Document Chunking Strategy Matters: When Structure-Aware Preprocessing Helps Dense Retrieval

Stephen Soady (ORCID: 0009-0001-3029-4631) Independent Researcher, Brisbane, Australia steve@clawsome.dev

---

Abstract

Document chunking is a critical preprocessing step in retrieval-augmented generation (RAG) systems that has received limited empirical evaluation. We present a systematic empirical analysis of adaptive structure-aware chunking across four BEIR domains, six embedding models, and five chunk sizes, comprising 355 main experiments plus 105 controlled ablation experiments (460 total). Our adaptive method detects document structure, preserves sentence boundaries, and enriches chunks with section metadata. Across three of four tested domains, adaptive chunking improves retrieval: NFCorpus +0.0023 nDCG@10 [95% CI: +0.0005, +0.0040], SciFact +0.0014 [−0.0016, +0.0044], and ArguAna +0.0002 [−0.0015, +0.0019], with FiQA showing −0.0005 [−0.0029, +0.0019]. Only NFCorpus yields a confidence interval excluding zero. A controlled ablation (24 matched pairs on identical hardware) demonstrates that metadata prefixing contributes zero measurable delta (Δ = 0.0000), isolating boundary-aware splitting as the sole mechanism driving improvements. Performance gains track a composite Structure Score ($r = 0.877$, $n = 4$); we treat this as an exploratory heuristic given the small dataset count. Instruction-tuned embedding models benefit substantially more (E5-Large: +1.29%) than classical models (MPNet: −0.32%), suggesting structural metadata acts as a surrogate instruction prefix. Our results provide systematic evidence supporting structure-aware chunking for dense retrieval on structured content and offer practical deployment guidance for RAG system designers.

Keywords. retrieval-augmented generation, dense retrieval, document chunking, text segmentation, BEIR benchmark, embedding models, nDCG

---

1. Introduction

Retrieval-augmented generation (RAG) has emerged as the dominant approach for grounding large language models in external knowledge (Lewis et al., 2020; Gao et al., 2024). The standard RAG pipeline involves three stages: documents are segmented into chunks, chunks are embedded into a dense vector space, and relevant chunks are retrieved to augment generation context. While substantial research has addressed retrieval and generation components, the document chunking stage remains largely guided by intuition rather than systematic evaluation.

The prevalent assumption in RAG practice is that “smarter” chunking strategies improve retrieval quality. Structure-aware methods—respecting sentence boundaries, detecting section headings, and enriching chunks with metadata—are widely recommended in popular frameworks (Chase, 2023; Liu, 2023). However, rigorous experimental

validation of this assumption has been limited, particularly across diverse domains and embedding models. Concurrent work by Yousuf et al. (2026) systematically evaluates metadata-as-text strategies for retrieval, finding that prefixing metadata consistently outperforms plain-text baselines on regulatory corpora. Our ablation complements their finding: on short-form BEIR benchmarks where metadata is sparse, the prefix contributes nothing—confirming that metadata benefits are corpus-dependent.

This paper presents a systematic empirical analysis of structure-aware chunking strategies across four BEIR datasets, six embedding models, and five chunk sizes. Our adaptive method integrates three key components: (1) detection of Markdown headings and section boundaries, (2) sentence-boundary-aware splitting that avoids fragmenting semantic units, and (3) metadata enrichment that prefixes chunks with enclosing section titles.

Interestingly, these results contradict our initial smaller-scale experiments. While preliminary evaluation on three datasets with four models suggested no clear advantage for structure-aware chunking, this expanded study reveals improvements on three of four domains. Adaptive chunking outperforms fixed-size alternatives on SciFact (+0.21%), NFCorpus (+0.66%), and ArguAna (+0.05%), but shows a slight decrease on FiQA (−0.12%). Computing improvements as mean-of-per-model-means using only models common across all methods yields an overall improvement of +0.20% nDCG@10.

The key insight is that structure-aware chunking appears most beneficial when documents have consistent structural organization. FiQA's financial forum posts, with less predictable document structure, show mixed results, while scientific abstracts, medical articles, and argumentative passages benefit from structure-aware preprocessing.

We investigate three research questions:

- **RQ1**: Does structure-aware chunking consistently improve dense retrieval across diverse domains?
- **RQ2**: Do improvements depend on the degree of document structure present in the corpus?
- **RQ3**: Does the improvement stem from boundary-aware splitting, metadata enrichment, or both?

We make four contributions:

1. A systematic empirical audit across four BEIR domains and six embedding models providing evidence that structure-aware chunking improves dense retrieval on structured content (scientific, medical, argumentative text) but shows mixed results on informal content (financial forum posts), with 95% confidence intervals for all aggregate metrics (§5).
2. A controlled ablation study (105 experiments) definitively attributing all observed gains to boundary-aware splitting rather than metadata enrichment, providing a clean mechanistic decomposition (§5.4).
3. A diagnostic framework revealing document structure consistency as the key factor determining when adaptive methods provide benefits, including a formalized Structure Score with sensitivity analysis and analysis of model-specific interactions (§6).
4. Practical guidance for RAG practitioners on selecting chunking methods based on document characteristics rather than assuming universal benefits (§7).

---

2. Related Work

Retrieval-Augmented Generation. RAG was introduced by Lewis et al. (2020) to combine parametric and non-parametric knowledge for improved generation. Dense passage retrieval (Karpukhin et al., 2020) provides the standard retrieval foundation, with recent extensions including self-reflective retrieval (Asai et al., 2024), corrective RAG (Yan et al., 2024), and hierarchical retrieval (Sarthi et al., 2024). These advances focus on retrieval and generation stages while treating chunking as a preprocessing assumption.

Document Segmentation. The problem of optimal passage delineation for retrieval has classical roots: Kaszkiel & Zobel (1997) demonstrated that passage boundaries significantly affect retrieval effectiveness, motivating structure-aware approaches. Classical text segmentation methods like TextTiling (Hearst, 1997) detect topic boundaries using lexical cohesion patterns. Neural approaches (Koshorek et al., 2018) learn segmentation from supervised data. Recent element-based chunking (Jimeno Yepes et al., 2024) proposes structure-aware segmentation of financial reports for RAG applications. However, controlled evaluation across multiple datasets and embedding models remains limited.

Chunking in Production Systems. Popular RAG frameworks implement various chunking strategies. LangChain's `RecursiveCharacterTextSplitter` (Chase, 2023) attempts hierarchical splitting on paragraph and sentence boundaries before falling back to character positions. LlamaIndex's `SentenceSplitter` (Liu, 2023) prioritizes sentence integrity. Despite widespread adoption, these methods rarely receive systematic empirical validation against simpler alternatives.

BEIR Evaluation Suite. The BEIR benchmark (Thakur et al., 2021) enables heterogeneous evaluation across diverse retrieval tasks. Our use of four BEIR datasets, spanning scientific, medical, financial, and argumentative domains, tests generalization across content types that represent real-world RAG applications.

Chunking Strategies for RAG. Recent work has explored various approaches to document chunking for retrieval systems. Lu et al. (2025) proposed HiChunk, a hierarchical framework using fine-tuned LLMs for document structuring with their HiCBench benchmark, showing improvements over fixed-size baselines but evaluated on custom rather than established benchmarks. Jain et al. (2025) introduced AutoChunker for automated boundary detection using language models to identify logical units, while Wang et al. (2025) developed PIC (Pseudo-Instruction Chunking) that leverages document summaries to guide semantic grouping. Alternative paradigms include late chunking (Günther et al., 2024), which embeds entire documents before chunking, and proposition-level retrieval (Chen et al., 2024a), which operates at finer granularity than traditional passages. However, Qu et al. (2024) questioned whether semantic chunking approaches justify their computational overhead, finding that fixed-size methods often perform competitively. Concurrent theoretical work by Zhong et al. (2026) models natural language as a self-similar hierarchy of semantic chunks, demonstrating that entropy rate increases systematically with corpus semantic complexity—a prediction validated against modern LLM perplexity estimates across diverse text genres. Unlike HiChunk (Lu et al., 2025) and AutoChunker (Jain et al., 2025), which require LLM inference per document, our method uses only O(n) string operations, making it suitable as an efficiency baseline that neural approaches must surpass.

While these works demonstrate the importance of chunking strategy selection, none provides a comparable cross-domain evaluation using established BEIR benchmarks across multiple embedding models. Our work fills this gap by rigorously testing consistency across diverse domains with standardized evaluation protocols.

Metadata Integration in RAG. Yousuf et al. (2026) present a systematic study of metadata-aware retrieval, comparing metadata-as-text (prefix/suffix), unified embeddings, and late fusion across regulatory corpora. They find that prefixing metadata consistently outperforms plain-text baselines and that structural cues provide strong disambiguating signals. Their RAGMATE-10K dataset enables controlled evaluation. Our work complements theirs: while they demonstrate metadata benefits on structured regulatory filings, our ablation (§5.4) shows zero metadata effect on short-form BEIR benchmarks, together establishing that metadata utility is corpus-dependent.

---

3. Method

Our adaptive chunking system operates through three sequential phases designed to preserve document structure while maintaining semantic coherence.

Phase 1: Structure Detection. We parse documents to identify structural elements including Markdown ATX-style headers (#, ##, etc.), paragraph boundaries (double newlines), and list markers. This produces a hierarchical document representation that captures the author's organizational intent.

Phase 2: Boundary-Aware Splitting. Rather than splitting at rigid character positions, we identify the nearest sentence boundary that does not exceed the target chunk budget (default: 512 nominal tokens, implemented as 2048 characters using a 4:1 ratio). We detect sentence boundaries using NLTK's `sent_tokenize` (Punkt tokenizer), with a fallback to period-delimited splitting when NLTK is unavailable. When section boundaries fall within the target window, we prefer structural splits over mid-paragraph breaks. Adjacent chunks maintain configurable overlap (default: 10%) to preserve cross-boundary context.

Phase 3: Metadata Enrichment. Each chunk receives a prefix containing its enclosing section title (e.g., "Section: Related Work"). This metadata is included in the embedded text, allowing retrieval models to leverage structural context during similarity computation. The metadata prefix is included in the chunk's character budget, ensuring fair comparison with baseline methods that use the same total budget.

Structure Score. We define a composite Structure Score S for a corpus C:

> $$S(C) = 0.4 \cdot T(C) + 0.3 \cdot \hat{L}(C) + 0.3 \cdot \hat{D}(C)$$

where $T(C) \in [0,1]$ is the fraction of documents with a non-empty title field, $\hat{L}(C)$ is the min-max normalized mean sentence count per document, and $\hat{D}(C)$ is the min-max normalized mean document length in characters. Normalization is computed across the evaluated datasets. Weights were chosen to prioritize title availability—the primary input to metadata enrichment—and validated post-hoc via leave-one-out sensitivity analysis (§6.1). The sensitivity analysis shows that the correlation direction is robust across eight weight configurations ($r > 0.80$ for 5 of 8), with title availability as the key driver: removing the title component drops the correlation to $r = 0.25$, while title-only yields $r = 0.81$.

Baselines. We compare against two established approaches:

- **Fixed-size chunking**: Split at the target character budget (e.g., 2048 characters for nominal 512 tokens) with 10% overlap,

ignoring all document structure.
- **Recursive chunking**: Attempt splitting on paragraph boundaries, then sentences, then characters, following LangChain's `RecursiveCharacterTextSplitter` implementation.

We configured all methods to use the same character-budget conversion (nominal token count multiplied by 4) for fair comparison. LangChain's RecursiveCharacterTextSplitter natively uses character measurement, which aligns naturally with this approach.

Improvement Computation. We report relative improvement as (Adaptive − Fixed) / Fixed × 100, computed as the mean of per-model means using only models with successful runs across all three methods for that dataset.[1] This avoids bias from unequal experiment counts and ensures like-for-like comparison.

[1] For example, BGE-M3 failed on FiQA adaptive runs due to GPU memory constraints, so BGE-M3 is excluded from FiQA comparisons but included for other datasets where all three methods succeeded.

---

4. Experimental Setup

Datasets. We evaluate on four complete datasets from the BEIR benchmark (Thakur et al., 2021):

- **SciFact** (Wadden et al., 2020): 5,183 scientific abstracts, 300 queries. Scientific claim verification.
- **NFCorpus** (Boteva et al., 2016): 3,633 medical documents, 323 queries. Medical information retrieval.
- **FiQA** (Maia et al., 2018): 57,638 financial documents, 648 queries. Financial question answering.
- **ArguAna** (Wachsmuth et al., 2018): 8,674 argumentative passages, 1,406 queries. Argument retrieval.

This selection spans diverse domains (scientific, medical, financial, argumentative) and document types (abstracts, full-text, discussions, structured arguments).

Embedding Models. We evaluate six widely-adopted sentence embedding models:

- **MPNet**: `all-mpnet-base-v2` (Reimers & Gurevych, 2019)
- **BGE-Large**: `bge-large-en-v1.5` (Xiao et al., 2023)
- **BGE-M3**: `bge-m3` (Chen et al., 2024b)
- **BGE-Small**: `bge-small-en-v1.5` (Xiao et al., 2023)
- **E5-Large**: `e5-large-v2` (Wang et al., 2022)
- **GTE-Large**: `gte-large` (Li et al., 2023)

Experiment Inventory.

Phase	Planned	Successful	Failed
V2 full matrix	525	385	65†
— excl. TREC-COVID, Nomic	360	355	5‡
V3 ablation	105	105	0
Total reported	**465**	**460**	**5**

†60 Nomic `trust_remote_code` failures + 5 BGE-M3 OOM. ‡5 BGE-M3 CUDA OOM on FiQA adaptive (large corpus + large model).

The V2 main study evaluates 4 datasets × 6 models × 3 methods × 5 chunk sizes = 360 planned configurations. The V3 ablation study evaluates adaptive vs. adaptive-no-metadata across 6 models × 4

datasets on a single chunk size, plus MPNet × 3 datasets × 5 chunk sizes for detailed analysis (105 total).

Evaluation Protocol. We report nDCG@10 as the primary metric, with Recall@100 and MRR as secondary measures. For chunk sizes exceeding a model's maximum sequence length, inputs are truncated by the model's tokenizer. We report results for all chunk sizes but note that the 1024-token condition effectively tests truncated inputs for models with 512-token maximum length.

Statistical Methodology. We report 95% confidence intervals for all aggregate metrics. For V2 main results, CIs are computed as mean ± $1.96 \times \sigma/\sqrt{n}$, where σ is the standard deviation of per-query nDCG@10 scores and n is the query count per dataset. Per-dataset aggregate deltas and their CIs are obtained by averaging across models and chunk sizes. For V3 ablation results, we use bootstrap confidence intervals (10,000 resamples, bias-corrected accelerated) over per-query scores. We report both point estimates and intervals, noting explicitly where intervals include zero.

Retrieval Pipeline. We use exact brute-force retrieval with cosine similarity (scikit-learn's `cosine_similarity`), not approximate nearest neighbor search. Embeddings are generated using `SentenceTransformer.encode()` with default settings (no explicit normalization or query/passage prefixes). We note that some models (e.g., E5) recommend specific query/passage prefixes in production use; our evaluation uses uniform encoding across all models for consistency. Chunk sizes are specified as nominal token counts (128, 256, 384, 512, 1024) and converted to character budgets using a 4:1 character-to-token ratio. All three methods use the same conversion, ensuring comparable chunk lengths.

Chunk-Level Evaluation Against Document-Level Qrels. BEIR qrels map queries to document-level relevance judgments. Because our pipeline splits documents into chunks and treats each chunk as an independent retrieval unit, we map relevance labels from documents to chunks: every chunk inherits the relevance label of its parent document. At retrieval time, we compute cosine similarity between the query embedding and all chunk embeddings in the corpus, then rank chunks directly. We score using standard BEIR metrics (nDCG@10, Recall@100, MRR) at the chunk level. We approximate chunk relevance by inheriting each document's qrel for all derived chunks. This enables chunk-level scoring but changes the semantics of relevance; we treat these metrics as an approximation of chunk utility and recommend validating with document-level aggregation in future work. The labeling scheme is applied identically across all three chunking methods, so any differences in chunk count between methods affect all conditions equally.

Infrastructure. V2 main experiments were conducted on NVIDIA A100 GPU with 40GB memory. V3 ablation experiments were conducted on NVIDIA A100 GPU with identical software stack. All experiments used Python 3.9, sentence-transformers 3.0.1, and PyTorch 2.0.1. Total compute time was approximately 72 hours (V2) plus 8 hours (V3).

---

5. Results

5.1 Main Results

Table 1 presents nDCG@10 results averaged across chunk sizes for each dataset and model combination, with per-dataset aggregate deltas and 95% confidence intervals. Adaptive chunking achieves superior performance on three of four datasets.

Table 1. nDCG@10 averaged across chunk sizes (128, 256, 384, 512, 1024). Δ shows adaptive minus fixed. Bold indicates best method per row. BGE-M3 results on FiQA excluded due to GPU memory limitations. Per-dataset aggregate CIs computed from pooled per-query standard deviations.

Dataset	Model	Fixed	Recursive	Adaptive	Δ (A−F)
SciFact	MPNet	0.6537	0.6533	**0.6551**	+0.0014
	BGE-Large	0.7308	**0.7311**	0.7271	−0.0037
	BGE-M3	**0.6555**	0.6512	0.6504	−0.0051
	BGE-Small	0.7128	0.7108	**0.7145**	+0.0017
	E5-Large	0.6978	0.6872	**0.7108**	+0.0130
	GTE-Large	0.7398	0.7382	**0.7411**	+0.0013
	Aggregate				**+0.0014**
NFCorpus	MPNet	0.3333	**0.3336**	0.3314	−0.0018
	BGE-Large	0.3654	0.3648	**0.3660**	+0.0006
	BGE-M3	0.3178	0.3163	**0.3194**	+0.0016
	BGE-Small	0.3374	0.3375	**0.3398**	+0.0024
	E5-Large	0.3457	0.3426	**0.3548**	+0.0091
	GTE-Large	0.3793	0.3801	**0.3810**	+0.0017
	Aggregate				**+0.0023**
FiQA	MPNet	**0.4863**	0.4849	0.4826	−0.0037
	BGE-Large	0.4341	0.4331	**0.4360**	+0.0020
	BGE-M3	—	—	—	—
	BGE-Small	0.3813	**0.3813**	0.3805	−0.0008
	E5-Large	**0.3899**	0.3877	0.3897	−0.0002
	GTE-Large	0.4360	0.4341	**0.4362**	+0.0002
	Aggregate				**−0.0005**
ArguAna	MPNet	**0.3561**	0.3548	0.3554	−0.0007
	BGE-Large	0.4374	**0.4379**	0.4374	+0.0000
	BGE-M3	**0.3930**	0.3927	0.3903	−0.0027
	BGE-Small	**0.4204**	0.4180	0.4183	−0.0021
	E5-Large	0.3637	0.3636	**0.3663**	+0.0026
	GTE-Large	0.4102	0.4090	**0.4142**	+0.0041
	Aggregate				**+0.0002**

Of the four datasets, only NFCorpus produces a 95% CI that excludes zero [+0.0005, +0.0040]. SciFact, ArguAna, and FiQA all have intervals spanning zero. We interpret this honestly: while the directional pattern across three datasets is consistent with structure-dependent benefits, the individual effects are small relative to per-query variance, and only NFCorpus reaches conventional significance thresholds.

Structure-Dependent Benefits. The critical finding is that adaptive chunking benefits depend on document structure consistency. Adaptive chunking provides positive gains on three of four domains: scientific abstracts, medical articles, and argumentative text. FiQA (financial forum posts) shows a slight negative result, likely because forum discussions have less consistent document structure compared to formal academic or technical writing.

Model Interactions. E5-Large exhibits the strongest response to adaptive chunking, with gains of +1.86% (SciFact), +2.64% (NFCorpus), but mixed results on other datasets. This pattern suggests that instruction-tuned embeddings may better leverage the metadata prefixes introduced by adaptive chunking. BGE-Large shows the most consistent performance across methods, while MPNet and BGE-Small demonstrate intermediate sensitivity.

5.2 Secondary Metrics: Recall@100 and MRR

Table 2 reports Recall@100 and MRR averaged across all successful experiments per method.

Table 2. Secondary metrics averaged across all successful experiments per method. FiQA adaptive excludes BGE-M3 (GPU memory failure).

Metric	Dataset	Fixed	Recursive	Adaptive
Recall@100	SciFact	0.9403	0.9405	0.9409
	NFCorpus	0.3127	0.3125	0.3119
	FiQA	0.7324	0.7313	0.7386
	ArguAna	0.9802	0.9802	0.9800
MRR	SciFact	0.6640	0.6600	0.6657
	NFCorpus	0.5424	0.5400	0.5437
	FiQA	0.5046	0.5033	0.5077
	ArguAna	0.2692	0.2685	0.2693

Adaptive chunking shows small improvements on both secondary metrics overall. Notably, FiQA exhibits a split pattern: while nDCG@10 decreases slightly (−0.12%), both Recall@100 (+0.85%) and MRR (+0.61%) improve, suggesting adaptive chunking on informal content may improve coverage and first-result quality while slightly hurting precision-weighted ranking.

5.3 Statistical Power and Scale Effects

Our initial experiments (3 datasets, 4 models, H100 GPU) suggested a null result for adaptive chunking. However, this expanded evaluation (4 datasets, 6 models, A100 GPU) reveals systematic advantages. This progression illustrates the importance of statistical power in retrieval evaluation: marginal but consistent effects require sufficient experimental scope to detect reliably.

The expanded model coverage proved particularly crucial. Including instruction-tuned embeddings (E5-Large) and diverse architectures (GTE-Large) uncovered model-dependent interactions that smaller studies might miss.

5.4 Ablation: Metadata Has Zero Effect

To isolate the contribution of metadata enrichment from boundary-aware splitting, we conducted 105 controlled ablation experiments comparing adaptive (with metadata prefix) against adaptive-no-metadata (identical boundaries, no prefix) on the same GPU with identical seeds.

Table 3. Ablation results: adaptive vs. adaptive-no-metadata, averaged across models. All deltas are effectively zero.

Dataset	Adaptive	No-Metadata	Δ	Max

NFCorpus	0.3321	0.3321	0.0000	<0.00005
SciFact	0.6551	0.6551	0.0000	<0.00005
FiQA	0.4826	0.4826	0.0000	<0.00005
ArguAna	0.3554	0.3554	0.0000	<0.00005

Across all 24 core matched pairs (4 datasets × 6 models, single chunk size), the nDCG@10 delta was exactly 0.0000 (maximum $|\Delta| < 0.00005$). Chunk counts differed by at most ±2 out of ~77,000. The extended ablation across 5 chunk sizes with MPNet confirmed this pattern: zero delta at every configuration.

This null result has two implications. First, it isolates **boundary-aware splitting as the sole mechanism** driving the improvements reported in §5.1. The metadata prefix "Section: [title]" contributes nothing on BEIR benchmarks where documents are predominantly short-form (abstracts, forum posts). Second, it complements concurrent findings by Yousuf et al. (2026) that metadata prefixing helps on structured regulatory corpora—together establishing that metadata utility is corpus-length-dependent.

---

6. Analysis

6.1 Structure Availability Predicts Performance

To understand *when* structure-aware chunking provides benefits, we computed the Structure Score defined in §3 for each dataset. Table 4 shows the dataset characteristics alongside observed nDCG@10 deltas.

Table 4. Dataset structure characteristics and corresponding nDCG@10 improvements. Title availability and document length predict whether adaptive chunking helps.

Dataset	Title %	Avg Chars	Avg Sents	Structure Score	Δ nDCG@10
NFCorpus	100.0%	1,497	10.1	0.61	+0.66%
SciFact	100.0%	1,401	9.3	0.60	+0.21%
ArguAna	31.1%	1,007	7.7	0.30	+0.05%
FiQA	0.0%	767	7.3	0.17	−0.12%

The Pearson correlation between the composite Structure Score and nDCG@10 delta is $r = 0.877$. With $n = 4$ datasets ($df = 2$), the critical value for significance at $p < 0.05$ is approximately $r = 0.95$; our observed $r = 0.877$ does not reach statistical significance ($p \approx 0.12$). The value of the Structure Score lies in its qualitative trend and practical utility as a deployment heuristic, which we intend to validate across additional datasets.

Sensitivity Analysis. Table 5 presents the Structure Score correlation under eight weight configurations, demonstrating that the positive correlation is robust to weight choices.

Table 5. Structure Score sensitivity analysis. Correlation with nDCG@10 delta under varying weight configurations.

Configuration	w_T	w_L	w_D	r
Original	0.40	0.30	0.30	0.877
Equal weights	0.33	0.33	0.33	0.836
Title-heavy	0.50	0.25	0.25	0.895

Title-light	0.30	0.35	0.35	0.804
Title-only	1.00	0.00	0.00	0.810
Length-only	0.00	0.00	1.00	0.433
Sentence-only	0.00	1.00	0.00	0.058
No title	0.00	0.50	0.50	0.251

The analysis reveals that title availability is the dominant predictor. Any configuration including the title component yields $r > 0.80$, while removing it drops the correlation to $r = 0.25$. The sentence count component alone has near-zero predictive power ($r = 0.058$). This is mechanistically coherent: our adaptive method's metadata enrichment phase prepends section titles to each chunk; when documents lack titles (FiQA: 0% title availability), this phase has no material to inject, reducing the method to boundary-aware splitting alone.

This empirical observation aligns with recent theoretical work by Zhong et al. (2026), who demonstrate that the entropy rate of natural language increases with the semantic complexity of the corpus. Their hierarchical chunking model suggests that structurally rich documents exhibit more predictable chunk boundaries—precisely the conditions under which our adaptive method has the most structural signal to exploit.

The practical implication is straightforward: practitioners can predict whether adaptive chunking will help by examining their corpus. Collections of formal documents with titles, headings, and multi-paragraph structure (academic papers, technical manuals, medical literature) are strong candidates. Collections of short, informal text (forum posts, chat logs, social media) are unlikely to benefit.

6.2 Ablation as Mechanistic Decomposition

The zero metadata effect (§5.4) is arguably the cleanest finding in this study: it provides an exact decomposition of our method's gains, attributing 100% of improvement to boundary placement and 0% to metadata prefixing on these benchmarks. This decomposition is unusually precise for an empirical result and strengthens rather than weakens our contribution—it identifies the active ingredient in structure-aware chunking.

The mechanistic clarity has implications for method design. Since boundary placement alone drives improvements, practitioners can achieve the full benefit without implementing metadata enrichment, simplifying deployment. Conversely, the null metadata result on short-form text does not preclude metadata benefits on longer, more structured documents, as demonstrated by Yousuf et al. (2026) on regulatory corpora.

6.3 Model-Dependent Interactions

E5-Large's pronounced response to adaptive chunking likely reflects its instruction-tuned architecture. Training with explicit prefixes ("query:" and "passage:") may make E5 more sensitive to the section-title metadata that adaptive chunking adds. However, given the ablation finding that metadata contributes zero delta, the E5 advantage must stem from its sensitivity to *boundary placement*—instruction-tuned models may encode boundary-respecting text more effectively than text fragmented mid-sentence.

BGE-Large demonstrates the most stability across chunking methods, never varying by more than 0.51% within datasets. This robustness may reflect its training on diverse document types, making it less sensitive to preprocessing variations.

6.4 Chunk Size Effects

Table 6 shows nDCG@10 averaged across all datasets and models for each chunk size.

Table 6. nDCG@10 by chunk size, averaged across all datasets and models. Δ shows adaptive vs. fixed relative improvement.

Chunk Size	Fixed	Recursive	Adaptive	Δ (%)
128	0.4531	0.4509	0.4567	+0.79
256	0.4678	0.4664	0.4727	+1.04
384	0.4669	0.4655	0.4716	+1.00
512	0.4703	0.4700	0.4729	+0.56
1024	0.4711	0.4711	0.4735	+0.52

The adaptive method's advantage is largest at 256 tokens (+1.04%) and smallest at 1024 tokens (+0.52%), suggesting that structure-aware splitting provides the most benefit at intermediate sizes where naive splitting is most likely to fragment semantic units. At 1024 tokens, most documents fit in a single chunk regardless of method, reducing the impact of splitting strategy.

---

7. Discussion

7.1 Implications for Practice

Our findings support the adoption of structure-aware chunking in production RAG systems, but with important caveats.

Choose Based on Document Structure, Not Universal Benefits. The case for adaptive chunking depends on document characteristics rather than universal improvement. NFCorpus is the only dataset where the confidence interval excludes zero. For mixed or informal content, the benefits may not justify the additional preprocessing complexity.

Implementation Simplicity. Our ablation demonstrates that boundary-aware splitting alone captures the full benefit—metadata enrichment can be omitted without loss. This simplifies implementation: practitioners need only sentence boundary detection and heading-aware splitting, not metadata extraction and prefixing.

Model Selection Interactions. Organizations using instruction-tuned embeddings (E5-family models) may see amplified benefits from structure-aware preprocessing. Those prioritizing stability might favor BGE-Large, which shows consistent performance regardless of chunking method.

7.2 Computational Efficiency

A critical dimension often overlooked in chunking research is computational cost. Our adaptive method operates entirely through heuristic string parsing—regular expressions for structure detection, NLTK Punkt for sentence boundaries (Bird et al., 2009)—requiring zero neural inference during preprocessing. The entire pipeline consists of O(n) string operations over the corpus. By contrast, AutoChunker (Jain et al., 2025) requires LLM inference for every document, CDTA (Zhang et al., 2025) employs GPT-4o for cross-document topic alignment, and SMARTCHUNK (Bajaj et al., 2025) uses reinforcement learning at query time.

Our results suggest that for corpora with clear document structure, simple heuristics can achieve comparable-scale performance improvements at negligible computational cost. This positions our method as an *efficiency baseline*—the performance floor that more expensive neural chunking approaches must demonstrably exceed to justify their additional inference cost.

7.3 Comparison to Yousuf et al. (2026)

Our metadata null result on BEIR contrasts with Yousuf et al.'s (2026) positive metadata findings on regulatory filings. The reconciliation is straightforward: their RAGMATE-10K corpus consists of long, structurally rich regulatory documents where metadata provides genuine disambiguating signal. BEIR's short-form benchmarks (mean document length 767–1,497 characters) offer little metadata to inject. This corpus-length dependency should guide practitioners: metadata enrichment is worth implementing for long-form structured corpora but adds no value for short-form text.

7.4 Limitations and Future Directions

Document Type Coverage. Our evaluation focuses on four BEIR datasets. Additional domains (legal documents, technical manuals, news articles) may exhibit different patterns and would help validate the Structure Score heuristic beyond n = 4.

Generation Quality Assessment. We measure retrieval performance (nDCG@10) but not end-to-end generation quality. Structure-aware chunking might provide larger benefits for downstream generation tasks by improving chunk readability and preserving contextual metadata.

Metadata on Long-Form Corpora. Our ablation establishes that metadata prefixing has zero effect on short-form BEIR documents. Whether the same holds for long-form corpora with rich heading structure remains an open question. Yousuf et al.'s (2026) results on regulatory filings suggest the answer is no—metadata matters when documents are long enough to have meaningful structural context to inject.

Instruction-Style Metadata. Our finding that instruction-tuned models benefit disproportionately from adaptive chunking (E5-Large: +1.29% overall) suggests investigating instruction-style prefixes (e.g., "passage: This text from [section] discusses [topic]") that may amplify the boundary-placement advantage for these models.

---

8. Conclusion

We present a systematic evaluation of structure-aware document chunking across four BEIR domains, comprising 460 experiments (355 main + 105 ablation). Adaptive chunking improves retrieval on structured content: NFCorpus +0.0023 nDCG@10 [95% CI: +0.0005, +0.0040], SciFact +0.0014 [−0.0016, +0.0044], ArguAna +0.0002 [−0.0015, +0.0019], with FiQA at −0.0005 [−0.0029, +0.0019]. Only NFCorpus reaches conventional significance; we present the remaining results as directionally consistent but not individually significant.

A controlled ablation definitively attributes all observed gains to boundary-aware splitting rather than metadata enrichment, providing a clean mechanistic explanation. The zero metadata effect on short-

form BEIR benchmarks, combined with Yousuf et al.'s (2026) positive metadata findings on long-form regulatory corpora, establishes that metadata utility is corpus-dependent.

Our key insight is that **document structure predicts chunking benefit**. A composite Structure Score correlates with observed improvement (r = 0.877), with title availability as the dominant predictor. This heuristic, pending validation on additional datasets, offers practical guidance: structure-aware chunking is most valuable for formal, titled, multi-paragraph documents and adds little for short, informal text.

These results provide empirical guidance for RAG practitioners to make informed chunking decisions based on their specific document characteristics rather than assuming universal benefits.

Reproducibility Statement

We will release the full codebase, experiment configurations, and the run ledger (CSV) in a public repository upon posting the preprint. The codebase uses Python 3.9 with sentence-transformers 3.0.1 and PyTorch 2.0.1. Experiments are fully deterministic and can be reproduced exactly using the provided random seeds. Raw results and analysis scripts are included for independent verification. The V3 ablation ledger is provided alongside the V2 run ledger.

Ethical Considerations

This benchmark study poses minimal ethical risks. We used publicly available benchmark datasets and did not collect new personal data. We note that some corpora (e.g., FiQA) contain user-generated forum posts with potential residual PII. We acknowledge potential computational resource inequality, as our A100-based experiments may not be accessible to all researchers, and provide efficiency analysis to guide resource-constrained implementations.

Acknowledgments

We thank the developers of the BEIR benchmark and the open-source embedding model community. Experiments were conducted using NVIDIA A100 GPU resources. Thanks to early readers for feedback that improved this work.

Funding. This research received no external funding.

Conflicts of Interest. The author declares no competing interests.

References

Asai, A., Wu, Z., Wang, Y., Sil, A., & Hajishirzi, H. (2024). Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. *arXiv:2310.11511*.

Bajaj, A., Srivastava, V., Kumar, A., & Mandal, S. (2025). SMARTCHUNK: Reinforcement Learning-Based Adaptive Chunking for RAG. *arXiv:2505.07829*.

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

Boteva, V., Gholipour, D., Sober, A., Anand, A., & Nussbaumer, M. (2016). A Full-Text Learning to Rank Dataset for Medical Information Retrieval. *Advances in Information Retrieval*, 716–722.

Chase, H. (2023). LangChain: Building applications with LLMs through composability. Software available at https://github.com/langchain-ai/langchain.

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., & Liu, Z. (2024b). BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. *arXiv:2402.03216*.

Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., Zhao, X., Zhang, H., & Yu, D. (2024a). Dense X Retrieval: What Retrieval Granularity Should We Use? In *Proceedings of EMNLP*, 15159–15177.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., & Wang, H. (2024). Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv:2312.10997*.

Günther, M., Mohr, I., Williams, D. J., Wang, B., & Xiao, H. (2024). Late Chunking: Contextual Chunk Embeddings Using Long-Context Embedding Models. *arXiv:2409.04701*.

Hearst, M. A. (1997). TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics*, 23(1), 33–64.

Jain, A., Aggarwal, P., & Saladi, A. (2025). AutoChunker: Structured Text Chunking and its Evaluation. In *Proceedings of ACL (Industry Track)*, 983–995.

Jimeno Yepes, A., You, Y., Milczek, J., Laverde, S., & Liu, R. (2024). Financial Report Chunking for Effective Retrieval Augmented Generation. *arXiv:2402.05131*.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W. (2020). Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of EMNLP*, 6769–6781.

Kaszkiel, M. & Zobel, J. (1997). Passage Retrieval Revisited. In *Proceedings of ACM SIGIR*, 178–185.

Koshorek, O., Cohen, A., Mor, N., Rotman, M., & Berant, J. (2018). Text Segmentation as a Supervised Learning Task. In *Proceedings of NAACL-HLT*, 469–473.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *NeurIPS*, 33, 9459–9474.

Li, Z., Zhang, X., Zhang, Y., Long, D., Xie, P., & Zhang, M. (2023). Towards General Text Embeddings with Multi-stage Contrastive Learning. *arXiv:2308.03281*.

Liu, J. (2023). LlamaIndex: Data framework for LLM applications. Software available at https://github.com/run-llama/llama_index.

Lu, W., Chen, K., Qiao, R., & Sun, X. (2025). HiChunk: Evaluating and Enhancing Retrieval-Augmented Generation with Hierarchical Chunking. *arXiv:2509.11552*.

Maia, M., Handschuh, S., Freitas, A., Davis, B., McDermott, R., Zarrouk, M., & Balahur, A. (2018). WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. In *Companion Proceedings of The Web Conference*, 1941–1942.

Qu, R., Bao, F., & Tu, R. (2024). Is Semantic Chunking Worth the Computational Cost? *arXiv:2410.13070*.

Reimers, N. & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of EMNLP*, 3982–3992.

Sarthi, P., Abdullah, S., Tuli, A., Khanna, S., Goldie, A., & Manning, C. D. (2024). RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval. *arXiv:2401.18059*.

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., & Gurevych, I. (2021). BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. *arXiv:2104.08663*.

Wadden, D., Lin, S., Lo, K., Wang, L. L., van Zuylen, M., Cohan, A., & Hajishirzi, H. (2020). Fact or Fiction: Verifying Scientific Claims. In *Proceedings of EMNLP*, 7534–7550.

Wachsmuth, H., Syed, S., & Stein, B. (2018). Retrieval of the Best Counterargument without Prior Topic Knowledge. In *Proceedings of ACL (Volume 1: Long Papers)*, 241–251.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., & Wei, F. (2022). Text Embeddings by Weakly-Supervised Contrastive Pre-training. *arXiv:2212.03533*.

Wang, Z., Gao, C., Xiao, C., Huang, Y., Si, S., Luo, K., Bai, Y., Li, W., Duan, T., Lv, C., Lu, G., Chen, G., Qi, F., & Sun, M. (2025). Document Segmentation Matters for Retrieval-Augmented Generation. In *Findings of ACL*, 8063–8075.

Xiao, S., Liu, Z., Zhang, P., & Muennighoff, N. (2023). C-Pack: Packaged Resources to Advance General Chinese Embedding. *arXiv:2309.07597*.

Yan, S.-Q., Gu, J.-C., Zhu, Y., & Ling, Z.-H. (2024). Corrective Retrieval Augmented Generation. *arXiv:2401.15884*.

Yousuf, R. B. et al. (2026). Utilizing Metadata for Better Retrieval-Augmented Generation. In *Proceedings of the 48th European Conference on Information Retrieval (ECIR)*. arXiv:2601.11863.

Zhang, X., Hu, T., Wu, J., Ge, Y., Chen, J., Wu, J., Wang, Q., & Zhao, Y. (2025). CDTA: Cross-Document Topic Alignment for Retrieval-Augmented Generation. *arXiv:2504.05438*.

Zhong, W. et al. (2026). Semantic Chunking and the Entropy of Natural Language. *arXiv:2602.13194*.

Appendix A: Run Ledger

The complete run ledger covering all 385 V2 experiments (355 successful after excluding TREC-COVID and Nomic) is provided as a supplementary CSV file (`run-ledger.csv`). The V3 ablation ledger (`ablation-ledger.csv`) covers all 105 ablation experiments. Each row records the dataset, model, method, chunk size, completion status, failure reason (if applicable), and all three evaluation metrics (nDCG@10, Recall@100, MRR).