as academic articles, legal cases, or multi-turn dialogues. Standard pipelines that retrieve and concatenate only a few short passages (typically 100-300 tokens each) often suffer from information fragmentation or omission of critical context (Li et al., 2024b; Hsieh et al., 2024a). Furthermore, fixed-length context windows in most pretrained LLMs (e.g., 2K-4K tokens) severely limit the amount of retrievable evidence considered simultaneously. These bottlenecks have prompted shifts toward long-context RAG setups, leveraging larger contexts and improved retrieval for open-domain QA (Asai et al., 2023; Lee et al., 2019; Nakano et al., 2021), multi-hop reasoning (Zhong et al., 2023; Ho et al., 2020), and complex document understanding (Dua et al., 2019; Li et al., 2024a).

Long-Context RAG. Recent studies (Liu et al., 2024a; Zhang et al., 2024b; An et al., 2024; Liu et al., 2024b) have revealed critical limitations in how large language models utilize long-context inputs in RAG. Simply appending more retrieved text does not guarantee improved performance, potentially causing degradation due to positional biases, information dilution, and the "lost-in-the-middle" phenomenon (Liu et al., 2024a). Models often favor content at the beginning or end of the prompt, neglecting relevant information buried in the middle. This results in significant performance variance depending on the order of retrieved documents, even if the overall content remains unchanged (Liu et al., 2024b; Zhang et al., 2024b; An et al., 2024). Thus, the effectiveness of long-context RAG is influenced not only by the amount of available information but also by how it is ordered and integrated.

7 Conclusion

In this work, we expose the "Surface Brittleness" problem in long-context RAG. Semantically identical contexts can yield drastically different performance based solely on surface differences. Our mechanistic analysis reveals that this is not a failure of reasoning, but a perceptual bottleneck arising from the interplay of model's inductive biases. Building on these insights, we introduce C-NORM, a training-free calibration framework. By leveraging the proposed Attention Balance Score (ABS), C-NORM aligns the structural presentation of retrieved documents with the model's internal dynamics. Extensive experiments across both controlled evaluations and the real-world RAG benchmark demonstrate that C-NORM consistently improves the RAG performances. Gains are especially pronounced in challenging Hard and Long scenarios, where retrieval noise and positional biases pose the greatest hurdles.

Ultimately, our findings highlight that reliable grounding in RAG depends not only on what is retrieved, but also on how it is presented to the model. By reframing context presentation as a normalization problem, C-NORM opens a practical new direction for improving the stability and scalability of large language models.

8 Limitations

While C-NORM offers a robust, training-free mechanism for stabilizing long-context RAG, we acknowledge several limitations regarding applicability and scope in our current study. C-NORM relies on internal attention weights for the calibration phase, limiting its direct application to closed-source API models where such telemetry is inaccessible. However, for these scenarios, we hypothesize that the optimal formats identified on open-weights models of similar training dynamics could possibly serve as effective transfer proxies, a direction we leave for future exploration. Besides, in this work, we focused primarily on optimizing sentence-level delimiters. We did not exhaustively evaluate hierarchical formats such as JSON, XML, or Markdown structures, nor did we explore semantic-level rewrites. While our findings confirm that simple delimiter calibration yields significant gains, investigating the interaction between complex structural schemas and attention mechanisms remains a rich avenue for future work.

References

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2024. Make your llm fully utilize the context. *Advances in Neural Information Processing Systems*, 37:62160–62188.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2023. Task-aware retrieval with instructions. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 3650–3675. Association for Computational Linguistics.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *CoRR*, abs/2412.15204.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, and 9 others. 2022. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2368–2378. Association for Computational Linguistics.

Junqing He, Kunhao Pan, Xiaoqun Dong, Zhuoyang Song, Yibo Liu, Qianguo Sun, Yuxin Liang, Hao Wang, Enming Zhang, and Jiaxing Zhang. 2024. Never lost in the middle: Mastering long-context question answering with position-agnostic decompositional training. *Preprint*, arXiv:2311.09198.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing A multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6609–6625. International Committee on Computational Linguistics.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024a. RULER: what's the real context size of your long-context language models? *CoRR*, abs/2404.06654.

Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long Le, Abhishek Kumar, James Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, and Tomas Pfister. 2024b. Found in the middle: Calibrating positional attention bias improves long context utilization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14982–14995, Bangkok, Thailand. Association for Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 7036–7050. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical*

Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018, pages 66–71. Association for Computational Linguistics.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6086–6096. Association for Computational Linguistics.

Quinn Leng, Jacob Portes, Sam Havens, Matei Zaharia, and Michael Carbin. 2024. Long context RAG performance of large language models. In *Adaptive Foundation Models: Evolving AI for Personalized and Efficient Learning*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Huayang Li, Pat Verga, Priyanka Sen, Bowen Yang, Vijay Viswanathan, Patrick Lewis, Taro Watanabe, and Yixuan Su. 2024a. Alr^2: A retrieve-then-reason framework for long-context question answering. *CoRR*, abs/2410.03227.

Zhuowan Li, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024b. Retrieval augmented generation or long-context llms? a comprehensive study and hybrid approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 881–893.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024a. Lost in the middle: How language models use long contexts. *Trans. Assoc. Comput. Linguistics*, 12:157–173.

Tianyu Liu, Jirui Qi, Paul He, Arianna Bisazza, Mrinmaya Sachan, and Ryan Cotterell. 2024b. Likelihood as a performance gauge for retrieval-augmented generation. *arXiv preprint arXiv:2411.07773*.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Juraj Vladika and Florian Matthes. 2025. On the influence of context size and model choice in retrieval-augmented generation systems. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6724–6736.

Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilia Kulikov, and Zaid Harchaoui. 2024. From decoding to meta-generation: Inference-time algorithms for large language models. *Preprint*, arXiv:2406.16838.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Retrieval meets long context large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Gongbo Zhang, Zihan Xu, Qiao Jin, Fangyi Chen, Yilu Fang, Yi Liu, Justin F. Rousseau, Ziyang Xu, Zhiyong Lu, Chunhua Weng, and Yifan Peng. 2024a. A mapreduce approach to effectively utilize long context information in retrieval augmented language models. *Preprint*, arXiv:2412.15271.

Zhenyu Zhang, Runjin Chen, Shiwei Liu, Zhewei Yao, Olatunji Ruwase, Beidi Chen, Xiaoxia Wu, and Zhangyang Wang. 2024b. Found in the middle: How language models use long contexts better via plug-and-play positional encoding. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. 2023. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of*

the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023, pages 15686–15702. Association for Computational Linguistics.

A Analysis of Tokenization Length

In this section, we investigate the impact of tokenization length on the key-value extraction task to determine if performance gains are merely a result of shorter contexts.

For LLMs such as Qwen2.5-1.5B, which use a SentencePiece-based tokenizer (Kudo and Richardson, 2018), delimiter characters such as '-', ':', '&', '_', and '+' affect the token count of the input string significantly. To analyze this, we used 200 synthetic key-value samples, each consisting of 40 context pairs. The target (gold) key-value pair is inserted at each position. We report OAA as the aggregated metric.

As shown in Figure 7, the results reveal a relatively strong negative correlation (Pearson's $r = -0.82$) between the number of tokens produced and the corresponding OAA for Qwen2.5-1.5B. In other words, delimiters that yield shorter tokenized sequences (e.g., hyphens or colons) lead to higher accuracy.

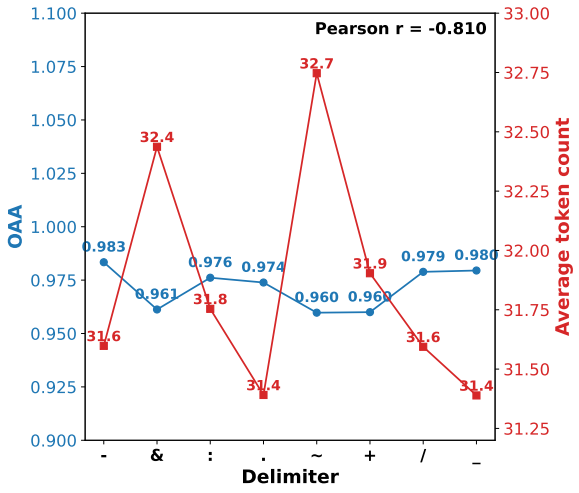


Figure 7: Qwen2.5-1.5B performance across delimiter configurations. Across settings, we observe a negative trend: configurations that inflate tokenization length tend to yield lower OAA.

Why this is not the full story: This behavior is not universal. For LLMs like LLaMA-2-7B, which tokenize many symbols (e.g., -, _, /, +) into single-character tokens, the number of tokens remains unchanged across different delimiters. In these cases, performance still varies significantly with different delimiters, but the effect cannot be attributed to token count. Therefore, while token efficiency is beneficial, it does not explain the format sensitivity phenomenon across all models, necessitating the attention-based approach proposed in the main text.

B Key-Value Extraction

We adopt a controlled **key-value extraction task** to study the effect of context formatting on retrieval-augmented generation. The task is defined as follows: given a long JSON-like object containing multiple key-value pairs, the model must return the value corresponding to a specified key. Unlike open-domain QA, this setup is free from world knowledge or semantic priors, since both keys and values are synthetic 32-character strings. As a result, performance directly reflects the LLM's ability to utilize and navigate long contexts rather than any memorized information. This task provides a minimal yet effective probe of long-context reasoning. Because all key-value pairs are semantically meaningless, the model cannot rely on prior knowledge; instead, it must depend entirely on the context provided. Success therefore reflects two abilities: (i) robust retrieval under distraction, as the model must locate the gold key among many distractors regardless of position, and (ii) sensitivity to formatting, since any performance difference arises solely from how identifiers are represented (e.g., hyphenated UUIDs versus plain texts). This isolation makes the task particularly well-suited for analyzing how structural cues in the input guide attention and grounding.

We design three variants of the input, differing only in the format of the identifiers:

- **UUID**: Keys and values are expressed as standard universally unique identifiers, represented as 32-character hexadecimal strings with hyphen delimiters.
- **Plain Text**: Identifiers are flattened into continuous 32-character strings without structural delimiters.
- **Modified UUID**: Identifiers are expressed as UUIDs but with hyphens replaced by alternative delimiters (e.g., the "&" symbol).

Prompt. The task prompt is shown below. The model is asked to extract the value associated with a given key.

Task Prompt

Extract the value corresponding to the specified key in the JSON object below.

```
# UUID:
550e8400-e29b-41d4-a716-446655440000:
123e4567-e89b-12d3-a456-426614174000

# Plain Text:
550e8400e29b41d4a716446655440000:
123e4567e89b12d3a456-426614174000

# Modified UUID:
550e8400&e29b&41d4&a716&446655440000:
123e4567&e89b&12d3&a456&426614174000
```

Key: xxxxxxx **Corresponding value:**

C Frequency-Controlled Token Replacement

To further analyze whether token exposure during fine-tuning contributes to the observed sensitivity of attention allocation to context format, we design a **Frequency-Controlled Token Replacement** experiment. Specifically, we focus on the Stanford-Alpaca-7B model and construct test cases where context tokens are systematically replaced with tokens of varying frequency in the fine-tuning corpus.

Settings. We evaluate the robustness and capacity of LLM on 100 samples from the NQ-Open dataset (Liu et al., 2024a). Each sample is paired with 6 retrieved documents, each containing approximately 100-300 tokens. To simulate long-context reasoning, we permute the position of the gold document across all possible positions. For token replacement, we sort tokens in the Alpaca fine-tuning data by frequency of occurrence and define replacement groups corresponding to the top-$k\%$ most frequent tokens and bottom-$k\%$ least frequent tokens ($k = 1, 5, 10$). Replacement is enforced by prompting the model to rewrite retrieved passages using only tokens from the allowed set, according to the following instruction:

Replacement Prompt

You are given a list of allowed tokens. Your task is to rewrite the text by replacing as many words as possible with the allowed tokens.

Rules:

1. Do **not** add or remove sentences.
2. Do **not** change the order or structure.
3. Only substitute words with allowed tokens when possible.
4. Keep the formatting exactly the same as the original.

Allowed tokens: [token list here]

Example:
Original text: `The cat is sleeping on the mat.`
Rewritten text: `a cat is sleeping on the mat`

Now rewrite the following text: [original text here]

Table 2: Performance under frequency-controlled token replacement on NQ-Open with Stanford Alpaca 7B. Top-k and bottom-k indicate substitution using the most and least frequent tokens from the fine-tuning corpus.

Setting	OAA	OPA
Stanford Alpaca (no replacement)	0.538	0.690
Top 10%	0.530	0.720
Top 5%	0.512	0.700
Top 1%	0.505	0.640
Bottom 10%	0.505	0.680
Bottom 5%	0.510	0.700
Bottom 1%	0.502	0.670

Results. Table 2 reports the overall averaged accuracy (OAA) and optimal-position accuracy (OPA) under different replacement groups. The baseline Alpaca model without replacement achieves an OAA of 0.538 and OPA of 0.690. Substituting with frequent tokens (top 10%) slightly reduces performance (OAA = 0.530, OPA = 0.720), while extreme substitution with the most frequent single token further degrades results (OAA = 0.505, OPA = 0.640). Similarly, replacing with least frequent tokens (bottom 10% / 5% / 1%) shows comparable degradation.

Discussion. The results suggest that token frequency alone does not provide a satisfactory expla-

nation for the different attention allocation patterns observed across context formats. Substitutions with both highly frequent and rarely seen tokens lead to similar levels of degradation, and no clear monotonic relationship is observed. This indicates that the grounding mechanism behind context sensitivity is more complex than exposure frequency, and is likely shaped jointly by pretraining dynamics and fine-tuning objectives.

D Prompt Templates

We provide the prompt templates used across different evaluation settings to ensure consistency and reproducibility. These prompts guide the model during evaluation in various configurations, including controlled chunk order permutations, and real RAG settings (LongBench-v2).

Controlled Setting

Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant). The search results are ordered randomly.

Search Results: {search_results}

Question: {question}

Answer:

Evaluation Prompt in LongBench-v2

Please read the following text and answer the question below.

<text> DOC </text>

Question: Q

Choices:
(A) C_A
(B) C_B
(C) C_C
(D) C_D

Format your response as follows: "The correct answer is (insert answer here)".

System Prompt (Alignment Instruction)

You are a helpful, respectful and honest assistant. Always answer as helpfully as possible, while being safe. Please ensure that your responses are socially unbiased and positive in nature. If a question does not make any sense, or is not factually coherent, explain why instead of answering something not correct. If you don't know the answer to a question, please don't share false information.