CONTEXT BAGGING: INFERENCE-TIME ENSEMBLING FOR ROBUST LONG-CONTEXT QA UNDER HARD DISTRACTORS

FARS
Analemma
fars@analemma.ai

ABSTRACT

Long-context language models struggle with hard distractors—semantically similar but misleading passages that cause systematic errors. Self-consistency, which samples multiple decoding trajectories from the same context, fails because errors are context-driven: all samples converge to the same wrong answer. We propose Context Bagging (CoBag), an inference-time ensembling method that allocates test-time compute to *context diversity* rather than decoding diversity. CoBag samples K diverse context subsets using relevance-weighted selection, randomly permutes paragraph order within each subset, generates answers via greedy decoding, and aggregates via majority voting. On MuSiQue with hard distractors, CoBag significantly outperforms self-consistency (+3.12 EM, p¡0.001). Surprisingly, ablation reveals that order diversity is the dominant mechanism (+1.44 EM), while subset diversity provides marginal additional benefit (+0.40 EM), suggesting that simple order shuffling may suffice for many applications.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Long-context language models are increasingly deployed in retrieval-augmented generation (RAG) systems, multi-document question answering, and tool-using agents that aggregate intermediate outputs. In these settings, the input context is rarely clean: it often contains irrelevant passages, conflicting evidence, or misleading text that is semantically similar to the question. Recent evaluations reveal that such *hard distractors* can cause catastrophic performance drops of up to 80% in state-of-the-art models (Lee et al., 2026), with the "lost in the middle" phenomenon (Liu et al., 2023) showing that models struggle to utilize information from middle positions of long contexts.

A natural approach to improve robustness is self-consistency (Wang et al., 2022): sample multiple decoding trajectories and select the most frequent answer through majority voting. However, Byerly & Khashabi (2024) demonstrated that self-consistency fundamentally fails for long-context problems. The core issue is that when the context contains misleading information, all samples are conditioned on the same biased context, producing correlated errors that converge to the same wrong answer. This suggests that the problem is *context-driven*, not *decoding-driven*.

We propose that the solution is *context-level diversity* rather than decoding-level diversity. By showing the model different views of the context—varying both which paragraphs are included and their order—we can decorrelate errors and enable effective majority voting. We introduce **Context Bagging (CoBag)**, an inference-time ensembling method that samples K diverse context subsets using relevance-weighted selection, randomly permutes paragraph order within each subset, generates answers via greedy decoding, and aggregates via majority voting.

We evaluate CoBag on a noisy split of MuSiQue (Trivedi et al., 2021) with hard distractors and find that it significantly outperforms self-consistency (+3.12 EM, p¡0.001). Surprisingly, our ablation reveals that order diversity alone captures most of the improvement (+1.44 EM), while subset diversity

[1] https://gitlab.com/fars-a/context-bagging-noisybench

provides marginal additional benefit (+0.40 EM). Our contributions are threefold. First, we propose Context Bagging (CoBag), an inference-time method that allocates test-time compute to context diversity rather than decoding diversity for robust long-context QA. Second, we demonstrate that context-level diversity significantly outperforms decoding-level diversity (self-consistency) under hard distractors, with CoBag achieving +3.12 EM improvement (p_i0.001). Third, we reveal through ablation that order diversity is the dominant mechanism, suggesting that simple order shuffling may be sufficient for many applications.

2 Related Work

Long-Context Robustness. Recent work has revealed significant limitations in how language models utilize long contexts. Liu et al. (2023) demonstrated the "lost in the middle" phenomenon, showing that LLM performance degrades substantially when relevant information appears in the middle of long contexts rather than at the beginning or end. Subsequent benchmarks including RULER (Hsieh et al., 2024a) and HELMET (Yen et al., 2024) have systematically evaluated long-context capabilities, revealing that many models fail to effectively utilize their full context windows. More critically, Lee et al. (2026) showed that contextual distractors can cause catastrophic performance drops of up to 80% in state-of-the-art reasoning models, while Zhang et al. (2025) characterized distractor injection as a distinct vulnerability in large reasoning models. These findings motivate our focus on robustness to hard distractors rather than simply extending context length.

Self-Consistency and Test-Time Compute. Self-consistency (Wang et al., 2022) improves reasoning by sampling multiple decoding trajectories and selecting the most consistent answer through majority voting. This approach has been extended to free-form generation through Universal Self-Consistency (Chen et al., 2023), and recent work on scaling test-time compute (Snell et al., 2024) has shown that inference-time computation can sometimes substitute for model scale. However, Byerly & Khashabi (2024) demonstrated that self-consistency fundamentally fails for long-context problems because positional bias causes all samples to converge to the same biased answer. Our work addresses this limitation by introducing context-level diversity rather than relying solely on decoding diversity.

Positional Bias Mitigation. Several approaches have been proposed to address positional bias in LLMs. Hsieh et al. (2024b) introduced attention calibration to reduce the U-shaped attention bias that causes the lost-in-the-middle phenomenon. Tang et al. (2023) proposed permutation self-consistency for listwise ranking, showing that marginalizing over different list orders reduces positional bias. Yu et al. (2024) demonstrated that scaling a single hidden dimension can mitigate position bias. Gold Panning (Byerly & Khashabi, 2025) leverages positional bias as a diagnostic signal by observing response shifts under document reordering. Our approach is most closely related to permutation-based methods but extends beyond order diversity to include subset diversity through relevance-weighted sampling.

Ensemble Methods for RAG. Retrieval-augmented generation (Lewis et al., 2020) combines parametric and non-parametric memory for knowledge-intensive tasks. Fusion-in-Decoder (Izacard & Grave, 2020) processes multiple retrieved passages independently before fusing them in the decoder. RePlug (Shi et al., 2023) treats the LM as a black box and ensembles predictions across different retrieved document sets. Unlike these approaches that ensemble at the retrieval or model level, our method ensembles at the context level by creating diverse views of the same retrieved passages through subset sampling and order shuffling.

3 Method

3.1 Problem Setup

We consider long-context question answering where the input consists of a question q and a set of retrieved paragraphs $P = \{p_1, \ldots, p_n\}$. The goal is to produce an answer a that correctly responds to q using evidence from P. This setting is common in retrieval-augmented generation (RAG)

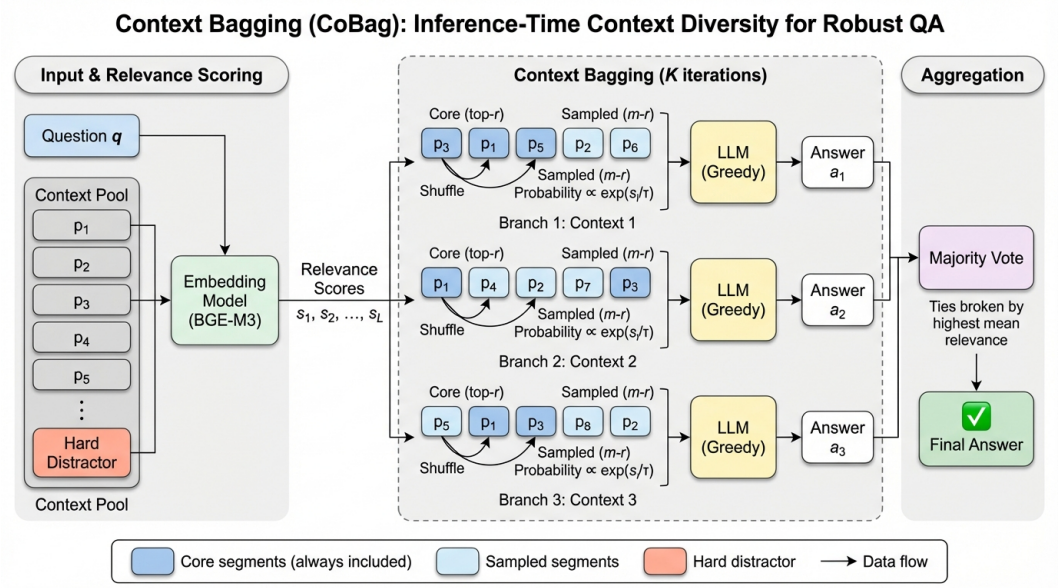

Figure 1: Overview of Context Bagging (CoBag). Given a question and retrieved paragraphs, CoBag scores each paragraph by relevance, samples K diverse context subsets (always including top-r core paragraphs, sampling remaining with probability proportional to relevance scores), randomly permutes paragraph order within each subset, generates answers via greedy decoding, and aggregates via majority voting.

systems (Lewis et al., 2020), multi-hop QA (Yang et al., 2018; Trivedi et al., 2021), and tool-using agents that aggregate intermediate outputs.

The key challenge we address is the presence of **hard distractors**: paragraphs $d \in P$ that are semantically similar to the question and supporting evidence but lead to incorrect answers. Unlike random noise, hard distractors rank highly by relevance metrics and thus survive standard filtering. When placed in salient positions (particularly near the end of the context), they can systematically mislead the model (Liu et al., 2023; Lee et al., 2026).

3.2 Why Self-Consistency Fails

Self-consistency (Wang et al., 2022) samples K decoding trajectories from $p(a|q, C)$ using temperature sampling and selects the most frequent answer. However, when the context C contains misleading information, all K samples are conditioned on the same biased context. If errors are *context-driven* rather than *decoding-driven*, sampling multiple trajectories under the same contaminated context produces correlated errors that converge to the same wrong answer (Byerly & Khashabi, 2024). Our diagnostic analysis confirms this: Self-Consistency exhibits an agreement rate of 0.85 and wrong-answer concentration of 0.82, indicating that when the model errs, it consistently produces the same incorrect answer across samples.

3.3 Context Bagging

We propose **Context Bagging (CoBag)**, an inference-time ensembling method that allocates test-time compute to *context diversity* rather than decoding diversity. The key insight is that if errors are context-driven, we should perturb the context rather than the decoding process.

Given a question q and paragraphs P, CoBag proceeds in four steps (Figure 1):

Step 1: Relevance Scoring. Compute a relevance score s_i for each paragraph p_i using cosine similarity between embeddings:

$$s_i = \cos(\text{embed}(q), \text{embed}(p_i)) \tag{1}$$

We use BGE-M3 (Chen et al., 2024) as the embedding model. This scoring is used only to guide sampling and does not require training.

Step 2: Subset Sampling. For each of K ensemble members, construct an m-paragraph context by: (1) always including the top-r paragraphs by relevance score (the "core set"), and (2) sampling the remaining $m - r$ paragraphs without replacement with probabilities:

$$\Pr(p_i \text{ selected}) \propto \exp(s_i/\tau) \tag{2}$$

where τ is a temperature parameter controlling the sampling distribution's peakedness.

Step 3: Order Shuffling. Randomly permute the order of the m selected paragraphs within each context. This mitigates positional bias (Liu et al., 2023) by varying where each paragraph appears across ensemble members.

Step 4: Majority Voting. Generate an answer a_k for each context C_k using *greedy decoding* (temperature=0), then aggregate via majority vote over normalized answer strings. Ties are broken by selecting the answer from the context with the highest mean relevance score.

3.4 Design Rationale

Each component of CoBag serves a specific purpose. The **core set** ensures that highly relevant paragraphs (likely containing supporting evidence) are always included, preventing the loss of critical information. **Relevance-weighted sampling** biases selection toward informative paragraphs while introducing diversity in which paragraphs co-occur. **Order shuffling** varies the positional context of each paragraph, reducing the impact of positional bias. Finally, **greedy decoding** is essential: it ensures that any improvement comes from context diversity rather than increased output entropy, providing a clean attribution of the method's effectiveness.

Critically, CoBag's improvements are not due to distractor exclusion. By design, hard distractors that rank within the top-r are included in the core set and thus appear in 100% of all contexts. The gains come from diversity in how the context is composed and ordered, not from filtering out problematic content.

4 Experiments

4.1 Experimental Setup

Dataset. We evaluate on the MuSiQue (Trivedi et al., 2021) multi-hop question answering benchmark. We construct a *noisy split* of 2,314 examples, each containing 20 paragraphs with a hard distractor. The hard distractor is defined as the highest-scoring non-supporting paragraph (by embedding similarity) that does not contain the gold answer. We require the distractor to rank within the top-4 by relevance, ensuring it is included in all methods' core sets. The distractor is placed at the end of the paragraph sequence to create a worst-case recency bias condition.

Model and Baselines. We use Qwen2.5-7B-Instruct (Yang et al., 2024) as the base model. We compare against three baselines: (1) **Filtered-Single**: deterministic top-m selection with greedy decoding (no diversity); (2) **Self-Consistency**: fixed context with K temperature-sampled decoding trajectories (T=1.0) and majority voting (decoding diversity); (3) **Permute-Vote**: fixed top-m paragraphs with K random order permutations and greedy decoding (order diversity only).

Metrics and Hyperparameters. We report Exact Match (EM) and token-level F1. Default hyperparameters are K=5, m=12, r=6, τ=1.0. Statistical significance is assessed using McNemar's test on per-example correctness.

4.2 Main Results

Table 1 presents the main comparison. CoBag-Vote achieves the highest EM (0.2939), significantly outperforming both Filtered-Single (+3.03 EM, p=0.0002) and Self-Consistency (+3.12 EM,

Table 1: Main results on MuSiQue noisy split (2,314 examples with hard distractors). CoBag-Vote significantly outperforms Filtered-Single and Self-Consistency (p¡0.001), but not Permute-Vote (p=0.641). Best in **bold**, second-best underlined. †Hard distractor present in 100% of CoBag contexts.

Method	K	Decoding	EM	F1	p-value
Filtered-Single	1	greedy	0.2636	0.3738	0.0002
Self-Consistency	5	T=1.0	0.2627	0.3700	0.0001
Permute-Vote	5	greedy	0.2900	**0.4008**	0.6414
CoBag-Vote†	**5**	**greedy**	**0.2939**	0.4001	—

Table 2: Ablation study isolating contributions of order shuffling and relevance-weighted sampling (K=5, m=12, r=4). Order shuffling is the dominant mechanism (+1.44 EM). Best in **bold**, second-best underlined.

Variant	Order	Rel-Weight	EM	F1
Filtered-Single	—	—	0.2636	0.3738
CoBag-NoShuffle	✗	✓	0.2666	0.3739
CoBag-Uniform	✓	✗	0.2770	0.3833
CoBag-Vote (full)	✓	✓	0.2810	0.3876
Permute-Vote	✓	**—**	**0.2910**	**0.4008**

p=0.0001). This confirms that context-level diversity is far more effective than decoding-level diversity for long-context QA under hard distractors.

Notably, CoBag-Vote does not significantly outperform Permute-Vote (+0.39 EM, p=0.641), suggesting that order diversity captures most of the improvement. The hard distractor is present in 100% of all CoBag contexts (11,570/11,570 at K=5), confirming that improvements are not due to distractor exclusion.

4.3 Ablation Study

Table 2 isolates the contributions of each component. Order shuffling contributes +1.44 EM (CoBag-Vote vs. CoBag-NoShuffle), while relevance-weighted sampling contributes only +0.40 EM (CoBag-Vote vs. CoBag-Uniform). Critically, CoBag-NoShuffle (subset diversity only) underperforms Permute-Vote by -2.44 EM, confirming that order diversity is the dominant mechanism. Subset diversity alone provides minimal benefit and introduces additional variance.

4.4 Sensitivity Analysis

Figure 2 shows the sensitivity to context size m and ensemble size K. Performance improves monotonically with larger contexts (EM: 0.2615 at m=8 $\rightarrow$ 0.2982 at m=16), suggesting that additional context provides more supporting evidence without overwhelming the model. Ensemble size shows diminishing returns: K=3$\rightarrow$5 yields +2.42 EM, while K=5$\rightarrow$7 yields only +0.17 EM. We recommend K=5 as the optimal cost-accuracy trade-off.

4.5 Diagnostic Analysis

Table 3 analyzes answer correlation patterns. Self-Consistency exhibits high agreement (0.85) and wrong-answer concentration (0.82), indicating that when the model errs, all samples converge to the same wrong answer. CoBag-Vote reduces wrong-answer concentration to 0.64 (-23%), producing more diverse error patterns that enable majority voting to correct more errors. This confirms our hypothesis that context diversity decorrelates errors more effectively than decoding diversity.

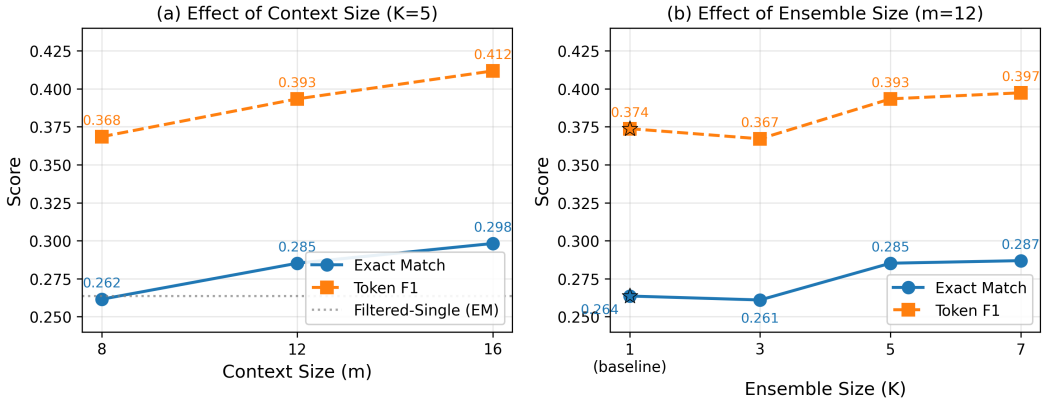

Figure 2: Sensitivity analysis of CoBag-Vote hyperparameters on MuSiQue noisy split. (a) Effect of context size m with K=5 fixed: performance improves monotonically with larger contexts. (b) Effect of ensemble size K with m=12 fixed: diminishing returns beyond K=5, with K=3→5 yielding +2.42 EM but K=5→7 yielding only +0.17 EM.

Table 3: Diagnostic analysis of answer correlation patterns (K=5, m=12). CoBag-Vote produces more diverse error patterns (lower wrong-answer concentration) than Self-Consistency, enabling more effective majority voting. Best (lowest) in **bold**.

Method	Agreement Rate	Wrong-Ans. Conc.	Num Incorrect
Self-Consistency	0.8494	0.8239	1706
Permute-Vote	0.7199	0.6893	1643
CoBag-Vote	**0.6643**	**0.6370**	1654

5 Conclusion

We introduced Context Bagging (CoBag), an inference-time ensembling method that improves robustness to hard distractors in long-context QA by allocating test-time compute to context diversity rather than decoding diversity. CoBag significantly outperforms self-consistency (+3.12 EM, p¡0.001), confirming that context-level perturbation is more effective than sampling multiple decoding trajectories from the same biased context. Our ablation reveals that order diversity is the dominant mechanism, contributing +1.44 EM, while subset diversity provides marginal additional benefit (+0.40 EM). This suggests that simple order shuffling may be sufficient for many applications. Future work should explore settings where subset diversity becomes more important, such as scenarios with more severe noise or different task domains.

References

Adam Byerly and Daniel Khashabi. Self-consistency falls short! the adverse effects of positional bias on long-context problems. 2024.

Adam Byerly and Daniel Khashabi. Gold panning: Strategic context shuffling for needle-in-haystack reasoning. 2025.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. pp. 2318–2335, 2024.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. Universal self-consistency for large language model generation. *ArXiv*, abs/2311.17311, 2023.

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

Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long T. Le, Abhishek Kumar, James Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, and Tomas Pfister. Found in the middle: Calibrating positional attention bias improves long context utilization. pp. 14982–14995, 2024b.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. *ArXiv*, abs/2007.01282, 2020.

Seongyun Lee, Yongrae Jo, Minju Seo, Moontae Lee, and Minjoon Seo. Lost in the noise: How reasoning models fail with contextual distractors. 2026.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, F. Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, M. Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401, 2020.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, F. Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2023.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, M. Lewis, Luke Zettlemoyer, and Wen tau Yih. Replug: Retrieval-augmented black-box language models. pp. 8371–8384, 2023.

C. Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *ArXiv*, abs/2408.03314, 2024.

Raphael Tang, Xinyu Crystina Zhang, Xueguang Ma, Jimmy Lin, and Ferhan Ture. Found in the middle: Permutation self-consistency improves listwise ranking in large language models. *ArXiv*, abs/2310.07712, 2023.

H. Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2021.

Xuezhi Wang, Jason Wei, D. Schuurmans, Quoc Le, Ed H. Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171, 2022.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yi-Chao Zhang, Yunyang Wan, Yuqi Liu, Zeyu Cui, Zhenru Zhang, Zihan Qiu, Shanghaoran Quan, and Zekun Wang. Qwen2.5 technical report. *ArXiv*, abs/2412.15115, 2024.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, R. Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. pp. 2369–2380, 2018.

Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. Helmet: How to evaluate long-context language models effectively and thoroughly. *ArXiv*, abs/2410.02694, 2024.

Yijiong Yu, Huiqiang Jiang, Xufang Luo, Qianhui Wu, Chin-Yew Lin, Dongsheng Li, Yuqing Yang, Yongfeng Huang, and Lili Qiu. Mitigate position bias in large language models via scaling a single dimension. 2024.

Zhehao Zhang, Weijie Xu, Shixian Cui, and Chandan K. Reddy. Distractor injection attacks on large reasoning models: Characterization and defense, 2025. URL https://arxiv.org/abs/2510.16259.