# Beyond Full-Document Contextualization for Contextual Retrieval

**Anonymous ACL submission**

## Abstract

Dense retrieval systems commonly handle long documents by splitting them into short chunks and embedding each chunk independently. However, chunking creates a mismatch between isolated chunk representations and context-dependent relevance. A chunk may contain the correct evidence, but its relevance often depends on surrounding context that resolves entities, events, or discourse relations. When such context is split across chunk boundaries, the correct chunk can become underspecified, while superficially similar but contextually incorrect chunks may appear equally relevant. We refer to this failure mode as **semantic fragmentation**. To systematically evaluate this problem, we introduce **SFR-Bench**, a multi-domain benchmark targeting context-dependent retrieval across structural, discourse-level, entity-centric, and long-range dependency scenarios. We further propose **Progressive Contextualized Window (PCW)**, a token-efficient contextual retrieval method that progressively enriches chunk representations by reconstructing only the necessary neighboring context, rather than contextualizing the entire document. This design preserves cross-chunk dependencies while substantially reducing contextual generation overhead. In addition, we develop a **retrieval-oriented optimization framework** that combines similarity-guided filtering with GRPO-based reinforcement learning to align generated context with downstream retrieval objectives. Experiments across SFR-Bench and diverse retrieval benchmarks show that PCW achieves a stronger cost-performance trade-off than full-document contextualization, maintaining or improving retrieval effectiveness while significantly reducing contextual generation cost.

## 1 Introduction

Information retrieval (IR) plays a critical role in modern AI systems, supporting applications such as **semantic search** (Karpukhin et al., 2020) and **Retrieval-Augmented Generation (RAG)** (Lewis et al., 2020). In many real-world scenarios, retrieval systems must process long documents that exceed the input limits of dense retrieval encoders (Chen et al., 2024). To accommodate these constraints, documents are typically partitioned into smaller chunks, each of which is encoded independently into dense representations. However, this chunking strategy can separate semantically dependent information across chunk boundaries, causing individual chunks to lose critical contextual signals. As illustrated in Figure 1, entity references may become detached from their semantic anchors after chunking, making multiple chunks appear semantically similar despite referring to different entities. As a result, retrieval systems may fail to identify the target chunk when essential semantic evidence falls outside the local chunk scope.

To mitigate semantic fragmentation, prior work has explored approaches that enhance contextual modeling during embedding generation. Representative methods include late chunking (Günther et al., 2024), which postpones chunk partitioning until after contextual encoding, and context-aware embedding optimization (Conti et al., 2025), which improves the ability of chunk representations to capture surrounding semantics. While these methods provide a mechanism for incorporating information beyond the local chunk scope during embedding generation, they may also introduce **biases** that traditional embedding models do not typically encounter, potentially affecting generalization across diverse retrieval scenarios.

Recent Contextual Retrieval (Anthropic, 2024) method further augment chunk representations with LLM-generated document-level contextual descriptions prior to embedding generation. While such approaches alleviate semantic fragmentation, they often require repeatedly conditioning on the full

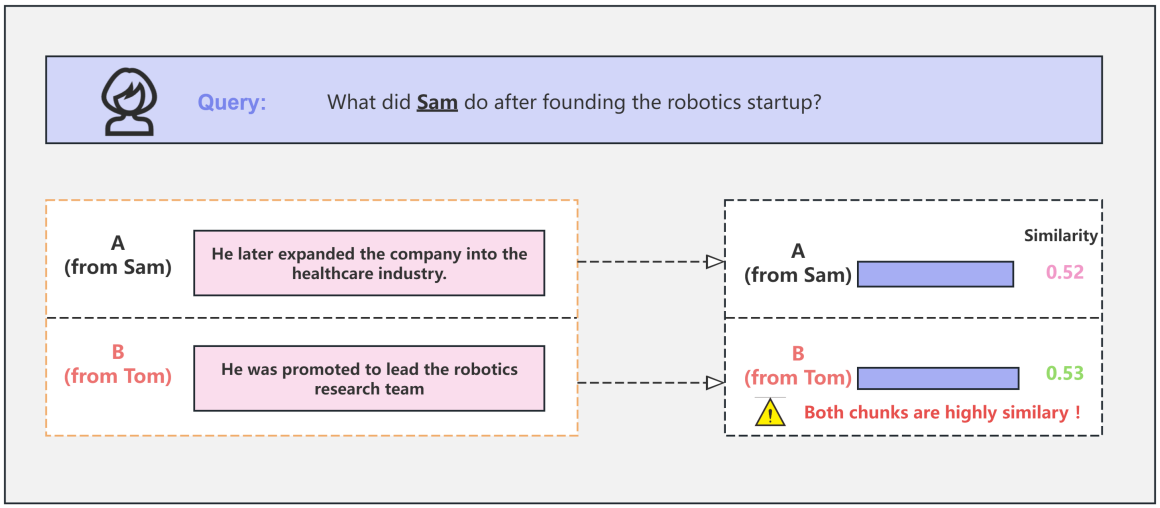


Figure 1: Illustration of semantic fragmentation caused by document chunking. After chunking, entity anchors are separated from local chunks, leaving multiple semantically similar chunks that contain only pronouns such as "he". As a result, embedding similarity alone may fail to correctly associate the query entity with the target chunk.

document during contextual generation, leading to substantial token consumption for long documents.

The development of contextual enhancement methods is also limited by the lack of benchmarks that accurately reflect context-dependent retrieval scenarios. Although benchmarks such as MTEB (Muennighoff et al., 2023) and BEIR (Thakur et al., 2021) have substantially advanced retrieval evaluation, they primarily rely on self-contained retrieval settings where relevant information is locally available within individual chunks, making the evaluation of semantic fragmentation inherently limited.

In this work, we focus on more token-efficient contextual augmentation within contextual retrieval frameworks.

To this end, we first construct **SFR-Bench**, a benchmark specifically designed to evaluate whether contextual retrieval methods can effectively integrate non-local contextual information under semantic fragmentation.

Building upon this benchmark, we propose **Progressive Contextualized Window (PCW)**, an improved contextual retrieval framework that progressively propagates contextual information across neighboring chunks using local contextual windows. Unlike prior contextual retrieval methods that generate chunk context from the full source document, PCW performs contextual augmentation locally, substantially reducing contextual generation cost while preserving retrieval effectiveness.

Furthermore, we introduce a retrieval-aligned optimization framework based on GRPO to improve the **discriminability** of generated contextual information under hard-negative retrieval settings.

Our primary contributions are summarized as follows:

- **SFR-Bench**, a benchmark for evaluating retrieval robustness under semantic fragmentation.
- **PCW**, a token-efficient contextual retrieval framework based on progressive contextual propagation.
- **A GRPO-based retrieval optimization framework** for improve the discriminability of generated contextual information under hard-negative retrieval settings.

## 2 Problem Definition & Related Work

### 2.1 Chunk-based Dense Retrieval

Modern retrieval systems typically adopt a chunk-based dense retrieval pipeline for long-form documents. Given a document corpus

$$\mathcal{D} = \{D_1, D_2, \ldots, D_m\},$$

each document is partitioned into smaller text chunks:

$$D = \{c_1, c_2, \ldots, c_n\},$$

where each chunk satisfies the maximum input length constraint of the embedding model.

Each chunk is independently encoded into a dense vector representation through an embedding encoder $\phi$:

$$v_i = \phi(c_i), \quad v_i \in \mathbb{R}^d.$$

During retrieval, a user query $q$ is similarly embedded:

$$v_q = \phi(q),$$

and retrieval is performed via nearest-neighbor similarity search:

$$s(q, c_i) = \text{sim}(v_q, v_i),$$

where $\text{sim}(\cdot)$ denotes cosine similarity or inner product similarity.

Recent **Contextual Retrieval** methods extend this paradigm by augmenting each chunk with additional document-level context prior to embedding generation. Instead of directly encoding the isolated chunk $c_i$, these approaches first generate contextual information

$$context_i = \mathcal{G}(c_i, \mathcal{D}),$$

which is concatenated with the original chunk:

$$\tilde{c}_i = [context_i \oplus c_i].$$

The contextualized chunk is then encoded as

$$v_i = \phi(\tilde{c}_i).$$

This modification aims to alleviate semantic fragmentation by incorporating contextual signals beyond the local chunk scope. However, contextual retrieval methods that generate chunk-specific context by conditioning on the full source document incur substantial **token consumption** and **inference cost**. We refer to this full-document contextual retrieval paradigm as **Full-Document Contextualization (FDC)**.

### 2.2 Mitigating Semantic Fragmentation

Document chunking enables scalable retrieval over long-form corpora, but it may separate chunks from the contextual information required for accurate semantic interpretation. To address this issue, prior work has explored several approaches for mitigating semantic fragmentation.

Early efforts focused on semantic-aware chunking strategies (Kiss et al., 2025; Qu et al., 2025) that preserve coherent semantic units during document partitioning. More recent methods enhance contextual awareness during embedding generation.Approaches such as LongEmbed (Zhu et al., 2024), Late Chunking (Günther et al., 2024), and InSeNT (Conti et al., 2025) improve dependency modeling beyond local chunk boundaries through long-context encoding or context-aware representation learning, but may also introduce contextual biases that are less commonly encountered in traditional embedding models.In parallel, contextual augmentation methods such as Contextual Retrieval (Anthropic, 2024) prepend LLM-generated contextual descriptions to chunks before indexing. Unlike context-aware representation learning approaches, these methods avoid directly altering the embedding model itself, but often rely on full-document contextualization, leading to substantial token consumption for long documents.

### 2.3 Context-Dependent Retrieval Benchmarks

Most existing retrieval benchmarks primarily evaluate self-contained retrieval scenarios, where relevant semantic evidence is locally available within individual chunks. Recent efforts such as ConTEB (Conti et al., 2025) attempt to introduce context-aware evaluation settings. However, these benchmarks remain limited by labor-intensive construction and benchmark corpora that are not entirely constructed around strongly context-dependent retrieval scenarios.

## 3 SFR-Bench: Semantic Fragmentation Retrieval Benchmark

Existing retrieval benchmarks largely assume that document chunks are independently interpretable(Chen et al., 2025), where queries are typically constructed to be answerable using information contained within a single chunk.. However, real-world chunking frequently breaks semantic dependencies such as coreference, discourse continuity, and long-range topical grounding, causing retrieval to depend on contextual information beyond isolated chunks. We refer to this retrieval failure mode as *semantic fragmentation*. To evaluate retrieval robustness under such context-deprived settings, we construct a benchmark grounded in naturally context-dependent corpora. Since large-scale manual annotation for contextual retrieval is both labor-intensive and difficult to scale, recent studies have shown that LLM-based synthesis can produce reliable retrieval benchmarks when combined with appropriate filtering and verifica-

| Dataset | Statistics | | Avg. Length | | Document Info | |
|---|---|---|---|---|---|---|
| | #Queries | #Chunks | Query | Chunk | Avg. Doc Len | #Docs |
| Actor | 4330 | 9048 | 13 | 123 | 2237 | 499 |
| Bird | 1505 | 3177 | 14 | 122 | 1938 | 201 |
| Financial | 106 | 561 | 19 | 118 | 629 | 106 |
| Paper | 563 | 2814 | 19 | 178 | 2134 | 236 |
| Dialogue | 612 | 4318 | 14 | 141 | 2989 | 204 |
| State | 254 | 531 | 12 | 125 | 66800 | 1 |

Table 1: Statistics of the proposed semantic fragmentation benchmark across six domains. Query and chunk lengths are reported in average token counts.

tion mechanisms (Bonifacio et al., 2022; Dai et al., 2022; Cui et al., 2023). The strong semantic understanding and controllable generation capabilities of modern LLMs(Achiam et al., 2023) further make scalable synthetic benchmark construction increasingly feasible. Motivated by these findings, we leverage an LLM-based synthesis pipeline to construct our benchmark.

### 3.1 Dependency-aware Corpus Curation

To evaluate retrieval robustness under semantic fragmentation, we curate naturally context-dependent corpora spanning multiple dependency regimes.

- **Financial & Dialogue:** Financial reports[1] and dialogue corpora[2] contain strong attribution and discourse dependencies, where localized evidence relies on globally established entities or conversational history.
- **Paper:** Academic papers[3] exhibit structural dependencies between tables, captions, and explanatory paragraphs. We further amplify this fragmentation by separating related structural components during chunking.
- **Actor & Bird:** Entity-centric corpora from Wikipedia[4] contain dense coreference dependencies requiring surrounding document context for entity resolution.
- **State:** We construct an long corpus exceeding 60k tokens by concatenating multiple entity-centered documents, creating long-range retrieval scenarios with substantial contextual interference.

By grounding the benchmark in naturally occurring contextual dependencies, retrieval difficulty shifts from shallow semantic matching toward contextual disambiguation under semantic fragmentation.

### 3.2 Context-Amplified Query Synthesis

The objective of query synthesis is not to artificially create retrieval difficulty, but to expose latent contextual dependencies already present in the source documents. To achieve this, we use GPT-5 to synthesize queries grounded in document-level semantics while omitting identifiers that can be directly matched from isolated chunks.

We construct query–chunk pairs using two complementary strategies corresponding to different dependency regimes.

For corpora that naturally exhibit strong **contextual dependency**, including Financial and Dialogue, queries are first synthesized from the full-document perspective using global document semantics. The documents are then chunked using the sentence-based splitter provided by LlamaIndex[5]. and an independent LLM identifies the positive chunk $c_{pos}$ containing the evidence necessary to answer the query. This setting reflects realistic retrieval scenarios where queries are formulated with respect to globally coherent documents while retrieval operates on fragmented local chunks.

For corpora with stronger **entity-centric or structural dependency**, including Actor, Bird, and Paper, we instead begin from candidate chunks and apply lightweight heuristic filtering to retain only semantically incomplete chunks lacking sufficient local contextual anchors. Queries are then synthesized specifically for these filtered chunks, ensuring

[1] https://huggingface.co/datasets/dreamerdeo/finqa
[2] https://www.kaggle.com/datasets/shuyangli94/interview-npr-media-dialog-transcripts/data?select=utterances-2sp.csv
[3] https://huggingface.co/datasets/kasnerz/scigen
[4] https://www.wikimedia.org/
[5] https://github.com/run-llama/llama_index

that successful retrieval requires recovering missing entity or structural context beyond the isolated chunk itself.

The State corpus is constructed using the same chunk filtering and query synthesis procedure as the Actor and Bird corpora. To further amplify long-range dependency, we additionally concatenate multiple entity-centered documents into ultra-long sequences, introducing substantial contextual interference across distant document regions.

Together, these strategies produce retrieval settings in which successful retrieval requires contextual reconstruction beyond isolated chunk semantics.

**Quality Filtering.** Following the quality control philosophy adopted in AIR-Bench (Chen et al., 2025), we apply an additional verification stage to ensure that retrieval difficulty originates from contextual insufficiency rather than arbitrary query obfuscation.

For each synthesized query–document pair, an independent LLM evaluates whether the designated positive chunk $c_{pos}$ remains factually relevant and answerable under full contextual information. Samples failing this verification are discarded.

This filtering stage ensures that the benchmark evaluates contextual dependency instead of general retrieval difficulty introduced by noisy synthesis artifacts.

We also include **AirBench** as a standard self-contained retrieval benchmark, allowing us to evaluate whether contextualized retrieval methods maintain performance on self-contained retrieval scenarios.

# 4 Methodology

## 4.1 Progressive Contextualized Window

While Full-Document Contextualization (FDC) effectively alleviates semantic fragmentation, it incurs substantial contextual generation cost for long documents. A natural token-efficient alternative is **Naive Sliding Window (NSW)** contextualization using neighboring chunks within a fixed local window:

$$context_i = \mathcal{G}(c_i, c_{i-k}, \ldots, c_{i+m}),$$

which avoids repeatedly processing the entire document.

However, NSW restricts contextualization within a **fixed local window**, preventing chunks from capturing dependencies beyond the window boundary. As a result, semantically relevant information outside the accessible window cannot participate in downstream contextualization.

To alleviate this limitation, we propose **Progressive Contextualized Window (PCW)**, which recursively reuses previously **contextualized neighboring chunks** during contextual generation. Unlike NSW, which only conditions on raw local chunks, contextual information introduced in earlier windows may reappear in later contextualization steps, allowing dependencies to propagate beyond the original local window.A detailed example illustrating long-range contextual propagation under recursive contextualization is provided in Appendix C.3.

Formally, PCW recursively contextualizes each chunk using previously contextualized neighboring representations:

$$context_i^{PCW} = \mathcal{G}(c_i, e_{i-k}, ..., e_{i-1}, c_{i+1}, ..., c_{i+m}),$$

where

$$e_j = [context_j^{PCW} \oplus c_j]$$

denotes the contextualized representation of chunk $c_j$.

## 4.2 Retrieval-oriented Optimization

Although PCW improves long-range contextual propagation, semantically coherent contextualization does not necessarily imply retrieval utility. Context generated by language models may remain linguistically plausible while failing to maximize retrieval discriminability in embedding space.

To better align contextual generation with downstream retrieval objectives, we introduce a two-stage retrieval-oriented optimization framework consisting of supervised fine-tuning with similarity-based rejection sampling and retrieval-aware reinforcement learning.

### 4.2.1 Similarity-based Rejection Sampling

Directly using randomly generated contextual descriptions for supervised fine-tuning (SFT) (Ouyang et al., 2022) often introduces semantically noisy or retrieval-irrelevant supervision signals.Our training corpora are constructed from publicly available web documents, including long-form novels, financial reports, and dialogue data. Queries are synthesized using the same contextual dependency construction strategy as SFR-Bench. Intuitively, contextual descriptions that improve

query-context similarity after augmentation are more likely to provide informative retrieval signals, as embedding similarity strongly correlates with retrieval effectiveness in dense retrieval systems (Li et al., 2025). We therefore construct high-quality SFT data through similarity-based rejection sampling.

For each chunk, we sample multiple contextual candidates from the Qwen3-235B-A22B generator (Yang et al., 2025):

$$\{context_i^{(1)}, context_i^{(2)}, \ldots, context_i^{(N)}\}.$$

Each candidate is concatenated with the original chunk:

$$e_i^{(j)} = [context_i^{(j)} \oplus c_i],$$

and evaluated using query-document similarity in embedding space.

We retain contextual candidates that maximize retrieval similarity while filtering semantically noisy generations:

$$context_i^* = \arg \max_{context_i^{(j)}} \mathrm{sim}(\phi(q), \phi(e_i^{(j)})).$$

This objective encourages the model to learn contextual descriptions that improve retrieval-oriented semantic alignment rather than purely linguistic fluency.

#### 4.2.2 Retrieval-aligned Policy Optimization

While contextual augmentation improves query-chunk semantic alignment, it may also increase similarity among neighboring or semantically related chunks, reducing **embedding-space discriminability** between positive chunks and hard negatives.

To mitigate representation homogenization introduced by recursive contextual propagation, we formulate contextual generation as a discriminability-preserving policy optimization problem. Specifically, we adapt GRPO (Shao et al., 2024) with retrieval-aware rewards that encourage positive-negative separability while preserving inter-chunk discriminability in embedding space.

For each query $q$, we construct a candidate set consisting of one positive chunk $c^+$ and $N$ hard negative chunks:

$$\mathcal{S}_q = \{c^+\} \cup \{c_1^-, c_2^-, \ldots, c_N^-\}.$$

During each rollout, the policy model generates contexts for all chunks in $\mathcal{S}_q$:

$$context_j \sim \pi_\theta(\cdot \mid x_j), \quad c_j \in \mathcal{S}_q,$$

and constructs contextualized chunks:

$$e_j = [context_j \oplus c_j].$$

We then encode the query and all contextualized candidates, compute retrieval scores, and rank the candidate set:

$$s(q, e_j) = \mathrm{sim}(\phi(q), \phi(e_j)).$$

To mitigate *reward hacking* behaviors such as excessive query copying or uncontrolled context expansion, we further introduce retrieval regularization penalties into the reward formulation. The final retrieval reward is defined as:

$$\begin{aligned}\tilde{r}(q) = \mathrm{nDCG@10}(q, \mathcal{S}_q) \\ - \alpha \cdot P_{\mathrm{copy}} \\ - \mathbb{I}_{\bar{L} > L_{\mathrm{tol}}} \cdot \lambda \tanh\left(\frac{\bar{L} - L_{\mathrm{tol}}}{\tau}\right).\end{aligned}$$

where

$$P_{\mathrm{copy}} = \frac{|\mathrm{Tok}(context) \cap \mathrm{Tok}(q)|}{|\mathrm{Tok}(q)|},$$

denotes the lexical overlap ratio between the generated contextual descriptions and the query, and the length regularization term penalizes excessively verbose contextual generations beyond the tolerance threshold $L_{\mathrm{tol}}$.

Following GRPO, multiple rollouts are sampled for the same query, and rewards are normalized within the rollout group to obtain relative advantages:

$$A^{(k)} = \frac{\tilde{r}^{(k)} - \mathrm{mean}(\{\tilde{r}^{(l)}\}_{l=1}^K)}{\mathrm{std}(\{\tilde{r}^{(l)}\}_{l=1}^K) + \epsilon}.$$

The generator is optimized to increase the likelihood of rollouts that produce higher retrieval rewards:

$$\begin{aligned}\mathcal{L}_{\mathrm{GRPO}} = -\frac{1}{K}\sum_{k=1}^{K} A^{(k)} \\ \cdot \sum_{c_j \in \mathcal{S}_q} \log \pi_\theta\left(context_j^{(k)} \mid x_j\right).\end{aligned}$$

This objective directly rewards contextual generations that rank the positive chunk above hard negatives, thereby aligning context generation with retrieval discriminability rather than linguistic plausibility alone.

## 5 Results and Analysis

### 5.1 Main Results on Heterogeneous Benchmark

We evaluate the effectiveness of our **PCW** method against existing contextualization paradigms. Specifically, we compare **FDC** (full-document contextualization), **NSW** (raw sliding window contextualization), and our **PCW** (Progressive Contextualized Window). For fair comparison, all contextual descriptions are generated using **Qwen3-4B**.

As shown in Table 2, PCW consistently outperforms both global and static local contextualization baselines across most context-dependent retrieval scenarios.

The experimental results yield several key observations:

- **Limitations of Naive Windowing:** Although *NSW* reduces contextual augmentation cost, fixed local windows fail to capture dependencies beyond the accessible context, resulting in inferior retrieval performance.
- **Effectiveness of Progressive Propagation:** *PCW* achieves comparable or stronger retrieval performance while substantially reducing token consumption. Through recursive propagation, PCW captures long-range dependencies more effectively than static local windowing without requiring full-document conditioning.
- **Benefits of Retrieval-aligned Optimization:** Reinforcement learning further improves retrieval performance, suggesting that retrieval-aligned optimization helps contextual augmentation produce more discriminative semantic representations.

### 5.2 Robustness Across Embedding Models

We further evaluate whether PCW generalizes across different embedding architectures. Results show that PCW consistently outperforms the naive sliding-window baseline (NSW) across all tested retrievers, including BGE-m3, Qwen3-embedding-0.6B, and mE5-large (Appendix Table 5).

The consistent improvements suggest that progressive contextual propagation improves retrieval robustness beyond model-specific embedding characteristics, indicating that PCW serves as a generally effective contextual retrieval strategy rather than a retriever-specific optimization.

### 5.3 Ablation Study

To verify the contribution of each training stage, we evaluate the performance of **PCW** across different configurations. As shown in Table 3, while the **Zero-shot** baseline establishes the feasibility of recursive propagation, **Stage 1 (SFT)** provides a substantial performance uplift by stabilizing semantic coherence. The transition to **Stage 2 (GRPO)** further refines the results, demonstrating that aligning the generator with retrieval metrics via contrastive rewards is essential for producing highly discriminative context.

### 5.4 Query-copy Regularization Analysis

As shown in Table 4, we evaluate reward regularization on both the semantic fragmentation benchmark and the self-contained retrieval benchmark. Although GRPO without regularization improves retrieval performance on SFR-Bench, it degrades performance on the self-contained benchmark, indicating reduced generalization of the generated contextual descriptions.

This behavior suggests that retrieval optimization may become biased toward fragmentation-heavy settings, where retrieval failures are primarily caused by missing contextual anchors. Under such scenarios, injecting highly retrievable lexical cues, such as entity mentions or recurring semantic anchors, can partially compensate for missing dependencies and improve retrieval ranking. However, such shortcut-oriented contextualization generalizes poorly to conventional self-contained retrieval, where semantic completeness is already preserved within individual chunks.

By suppressing excessive lexical overlap during training, the proposed regularization encourages more semantically grounded contextualization and improves overall retrieval generalization.

### 5.5 Parameter Sensitivity

We evaluate the impact of **window size** on the precision-efficiency trade-off. As shown in Figure 2, increasing window size generally improves $nDCG@10$ by capturing broader context. Notably, our **PCW** exhibits superior scalability over **NSW**, and even surpasses **FDC** when the window size increases to a moderate range. This suggests that progressive recursive contextualization propagates long-range dependencies more effectively than static local windowing. We select an optimized window size to balance semantic fidelity

| Method | Fragmentation Benchmarks | | | | | | | General Retrieval | |
|---|---|---|---|---|---|---|---|---|---|
| | Structural | Contextual | | Entity Dep. | | Long-range | Avg. | Self-contained | Cost↓ |
| | Paper | Financial | Dialogue | Actor | Bird | State | | AirBench | |
| BM25 | 49.10 | 26.07 | 43.22 | 27.09 | 39.79 | 60.51 | 40.96 | 54.81 | – |
| No Context | 50.04 | 21.65 | 49.49 | 26.71 | 43.33 | 68.40 | 42.27 | 57.42 | – |
| FDC | 71.74 | 37.91 | 54.31 | 44.64 | 67.23 | 59.67 | 55.92 | 57.35 | 1.00× |
| NSW | 71.49 | 31.03 | 52.52 | 31.72 | 49.46 | 68.47 | 50.78 | 56.14 | 0.20× |
| **PCW** | 70.43 | 39.25 | 54.34 | 44.32 | 68.35 | 78.28 | 59.18 | 58.40 | 0.27× |
| **PCW + SFT** | 73.32 | 42.83 | **55.61** | 59.46 | 72.84 | 80.71 | 64.13 | **58.98** | 0.27× |
| **PCW + SFT + RL** | **77.16** | **49.78** | 52.38 | **66.68** | **78.29** | **82.01** | **67.72** | 58.23 | 0.27× |

Table 2: Main retrieval results ($nDCG@10$) using BGE-m3. Datasets are grouped by dependency type. Frag. Avg. reports the average performance on our semantic fragmentation benchmark, while AirBench evaluates standard self-contained retrieval. Cost measures relative **token consumption** during contextual generation, normalized to FDC. FDC denotes full-document contextualization, NSW denotes Naive Sliding Window contextualization, and PCW denotes Progressive Contextualized Window.

| Configuration | Avg. $nDCG@10$ |
|---|---|
| **PCW (Full Framework)** | **67.72** |
| w/o GRPO | 64.13 |
| w/o SFT | 64.35 |

Table 3: Ablation study on training stages.

| Configuration | SFBench | AirBench |
|---|---|---|
| PCW + SFT | 64.13 | 58.98 |
| w/o Query-copy Penalty | 65.89 | 56.07 |
| w/ Query-copy Penalty | **67.72** | 58.23 |

Table 4: Effect of query-overlap regularization on retrieval generalization under retrieval-oriented reinforcement learning.

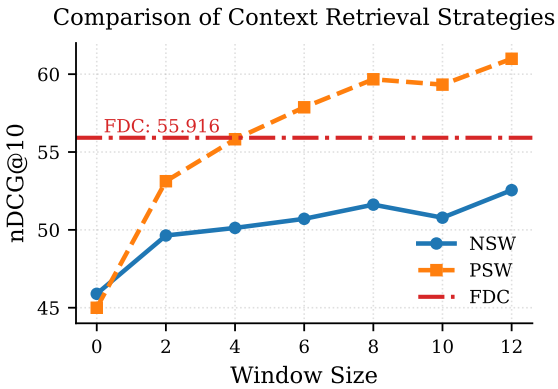


Figure 2: Performance vs. window size. PCW consistently outperforms NSW and exceeds the No Window baseline.

with indexing throughput.

## 6 Conclusion

In this work, we study retrieval robustness under semantic fragmentation caused by document chunking and construct **SFR-Bench**, a multi-domain benchmark covering diverse context-dependent retrieval scenarios.

We further propose **Progressive Contextualized Window (PCW)**, a token-efficient contextual retrieval mechanism that progressively reconstructs contextual dependencies through recursively contextualized local windows. Compared with Full-Document Contextualization (FDC), PCW substantially reduces contextual generation cost while maintaining strong retrieval performance.

To better align contextual generation with downstream retrieval objectives, we additionally introduce a retrieval-aligned optimization framework based on similarity-guided supervision and GRPO-based policy optimization, further improving positive-negative discriminability in embedding space.

Overall, our findings suggest that progressively contextualized local retrieval, combined with retrieval-aligned optimization, provides an effective and scalable alternative to full-document contextualization for long-document retrieval systems.

## Limitations

**Long-context Evaluation.** In addition, while the benchmark captures a range of naturally occurring contextual dependencies under semantic fragmentation, the current evaluation setting does not yet extend to ultra-long document scenarios exceeding 100k tokens. Evaluating contextual retrieval robustness under substantially longer contexts and more

complex cross-document dependencies remains an important direction for future work.

**Downstream Evaluation.** Our evaluation primarily focuses on retrieval quality rather than downstream generation performance. Future work may further investigate how semantic fragmentation affects end-to-end RAG systems.

## References


Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anthropic. 2024. Introducing contextual retrieval. https://www.anthropic.com/news/contextual-retrieval. Accessed: 2025-05-19.

Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. Inpars: Data augmentation for information retrieval using large language models. *arXiv preprint arXiv:2202.05144*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*, 4(5).

Jianlyu Chen, Nan Wang, Chaofan Li, Bo Wang, Shitao Xiao, Han Xiao, Hao Liao, Defu Lian, and Zheng Liu. 2025. Air-bench: Automated heterogeneous information retrieval benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19991–20022.

Max Conti, Manuel Faysse, Gautier Viaud, Antoine Bosselut, Céline Hudelot, and Pierre Colombo. 2025. Context is gold to find the gold passage: Evaluating and training contextual document embeddings. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22605–22619.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback.

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B Hall, and Ming-Wei Chang. 2022. Promptagator: Few-shot dense retrieval from 8 examples. *arXiv preprint arXiv:2209.11755*.

Michael Günther, Isabelle Mohr, Daniel James Williams, Bo Wang, and Han Xiao. 2024. Late chunking: contextual chunk embeddings using long-context embedding models. *arXiv preprint arXiv:2409.04701*.

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.

Csaba Kiss, Marcell Nagy, and Péter Szilágyi. 2025. Max–min semantic chunking of documents for rag application. *Discover Computing*, 28(1):117.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Chaofan Li, Jianlyu Chen, Yingxia Shao, Chaozhuo Li, Quanqing Xu, Defu Lian, and Zheng Liu. 2025. Reinforced ir: A self-boosting framework for domain-adapted information retrieval. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22061–22073.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. Mteb: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Renyi Qu, Ruixuan Tu, and Forrest Bao. 2025. Is semantic chunking worth the computational cost? In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 2155–2177.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Dawei Zhu, Liang Wang, Nan Yang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. 2024. Longembed:

Extending embedding models for long context retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 802–816.

# A Benchmark Construction Details

## A.1 Financial & Dialogue Corpora

For the Financial corpus, we directly use the original human-written queries from the source dataset. For the Dialogue corpus, queries are synthesized from the full dialogue context using an LLM. Specifically, we employ the following prompt template for query generation:

```
Here is a dialogue:

{{dialogue}}

Generate {{num_questions}}
retrieval-oriented questions based on the
dialogue.

Requirements:
- Each question should target a specific
fact mentioned by only one speaker.
- The speaker name should appear naturally
in the question, but not as the first token.
- The question length should be
approximately {{query_length}}.
```

After chunking, an independent LLM identifies the positive chunk $c_{pos}$ containing the evidence necessary to answer each query.The corresponding prompt for positive chunk identification is given below:

```
Here are consecutive, ordered chunks from a
{{corpus}}.
Each chunk is labeled with a 0-based index.

Chunks:
{{paragraphs}}

Question:
{{question}}

Task:
- Select the single chunk that can most
directly answer the question.

Output:
Only the index before the paragraph (an
integer).
Do not output anything else.
```

## A.2 Paper Corpus

Academic papers exhibit strong structural dependency between tables, captions, and explanatory paragraphs. To amplify semantic fragmentation, we intentionally separate these structurally related components during chunk construction using LLM. As a result, many chunks contain only partial structural evidence, requiring retrieval systems to recover missing contextual relations beyond isolated chunk semantics.

Queries are synthesized from the global document context, while positive chunks are identified after chunking using the same procedure described previously.

## A.3 Actor & Bird Corpora

The Actor and Bird corpora exhibit strong entity-centric dependency, where many chunks contain implicit references whose semantic interpretation depends on globally established entity context. To amplify semantic fragmentation, we first filter candidate chunks and retain only semantically incomplete chunks lacking sufficient local entity anchors.

The following prompt is used for chunk filtering:

```
Here is a chunk of text:

{{chunk}}

Task:
Using only the information in this chunk,
decide whether you can clearly identify which
specific person this text is describing.

If you cannot tell who the person is, answer
"no".
If the person is explicitly named, answer
"yes".

Do not include any explanations or
additional text outside the output.
```

Queries are then synthesized specifically for the filtered chunks to ensure that successful retrieval requires recovering missing entity context beyond isolated chunk semantics.

The following prompt is used for query synthesis:

```
Here is a chunk of a Wikipedia article for a
US actor named {{name}}:

{{chunk}}

Generate a question about the chunk with a
length of {{length}}.

Requirements:
- The question can be answered by the chunk.
- The question must mention the name of the
actor.

Your output should be a string without any
other text.
```

### A.4 State Corpus

The State corpus introduces long-range structural dependency by concatenating multiple documents into ultra-long sequences, creating substantial contextual interference across distant document regions. Queries are synthesized from filtered target chunks in the same manner as the Actor and Bird corpora, while retrieval requires recovering entity context across much longer document spans.

### A.5 AirBench

We additionally evaluate retrieval generalization on **AirBench**, a standard self-contained retrieval benchmark. Since the original AirBench release does not provide the complete source documents together with the original chunk ordering, we reconstruct the benchmark following the same construction procedure and using the same long-document corpora described in AirBench. Unlike the original AirBench construction, we preserve all chunks in the reconstructed corpus. As a result, our reproduced evaluation setting may differ from the officially reported AirBench results.

## B Cross-Model Generalization

We additionally evaluate PCW across different embedding architectures to verify that the observed improvements are not tied to a specific retriever family.

| Retriever | NSW | PCW (Ours) | Relative Gain |
|---|---|---|---|
| BGE-m3 | 50.78 | 59.18 | +16.5% |
| Qwen3-0.6B | 49.55 | 57.03 | +15.1% |
| mE5-large | 39.13 | 52.78 | +34.9% |

Table 5: Cross-model performance comparison measured by average $nDCG@10$ across six domains. Relative Gain indicates the improvement of PCW over the naive sliding-window baseline (NSW).

Results show that PCW consistently improves retrieval performance across all evaluated embedding architectures, suggesting that progressive contextual propagation provides a generally effective contextual retrieval strategy rather than a retriever-specific optimization.

## C Method Details

### C.1 Window size

Unless otherwise specified, we use a window configuration with $k_p = 10$ preceding chunks and $k_s = 5$ succeeding chunks. This setting is chosen to balance contextual generation quality and token consumption.signals.

### C.2 Context Concatenation

To avoid ambiguity between original chunk content and generated contextual descriptions during recursive contextual propagation, we explicitly separate the two components using bracket-style concatenation. Specifically, the generated contextual summary is appended as an auxiliary contextual suffix, while the original chunk content is preserved in its original form. This structured formatting helps the generator distinguish intrinsic document semantics from externally generated contextual augmentation when recursively contextualized chunks are reused in subsequent propagation windows. By maintaining an explicit boundary between generated context and source content, our design stabilizes recursive contextual propagation while preserving the semantic identity of individual chunks.

An example is shown below:

```
His third Emmy win was for Guest Actor in a
Drama Series,for an episode of *Picket
Fences*, in which he had a recurring roleas
the father of main character Jill Brock
(Kathy Baker).He also received an Emmy
nomination for portraying Chief JusticeEarl
Warren in the 1991 miniseries *Separate but
Equal*dramatizing *Brown vs. Board of
Education*.[This chunk discusses Richard
Kiley's third Emmy winfor his role in
*Picket Fences* and his Emmy nominationfor
his portrayal of Chief Justice Earl Warren
in the miniseries *Separate but Equal*.]
```

### C.3 Long-range Dependency Propagation Analysis

The following example illustrates progressive semantic propagation under long-range dependency. Although the contextual window of the positive chunk does not directly include Chunk 21, which contains the entity anchor “Richard Kiley”, the entity information is progressively propagated through recursively contextualized neighboring chunks. Consequently, the final contextualized representation successfully restores the missing entity reference and resolves the semantic dependency required for retrieval.

**Chunk 21**

Richard Kiley was an American actor known for stage, film, and television roles.

[The chunk introduces Richard Kiley, an American actor known for stage, film, and television roles.]

**Recursive Contextual Window**

**Chunk 22**

In the 1991 miniseries *Separate but Equal*, he portrayed Chief Justice Earl Warren.

[The chunk describes Richard Kiley portraying Chief Justice Earl Warren in *Separate but Equal*.]

**Chunk 23**

The role centered on the Supreme Court's handling of *Brown v. Board of Education*.

[The chunk discusses Richard Kiley's role in *Separate but Equal* concerning *Brown v. Board of Education*.]

**Chunk 24**

Critics praised the performance for its restraint and judicial authority.

[The chunk discusses critics praising Richard Kiley's performance in *Separate but Equal*.]

**Chunk 25 (Positive Chunk)**

For this performance, he received an Emmy nomination for Outstanding Lead Actor.

[The chunk states that Richard Kiley received an Emmy nomination for **Outstanding Lead Actor** for his performance in *Separate but Equal*.]