One Anchor Is Enough: Cross-Document Coverage in Multi-Hop Retrieval

Anonymous ACL submission

Abstract

Multi-hop question-answering (QA) requires evidence from multiple documents, yet prior work on retrieval has largely focused on encoders and reranking while treating retrieval unit construction as fixed. We study how retrieval unit design affects multi-hop retrieval and downstream QA by holding the encoder and scoring function fixed and varying chunking hyperparameters within Query-Anchored Span Selection (QASS). Across HotPotQA and 2WikiMultihopQA, we find that the number of anchors per document dominates all other chunking choices. Restricting QASS to a single anchor per document improves Exact Match by $+4.26$ and $+2.77$ at $K{=}5$ over a three-anchor baseline ($p < 0.001$). This constraint guarantees that the top-K retrieved units originate from K distinct documents, yielding average unique document counts of 4.99 and 5.00 and approaching the structural maximum. Neither adaptive anchor selection nor stronger encoders alter this pattern. Our results suggest that, for multi-hop retrieval, enforcing cross-document coverage during retrieval unit construction can recover benefits typically attributed to post-hoc reranking, without additional reranking hyperparameters or a separate ranking stage.

1 Introduction

Multi-hop question answering requires evidence from multiple documents to produce a single answer (Yang et al., 2018; Ho et al., 2020). A retrieval system that supports this task must return a small set of spans, typically the top-K, that together contain the evidence the reader needs to compose an answer. When the answer depends on combining information across documents, the top-K result must distribute its evidence across distinct sources. A retrieval result that places multiple spans from one document at the top of the ranking, however high-scoring those spans are, leaves the reader without one of the hops the question requires.

Research on multi-hop retrieval has so far concentrated on two design dimensions. The first is the encoder, including dense retrievers (Karpukhin et al., 2020; Izacard et al., 2022), late-interaction models (Khattab and Zaharia, 2020), and iterative multi-hop methods (Xiong et al., 2020; Khattab et al., 2021). The second is post-hoc reranking, where diversity-aware scoring methods (Carbonell and Goldstein, 1998; Zhai et al., 2003; Clarke et al., 2008; Agrawal et al., 2009) penalize redundancy among already-retrieved candidates. Yet both approaches accept the retrieval unit, the passage or chunk that gets ranked, as given. Chunk boundaries are fixed at indexing time, query-agnostically, and the same partition serves every query that arrives later.

We take a different angle. Holding the encoder and the scoring function fixed, we ask how retrieval unit construction itself affects multi-hop ranking and downstream QA. To make this question tractable, we use Query-Anchored Span Selection (QASS) as a controlled test bed. QASS selects sentences relevant to the query as anchors and expands each into a context-preserving span, exposing the chunking decision through a small set of hyperparameters. This setup lets us vary the retrieval unit while keeping every other component of the pipeline unchanged, and compare against fixed-size chunking and semantic segmentation under the same encoder.

Our experiments on HotPotQA (Yang et al., 2018) and 2WikiMultihopQA (2Wiki; Ho et al., 2020) reveal that one of the QASS hyperparameters dominates the others. The number of anchors per document, M, produces a substantial swing in downstream QA. Changing M from 3 to 1 yields an Exact Match gain of $+4.26$ on HotPotQA and $+2.77$ on 2Wiki at $K{=}5$ ($p < 0.001$). The mechanism is direct. With $M{=}1$, each document contributes at most one span to the candidate pool, so the top-K comes from K distinct documents

as an algorithmic property rather than a scoring outcome, reaching 4.99 unique documents on HotPotQA and 5.00 on 2Wiki at $K{=}5$. Four heuristics that vary M based on local query-relevance signals all underperform QASS fixed $M{=}1$, with NDCG decreasing monotonically as the average M rises above 1. The pattern persists with a strong encoder, `BGE-large-en-v1.5` (Xiao et al., 2023). The NDCG@5 advantage of $M{=}1$ shrinks but remains significant on both datasets ($p < 0.001$).

This study makes three contributions. We provide a controlled empirical comparison of retrieval unit designs in multi-hop QA, with grid search on a held-out development split, paired bootstrap for significance, and evaluation across two datasets and two encoders. Cross-document coverage emerges as the structural property that explains a substantial portion of the variation in multi-hop ranking and QA quality, and we measure this property directly through unique document counts. The property itself can be enforced at the chunking stage through a simple structural constraint, and no adaptive refinement we tested improves upon the constraint. At equal unique document counts, the chunking-stage constraint of QASS $M{=}1$ retains a NDCG@5 advantage of $+0.019$ over Fixed + cap on HotPotQA and $+0.062$ on 2Wiki, indicating that the contribution of chunking-stage construction extends beyond per-document coverage alone.

2 Related Work

Multi-hop retrieval and question answering. Advances in multi-hop QA have largely come from better encoders and retrieval strategies rather than from considering the retrieval unit itself. Dense retrievers such as Dense Passage Retrieval (DPR; Karpukhin et al., 2020) and Contriever (Izacard et al., 2022) learn increasingly effective query-passage representations. ColBERT (Khattab and Zaharia, 2020) and other late-interaction models introduce token-level matching. Extensions for multi-hop reasoning further refine retrieval through iterative or path-based strategies, including Multi-hop Dense Retrieval (MDR; Xiong et al., 2020), Baleen (Khattab et al., 2021), and graph-based reasoning (Asai et al., 2020). Effective as these methods are at refining scoring and multi-step reasoning, they all take passage boundaries as given, inheriting whatever segmentation was chosen at preprocessing time. Our study takes the orthogonal view, holding the encoder fixed and asking how retrieval unit construction affects multi-hop ranking.

Document segmentation and passage retrieval. The effect of document segmentation on passage retrieval has a long history in QA and information retrieval. Kaszkiel and Zobel (1997, 2001) examined how different passage granularities affect retrieval effectiveness, while Roberts and Gaizauskas (2004) compared architectural approaches to passage retrieval. Tiedemann and Mur (2008) compared linguistically motivated segmentation strategies with simple fixed-size windows and found that small, uniform windows can outperform semantic segmentation in one-step passage retrieval. Recent work in RAG has renewed interest in chunking strategies. Bhat et al. (2025) analyzes the impact of chunk size and segmentation heuristics on retrieval and downstream generation. Pseudo-Instruction for document Chunking (PIC; Wang et al., 2025) and Coreference-Linked Augmentation for Passage Retrieval (CLAP; Xu et al., 2025) use document-level semantics, summaries, or coreference to compose more coherent chunks, while chunkRAG (Singh et al., 2025) applies LLM-based filtering at the chunk level. All of these determine chunk boundaries at indexing or preprocessing time and remain query-agnostic. We instead construct units at query time, which exposes a structural property (cross-document coverage) that fixed boundaries cannot guarantee.

Query-aware sentence selection. Query-biased sentence selection dates back to early IR work on passage summarization. Tombros and Sanderson (1998) and Allan et al. (2003) showed that selecting query-relevant sentences improves user relevance judgements and task performance. Subsequent studies explored query-focused and task-oriented summarization in QA and cross-lingual IR (Williams et al., 2014). More recently, neural models have revisited query-aware sentence selection for summarization and answer-focused generation (Vig et al., 2022). These methods confine the role of sentences to display. Sentences are selected for summaries or snippets, but the underlying retrieval still operates over pre-segmented passages. We take the same intuition of anchoring on query-relevant sentences and use it as the seed for retrieval unit construction itself.

Diversity in retrieval. Maximal Marginal Relevance (MMR; Carbonell and Goldstein, 1998) penalizes redundancy in the selected set and remains

the most widely used diversity mechanism. Subsequent work has formalized diversity as a subtopic coverage problem (Zhai et al., 2003), proposed explicit modeling (Agrawal et al., 2009), and developed evaluation metrics that account for redundancy (Clarke et al., 2008). These efforts target query ambiguity in web search, where multiple subtopics need separate coverage. Multi-hop retrieval shares the surface goal of representing distinct evidence but differs in mechanism. The units that should be distinct are documents, not topics, and the constraint can be enforced at the chunking stage rather than through post-hoc reranking. We make this contrast explicit in §6.1.

3 Method

We use QASS as the controlled test bed for our analysis. Rather than introducing a new retrieval algorithm, our goal is to expose the chunking decision in a parameterized form so that we can isolate which design choices matter for multi-hop retrieval.

3.1 QASS as design lever

Most chunking strategies fix the retrieval unit before any query arrives: sentences, fixed-size windows (Kaszkiel and Zobel, 1997), or topic-segmented passages (Hearst, 1997). Each query must work with the same partition. QASS instead constructs spans *at query time* by anchoring on sentences relevant to the query and expanding around them. This keeps the encoder and scoring fixed, and makes the chunking decision the only variable in our experiments.

Concretely, QASS exposes four hyperparameters that together define a family of chunking strategies: anchors per document (M), window radius (w), minimum gap between anchors (τ), and the interpolation weight between anchor- and span-level scores (α). Different settings of these parameters produce different chunking behaviors, from near sentence-level retrieval (M large, w=0) to large coherent passages (M small, w large). We study how each parameter interacts with multi-hop retrieval performance, and find that one of them, M, dominates the others.

3.2 Algorithm

Let q denote the query and $\mathcal{D} = \{d_1, \ldots, d_n\}$ the candidate documents. Each document d_i is a sequence of sentences $\{s_{i,1}, \ldots, s_{i,|d_i|}\}$. We encode every text unit using a pretrained encoder $\mathrm{Enc}(\cdot)$ and score relevance via cosine similarity:

$$f(q, x) = \cos\big(\mathrm{Enc}(q),\ \mathrm{Enc}(x)\big). \qquad (1)$$

QASS proceeds in three steps for each query.

Algorithm 1 Query-Anchored Span Selection

Require: Query q, documents $\mathcal{D}$, parameters M, w, τ, α
Ensure: Top-K ranked spans
for each document $d_i \in \mathcal{D}$ **do**
  Compute $f(q, s_{i,j})$ for all sentences $s_{i,j}$
  $\mathcal{A}_i \leftarrow \emptyset$
  for j in descending order of $f(q, s_{i,j})$ **do**
    if $|j - p| \geq \tau$ for all $p \in \mathcal{A}_i$ **then**
      $\mathcal{A}_i \leftarrow \mathcal{A}_i \cup \{j\}$
    end if
    if $|\mathcal{A}_i| \geq \min(M, |d_i|)$ **then**
      break
    end if
  end for
  for each anchor $j \in \mathcal{A}_i$ **do**
    $S_{i,j} \leftarrow \{s_{i,k} : \max(1, j - w) \leq k \leq \min(|d_i|, j + w)\}$
    $\mathcal{S}(S_{i,j}) \leftarrow \alpha\, f(q, s_{i,j}) + (1 - \alpha)\, f(q, S_{i,j})$
  end for
end for
return top-K spans by $\mathcal{S}(\cdot)$ across all documents

The procedure has three steps. **Anchor selection** ranks each document's sentences by $f(q, s_{i,j})$ and greedily picks up to M anchors with minimum index distance τ to avoid redundant spans. **Window expansion** extends each anchor index j to a span $S_{i,j}$ of w sentences on either side, clipped at document boundaries. **Span scoring** interpolates anchor-level and span-level relevance:

$$\mathcal{S}(S_{i,j}) = \alpha\, f(q, s_{i,j}) + (1-\alpha)\, f\big(q, \mathrm{concat}(S_{i,j})\big). \qquad (2)$$

All spans are pooled across documents and ranked by $\mathcal{S}(\cdot)$, returning the top-K.

3.3 Design space

The four hyperparameters control different aspects of the chunking behavior. Among them, M has a qualitatively different effect from the others, which we analyze in detail in §6.

- M controls how many anchors a single document can contribute to the candidate pool. With M=1, each document has at most one span in the pool; with larger M, a document can contribute multiple overlapping spans.
- w controls the size of each span. w=0 reduces QASS to sentence-level retrieval; larger w produces longer, more coherent spans at the cost of including more peripheral content.

Method	HotPotQA				2WikiMultihopQA			
	R@5	NDCG@5	MRR	MHCov@5	R@5	NDCG@5	MRR	MHCov@5
BM25	.552	.525	.706	.480	.387	.352	.511	.275
Sentence	.572	.539	.705	.532	.587	.562	.734	.498
Contriever	.542	.499	.658	.512	.534	.485	.633	.505
ColBERT-style	.646	.576	.741	.406	.681	.626	.812	.476
Semantic* (p=20)	.678	.630	.770	.594	.681	.657	.829	.497
QASS default (M=3, w=2)	.765	.695	.861	.589	.747	.723	.914	.533
Fixed* (cs=5)	.809	.761	.876	.646	.771	.756	.915	.557
QASS* (M=1)	**.822*****	**.778*****	**.892*****	**.660*****	**.786*****	**.805*****	**.931*****	**.595*****

Table 1: Retrieval performance on the test split. * marks configurations selected by grid search on the dev split (Fixed: $cs \in \{1, \ldots, 5\}$; Semantic: $p \in \{20, \ldots, 50\}$; QASS: $M \in \{1, 2, 3\} \times w \in \{2, \ldots, 6\}$); best QASS uses w=6 (HotPotQA) and w=5 (2Wiki). $^{***}p < 0.001$ over Fixed by paired bootstrap (n_{iter}=10,000).

- τ prevents nearby anchors from producing nearly identical spans within a document.
- α balances anchor-level scoring, which is precise but local, against span-level scoring, which is coherent but diluted.

We also explore adaptive variants of M that vary across documents based on local query-relevance signals, evaluating four such heuristics against QASS fixed M=1 in Section 6.3.

4 Experimental Setup

Datasets. We evaluate on two multi-hop QA benchmarks in the distractor setting: HotPotQA (Yang et al., 2018) with 7,405 validation examples (5,918 bridge and 1,487 comparison), and 2Wiki (Ho et al., 2020) with 12,576 examples across four types (5,236 compositional, 3,040 comparison, 2,751 bridge-comparison, 1,549 inference). Both datasets provide sentence-level supporting fact annotations. We split each dataset into 20% dev and 80% test (seed 42); all hyperparameters are selected on dev, and test set results are reported only once.

Methods. We compare seven retrieval methods. Three operate at sentence level: **Sentence** retrieves individual sentences by dense cosine similarity. **BM25** (Robertson and Zaragoza, 2009) uses lexical matching (k_1=1.5, b=0.75), and **Contriever** (Izacard et al., 2022) uses its own pretrained encoder. **ColBERT-style** approximates token-level late interaction with sentence-level units. Two methods retrieve multi-sentence units with fixed boundaries: **Fixed** partitions documents into non-overlapping chunks of cs sentences (chunk size), and **Semantic** inserts boundaries at low adjacent-similarity points (percentile p). **QASS** constructs spans at query time with the algorithm described in §3.2. For Fixed, Semantic, and QASS, all hyperparameters are selected by grid search on dev. The search ranges are: $cs \in \{1, 2, 3, 4, 5\}$ for Fixed, $p \in \{20, 30, 40, 50\}$ for Semantic, and $M \in \{1, 2, 3\} \times w \in \{2, 3, 4, 5, 6\}$ for QASS. We fix τ=2 and α=0.5.

Implementation. All embedding-based methods except Contriever use `all-MiniLM-L6-v2` (Wang et al., 2020) as the encoder, with cosine similarity over L_2-normalized embeddings. We additionally evaluate Fixed and QASS with `BGE-large-en-v1.5` (Xiao et al., 2023) as a robustness check on encoder choice; full results are in Appendix C. For downstream QA, we feed top-K retrieved spans to `gpt-5-nano` via the OpenAI Batch API using the prompt template in Appendix A.

Metrics and evaluation. For retrieval, we report Recall@K, span-level NDCG@K (with IDCG computed over the full candidate pool to keep DCG and IDCG on the same unit), MRR, and multi-hop coverage (MHCov@K), defined as 1 if all gold document titles appear in the top-K retrieved spans. A span counts as relevant if it contains at least one annotated supporting sentence. For QA, we report Exact Match (EM). All pairwise comparisons use paired bootstrap with n_{iter}=10,000; adaptive-M comparisons in §6.3 additionally use paired t-tests.

5 Main Results

We evaluate seven retrieval methods on the test split of HotPotQA and 2Wiki. All chunking-based baselines (Fixed, Semantic, QASS) use configurations selected through grid search on the dev split. We report results at $K = 5$ for retrieval (Table 1), test cross-dataset robustness (Table 2), and mea-

sure downstream QA across multiple values of K (Table 3).

5.1 Retrieval performance

Table 1 reports retrieval performance for all methods. Two findings stand out. First, QASS with single-anchor selection $M{=}1$ achieves the highest on every metric across both datasets, and improvements over the best-tuned Fixed baseline are statistically significant at $p < 0.001$. The NDCG@5 gap is small on HotPotQA ($+0.017$) but substantial on 2Wiki ($+0.049$). Second, the QASS default configuration ($M{=}3, w{=}2$) underperforms Fixed on both datasets despite using the same encoder. This is the first signal that the choice of M is not a minor hyperparameter. We examine the gap between $M{=}1$ and $M{=}3$ in detail in Section 6.

5.2 Cross-dataset transfer

A natural concern is whether the $M{=}1$ finding reflects dataset-specific tuning rather than a general property of multi-hop retrieval. To test this, we apply each dataset's best dev configuration to the other dataset's test split.

Dev source → Test	HotPot test	2Wiki test
Best QASS		
From HotPot dev (M=1, w=6)	.778	.805
From 2Wiki dev (M=1, w=5)	.777	.805
Δ	0.001	0.000
Best Fixed		
From HotPot dev (cs=5)	.761	.756
From 2Wiki dev (cs=5)	.761	.756

Table 2: NDCG@5 of best dev configurations applied to each test split. QASS transfers between datasets with negligible loss (≤ 0.001).

Table 2 shows that the best QASS configuration on HotPot dev applied to 2Wiki test yields NDCG@5 of 0.805, which is identical to the in-dataset best. The reverse direction shows the same pattern (HotPot test: 0.778 vs. 0.777; difference 0.001). The same holds for Fixed: chunk size 5 is optimal on both dev splits. We interpret this as evidence that the design choices we identify are not artifacts of HotPot-specific or 2Wiki-specific tuning.

5.3 Encoder robustness

To verify that the $M{=}1$ advantage is not specific to our choice of encoder, we additionally measure Fixed with $cs{=}5$ and the best QASS ($M{=}1$ with $w{=}6$ on HotPotQA and $w{=}5$ on 2Wiki) using `BGE-large-en-v1.5` (Xiao et al., 2023), a strong general-purpose encoder. The NDCG@5 advantage of QASS over Fixed remains statistically significant on both datasets: $+0.005$ on HotPotQA ($p < 0.001$) and $+0.024$ on 2Wiki ($p < 0.001$), both via paired bootstrap. The unique document count of QASS is identical to the `all-MiniLM-L6-v2` result (4.99 on HotPotQA, 5.00 on 2Wiki at $K{=}5$), confirming that the structural guarantee of $M{=}1$ is independent of the encoder. Full results are in Appendix C.

5.4 Downstream QA

We feed the top-K retrieved spans to `gpt-5-nano` with a short prompt requesting only the answer. Table 3 reports Exact Match (EM) at $K \in \{1, 2, 3, 5\}$ for the three methods of central interest. Significance is computed by paired bootstrap with $n_{\text{iter}}{=}10{,}000$.

Table 3 reveals a pattern that motivates our analysis in Section 6. Within QASS, switching from $M{=}3$ to $M{=}1$ produces an EM gain of $+4.26$ on HotPotQA and $+2.77$ on 2Wiki at $K{=}5$, with all differences across K values significant at $p < 0.001$. The $M{=}3$ configuration itself underperforms Fixed by -3.67 EM on HotPotQA and -1.93 EM on 2Wiki at $K{=}5$, also at $p < 0.001$. The cumulative effect is direct. A single hyperparameter, swept from $M{=}3$ to $M{=}1$, moves QASS from clearly worse than Fixed to matching or exceeding it.

The asymmetry between datasets, with $M{=}1$ outperforming Fixed by $+1.84$ EM at $K{=}2$ on 2Wiki ($p < 0.001$) but matching it on HotPotQA, is not noise: Section 6 attributes it to how cross-document coverage matters most when fixed chunking already struggles with it.

6 Analysis

The EM swing between $M{=}1$ and $M{=}3$ shown in Table 3 requires explanation. We argue that this swing reflects a single structural property: cross-document coverage in the top-K retrieved set. Single-anchor selection guarantees this property by construction; multi-anchor selection breaks it. We support this claim from three angles. Section 6.1 measures the property directly through unique document counts. Section 6.2 shows that the advantage scales with how much fixed chunking already struggles with this coverage. Section 6.3 shows that no

Method	HotPotQA				2WikiMultihopQA			
	K=1	K=2	K=3	K=5	K=1	K=2	K=3	K=5
Absolute EM (%)								
QASS default (M=3)	17.33	22.83	25.13	27.87	2.51	13.14	15.66	19.87
Fixed (cs=5)	19.20	26.53	28.78	31.70	2.98	14.69	18.02	21.94
QASS (M=1)	**19.78**	**26.48**	**29.27**	**32.16**	**3.16**	**16.64**	**19.23**	**22.79**
Pairwise Δ EM (%)								
M=1 vs Fixed	$+0.57$	-0.07	$+0.64$	$+0.36$	$+0.15$	$+1.84^{***}$	$+1.21^{***}$	$+0.84^{*}$
M=1 vs M=3	$+2.33^{***}$	$+3.54^{***}$	$+4.19^{***}$	$+4.26^{***}$	$+0.63^{***}$	$+3.42^{***}$	$+3.55^{***}$	$+2.77^{***}$
M=3 vs Fixed	-1.80^{***}	-3.60^{***}	-3.49^{***}	-3.67^{***}	-0.45^{**}	-1.51^{***}	-2.32^{***}	-1.93^{***}

Table 3: Downstream QA performance using gpt-5-nano as the reader. Top section reports absolute EM on each method's valid predictions (sample sizes differ by around 1% due to API errors). Bottom section reports paired differences on the intersection of valid predictions, with significance from paired bootstrap (n_{iter}=10,000; $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$). The 3–4 EM swing between M=1 and M=3 is analyzed in Section 6.

adaptive variant of M recovers the lost coverage, ruling out the possibility that M=1 is incidentally well-tuned rather than structurally robust.

6.1 Cross-document coverage

We measure the number of distinct documents represented in the top-K spans for each method. This metric directly captures how much cross-document evidence the reader actually received.

Method	HotPotQA			2Wiki		
	K=2	K=3	K=5	K=2	K=3	K=5
QASS M=3	1.87	2.70	4.24	1.84	2.61	4.13
Fixed cs=5	1.98	2.94	4.80	1.94	2.84	4.61
QASS M=1	**2.00**	**3.00**	**4.99**	**2.00**	**3.00**	**5.00**

Table 4: Average unique documents in top-K retrieved spans (test split). M=1 approaches the structural maximum while M=3 falls below even Fixed.

Table 4 shows the pattern. M=1 selects exactly one anchor per document, which by construction guarantees that the top-K spans come from K distinct documents up to the candidate pool size. On 2Wiki at K=5, this yields a perfect 5.00 unique documents. On HotPotQA, the value is 4.99: 31 out of 5,924 queries (0.52%) have candidate pools too small to provide K distinct documents.

In contrast, M=3 allows a single document to occupy multiple slots if its anchors all rank highly. This redundancy is not theoretical. The average unique document count of M=3 at K=5 falls to 4.13 on 2Wiki and 4.24 on HotPotQA, below even Fixed chunking (4.61 and 4.80 respectively). Each top-K slot occupied by a redundant span from the same document comes at the cost of a slot that could have carried evidence from a different document.

This explains the M=3 underperformance versus Fixed in Table 3, despite both retrieving roughly the same set of relevant documents. The difference lies in how those documents are distributed across the top-K.

The same coverage property can be enforced through simpler mechanisms applied to any retriever. Restricting top-K selection to at most one unit per document, which we refer to as a per-document cap, reaches the same unique document count when applied to Fixed chunks. MMR reranking (Carbonell and Goldstein, 1998) produces a similar redistribution by penalizing similarity to already-selected items at scoring time. We compare directly against five such coverage-enforcing baselines (Sentence + cap, Fixed + cap, Fixed + MMR, doc-then-span, BM25 + cap) in Appendix D. Per-document coverage is necessary but not sufficient. At equal unique document counts, NDCG@5 remains $+0.019$ higher for QASS M=1 over Fixed + cap on HotPotQA, and $+0.062$ higher on 2Wiki. What QASS adds beyond a per-document cap is the query-anchored span boundary itself. Fixed chunk boundaries do not consider where the supporting evidence lies in the document, so a relevant sentence may sit at a chunk boundary or be surrounded by unrelated context. QASS centers each span on a query-relevant anchor, keeping the supporting sentence in the middle of its local context. Post-hoc reranking can re-order what the chunker produces but cannot reshape the units themselves.

6.2 Task complexity

The QA gains from M=1 are not uniform across question types. We break down EM at K=5 by type to identify where the advantage is largest.

Question Type	QASS $M{=}3$	Fixed cs=5	QASS $M{=}1$
HotPotQA			
Bridge ($n{=}4737$)	24.67	28.26	**28.84**
Comparison ($n{=}1187$)	40.66	**45.45**	45.44
2WikiMultihopQA			
Bridge-comp ($n{=}2212$)	7.04	10.63	**12.07**
Comparison ($n{=}2440$)	51.02	54.59	**56.04**
Compositional ($n{=}4164$)	13.66	14.37	**15.05**
Inference ($n{=}1245$)	3.22	**3.71**	3.15

Table 5: EM at $K{=}5$ by question type. $M{=}1$ matches or outperforms Fixed on most categories, with the largest gains on bridge-comp and comparison in 2Wiki. The only substantial reversal is inference; HotPotQA comparison is a near-tie.

Table 5 reveals two patterns. First, the gap between $M{=}1$ and $M{=}3$ is consistent and large across nearly all question types, ranging from $+5.0$ EM (2Wiki Bridge-comp, comparison) to $+1.39$ EM (compositional). The sole near-tie is 2Wiki inference. The consistency suggests that intra-document redundancy hurts every question type that requires multi-hop reasoning, with the magnitude of the hurt determined by how much redundancy displaces the actual evidence.

Second, the gap between $M{=}1$ and Fixed is much smaller on HotPotQA than on 2Wiki. HotPotQA bridge questions show a $+0.58$ EM advantage for $M{=}1$ over Fixed; HotPot comparison shows essentially no difference (-0.01). On 2Wiki, the corresponding gaps reach $+1.44$ EM (bridge-comp) and $+1.45$ EM (comparison). This asymmetry mirrors the unique-document-count gap in Table 4. The distractor setting of HotPotQA has 10 candidate documents per query and bridge questions typically require only 2 supporting documents, so Fixed at $K{=}5$ already covers them with high probability (4.80 unique docs at $K{=}5$). 2Wiki compositional and inference questions require evidence from documents that lie further apart in the candidate pool, so the unique-document gap between Fixed (4.61) and QASS $M{=}1$ (5.00) leaves more room for the structural guarantee to matter.

The asymmetry is the empirical signature of the cross-document coverage mechanism. The $M{=}1$ advantage scales with how much fixed chunking already struggles with coverage.

6.3 Robustness to adaptive selection

The previous subsections leave open one possibility. $M{=}1$ might happen to be a strong default, but a smarter per-document selection rule could outperform it. To test this, we evaluated four heuristics that adaptively determine the number of anchors per document based on its query-relevance distribution (defined in Appendix E). Each heuristic increases M when the document's query-similarity scores suggest multiple equally relevant sentences, on the intuition that such documents may contain more than one informative span.

The result, shown in Table 6, is unequivocal. Across both datasets, every adaptive variant underperforms QASS fixed $M{=}1$ in NDCG@5, and the gap is monotonic in average M. The most conservative heuristic (adaptive_combined, raising M above 1 in 17.1% of documents on HotPotQA and 8.1% on 2Wiki) loses 0.003 NDCG on HotPotQA and 0.010 on 2Wiki. The most aggressive (adaptive_mass, raising M in 63.2% of HotPotQA documents and 30.8% of 2Wiki documents) loses 0.011 and 0.051 respectively. All differences are significant at $p < 0.001$ by paired t-test.

The pattern is informative. R@5 actually rises slightly on 2Wiki for the gap and combined heuristics ($+0.004$ for adaptive_gap, $p < 0.001$), since allowing more anchors retrieves additional spans that contain relevant evidence. However, the rank order in the top-K degrades much more, lowering NDCG. This pattern is consistent with the cross-document coverage hypothesis: redundant intra-document spans displace second-hop evidence in the top of the ranking, and none of the anchor-selection heuristics tested compensate for the resulting structural loss.

We read this as strong support for the claim of Section 6.1. $M{=}1$ works because it prevents intra-document redundancy at the algorithm level, and any rule that allows even occasional $M{>}1$ reintroduces the redundancy without compensation. The single-anchor principle is robust to refinement because the property it enforces is structural, not score-based.

7 Discussion

7.1 Diversity by design vs. post-hoc reranking

Two paradigms can be distinguished for ensuring diversity in multi-hop retrieval. Post-hoc reranking, surveyed in §2, treats diversity as an optimization problem at the ranking stage by penalizing redundancy among retrieved candidates. A chunking-stage structural constraint treats it as a design problem in how chunks are formed, by restricting which units enter the candidate pool before any ranking

Method	avg M	HotPotQA NDCG@5	HotPotQA Δ	2Wiki NDCG@5	2Wiki Δ
Fixed $M{=}1$	1.00	**.777**	—	**.805**	—
adaptive_combined	1.18	.774	$-.003^{***}$	.795	$-.010^{***}$
adaptive_gap	1.39	.773	$-.004^{***}$	.780	$-.025^{***}$
adaptive_doc_rel	1.66	.770	$-.007^{***}$	.780	$-.025^{***}$
adaptive_mass	1.79	.766	$-.011^{***}$	.754	$-.051^{***}$

Table 6: Adaptive selection vs. fixed $M{=}1$ on the test split. avg M is averaged across HotPotQA and 2Wiki to summarize each heuristic's tendency to increase M above 1. All adaptive variants underperform fixed $M{=}1$ in NDCG@5 on both datasets, with significance from paired t-test ($^{***}p < 0.001$). The drop is monotonic in avg M.

decision is made.

Our results in §6 and Appendix D show that for the specific subgoal of multi-hop document coverage, the chunking-stage constraint matches what per-document cap produces in terms of unique document count, while MMR reranking moves in the same direction but does not fully match it. The chunking-stage constraint goes further by selecting query-relevant anchors and expanding around them, controlling both the candidate pool and the boundaries of each unit. Reranking optimizes the ordering of an already-retrieved candidate set, while the chunking-stage constraint determines what enters the pool in the first place.

This finding does not imply that post-hoc reranking is unnecessary in general. Diversity in retrieval covers a broader set of concerns, including the relevance-diversity trade-off, freshness, and personalization, which cannot be reduced to document coverage. Reranking remains a useful tool for those signals. What our analysis argues for is explicit separation. Subgoals that can be guaranteed structurally should be handled at the chunking stage, while those requiring scoring-based judgment should be handled at the ranking stage.

7.2 Implications for retrieval evaluation

Our findings also reveal a limitation in how multi-hop retrieval is commonly evaluated. NDCG measures whether relevant spans appear near the top of the ranking, but it does not measure how those spans are distributed across documents. Two retrieval results with the same NDCG can offer the reader very different evidence depending on this distribution.

The encoder robustness results in §5.3 make this concrete. On 2Wiki with BGE-large, R@5 and NDCG@5 point in opposite directions for the same comparison. QASS underperforms Fixed in R@5 but outperforms in NDCG, both significantly. Only the downstream QA comparison clarifies which method serves the reader better.

The mismatch arises because NDCG rewards a relevant span at a high rank but does not penalize the fact that the same document occupies multiple ranks. In multi-hop QA, the reader needs to combine evidence from K slots to produce an answer, and intra-document redundancy reduces the number of distinct sources available regardless of where in the ranking the spans appear. Coverage-aware diagnostics, such as the unique document count we report, capture this structural dimension that NDCG by itself misses. Designing more principled coverage-aware metrics for multi-hop retrieval is a natural direction for follow-up work.

8 Conclusion

We presented a controlled empirical study of retrieval unit design in multi-hop QA. Within the same algorithm, varying a single hyperparameter (M) produces an EM swing of $+4.26$ on HotPotQA and $+2.77$ on 2Wiki at $K{=}5$, and this swing traces directly to cross-document coverage as a structural property of the retrieved set. The advantage of $M{=}1$ resists refinement by four adaptive selection heuristics and holds under a stronger encoder. For multi-hop retrieval, the design principle that emerges is direct. Structural constraints at the chunking stage match what post-hoc reranking would achieve for cross-document coverage, while preserving the query-anchored span boundaries that improve ranking quality, without additional reranking hyperparameters or a separate ranking pass.

Limitations

The evaluation is conducted only in the distractor setting. HotPotQA and 2Wiki each provide around ten candidate documents per query with a relatively high proportion of supporting documents. Open-domain retrieval expands the candidate pool

by several orders of magnitude and reduces the proportion of supporting documents accordingly. Whether the cross-document coverage mechanism we identify behaves the same way in this setting remains an open question.

Beyond the dataset choice, our encoder selection is also limited. We evaluate retrieval unit design with only two encoders, `all-MiniLM-L6-v2` and `BGE-large-en-v1.5`. The advantage of $M{=}1$ shrinks with the stronger encoder, as observed in §5.3, suggesting that even stronger encoders may continue this trend. How retrieval unit design interacts with encoder capacity remains underexplored beyond the range we tested.

The retrieval setting is also restricted to single-shot. Iterative multi-hop methods such as MDR (Xiong et al., 2020) and Baleen (Khattab et al., 2021) produce the second-hop query from first-hop results, and the role of $M{=}1$ in such iterative pipelines is not addressed here.

Finally, all downstream QA results use a single reader, `gpt-5-nano`. Whether the same EM patterns hold with other readers, especially those with different scales or architectures, requires additional verification.

References

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *Proceedings of the second ACM international conference on web search and data mining*, pages 5–14.

James Allan, Courtney Wade, and Alvaro Bolivar. 2003. Retrieval and novelty detection at the sentence level. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, page 314–321, New York, NY, USA. Association for Computing Machinery.

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. *Preprint*, arXiv:1911.10470.

Sinchana Ramakanth Bhat, Max Rudat, Jannis Spiekermann, and Nicolas Flores-Herr. 2025. Rethinking chunk size for long-document retrieval: A multi-dataset analysis. *Preprint*, arXiv:2505.21700.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 335–336, New York, NY, USA. Association for Computing Machinery.

Charles L.A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, page 659–666, New York, NY, USA. Association for Computing Machinery.

Marti A Hearst. 1997. Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

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

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Preprint*, arXiv:2112.09118.

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, Online. Association for Computational Linguistics.

Marcin Kaszkiel and Justin Zobel. 1997. Passage retrieval revisited. *SIGIR Forum*, 31(SI):178–185.

Marcin Kaszkiel and Justin Zobel. 2001. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology*, 52(4):344–364.

Omar Khattab, Christopher Potts, and Matei Zaharia. 2021. Baleen: robust multi-hop reasoning at scale via condensed retrieval. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

Ian Roberts and Robert Gaizauskas. 2004. Evaluating passage retrieval approaches for question answering. In *European Conference on Information Retrieval*, pages 72–84. Springer.

Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*, volume 4. Now Publishers Inc.

Ishneet Sukhvinder Singh, Ritvik Aggarwal, Ibrahim Allahverdiyev, Muhammad Taha, Aslihan Akalin, Kevin Zhu, and Sean O'Brien. 2025. Chunkrag: Novel llm-chunk filtering method for rag systems. *Preprint*, arXiv:2410.19572.

Jörg Tiedemann and Jori Mur. 2008. Simple is best: Experiments with different document segmentation strategies for passage retrieval. In *Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering*, pages 17–25, Manchester, UK. Coling 2008 Organizing Committee.

Anastasios Tombros and Mark Sanderson. 1998. Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 2–10, New York, NY, USA. Association for Computing Machinery.

Jesse Vig, Alexander Fabbri, Wojciech Kryscinski, Chien-Sheng Wu, and Wenhao Liu. 2022. Exploring neural models for query-focused summarization. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1455–1468, Seattle, United States. Association for Computational Linguistics.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Zhitong Wang, Cheng Gao, Chaojun Xiao, Yufei Huang, Shuzheng Si, Kangyang Luo, Yuzhuo Bai, Wenhao Li, Tangjian Duan, Chuancheng Lv, Guoshan Lu, Gang Chen, Fanchao Qi, and Maosong Sun. 2025. Document segmentation matters for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8063–8075, Vienna, Austria. Association for Computational Linguistics.

Jennifer Williams, Sharon Tam, and Wade Shen. 2014. Finding good enough: A task-based evaluation of query biased summarization for cross-language information retrieval. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 657–669, Doha, Qatar. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, and 1 others. 2020. Answering complex open-domain questions with multi-hop dense retrieval. *arXiv preprint arXiv:2009.12756*.

Huanwei Xu, Lin Xu, and Liang Yuan. 2025. Clap: Coreference-linked augmentation for passage retrieval. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, CIKM '25, page 3612–3622. ACM.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Cheng Xiang Zhai, William W. Cohen, and John Lafferty. 2003. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, page 10–17, New York, NY, USA. Association for Computing Machinery.

Appendix

A QA Prompt Template

All downstream QA experiments (§5.4) use `gpt-5-nano` as reader. The top-K retrieved spans are concatenated into a single context string, separated by newlines, and passed to the model with the following prompt:

```
Answer the following question based on
the provided context passages.
Give ONLY the answer, nothing else. Keep
it as short as possible.

Context: {context}
Question: {question}
Answer:
```

Predictions are evaluated using EM against the gold answer.

B Grid Search Results

This appendix reports the full development-set results for the search ranges described in §4. Configurations are selected by NDCG@5. Tables 7 and 8 report development-set results for all configurations on HotPotQA and 2Wiki, sorted by NDCG@5 within each method. The selected configurations (top of each block) are evaluated on the test split, as reported in the main paper.

C BGE-large Encoder Results

Using `BGE-large-en-v1.5` (Xiao et al., 2023) as the encoder, we report retrieval performance across $K \in \{1, 2, 3, 5\}$ with paired bootstrap significance

Config	R@5	NDCG@5	MHCov@5
QASS			
qass_M1_w6 †	.827	**.780**	.666
qass_M1_w5	.824	.779	.664
qass_M2_w6	.825	.777	.659
qass_M1_w4	.817	.777	.662
qass_M3_w6	.823	.776	.656
qass_M1_w3	.808	.773	.664
qass_M2_w5	.819	.773	.650
qass_M3_w5	.817	.769	.648
qass_M2_w4	.809	.762	.636
qass_M1_w2	.773	.762	.653
qass_M3_w4	.806	.751	.630
qass_M2_w3	.797	.747	.622
qass_M3_w3	.791	.727	.608
qass_M2_w2	.771	.714	.600
qass_M3_w2	.769	.697	.591
Fixed			
fixed_cs5 †	.806	**.753**	.639
fixed_cs4	.789	.739	.625
fixed_cs3	.757	.709	.593
fixed_cs2	.693	.654	.562
fixed_cs1	.575	.545	.525
Semantic			
semantic_p20 †	.685	**.635**	.593
semantic_p30	.677	.629	.591
semantic_p40	.663	.616	.578
semantic_p50	.639	.596	.566

Table 7: Grid search on HotPotQA dev split (N=1481). † denotes the configuration selected by NDCG@5.

Config	R@5	NDCG@5	MHCov@5
QASS			
qass_M1_w5 †	.789	**.806**	.615
qass_M1_w6	.796	.806	.613
qass_M1_w4	.775	.805	.614
qass_M1_w2	.735	.804	.617
qass_M1_w3	.760	.804	.617
qass_M2_w6	.801	.780	.590
qass_M2_w5	.794	.776	.584
qass_M2_w4	.783	.770	.577
qass_M2_w3	.769	.765	.575
qass_M3_w6	.795	.760	.576
qass_M2_w2	.749	.756	.563
qass_M3_w5	.789	.751	.567
qass_M3_w4	.779	.739	.560
qass_M3_w3	.763	.727	.554
qass_M3_w2	.744	.723	.543
Fixed			
fixed_cs5 †	.765	**.750**	.559
fixed_cs4	.759	.747	.563
fixed_cs3	.742	.731	.564
fixed_cs2	.707	.692	.563
fixed_cs1	.580	.555	.511
Semantic			
semantic_p20 †	.684	**.658**	.508
semantic_p30	.675	.651	.510
semantic_p40	.667	.644	.520
semantic_p50	.651	.630	.517

Table 8: Grid search on 2Wiki dev split (N=2515). † denotes the configuration selected by NDCG@5.

(n_{iter}=10,000). Tables 9 and 10 summarize results on the test split. The QASS NDCG advantage persists across all K values, with the largest gaps on 2Wiki at K=2 (+0.033) and K=3 (+0.027), both significant at $p < 0.001$.

D Coverage Baselines

We compare QASS M=1 against five coverage-enforcing baselines applied to simpler retrievers.

- **Per-document cap.** In top-K selection, candidates are considered in score order and any candidate from a document title already in the selected set is skipped, enforcing at most one unit per document.
- **Sentence + cap.** Sentence retrieval with per-document cap.
- **Fixed + cap.** Fixed chunking (cs=5) with per-document cap.
- **Fixed + MMR.** Fixed chunks reranked with MMR (Carbonell and Goldstein, 1998) at λ=0.7, using cosine similarity between MiniLM embeddings.
- **Doc-then-span.** Each document is scored by its maximum sentence similarity; the top-K documents are selected with the best sentence per document.
- **BM25 + cap.** BM25 sentence retrieval with per-document cap.

Table 11 reveals two patterns.

First, every baseline that reaches the structural maximum unique document count (sentence + cap, Fixed + cap, doc-then-span, BM25 + cap, all at 4.99/5.00) does so by a different retrieval unit type. The single-anchor construction in QASS, however, is the only one that maintains NDCG@5 above 0.77. Per-document coverage in itself is necessary but not sufficient. The unit type matters too.

Second, Fixed + MMR (λ=0.7) performs essentially the same as Fixed + cap (NDCG@5 difference within 0.01 on both datasets). For multi-hop document coverage in the distractor setting, the discrete cap and the continuous MMR penalty converge to similar outcomes. The gap to QASS M=1 is preserved regardless of which post-hoc mechanism is used.

K	Method	R@K	NDCG@K	MRR	MHCov@K	uDocs	ΔNDCG@K	p
1	Fixed	.458	.910	.910	.000	1.00	—	—
	QASS	.457	.914	.914	.000	1.00	+.003	.25
2	Fixed	.781	.811	.942	.595	1.99	—	—
	QASS	.785	.814	.943	.598	2.00	+.003	.09
3	Fixed	.868	.863	.946	.758	2.96	—	—
	QASS	.872	.868	.948	.758	3.00	+.005**	.001
5	Fixed	.936	.897	.948	.882	4.87	—	—
	QASS	.940	.902	.950	.885	4.99	+.005***	.0001

Table 9: BGE-large results on HotPotQA test split (N=5924). ΔNDCG@K is QASS minus Fixed. p-value from paired bootstrap with n_{iter}=10,000.

K	Method	R@K	NDCG@K	MRR	MHCov@K	uDocs	ΔNDCG@K	p
1	Fixed	.423	.954	.954	.000	1.00	—	—
	QASS	.425	.957	.957	.000	1.00	+.004*	.03
2	Fixed	.675	.818	.971	.420	1.95	—	—
	QASS	.684	.851	.973	.429	2.00	+.033***	<.001
3	Fixed	.766	.832	.973	.545	2.81	—	—
	QASS	.764	.860	.975	.551	3.00	+.027***	<.001
5	Fixed	.866	.857	.974	.721	4.53	—	—
	QASS	.853	.881	.975	.730	5.00	+.024***	<.001

Table 10: BGE-large results on 2Wiki test split (N=10061). ΔNDCG@K is QASS minus Fixed. p-value from paired bootstrap with n_{iter}=10,000.

E Adaptive Anchor Selection: Heuristic Details

The main paper (§6.3) reports that four adaptive heuristics underperform QASS fixed M=1. This appendix defines the heuristics and provides dataset-level detail. The window radius is fixed at w=5 throughout, matching the QASS configuration used on 2Wiki.

E.1 Heuristic Definitions

For each document d_i, let $s_{i,1} \geq s_{i,2} \geq \ldots \geq s_{i,n_i}$ denote sentence-level query relevance scores in descending order. Each heuristic outputs $M_i \in \{1, 2, 3\}$ based on a different signal. Thresholds are chosen heuristically and held fixed across datasets.

gap (θ=0.05). Absolute score difference between consecutive top sentences:

$$M_i = \begin{cases} 1 & \text{if } s_{i,1} - s_{i,2} > \theta \\ 3 & \text{if } s_{i,1} - s_{i,2} \leq \theta,\ s_{i,2} - s_{i,3} < \theta \\ 2 & \text{otherwise} \end{cases}$$

mass (θ=0.7). Normalized cumulative score mass near the top. Scores are shifted to be non-negative and rescaled to sum to one, giving $\tilde{s}_{i,j}$:

$$M_i = \begin{cases} 1 & \text{if } \tilde{s}_{i,1} > \theta \\ 2 & \text{if } \tilde{s}_{i,1} + \tilde{s}_{i,2} > \theta + 0.15 \\ 3 & \text{otherwise} \end{cases}$$

doc_rel (θ=0.4). Maximum sentence score as a proxy for document-level query relevance:

$$M_i = \begin{cases} 1 & \text{if } s_{i,1} < \theta \\ 2 & \text{if } \theta \leq s_{i,1} < \theta + 0.15 \\ 3 & \text{if } s_{i,1} \geq \theta + 0.15 \end{cases}$$

combined. Conjunction of *doc_rel* and *gap*. Additional anchors are added only when the document has high overall relevance ($s_{i,1} \geq 0.4$) *and* small gap between top sentences ($s_{i,1} - s_{i,2} \leq 0.05$). Otherwise M_i=1.

E.2 Dataset-Level Detail

Table 12 reports per-dataset $\bar{M}$, R@5, NDCG@5, and MHCov@5. The summary in the main paper averages $\bar{M}$ across the two datasets; the breakdown here shows that the same heuristic produces a quite different anchor count on each dataset.

Method	HotPotQA				2Wiki			
	R@5	NDCG@5	MHCov@5	uDocs@5	R@5	NDCG@5	MHCov@5	uDocs@5
Sentence + cap	.512	.500	.620	4.99	.532	.532	.626	5.00
Doc-then-span	.511	.500	.619	4.99	.532	.532	.625	5.00
BM25 + cap	.516	.505	.540	4.99	.367	.343	.350	5.00
Fixed + cap	.806	.759	.659	4.99	.746	.743	.583	5.00
Fixed + MMR	.807	.760	.642	4.85	.763	.753	.550	4.69
Fixed cs=5 (Table 1)	.809	.761	.646	4.80	.771	.756	.557	4.61
QASS M=1 (Table 1)	**.822**	**.778**	.660	**4.99**	**.786**	**.805**	.595	**5.00**

Table 11: Coverage baselines on the test split. Fixed + cap reaches the same unique document count as QASS M=1 (4.99 and 5.00) but NDCG@5 remains lower by 0.019 and 0.062 respectively. Sentence-level baselines fall well below multi-sentence span methods.

Heuristic	$\bar{M}$	R@5	NDCG@5	MHCov@5
HotPotQA				
fixed M=1	1.00	**.820**	**.777**	**.661**
combined	1.24	.817	.774	.653
gap	1.48	.816	.773	.651
doc_rel	1.82	.815	.770	.646
mass	2.06	.812	.766	.640
2Wiki				
fixed M=1	1.00	.786	**.805**	**.595**
combined	1.11	.788	.795	.583
gap	1.29	**.790**	.780	.583
doc_rel	1.50	.786	.780	.561
mass	1.52	.787	.754	.559

Table 12: Adaptive heuristics vs. fixed M=1, dataset-level detail on test split.

E.3 Distribution of Selected Anchor Counts

Table 13 reports the percentage of documents assigned each M_i under each heuristic. The conservativeness ranking aligns with the NDCG@5 ranking in §6.3: heuristics that stay close to M=1 (combined) lose the least, while heuristics that send many documents to M=3 (mass, doc_rel) lose the most.

Heuristic	% M=1	% M=2	% M=3
HotPotQA			
combined	82.8	10.7	6.4
gap	66.6	18.2	15.2
doc_rel	38.4	40.9	20.7
mass	36.8	20.1	43.1
2Wiki			
combined	91.9	5.5	2.6
gap	79.4	12.5	8.1
doc_rel	63.5	23.3	13.3
mass	69.3	9.4	21.4

Table 13: Per-document anchor count distribution under each heuristic. Rows ordered by $\bar{M}$ within each dataset.

Two further observations from the distribution. First, mass is the most aggressive on HotPotQA (sending 43.1% of documents to M=3) but not on 2Wiki, where it stays at 21.4%. The score-mass signal responds differently to the two datasets even at the same threshold. Second, combined, which combines *doc_rel* and *gap*, is much more conservative than either component alone, since both conditions must fail simultaneously to add anchors.

F Retrieval Performance by Question Type

This appendix provides the per-type retrieval performance underlying §6.2.

Method	R@5	NDCG@5	MHCov@5
Bridge (n=4,737)			
Sentence	.528	.499	.479
BM25	.541	.522	.470
Contriever	.507	.474	.469
ColBERT-style	.609	.546	.349
Semantic	.637	.593	.536
QASS default	.729	.663	.526
Fixed	.778	.733	.587
QASS (M=1)	**.792**	**.752**	**.604**
Comparison (n=1,187)			
Sentence	.749	.699	.743
BM25	.596	.537	.519
Contriever	.680	.599	.686
ColBERT-style	.791	.695	.636
Semantic	.839	.777	.826
QASS default	.911	.822	.842
Fixed	.935	.872	.881
QASS (M=1)	**.939**	**.880**	**.883**

Table 14: HotPotQA test split, per-type retrieval performance at K=5. QASS uses M=1, w=6.

Method	R@5	NDCG@5	MHCov@5
Bridge-comp (n=2,212)			
Sentence	.492	.551	.087
BM25	.348	.377	.014
Contriever	.405	.425	.059
ColBERT-style	.546	.590	.072
Semantic	.569	.627	.107
QASS default	.618	.680	.104
Fixed	.684	.734	.187
QASS (M=1)	**.694**	**.759**	**.213**
Comparison (n=2,440)			
Sentence	.835	.780	.841
BM25	.646	.556	.574
Contriever	.792	.694	.806
ColBERT-style	.856	.765	.766
Semantic	.875	.817	.852
QASS default	.922	.849	.867
Fixed	.922	.869	.871
QASS (M=1)	**.931**	**.900**	**.893**
Compositional (n=4,164)			
Sentence	.515	.481	.518
BM25	.289	.262	.225
Contriever	.471	.430	.553
ColBERT-style	.659	.589	.521
Semantic	.639	.604	.497
QASS default	.720	.695	.559
Fixed	.733	.716	.565
QASS (M=1)	**.756**	**.790**	**.616**
Inference (n=1,245)			
Sentence	.510	.423	.492
BM25	.274	.207	.319
Contriever	.463	.368	.549
ColBERT-style	.650	.548	.473
Semantic	.641	.578	.497
QASS default	.725	.645	.552
Fixed	.754	.706	.573
QASS (M=1)	**.768**	**.750**	**.621**

Table 15: 2Wiki test split, per-type retrieval performance at K=5 (QASS uses M=1, w=5).

QASS with M=1 achieves the highest NDCG@5 across all retrieval methods on every question type. The largest retrieval gains over Fixed appear on 2Wiki compositional (+0.074) and inference (+0.044), while the QA gains in §6.2 are largest on bridge-comp and comparison.