
VL-DocIR: A Benchmark for Vision-Based Long Document Information Retrieval

Simon Ott^{1,*}, Yufan Chen^{1,*,†}, Ruiping Liu¹, Junwei Zheng¹, Jiale Wei¹
 Di Wen¹, Kunyu Peng¹, Jiaming Zhang², Rainer Stiefelhagen¹
¹Karlsruhe Institute of Technology ²Hunan University
 simon.ott@student.kit.edu, yufan.chen@kit.edu

Abstract

Vision-based document retrieval has improved rapidly, but current evaluation settings still emphasize short documents and single relevant pages. This leaves an important gap between leaderboard performance and realistic long-document retrieval, where the evidence needed to answer a query may span multiple pages or even multiple documents. We present VL-DocIR, a page-level benchmark for evidence-complete visual long-document retrieval. VL-DocIR is built from 29,641 HTML-born documents rendered into 388,548 page images from Wikipedia, arXiv, PubMed, and SEC proxy statements. It contains 271,760 queries across 23 domains and six evidence structures, covering single-page, within-document multi-page, and cross-document evidence configurations. Evidence annotations are grounded to rendered pages and HTML element identifiers, and queries are filtered with a cleaning pipeline targeting clarity, correctness, closedness, and the absence of explicit layout references. We evaluate leading single-vector and multi-vector visual retrievers together with diagnostic retrieval settings such as document-context scoring, query splitting, reranking, and round-robin ensembles. The strongest evaluated configuration reaches 69.3 nDCGAll@10 and 80.6 RecallAll@10 under structure-balanced Macro evaluation, but source- and evidence-structure breakdowns show that cross-document retrieval remains substantially harder than single-page retrieval, and that document-context scoring is especially helpful for within-document multi-page queries. VL-DocIR exposes these failure modes and provides a more demanding target for visual long-document retrieval. The new dataset and code will be made publicly available at VL-DocIR Project.

1 Introduction

Document retrieval is increasingly moving from OCR-heavy text pipelines to page-image retrieval, due to information lost in plain-text representations for charts, tables, figures, and layouts. This shift has produced strong retrievers for visual documents, while the evaluation remains incomplete. A system can perform well at the page level while still failing on the long documents that matter in practice, where the evidence for a query is distributed across multiple pages, mixed across text and visuals, or split across separate documents.

This gap is especially acute for scientific articles, encyclopedic web pages, and financial reports. These documents are long, visually heterogeneous, and structurally uneven, *i.e.*, relevant information may begin in prose, continue in a table, and be disambiguated by a

*Equal contribution.

†Corresponding author: Yufan Chen (yufan.chen@kit.edu).

later figure. Benchmarks that reward retrieving any relevant page do not test whether a system can recover the full evidence set required to answer queries.

We introduce **VL-DocIR**, a benchmark for vision-based long document retrieval, which contains 29,641 rendered documents, 388,548 page images, and 271,760 queries over 23 domains. Each benchmark item consists of a natural-language query, multiple evidence pages, and the image corpus. The benchmark covers six query types ranging from single-page single-evidence queries to cross-document queries whose evidence spans multiple documents. This structure can evaluate not only overall retrieval quality, but also the performance for distributed evidence.

With the VL-DocIR benchmark, we have evaluated leading single-vector and multi-vector document retrievers together. Cross-document retrieval remains the hardest setting by a wide margin, while document-context scoring is most beneficial when evidence remains inside the same long document. These results show that long document visual retrieval is not only a scale problem, but also a different evaluation regime with different bottlenecks.

Our contributions are fourfold:

- We introduce a large-scale vision-based long document retrieval benchmark with 29,641 documents, 388,548 rendered pages, 271,760 queries, and 23 domains.
- We define a benchmark construction pipeline that combines HTML-grounded page rendering, evidence-page configuration, VLM-based query generation, and query cleaning to produce page-level retrieval supervision at scale.
- We provide a systematic evaluation of leading visual document retrievers and complementary OCR-based, text-based, and hybrid baselines, together with benchmark-oriented analysis settings, including query splitting, document-context scoring, reranking, and simple ensembles.
- We show that multi-vector retrieval is consistently stronger than single-vector retrieval, that document-context scoring is especially effective for within-document multi-page queries, and that cross-document evidence coverage remains limited among the evaluated configurations.

2 Related Work

2.1 Vision-based Document Retrieval

Visual document retrieval has shifted from OCR-first pipelines toward direct page-image embedding. DSE treats screenshots as retrieval units [1], while ColPali introduced late interaction over visual page embeddings together with the ViDoRe benchmark [2]. The ViDoRe line has continued with harder visual retrieval and RAG settings, including ViDoRe V2 and V3 [3, 4]. Recent visual document retrieval work further studies multilingual and long-document retrieval [5, 6, 7, 8, 9] and efficient representations [10, 11]. Broader reasoning-intensive multimodal retrieval benchmarks and pipelines show that retrieval still struggles when visual inputs require reasoning beyond semantic matching [12, 13]. Visual RAG systems [14] and recent multimodal retrieval methods [15, 16, 17, 18, 19, 20, 21] further strengthen page-image retrieval.

2.2 Long Document and Visual Document Understanding

Document VQA benchmarks [22, 23, 24, 25, 26, 27, 28, 29] evaluate answering over document, scene-text, infographic, and chart images. Structured-document QA datasets [30, 31, 32, 33] extend the setting to web pages, slides, and mixed textual-visual evidence. MP-DocVQA extends document VQA to multi-page documents [34], while long document benchmarks [35, 36, 37] test long-context multimodal understanding, reasoning, and locating. These benchmarks establish that layout, tables, figures, and cross-page evidence matter for downstream understanding.

2.3 Document AI and Vision-Language Models

Visual document systems build on Document AI models that combine text, layout, and image signals. LayoutLM series [38, 39, 40] established pretraining objectives for visually rich document understanding, while generative methods [41, 42, 43, 44, 45] explore multimodal and OCR-free document understanding modeling. General vision-language models [46, 47, 48, 49, 50, 51] provide the perception and language capabilities increasingly used for document parsing, query generation, and visual retrieval.

2.4 Retrieval and Embedding Benchmarks

RAG systems [52, 53, 54, 55, 56, 57, 58] rely on sparse, dense, and late-interaction retrieval foundations. Text retrieval benchmarks [59, 60] broaden evaluation across tasks and domains. VLM2Vec [61] extends embedding evaluation to multimodal inputs, and later universal multimodal embedding work [62, 63] further adapts MLLMs to retrieval settings. Long-context benchmarks [64, 65] test reasoning over multiple textual contexts. Document-image benchmarks [6, 5, 1] further shift evaluation toward retrieving evidence. These resources motivate heterogeneous evaluation and expose the limits of single-domain leaderboards.

3 Benchmark

VL-DocIR is a page-level document retrieval benchmark for visually-rich long documents. It pairs a corpus of rendered document pages with natural-language questions whose supporting evidence is localized to one or more pages. Unlike standard page-level retrieval evaluations that reward retrieving any relevant page, VL-DocIR measures the evidence-complete retrieval, *i.e.*, a system succeeds only when it retrieves the full set of evidence pages.

3.1 Task Definition and Stratification

Each benchmark item is a tuple $(q, E(q), \mathcal{C})$, where q is a natural-language query, $E(q)$ is the set of one or more ground-truth evidence pages, and $\mathcal{C}$ is the full corpus of rendered page images. At evaluation time, the retriever ranks pages from $\mathcal{C}$, and the item is considered evidence-complete at cutoff k only if every page in $E(q)$ appears in the top- k list. VL-DocIR stratifies benchmark items along three dimensions, *i.e.*, evidence structure, source provenance, and domain coverage.

Evidence structure. Evidence structure defines the retrieval target and forms the task taxonomy with six compact evidence types, *i.e.*, **Page-Single** for one evidence on one page, **Page-Multi** for multiple evidences on one page, **Doc-Adjacent** for adjacent pages, **Doc-Near** for nearby non-adjacent pages, **Doc-Far** for far-apart pages in the same document, and **Cross-Doc** for pages across documents. This taxonomy separates ordinary page lookup from retrieval settings where the complete evidence set is distributed across pages or documents.

Source provenance. Our VL-DocIR benchmark aims to evaluate vision-based information retrieval methods on an extensive, diverse document corpus. To this end, we select documents from four data sources: *Wikipedia*, *arXiv*, *PubMed*, and *SEC Proxy Statements*. These sources differ in document length, layout regularity, visual density, and distractor density. Source-level evaluation provides diagnostic slices of retrieval performance under different document regimes rather than merely reporting metadata.

Domain coverage. Domain coverage mainly focuses on the semantic setting of each benchmark query across 23 domains. It is used to analyze whether retrieval performance generalizes across content regimes, not only across evidence structures or data sources. Domain labels prevent the benchmark from being dominated by a single application area, support domain-level failure analysis, and allow future work to report robustness beyond aggregate retrieval scores. Domain-level comparisons should therefore be interpreted with source provenance.

3.2 Visual Document Database Creation Pipeline

The intention behind creating the visual document database is to build a large corpus of long document images with rich visual content, including figures, tables, formulas, diagrams, and structured layouts. We render HTML-born documents directly so that visual layout and document structure remain available as retrieval signals. The retrieval unit in VL-DocIR is a rendered page image. In the benchmark, models are allowed to use visual, OCR-based, text-only, or hybrid internal representations to return ranked page identifiers from the same corpus. In this paper, our primary experiments focus on vision-based retrievers.

Document Index Creation. We first build a source-specific document index that records the source, document identifier, domain label, and collection metadata needed for HTML document rendering and evidence grounding. The indexing stage is designed to provide broad domain coverage, related-document neighborhoods for cross-document question generation, and documents that can be rendered with element-level evidence alignment. The collection strategy is source-specific, *e.g.*, Wikipedia is expanded from seed articles through linked-article selection, arXiv combines survey-seed expansion with category-level sampling to ensure enough papers, PubMed uses open-access articles with license metadata, and SEC uses one recent proxy statement per selected company to reduce year-to-year duplication. Domain labels are assigned before rendering and propagated to pages and generated questions.

Document Rendering and Page Construction. We render documents with a headless Chrome browser using Playwright³. Before rendering, we trigger lazy-loaded content, remove invisible HTML elements, remove source-specific clutter such as banners, inject identifiers into HTML elements, and record their bounding boxes. These identifiers and bounding boxes are later used to map VLM-selected evidence elements to rendered pages. We then split each full-document screenshot into page images of 2000×4000 pixels, using a rendering scale factor of 2 for sharper images. A naive fixed-height cut can split tables, figures, or text blocks across pages. To reduce such artifacts, the splitter uses whitespace and bounding-box heuristics to choose page boundaries. When necessary, it adds whitespace at the bottom of a page to move cut-off content to the next page. The stored bounding boxes are updated after this adjustment so that the rendered image and element coordinates remain aligned.

Visual Document Data Statistics. The final document dataset contains 29,641 documents and 388,548 rendered pages. As shown in Tab. 1, Wikipedia contributes many short documents, with an average length of 6.49 pages. arXiv and PubMed documents are longer, averaging 15.50 and 12.26 pages, respectively. SEC proxies are fewer in number but much longer, averaging 41.97 pages. This source-level length variation allows VL-DocIR to distinguish ordinary page lookup from retrieval in long and structurally repetitive documents. The domain distribution is broad but not uniform. The largest domains include Astrophysics, Science, and Sports, while smaller domains remain represented for evaluation.

Source	Docs	Avg. Pages
Wikipedia	11,988	6.49
arXiv	10,846	15.50
PubMed	4,815	12.26
SEC Proxy	1,992	41.97
Total	29,641	13.11

Table 1: Benchmark statistics.

3.3 Benchmark Query Creation Pipeline

Our benchmark query creation pipeline consists of three stages: evidence-configuration sampling, VLM-based query generation, and query cleaning. Figure 1 summarizes the pipeline. The goal is to generate natural-language queries whose evidence can be localized to one or more rendered document pages.

Evidence-configuration Sampling. The first stage selects the evidence pages used to generate queries. The sampling strategy depends on the evidence structure. For *Page-Single* and *Page-Multi* queries, we sample from the top five pages of a document according to a visual-content heuristic score. This biases generation toward visually rich pages while still allowing text-based evidence. For *Page-Adjacent*, *Page-Near*, and *Page-Far* queries, we further select candidate page pairs using TF-IDF cosine similarity [66] and add a small content-richness term to avoid low-content pages. For *Cross-Doc* queries, we first pair nearby

³<https://playwright.dev/>

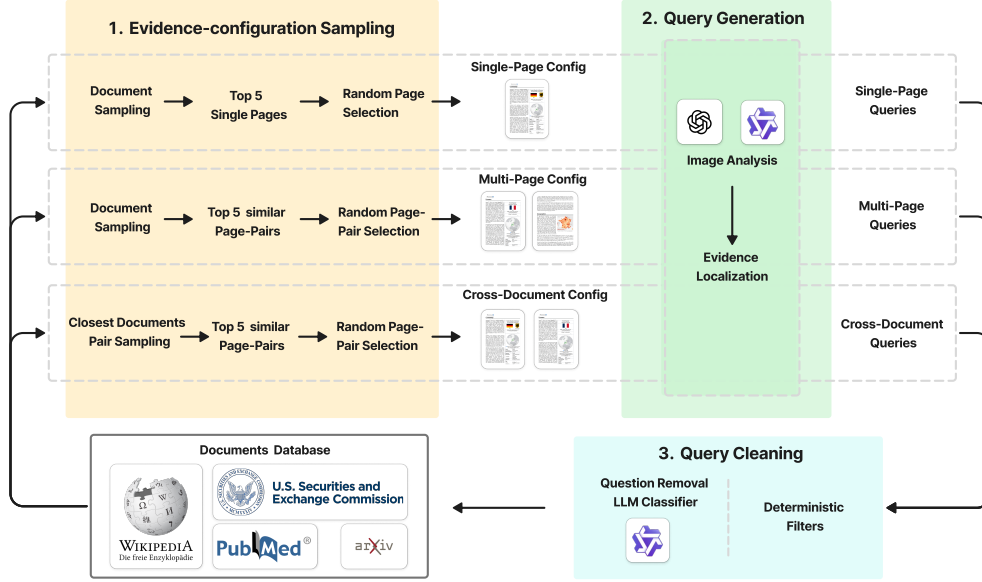


Figure 1: Benchmark query creation process overview.

documents using text embeddings of the document start text, and then apply the same page-pair selection strategy as in the within-document setting.

Query Generation. We generate the queries and corresponding answers based on the document pages with the help of Vision Language Models (VLMs). For each selected evidence configuration, the pipeline first creates a detailed analysis of the relevant page images. A VLM then receives the rendered page images, the underlying HTML with injected element identifiers, and the page-image analysis. The prompt asks the model to produce queries, answers, and evidence element identifiers. Prompts are specialized by source and evidence structure so that single-page, multi-page, and cross-document queries follow different generation constraints. After generation, we resolve valid HTML element identifiers to page-relative bounding boxes and evidence pages. Invalid generations, including malformed JSON outputs or invalid element identifiers, would be discarded.

Query Cleaning. Since VLM-generated queries can be ambiguous, underspecified, or incorrectly grounded, we apply a cleaning pipeline before constructing the final benchmark set. We evaluate each query using four criteria, *i.e.*, query clarity, answer correctness or plausibility when direct verification is unavailable, closedness, and absence of explicit layout references such as “in the table” or “in the figure”. We use LLM-generated criterion scores and an LLM-based binary removal classifier as the final filtering strategy. To evaluate the removal strategy, we manually annotate a query-cleaning evaluation set with 1,438 samples. As an additional deterministic filter, multi-page and Cross-Doc queries must contain evidence across different pages. After cleaning, the dataset is reduced from 388,389 generated queries to 271,760 final benchmark queries.

Benchmark Query Statistics. Table 2 shows the final query distribution by source and evidence configuration. The dataset contains 206,820 single-page queries and 64,940 distributed-evidence queries. This mixture preserves a large base of page-level retrieval examples while explicitly testing whether systems can recover complete evidence sets that span multiple pages or documents.

Evidence Structure	Data Source				Total
	Wiki	arXiv	PubMed	SEC	
Page-Single	26,306	14,027	14,128	31,735	86,196
Page-Multi	34,222	26,990	29,620	29,792	120,624
Doc-Adjacent	3,283	2,556	3,360	2,773	11,972
Doc-Near	1,172	634	1,000	586	3,392
Doc-Far	255	461	613	1,234	2,563
Cross-Doc	14,598	9,001	9,743	13,671	47,013
Total	79,836	53,669	58,464	79,791	271,760

Table 2: Benchmark Query Distribution Overview.

3.4 Annotation Quality and Adjudication

We conducted a stratified manual review of 800 entries covering all combinations of data source and evidence structure. For each source, we sampled 50 Cross-Doc entries and 30 entries for each of the other five structures to evaluate query construction and the correctness and necessity of the annotated evidence pages. Overall, 746 entries were entirely correct, yielding an observed accuracy of 93.25%, while the corresponding accuracy for Cross-Doc entries was 88.5%. Additionally, we evaluated the consistency between page mappings and bounding boxes for all 47,013 Cross-Doc queries and their 100,504 evidence elements. For 84.23% of the queries, the stored and configured page assignments matched perfectly with the assignments derived from the bounding boxes.

4 Experiment

Our experiments are designed for three purposes:

- How current visual retrievers perform when all evidence pages must be retrieved.
- Whether multi-vector page-image matching improves over single-vector embedding.
- For which retrieval-stage modifications help evidence coverage.

4.1 Experimental Setup

Evaluation Metrics. Standard Recall@k and nDCG@k [67] are informative for page-level relevance, but they can overstate success for distributed-evidence queries. For example, retrieving one of two required evidence pages is useful, but it is not sufficient for answering the item. We therefore report evidence-complete variants, **RecallAll@k** and **nDCGAll@k**.

For a query q , let $\pi_k(q) = (p_1, \dots, p_k)$ be the ranked top- k list returned by a retriever, and let $R_k(q) = \{p_1, \dots, p_k\}$ be the corresponding set of retrieved pages. Let $E(q)$ denote the set of ground-truth evidence pages. We define the evidence-completion indicator

$$c_k(q) = \mathbb{1}[E(q) \subseteq R_k(q)]. \quad (1)$$

RecallAll@k is the item-level completion indicator:

$$\text{RecallAll@k}(q) = c_k(q). \quad (2)$$

For nDCGAll@k, we first compute binary-relevance nDCG@k over the ranked list $\pi_k(q)$, where evidence pages have gain 1, and all other pages have gain 0:

$$\text{DCG@k}(q) = \sum_{j=1}^k \frac{\mathbb{1}[p_j \in E(q)]}{\log_2(j+1)}, \quad (3)$$

$$\text{IDCG@k}(q) = \sum_{j=1}^{\min(|E(q)|, k)} \frac{1}{\log_2(j+1)}, \quad (4)$$

$$\text{nDCG@k}(q) = \frac{\text{DCG@k}(q)}{\text{IDCG@k}(q)}. \quad (5)$$

The evidence-complete version is

$$\text{nDCGAll@k}(q) = c_k(q) \cdot \text{nDCG@k}(q). \quad (6)$$

Aggregation protocol. Our primary aggregate is structure-balanced Macro, which gives equal weight to the six evidence structures. For an item-level metric $m(q)$ and the query set Q_e of structure e , we compute

$$\text{Macro}(m) = \frac{100}{6} \sum_{e=1}^6 \frac{1}{|Q_e|} \sum_{q \in Q_e} m(q). \quad (7)$$

Micro is the mean over all queries multiplied by 100 and is reported as a secondary aggregate. Table 3 presents both aggregates at $k \in \{3, 5, 10\}$ for the same retrieval runs.

Evaluated Models. We evaluate the following information retrieval methods:

- **Single-Vector Visual Models:** GME Qwen2-VL [15], Jina v4 Single-Vector [16];
- **Multi-Vector Visual Models:** ColPali v1.3 [2], ColQwen2.5 [2], ColSmol500 [2], Jina v4 Multi-Vector [16], ColModernVBERT [2], Colnomic 7B [18], Tomoro ColQwen3 [19];
- **Text-based Models:** Tesseract 5.5.1 [68] with BM25 [69] and BGE-M3 [70], DOM with BM25 [69];
- **Hybrid Methods:** Colnomic 7B [18] with BM25 [69] by equal-weight Reciprocal Rank Fusion (RRF) [71].

All models are retrieved from the same corpus and receive only the query at evaluation.

Diagnostic retrieval settings. In addition to base retrievers, we evaluate four retrieval-stage diagnostic settings. These are used to characterize which failure modes current retrievers expose on our VL-DocIR benchmark.

- *Query Splitting* decomposes a query into sub-queries with an LLM and retrieves pages for each sub-query independently. We merge sub-query ranked lists in round-robin order while removing duplicate pages. This setting tests whether explicit query decomposition helps when a question requires evidence from multiple pages or documents.
- *Document-context scoring* augments page-level late interaction with a document-level context score. For each document, we cluster its page embeddings into a fixed number of document-context vectors using spherical k -means. At query time, a page p from document d receives the score

$$s(q, p, d) = \underbrace{\sum_{i=1}^{N_q} \max_{j \leq N_p} \langle E_q^{(i)}, E_p^{(j)} \rangle}_{s_{\text{page}}(q, p)} + \underbrace{\sum_{i=1}^{N_q} \max_{j \leq N_d} \langle E_q^{(i)}, E_d^{(j)} \rangle}_{s_{\text{doc}}(q, d)}. \quad (8)$$

Here N_q , N_p , and N_d denote the number of query vectors, page vectors, and document-context vectors. $E_q^{(i)}$, $E_p^{(j)}$, and $E_d^{(j)}$ are normalized query, page, and document-context embeddings.

- *Reranking* retrieves the top 25 pages with a multi-vector retriever and reranks these candidates with Jina Reranker M0 [72].
- *Round-robin ensembles* combine ranked lists from multiple settings by alternating between lists and skipping duplicate pages.

4.2 Results and Analysis

We organize the results into four diagnostic views. First, we compare representative retrievers across standard and evidence-complete metrics to quantify the overall performance gap. Second, we analyze retrieval-stage settings, including query splitting, document-context scoring, reranking, and round-robin ensembles. Third, we break down the average result by source and evidence structure to expose the long-document stress cases. Fourth, we aggregate performance by evidence structure to show the transition from single-page lookup to distributed-evidence retrieval. The full table over all models, metrics, and $k \in \{3, 5, 10\}$ is reported in the Appendix.

Overall performance. Table 3 shows that evidence-complete retrieval remains substantially harder than ordinary page-level relevance. The comparison between Jina V4 single-vector and multi-vector variants shows that the multi-vector version improves nDCG@10 and RecallAll@10 by 11.6 and 10.5 percentage points under Macro, respectively (12.9 and 10.0 under Micro), suggesting that token-level or patch-level matching is important

Model	Aggregate	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
		@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColSmol 500	Macro	49.43	53.36	56.26	52.89	61.04	68.62	35.30	42.11	47.33	39.01	49.10	58.15
	Micro	54.98	57.76	59.95	61.35	67.70	73.97	50.63	53.87	56.58	57.03	63.45	69.81
ColModernVBERT	Macro	47.33	50.91	53.72	50.48	57.93	65.30	33.28	39.31	44.29	36.73	45.70	54.35
	Micro	51.69	54.39	56.55	57.73	63.92	70.11	47.45	50.57	53.20	53.51	59.73	65.96
GME Qwen2-VL	Macro	55.27	59.51	62.72	59.35	68.10	76.42	41.08	48.82	55.10	45.33	56.70	67.34
	Micro	60.18	63.16	65.44	67.20	73.94	80.35	55.34	59.01	62.11	62.37	69.38	76.22
Jina V4 SV	Macro	48.31	52.61	55.80	52.40	61.33	69.69	34.70	42.09	48.03	38.80	49.86	60.24
	Micro	52.55	55.75	58.24	59.58	66.87	74.01	48.17	51.93	55.10	55.16	62.60	70.00
Jina V4 MV	Macro	60.92	64.99	67.79	64.43	72.78	79.99	45.69	53.56	59.59	49.61	60.92	70.77
	Micro	67.24	69.83	71.74	73.36	79.17	84.47	61.80	65.15	67.98	68.01	74.16	80.03
ColQwen3	Macro	65.54	69.09	71.36	68.15	75.35	81.13	50.39	57.86	63.20	53.74	64.11	72.42
	Micro	72.54	74.53	75.89	77.44	81.80	85.49	67.13	69.92	72.15	72.21	77.00	81.24
ColPali v1.3	Macro	42.03	45.47	48.26	45.04	52.18	59.52	28.93	34.55	39.41	32.14	40.58	49.13
	Micro	46.10	48.74	50.92	51.76	57.80	64.08	42.21	45.20	47.85	47.87	53.90	60.26
ColPali v1.3 (Rerank)	Macro	57.25	59.73	60.92	58.96	63.94	66.94	43.30	48.42	51.12	45.88	52.93	57.16
	Micro	62.97	64.26	64.95	66.38	69.19	71.03	58.31	60.18	61.33	61.93	65.05	67.17
ColQwen2.5	Macro	58.68	62.56	65.15	62.08	70.02	76.72	44.15	51.48	56.72	47.93	58.51	67.23
	Micro	64.94	67.41	69.21	70.84	76.37	81.40	59.84	62.97	65.51	65.82	71.63	77.04
ColQwen2.5 (Rerank)	Macro	65.82	69.22	71.15	68.55	75.45	80.31	51.33	58.45	62.88	54.69	64.60	71.62
	Micro	72.61	74.49	75.65	77.26	81.38	84.50	67.33	70.02	71.94	72.14	76.73	80.36
ColNomic	Macro	63.85	67.61	70.05	66.84	74.53	80.78	48.73	56.38	61.82	52.36	63.06	71.76
	Micro	70.46	72.75	74.35	75.95	81.05	85.46	65.06	68.17	70.68	70.71	76.24	81.24
ColNomic (Ctx)	Macro	64.85	69.37	72.57	68.26	77.51	85.72	49.33	58.64	66.03	53.28	66.31	78.19
	Micro	69.75	72.51	74.48	75.70	81.88	87.38	63.50	67.15	70.07	69.69	76.28	82.26
ColNomic (Rerank)	Macro	66.82	70.50	72.76	69.77	77.24	83.00	52.25	59.84	65.00	55.77	66.37	74.62
	Micro	73.77	75.87	77.29	78.72	83.33	87.15	68.44	71.42	73.76	73.52	78.63	83.11
ColNomic (Ensemble)	Macro	68.42	72.68	75.45	72.03	80.67	87.73	53.58	62.60	69.28	57.43	70.04	80.64
	Micro	75.00	77.39	79.03	80.59	85.85	90.30	69.43	72.89	75.64	75.02	80.95	86.23
OCR-BM25	Macro	51.30	55.40	58.47	54.99	63.50	71.53	37.56	44.59	50.24	41.45	51.91	61.69
	Micro	56.06	59.16	61.53	62.70	69.76	76.48	51.55	55.20	58.26	58.18	65.36	72.34
OCR-dense (BGE-M3)	Macro	22.90	24.97	26.59	24.58	28.90	33.16	14.14	16.92	19.35	15.92	20.26	24.70
	Micro	23.37	24.95	26.24	26.66	30.28	34.00	20.81	22.45	23.86	24.10	27.55	31.12
DOM-BM25	Macro	47.81	51.49	54.34	51.12	58.75	66.19	35.24	41.67	46.96	38.82	48.37	57.44
	Micro	51.11	53.86	56.00	57.07	63.34	69.41	47.04	50.31	53.07	53.01	59.44	65.72
ColNomic (Ctx) with OCR-BM25 (RRF)	Macro	62.88	67.70	71.23	67.11	77.01	86.10	47.87	57.14	65.13	52.30	65.60	78.67
	Micro	67.74	70.97	73.26	75.00	82.26	88.62	62.27	66.34	69.75	69.53	77.13	84.24
ColNomic (Rerank) with OCR-BM25 (RRF)	Macro	61.38	65.84	69.10	65.41	74.60	82.96	46.78	54.95	61.86	51.07	62.90	74.39
	Micro	67.30	70.39	72.61	74.29	81.25	87.37	62.16	66.07	69.34	69.16	76.47	83.30

Table 3: **Average Retrieval Results** on our VL-DocIR benchmark. Each model family is reported with the best variants under structure-balanced Macro (primary) and query-averaged Micro (secondary). SV/MV denotes single-vector and multi-vector retrieval. Ensemble denotes a combined setting. Bold indicates the best score within each aggregation protocol. The more detailed results and analysis are reported in the Appendix.

for recovering complete evidence sets. Among base visual retrievers, ColQwen3 and ColNomic are the strongest, while the strongest evaluated configuration, ColNomic (Ensemble), reaches 69.28 nDCGAll@10 and 80.64 RecallAll@10 under Macro (75.64 and 86.23 under Micro). The gain comes from combining page-local matching, document context, and reranking within the existing retrieval backbone. Among the text baselines, OCR-BM25 achieves the highest Macro RecallAll@10 of 61.69. Its fusion with ColNomic (Ctx) increases Macro RecallAll@10 from 78.19 to 78.67, while reducing nDCGAll@10 from 66.03 to 65.13.

Retrieval-stage setting effects. Table 4 shows that retrieval-stage modifications improve different aspects of the task. Reranking generally improves nDCGAll over the base retrievers by refining candidate ordering. Document-context scoring often improves RecallAll by helping retrieve a more complete evidence set when relevant pages belong to the same document. For ColPali, Ctx improves Macro RecallAll@10 from 49.13 to 65.69 (from 60.26 to 69.90 under Micro), while Rerank gives the best nDCGAll@10 under both protocols. For ColQwen2.5, Ctx gives the best Macro RecallAll@10, whereas B+Ctx gives the best Micro RecallAll@10; Rerank gives the best nDCGAll@10 under both protocols. For ColNomic, B+Ctx gives the best Macro nDCGAll@10 and RecallAll@10 among the settings in this table. Under Micro, Rerank gives the best nDCGAll@10, while B+Ctx gives the best RecallAll@10. The three-list ColNomic ensemble further improves both metrics under both protocols, as

Family	Aggregation	Base	Q-Split	Ctx	B+Ctx	Rerank
ColPali	Macro	39.41/49.13	–	50.87/ 65.69	48.69/62.55	51.12 /57.16
	Micro	47.85/60.26	–	53.48/ 69.90	53.14/69.12	61.33 /67.17
ColQwen2.5	Macro	56.72/67.23	57.48/69.19	62.80/ 75.78	62.74/75.68	62.88 /71.62
	Micro	65.51/77.04	65.64/78.63	66.85/80.10	67.81/ 81.41	71.94 /80.36
ColNomic	Macro	61.82/71.76	–	66.03/78.19	67.06 / 79.60	65.00/74.62
	Micro	70.68/81.24	–	70.07/82.26	71.96/ 84.81	73.76 /83.11

Table 4: **Retrieval-stage setting effects** measured by nDCG@10 and Recall@10. Ctx denotes document-context scoring, B+Ctx combines the ranked lists from base and document-context retrieval, Rerank denotes reranking, and Q-Split denotes query splitting. Bold indicates the best value for each metric.

shown in Table 3. These differences highlight distinct challenges in candidate generation, document-level coverage, and candidate ordering.

Source and evidence-structure stress.

Figure 2 shows that aggregate retrieval performance hides the main stress cases. Single-page queries are close to saturated for the strongest configuration across all sources, but distributed-evidence queries are much harder. Within-document multi-page retrieval remains challenging, especially for Wikipedia and SEC proxy statements, where long documents contain repeated structures and visually plausible distractors. In Cross-Doc retrieval, Recall@5 of ColNomic’s best variant drops to 49.8 on Wikipedia, 26.6 on SEC, 14.1 on arXiv, and 20.0 on PubMed. Across all evaluated settings, the best Cross-Doc Recall@5 reaches 50.9 on Wikipedia, 26.6 on SEC, 18.0 on arXiv, and 28.0 on PubMed, showing that query splitting can be more effective than reranking for cross-document regimes.

		Page-Single	Page-Multi	Doc-Adjacent	Doc-Near	Doc-Far	Cross-Doc
Wikipedia		92.5	93.0	67.2	64.4	67.1	50.9
	arXiv	95.8	97.1	79.5	77.3	75.5	18.0
	PubMed	96.3	97.4	82.9	79.6	73.7	28.0
	SEC	89.0	90.2	62.1	59.7	57.3	26.6

Figure 2: Source and evidence-structure-specific Recall@5 for the best evaluated configuration. Each cell shows the Recall@5 for a given source and evidence structure. Darker cells indicate higher Recall@5.

Evidence-structure aggregate. Table 5 summarizes the same trend at the evidence structure level. The benchmark contains a large base of single-page queries, but the source-macro Recall@5 drops sharply once evidence must be recovered from multiple pages. The hardest transition is not from Page-Single to Page-Multi, but from within-document evidence to cross-document evidence. This supports the central motivation of VL-DocIR, *i.e.*, long-document retrieval cannot be reduced to single-page ranking over a larger corpus.

Evidence structure	#Queries	Macro R@5
Page-Single	86,196	93.4
Page-Multi	120,624	94.4
Doc-Adjacent	11,972	71.4
Doc-Near	3,392	69.0
Doc-Far	2,563	66.3
Cross-Doc	47,013	27.6

Table 5: Evidence-structure aggregate view for ColNomic B+Ctx+Rank. R@5 denotes Recall@5 macro-averaged over the four data sources.

Scope of the experimental claims. The experiments support retrieval-stage claims about current visual retrievers on VL-DocIR, *i.e.*, multi-vector retrieval is stronger than single-vector retrieval, document-context scoring helps mainly when evidence remains within one document, and cross-document evidence coverage remains weak. They do not show that vision-only retrieval is superior to OCR-only, text-only, or hybrid retrieval, and they do not evaluate end-to-end question answering. A high Recall@10 means that all required pages are retrieved, which does not guarantee that a downstream task will answer correctly.

4.3 Qualitative Results and Analysis

Fig. 3 presents the top-5 document pages retrieved by ColQwen2.5, both with and without reranking, for a Cross-Doc query. This example demonstrates that reranking can substantially improve retrieval performance compared with the base model. Specifically, the base ColQwen2.5 model retrieves only one of the two relevant evidence pages, whereas the reranked results correctly identify both evidence pages. Although the pages retrieved by the base model are thematically plausible, as they also contain information about space missions, they refer to incorrect mission numbers. This indicates that the base retriever can capture high-level semantic relevance but may fail to distinguish fine-grained details. In contrast, the reranking stage better resolves such subtle distinctions, leading to more accurate retrieval of the required document evidence.

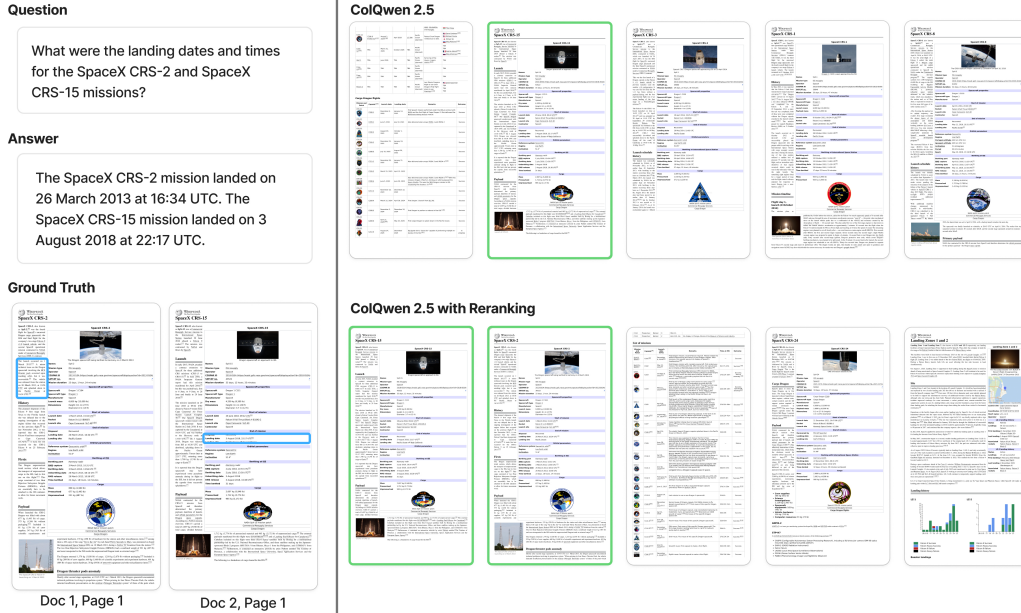


Figure 3: Cross-Doc query top-5 retrieval results of ColQwen2.5 and ColQwen2.5 with reranking. The blue rectangles indicate the evidence bounding-boxes. The green rectangles highlight correctly retrieved pages.

5 Conclusion

VL-DocIR introduces an evidence-complete page-retrieval benchmark for visually rich long documents, with single-page, multi-page, and cross-document evidence configurations. Unlike standard page-level retrieval evaluations, VL-DocIR requires systems to recover the complete set of evidence pages for each question. Our experiments show that multi-vector page-image retrieval is substantially stronger than single-vector retrieval, document-context scoring is most useful when evidence remains within one document, and cross-document evidence retrieval remains the most challenging regime. The strongest evaluated configuration reaches 75.6 nDCG@10 and 86.2 Recall@10, while source- and evidence-structure breakdowns show that aggregate scores still mask large cross-document failures.

Limitations. VL-DocIR evaluates evidence-page retrieval with page-level rather than exhaustive element level grounding. The current corpus focuses on rendered HTML-born documents, leaving scanned and handwritten documents outside scope. The query set is VLM-generated and automatically cleaned, so generator style and filtering criteria may still influence the benchmark distribution.

Acknowledgments and Disclosure of Funding

This work was supported in part by Helmholtz Association of German Research Centers, in part by the Ministry of Science, Research and the Arts of Baden-Württemberg (MWK) through the Cooperative Graduate School Accessibility through AI-based Assistive Technology (KATE) under Grant BW6-03, and in part by the National Natural Science Foundation of China under Grant No. 62503166, in part by the Hunan Provincial Research and Development Project under Grant number 2026QK3018, and in part by the Yuelushan Industrial Innovation Center. This work was partially performed on the HoreKa supercomputer funded by the MWK and by the Federal Ministry of Education and Research, partially on the HAICORE@KIT partition supported by the Helmholtz Association Initiative and Networking Fund, and partially on bwForCluster Helix supported by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG.

References

- [1] Xueguang Ma, Sheng-Chieh Lin, Minghan Li, Wenhui Chen, and Jimmy Lin. Unifying multi-modal retrieval via document screenshot embedding. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6492–6505, Miami, Florida, USA, November 2024. Association for Computational Linguistics.
- [2] Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, CELINE HUDELOT, and Pierre Colombo. Colpali: Efficient document retrieval with vision language models. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [3] Quentin Macé, António Loison, and Manuel Faysse. Vidore benchmark v2: Raising the bar for visual retrieval. *arXiv preprint arXiv:2505.17166*, 2025.
- [4] António Loison, Quentin Macé, Antoine Edy, Victor Xing, Tom Balough, Gabriel Moreira, Bo Liu, Manuel Faysse, Céline Hudelot, and Gautier Viaud. Vidore v3: A comprehensive evaluation of retrieval augmented generation in complex real-world scenarios. *arXiv preprint arXiv:2601.08620*, 2026.
- [5] Jian Chen, Ming Li, Jihyung Kil, Chenguang Wang, Tong Yu, Ryan Rossi, Tianyi Zhou, Changyou Chen, and Ruiyi Zhang. Visr-bench: An empirical study on visual retrieval-augmented generation for multilingual long document understanding, 2025.
- [6] Kuicai Dong, Yujing Chang, Xin Deik Goh, Dexun Li, Ruiming Tang, and Yong Liu. Mmdocir: Benchmarking multi-modal retrieval for long documents, 2025.
- [7] Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3docvqa: Multi-modal multi-page multi-document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 6237–6247, 2025.
- [8] Manan Suri, Puneet Mathur, Franck Dernoncourt, Kanika Goswami, Ryan A. Rossi, and Dinesh Manocha. Visdom: Multi-document qa with visually rich elements using multimodal retrieval-augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6088–6109, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.
- [9] Qiuchen Wang, Ruixue Ding, Zehui Chen, Weiqi Wu, Shihang Wang, Pengjun Xie, and Feng Zhao. Vidorag: Visual document retrieval-augmented generation via dynamic iterative reasoning agents, 2025.
- [10] Yibo Yan, Mingdong Ou, Yi Cao, Xin Zou, Shuliang Liu, Jiahao Huo, Yu Huang, James Kwok, and Xuming Hu. Beyond the grid: Layout-informed multi-vector retrieval with parsed visual document representations, 2026.
- [11] Yibo Yan, Mingdong Ou, Yi Cao, Jiahao Huo, Xin Zou, Shuliang Liu, James Kwok, and Xuming Hu. Visual late chunking: An empirical study of contextual chunking for efficient visual document retrieval, 2026.

- [12] Abdelrahman Abdallah, Mohamed Darwish Mounis, Mahmoud Abdalla, Mahmoud SalahEldin Kasem, Mostafa Farouk Senussi, Mohamed Mahmoud, Mohammed Ali, Adam Jatowt, and Hyun-Soo Kang. Mm-bright: A multi-task multimodal benchmark for reasoning-intensive retrieval, 2026.
- [13] Mahmoud SalahEldin Kasem, Mohamed Mahmoud, Mostafa Farouk Senussi, Mahmoud Abdalla, Abdelrahman Abdallah, and Hyun-Soo Kang. Marvel: Multimodal adaptive reasoning-intensive expand-rerank and retrieval, 2026.
- [14] Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, and Maosong Sun. VisRAG: Vision-based retrieval-augmented generation on multi-modality documents. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [15] Xin Zhang, Yanzhao Zhang, Wen Xie, Mingxin Li, Ziqi Dai, Dingkun Long, Pengjun Xie, Meishan Zhang, Wenjie Li, and Min Zhang. Gme: Improving universal multimodal retrieval by multimodal llms. *arXiv preprint arXiv:2412.16855*, 2024.
- [16] Michael Günther, Saba Sturua, Mohammad Kalim Akram, Isabelle Mohr, Andrei Ungureanu, Bo Wang, Sedigheh Eslami, Scott Martens, Maximilian Werk, Nan Wang, and Han Xiao. jina-embeddings-v4: Universal embeddings for multimodal multilingual retrieval. In *Proceedings of the 5th Workshop on Multilingual Representation Learning (MRL 2025)*, pages 531–550, Suzhuo, China, 2025. Association for Computational Linguistics.
- [17] Paul Teietche, Quentin Macé, Max Conti, Antonio Loison, Gautier Viaud, Pierre Colombo, and Manuel Faysse. Modernvbert: Towards smaller visual document retrievers. *arXiv preprint arXiv:2510.01149*, 2025.
- [18] Nomic Team. Nomic embed multimodal: Interleaved text, image, and screenshots for visual document retrieval. <https://nomic.ai/blog/posts/nomic-embed-multimodal> (last accessed: 21. April 2026), 2025.
- [19] Xin Huang and Kye Min Tan. Beyond text: Unlocking true multimodal, end-to-end rag with tomoro colqwen3. <https://tomoro.ai/insights/beyond-text-unlocking-true-multimodal-end-to-end-rag-with-tomoro-colqwen3> (last accessed: 21. April 2026), 2025.
- [20] Zilin Xiao, Qi Ma, Mengting Gu, Chun cheng Jason Chen, Xintao Chen, Vicente Ordonez, and Vijai Mohan. Metaembed: Scaling multimodal retrieval at test-time with flexible late interaction. In *The Fourteenth International Conference on Learning Representations*, 2026.
- [21] Omri Uzan, Asaf Yehudai, Roi Pony, Eyal Shnarch, and Ariel Gera. Guided query refinement: Multimodal hybrid retrieval with test-time optimization. In *International Conference on Learning Representations (ICLR)*, 2026.
- [22] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2200–2209, 2021.
- [23] Minesh Mathew, Viraj Bagal, Rubén Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.
- [24] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019.
- [25] Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluís Gomez, Marçal Rusinol, Minesh Mathew, C. V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. Scene text visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4291–4301, 2019.
- [26] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 947–952, 2019.

- [27] Jordy Van Landeghem, Łukasz Borchmann, Rubén Tito, Michał Pietruszka, Dawid Jurkiewicz, Rafał Powalski, Paweł Józiak, Sanket Biswas, Mickaël Coustaty, and Tomasz Stanisławek. Icdar 2023 competition on document understanding of everything (dude). In *Document Analysis and Recognition – ICDAR 2023*, pages 420–434. Springer Nature Switzerland, 2023.
- [28] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics.
- [29] Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. Plotqa: Reasoning over scientific plots. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1527–1536, March 2020.
- [30] Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 3277–3287, 2021.
- [31] Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. Multimodalqa: Complex question answering over text, tables and images. In *International Conference on Learning Representations*, 2021.
- [32] Xingyu Chen, Zihan Zhao, Lu Chen, JiaBao Ji, Danyang Zhang, Ao Luo, Yuxuan Xiong, and Kai Yu. Websrc: A dataset for web-based structural reading comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4173–4185, 2021.
- [33] Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13636–13645, 2023.
- [34] Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. Hierarchical multimodal transformers for multipage docvqa. *Pattern Recogn.*, 144(C), December 2023.
- [35] Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, Pan Zhang, Liangming Pan, Yu-Gang Jiang, Jiaqi Wang, Yixin Cao, and Aixin Sun. MMLONGBENCH-DOC: Benchmarking long-context document understanding with visualizations. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.
- [36] Zhaowei Wang, Wenhao Yu, Xiyu Ren, Jipeng Zhang, Yu Zhao, Rohit Saxena, Liang Cheng, Ginny Wong, Simon See, Pasquale Minervini, Yangqiu Song, and Mark Steedman. Mmlongbench: Benchmarking long-context vision-language models effectively and thoroughly, 2025.
- [37] Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and Cheng-Lin Liu. LongDocURL: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1135–1159, Vienna, Austria, July 2025. Association for Computational Linguistics.
- [38] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1192–1200, 2020.
- [39] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, Online, 2021. Association for Computational Linguistics.

- [40] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4083–4091, 2022.
- [41] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 993–1003, 2021.
- [42] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *Computer Vision – ECCV 2022*, pages 498–517. Springer, 2022.
- [43] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvasi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding. In *Proceedings of the 40th International Conference on Machine Learning*, pages 18893–18912. PMLR, 2023.
- [44] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. In *International Conference on Learning Representations*, 2024.
- [45] Dongsheng Wang, Natraj Raman, Mathieu Sibue, Zhiqiang Ma, Petr Babkin, Simerjot Kaur, Yulong Pei, Armineh Nourbakhsh, and Xiaomo Liu. Docllm: A layout-aware generative language model for multimodal document understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.
- [46] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.
- [47] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19730–19742. PMLR, 2023.
- [48] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [49] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023.
- [50] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report, 2025.
- [51] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.
- [52] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781, 2020.
- [53] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*, 2021.
- [54] Thibault Formal, Benjamin Piwowarski, and Stephane Clinchant. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2288–2292, 2021.
- [55] Omar Khattab and Matei Zaharia. 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*, pages 39–48, 2020.

- [56] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 3715–3734, 2022.
- [57] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022.
- [58] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktaschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.
- [59] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.
- [60] Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. MTEB: Massive text embedding benchmark. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.
- [61] Ziyang Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhui Chen. VLM2vec: Training vision-language models for massive multimodal embedding tasks. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [62] Ting Jiang, Minghui Song, Zihan Zhang, Haizhen Huang, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and Fuzhen Zhuang. E5-v: Universal embeddings with multimodal large language models, 2024.
- [63] Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. Mm-embed: Universal multimodal retrieval with multimodal llms, 2024.
- [64] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun’ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium, October–November 2018. Association for Computational Linguistics.
- [65] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench: A bilingual, multitask benchmark for long context understanding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3119–3137, Bangkok, Thailand, August 2024. Association for Computational Linguistics.
- [66] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November 1975.
- [67] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002.
- [68] Anthony Kay. Tesseract: an open-source optical character recognition engine. *Linux J.*, 2007(159):2, July 2007.
- [69] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009.
- [70] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.
- [71] Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’09, page 758–759, New York, NY, USA, 2009. Association for Computing Machinery.

- [72] JinaAI. jina-reranker-m0. <https://huggingface.co/jinaai/jina-reranker-m0> (last accessed: 21. April 2026), 2025. Hugging Face Model Card.
- [73] OpenAI. GPT-5 nano. <https://developers.openai.com/api/docs/models/gpt-5-nano> (last accessed: 6 May 2026), 2025.
- [74] OpenAI. GPT-4.1 mini. <https://developers.openai.com/api/docs/models/gpt-4.1-mini> (last accessed: 6 May 2026), 2025.

A Benchmark Details

We present in the following the benchmark creation details and further statistics for the document dataset and benchmark query dataset.

A.1 Document Collection

Wikipedia: For the collection of Wikipedia articles, we build an intelligent spider that recursively searches for new Wikipedia articles. As a starting point, we manually select 78 seed articles across domains such as geography, sports, biology, chemistry, arts, history, and economics. Additionally, we craft for each seed article a query instruction for further articles that are related to the seed article but likely do not represent a duplication of article content. For example, the query instruction for the Wikipedia article about "Mount Everest" is *"Search for articles which are about mountains. The title shall contain "Mount" or the name of a peak."*. Starting from the seed articles, the intelligent spider recursively searches in the Wikipedia articles for all referenced articles. The spider employs an LLM to decide which articles to select. The prompt for the LLM includes the query instruction together with the referenced articles' titles.

Wikipedia Spider Article Selection Prompt

```
System: You are a precise classifier. Your OUTPUT MUST be
valid JSON and nothing else. Do NOT provide chain-of-thought or
step-by-step reasoning. Short reasons only (<=10 words).

User: Description of desired wikipedia articles:
"<QUERY_INSTRUCTION>"

Max results: <INT>

Candidate articles (index: title):
<LIST_OF_ARTICLE_TITLE_NAMES_WITH_INDEX_NUMBERS>

Return JSON with this exact shape:
{ "matches": [ { "index": <int>, "score": <float between 0 and 1>,
"reason": "<short reason>" }, ... ] }

Only include articles you consider relevant (omit irrelevant ones).
Sort matches by score descending. If none match, return {"matches":
[]}.

```

arXiv: In contrast to the Wikipedia spider, the arXiv article indexer does not employ an LLM for determining which articles to visit next. Instead, the indexer simply searches for all available links to other arXiv articles. The reason for this decision is that scientific article titles are often ambiguous and do not fit into a clearly specified category, such as e.g. "Mount Everest" in the case of Wikipedia articles. Initially, we use a similar approach to the Wikipedia article indexing with 41 manually searched seed articles. The seed articles consist of recent survey articles. We select them because they contain references to closely related articles, are suitable for Cross-Doc query generation, and also contain more likely HTML versions due to their recency.

Since this approach leads to less than 500 arXiv articles, we extend the initial seed articles by 20,000 articles from the arXiv categories astrophysics, condensed matter, quantitative biology, and electrical engineering resulting in a total of 36,132 articles. We decide to create a larger initial index for arXiv to account for removal due to license restrictions and invalidly converted arXiv articles.

PubMed: In contrast to arXiv, PubMed articles contain fewer references with URLs to open-access articles. Therefore, we decide to retrieve a list of 9,488 open-access articles from

2025 to 2026 directly from PubMed Central instead of a recursive search. Furthermore, we collect for each article licensing information to later filter out restricted articles.

SEC Proxy Statements: We select about 2k large publicly listed companies’ most recent proxy statements, especially due to their high document length. Since proxy statements contain overlapping information between multiple years, leading to ambiguous retrieval problems, we decide to use only one proxy statement per company.

A.2 Document Items

The main components of a document item are the equally-sized page images, the underlying HTML with injected identifiers, and pixel-exact measured bounding-boxes. Figure 4 visualizes the bounding-boxes for an example document page cutout. Apart from this, we also collect abstracts, licensing, and citations of the document.



Figure 4: Bounding-box annotation visualization for the Wikipedia article about the Portugal national football team.

A.3 Document Dataset Statistics

In this subsection, we provide further insights into the structure of our document dataset. Figure 5 illustrates the domain distribution in the document dataset. Our VL-DocIR benchmark covers 23 domains: *Arts*, *Entertainment*, *Science*, *Technology*, *Geography*, *Sports*, *History*, *Politics*, *Economics*, *Astrophysics*, *Condensed Matter Physics*, *Quantitative Biology*, *Electrical Engineering*, *General Computer Science*, *Artificial Intelligence*, *Statistics*, *Surgical Case Reports*, *Public Health Reports*, *Environmental Reports*, *Medical Reports*, *Chemistry*, *Bioinformatics*, and *Financial Reports*. The most occurring domain is *Astrophysics* with 3,497 documents followed by the Wikipedia article domains *Science* and *Sports* with 3,053

and 3,007 documents. In general the document domain distribution is balanced and we have at least 50 documents per domain.

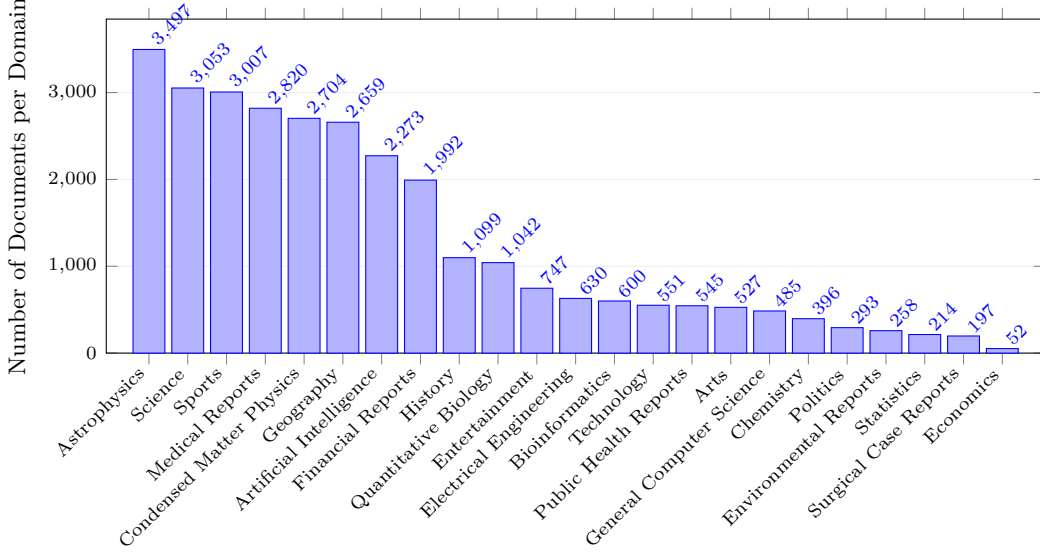


Figure 5: Domain distribution in the document dataset.

Figure 6 visualizes the distribution of pages per document for the four data sources. The distribution of Wikipedia highlights that most articles have a length between 1 and 10. Both scientific arXiv and PubMed article lengths are within 5 and 25. The distribution of SEC Proxy Statement page lengths deviates from the others, with the majority of documents being between 20 and 60 pages long.

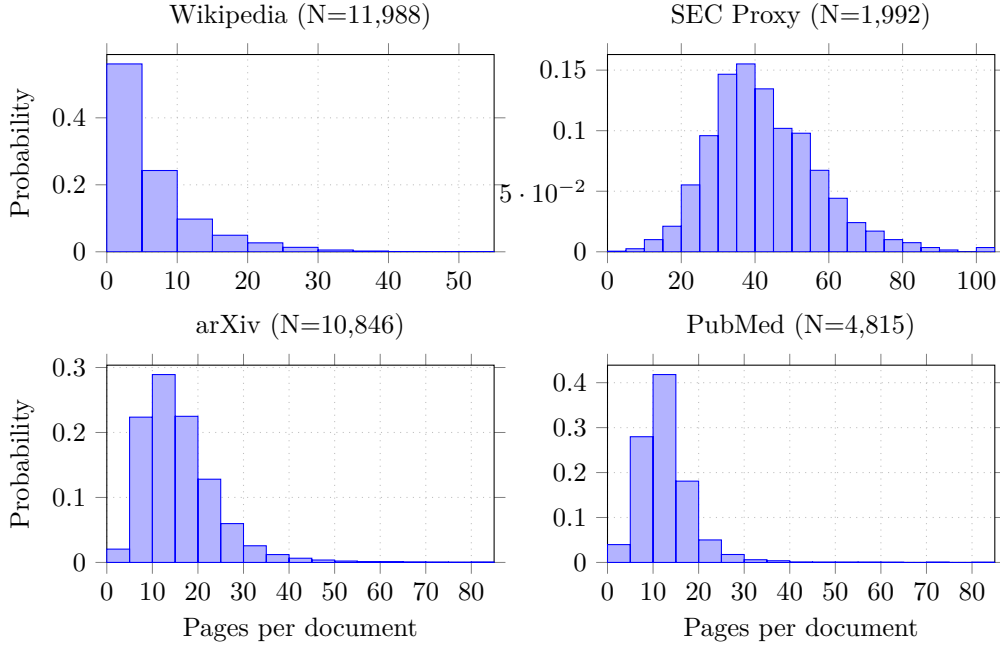


Figure 6: Page-count distributions with 5-page bins for every data source. The plots show that Wikipedia articles tend to be shorter documents, whereas SEC Proxy Statements are much longer. The page count distribution of arXiv and PubMed articles is similar, with the most articles having between 10 and 15 pages.

A.4 Query Generation

For each query generation configuration entry, the query generation consists of two steps. First, the page images are described using a VLM. In the second step, we prompt a VLM to generate 10 queries for the given input including the page summaries, underlying HTML with the injected element identifiers, and the raw page images. We use two different combinations of VLMs. The first combination is to use for both steps Qwen3-VL-32B [51], and the second combination uses for the first step the OpenAI GPT-5-nano model [73] and for the second step GPT-4.1-mini [74]. The cause for splitting into two different model combinations is that initial qualitative tests have revealed a better query quality for the OpenAI combination, however, being accompanied by higher costs. Therefore, we process for each data source and for each query evidence structure type 2,000 configuration entries with Qwen3-VL-32B and 500 configuration entries with the OpenAI models combination.

Page Image Analysis: For the page image analysis step, we use different prompts for the Qwen3-VL-32B and GPT-5-nano since GPT-5-nano is a reasoning model that requires fewer instructions than Qwen3-VL-32B. The multi-page and Cross-Doc query creators apply the page image analysis to each page individually. Since the query generation step utilizes also the HTML, we focus the prompt on describing visual content in the images.

Page Image Analysis Prompt for Qwen3-VL-32B (temperature: 0.8)

User: Describe in detail all visual elements (images, diagrams, graphics, photos, maps, charts, or tables) appearing on the document page shown in the attached image.

For each visual element, specify:

- A detailed description of what it depicts, including visible objects, people, text labels, numerical data, and symbols. Include i.e. facts, statistics, and concepts visible in pictures. Pay attention to details and name all of them.

Avoid summarizing or explaining meaning, just describe what is visibly present.

Page Image Analysis Prompt for GPT-5-nano

User: Describe in detail what the images/diagrams/graphics/photos/maps in the attached image include

General Query Generation: For the query generation step, the prompts differ depending on the data source, query evidence structure type, and VLM. However, they share in general all the same prompt structure consisting of the task description, output format description, and the input data. The goal of each VLM prompt is to get a list of queries together with the answers and HTML element identifiers as evidence.

Query Generation Prompt Structure

User:

<TASK_DESCRIPTION (high-level, criteria, good and bad examples)>
<OUTPUT_DESCRIPTION (thinking steps, JSON output format)>
<INPUT_DATA (article titles, page image descriptions, HTML with element identifiers)>

The task description always starts with a high-level instruction explaining the task to the VLM. Following, we provide a detailed description of the criteria for the queries to generate together with good and bad query examples. Furthermore, we formulate restrictions for the thinking process. The output description contains a definition of necessary thinking steps for the VLM, as well as the required JSON output format. We instruct the VLM to use a split string to separate thinking from JSON, since we do not use a reasoning model to reduce

compute time. The input data prompt part comprises article titles, the previously generated page image descriptions, and the HTML with element identifiers.

In case of missing splitters, invalid JSON format, or invalid HTML element identifier, we discard the generation output. For valid output, we extract the list of queries, answers, and HTML element identifiers. Thereon, we determine via the HTML element identifiers the exact page-relative bounding boxes and store the results.

Single-Page Query Generation Prompt for Wikipedia (Qwen3-VL-32B)

User:

Task:

You are given:

1. A **document page image**.
2. The corresponding **HTML snippet**, which provides element IDs and layout information.
3. A description of the document page image

Your task is to generate **10 questions** about the textual and visual content of the document on the given page.

Good questions are:

- clearly and precisely formulated
- unambiguous,
- fact-based,
- closed: include all required information in the question to answer it without the knowledge of the document page.
- about knowledge, facts, trends, statistics, ... contained in graphics, charts, photos, tables, diagrams, maps, pictures
- do not explicitly reference the page layout. Do NOT use terms such as 'in the table', 'in the figure', or 'according to the image'! Instead, try to extract from the tables and figures facts that are answerable in general.
- do not explicitly reference the document/article (do NOT use 'in the article', 'in the document', 'in the report', ...). Instead, use the name of the article, entities, events, ... to specify the context of each question.
- independent from each other, which means that the necessary context to answer them is included in each question

Good question examples:

- Good: What is the bond angle between the two hydrogen atoms in a water molecule?
- Good: What are the neighboring countries of France?
- Good: What was the total annual revenue of Apple Inc. in the fiscal year 2025?
- Good: Which aircraft did Fokker assemble for the Royal Netherlands Air Force that arrived for the Royal International Air Tattoo in England in 2014?

Bad question examples:

- Bad: What are the bony structures visible in the CT images of Figure 4, and how are they represented?
- Bad: What is the name of the galaxy cluster shown in the photograph at the top of the page?
- Bad: Which persons can be seen in the picture?

Furthermore, make no assumption that a question answerer knows about which document you are asking the questions.

Imagine that the person who answers the questions does not know which document the questions are about. Use precise questions with named methods, concepts, experiments, ...

<MULTI_EVIDENCE_USAGE_INSTRUCTION (optional)>

The document is a Wikipedia article about <ARTICLE_TITLE>.

```

### Output
The answer consists of multiple parts:
Start with:
1) Reflect on the document and extract general knowledge of the document
2) Draft a list of questions and answers (draft more than 10 questions so
that you can select thereafter the best ones)
3) Analyze each question with the criteria above
4) Select the questions that best match the question requirement from above

Then print exactly __SPLITTER__ (after __SPLITTER__ comes only json)

Create the final JSON result: Return a valid JSON array following this
exact schema:

[
  {
    "question": "<string>",
    "answer": "<string>",
    "evidence": ["<element_id_1>", "<element_id_2>", ...]
  },
  ...
]

---

### Input:
This is the cleaned HTML snippet associated with the attached image:
""<HTML_WITH_ELEMENT_IDS>""

There is also a detailed description of the document page image:
""<PAGE_IMAGE_DESCRIPTION>""

```

Multi-Page Queries: The prompts for multi-page query generation are similar to the single-page query generation prompts. The main difference is that there are two pages given as input, and the instruction to create only queries that are unanswerable with a single page of evidence. The further query prompts can be seen in our published benchmark creation program. Between Doc-Adjacent, Doc-Near, and Doc-Far queries, there is not difference in the generation process and we assign the query evidence structure afterwards.

Cross-Doc Queries: For Cross-Doc queries, the prompt differs more strongly compared to single-page and multi-page queries. Since we have two documents, the evidence localization requires determining not only the HTML element identifier, which are integers starting with 0, but also the document identifier. In the initial prompt evaluation, we find out that when naming the documents with the terms "Document 1" and "Document 2", the VLMs tend to use those also inside the queries itself. Since this is not suitable for an open-domain queries dataset, we use identifiers "12345" and "54321" for the document identifiers to reduce their usage in queries. Furthermore, we reference them by using "first" and "second" instead of a number. As in the case of the multi-page query generation prompt, we instruct the model to use evidence of both documents to create queries.

A.5 Query Cleaning Evaluation Dataset

In this subsection, we provide further details regarding the evaluation of our query removal methods. We evaluate each method on our query removal evaluation dataset, which consists of 1,438 manually labeled queries.

LLM-Rating Method: The query LLM rating method leverages a Qwen3-VL-32B as an evaluator. Instead of directly determining the removal class, we let the VLM rate the query together with the answer by the following four criteria with scores within 0, 0.25, 0.5, 0.75, 1:

- Clarity: not ambiguously formulated

- Answer correctness: in case that we do not know the answer, we use plausibility as a measure
- Closedness: no reference of articles / studies / documents without naming them (e.g. not "in the document")
- No layout reference: no terms such as "in the table"

For the removal of queries, we test thresholds of 0.75, 0.5, 0.25, and 0.0 for queries to be removed. If any of the four criteria scores falls below a threshold, then we consider a query as removed. The classification results can be seen in Table 6. The false omission rate, where omission means, in this case, non-removal of a query, is for all four threshold nearly the same. The removal with threshold 0.75 achieves the best false omission rate with 13.37%.

	P	N		P	N		P	N		P	N
P	279	161	P	281	159	P	281	159	P	294	146
N	65	933	N	71	927	N	77	921	N	128	870

(a) Threshold = 0.0
FOR = 0.1472

(b) Threshold = 0.25
FOR = 0.1464

(c) Threshold = 0.5
FOR = 0.1472

(d) Threshold = 0.75
FOR = 0.1437

Table 6: Confusion matrices for different question LLM-evaluation removal classification thresholds (total samples = 1438). Positive P stands for a question being removed. Negative N stands for a question not being removed. FOR is the false omission rate, where omission means non-removal.

LLM-Classifier Method: Similar to the LLM rating removal method, the LLM classifier method makes use of Qwen3-VL-32B to evaluate a query. However, instead of producing scores, the strategy directly prompts the LLM to classify a query into an accept-class or discard-class.

Compared to the LLM rating remover, the performance is with a false omission rate of 17.82%. A potential cause for this is that the score creation instructs the LLM to think in more detail about the query and be more critical.

	Predicted P	Predicted N
Actual P	237	203
Actual N	62	936

Table 7: Confusion matrix for LLM query removal classification strategy (total samples = 1438, false omission rate = 0.1782).

Furthermore, we analyze the impact of combining the LLM classifier method with the query LLM rating method. In the combination, we remove queries if at least one of the two methods predicts a removal. For the query LLM rating method we use the threshold 0.75 for the rating scores because the false omission rate is the best for this value. As the confusion matrix Table 8 depicts, the combination of both methods drops the false omission rate compared to the pure LLM rating method further to 13.60%.

	Predicted P	Predicted N
Actual P	305	135
Actual N	140	858

Table 8: Confusion matrix for combination of LLM classifier and Query LLM rating removal methods (total samples = 1438, false omission rate = 0.1360).

Removal Classifiers: The input to the classifiers are text embeddings of the queries created with Alibaba text-embedding-v4. We choose 800 samples from the query removal dataset for

the training set, leaving 638 samples for the evaluation. The results indicate that XGBoost is with a false omission rate of 12.46% slightly better than logistic regression with 13.07%. However, in terms of the resulting dataset, the queries count would be likely lower, because logistic regression predicts for the evaluation set 392 query non-removals, whereas XGBoost predicts 353 removals. The detailed classification results can be seen in the following Table 9.

	P	N		P	N
P	141	52	P	150	44
N	99	346	N	135	309

(a) Logistic Regression
FOR = 0.1307

(b) XGBoost
FOR = 0.1246

Table 9: Confusion matrices for the Logistic Regression and XGBoost classifier for query removal. Positive P stands for a query being removed. Negative N stands for a query not being removed. FOR is the false omission rate, where omission means non-removal.

Nearest-Neighbor Method: As for the classifiers, we use the query text embeddings generated with Alibaba text-embedding-v4 and 800 samples as representatives, with 638 samples being in the evaluation set. For a given sample, the method searches the five closest neighbors in terms of cosine similarity between the query embeddings and classifies the sample by the majority class of the neighbors. The results, however, show as presented in Table 10, that compared with both classifiers, the nearest neighbors method leads to a worse false omission rate of 19.77%.

	Predicted P	Predicted N
Actual P	116	85
Actual N	92	345

Table 10: Confusion matrix for five nearest neighbor question removal classification strategy (total samples = 638, FOR = 0.1977).

Final Query Cleaning Method: Although the classifiers logistic regression and XGBoost have a slightly lower false omission rate compared to the combination method of LLM classifier and query LLM rating, we decide not to use the former methods. The main cause for this is the relatively small query removal dataset, which could possibly mean higher variance for the classifiers for unknown query types. Moreover, the difference between the two approaches is quite small, and the LLM approach keeps in the evaluation set 69.05% of the queries and the classifiers 62.38% for logistic regression and XGBoost 55.33%.

A.6 Query Dataset Statistics

In this subsection, we present additional statistics for our query dataset. Table 11 shows the query distribution by evidence structure type and domain type. Across all query evidence structures, the Financial Reports domain contains the most queries with a total of 79,791 queries. The evidence structure type distribution is similar between the different domain types with single-page queries being the most common ones followed by Cross-Doc queries. Since there are for less represented domains, such as economics, only few queries, we recommend for further evaluations on our benchmark to treat results grouped by evidence structure type and domain type with caution.

Figure 7 illustrates the distribution of query evidence types and their co-occurrence. The most occurring evidence is pure text with 90,248 queries, followed by table row evidence with 40,955 queries, and the third most occurring combination is image+text with 28,932 queries. In general, this Figure 7 shows that our benchmark queries’ evidence types are well balanced across text, images and tabular data.

	Page-Single	Page-Multi	Doc-Adjacent	Doc-Near	Doc-Far	Cross-Doc	Total
Financial Reports	31,735	29,792	2,773	586	1,234	13,671	79,791
Medical Reports	7,835	17,006	1,902	523	416	6,324	34,006
Sports	5,891	10,017	1,067	436	69	7,026	24,506
Astrophysics	4,359	8,420	941	205	167	7,829	21,921
Science	6,006	7,080	632	212	41	2,566	16,537
Geography	6,064	6,957	442	206	85	2,267	16,021
Condensed Matter Physics	3,669	6,462	632	140	79	622	11,604
Artificial Intelligence	2,975	5,950	554	162	114	324	10,079
Bioinformatics	1,892	4,184	487	135	51	1,166	7,915
History	2,973	3,399	349	87	24	704	7,536
Entertainment	2,083	3,029	340	134	18	487	6,091
Public Health Reports	1,531	3,037	314	100	75	978	6,035
Chemistry	1,438	2,979	416	96	20	704	5,653
Quantitative Biology	1,298	2,768	215	59	51	84	4,475
Technology	1,257	1,420	159	32	10	1,258	4,136
Environmental Reports	725	1,690	132	71	36	214	2,868
Electrical Engineering	846	1,713	79	39	24	74	2,775
Politics	1,161	1,051	123	43	3	165	2,546
Arts	778	1,164	152	21	2	96	2,213
General Computer Science	659	1,248	101	24	18	54	2,104
Surgical Case Reports	707	724	109	75	15	357	1,987
Statistics	221	429	34	5	8	14	711
Economics	93	105	19	1	3	29	250
Total	86,196	120,624	11,972	3,392	2,563	47,013	271,760

Table 11: Distribution of dataset questions by question type and domain type.

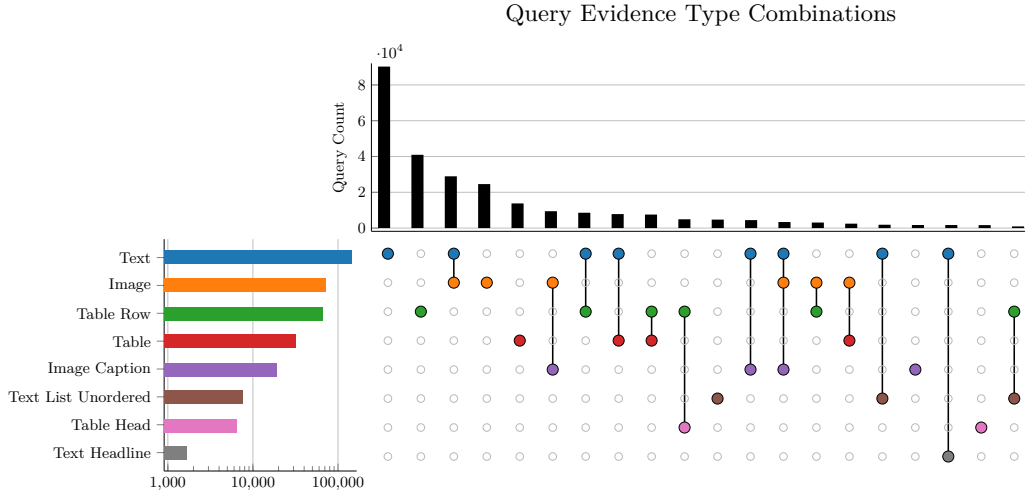


Figure 7: Co-occurrence of query evidence types and evidence type distribution. The most common occurring combinations are text, table row, and image with text.

Source	Evidence structure	Correct/Total	F	E	U	N
Wikipedia	Page-Single	29/30	1	0	0	0
	Page-Multi	30/30	0	0	0	0
	Doc-Adjacent	26/30	3	1	0	0
	Doc-Near	29/30	1	0	0	0
	Doc-Far	28/30	0	0	2	0
	Cross-Doc	45/50	4	0	1	0
arXiv	Page-Single	29/30	1	0	0	0
	Page-Multi	30/30	0	0	0	0
	Doc-Adjacent	28/30	0	1	1	0
	Doc-Near	27/30	0	1	2	0
	Doc-Far	29/30	1	0	0	0
	Cross-Doc	48/50	0	2	0	0
PubMed	Page-Single	29/30	1	0	0	0
	Page-Multi	30/30	0	0	0	0
	Doc-Adjacent	28/30	0	0	1	1
	Doc-Near	28/30	0	0	2	0
	Doc-Far	30/30	0	0	0	0
	Cross-Doc	40/50	0	10	0	0
SEC	Page-Single	30/30	0	0	0	0
	Page-Multi	29/30	1	0	0	0
	Doc-Adjacent	26/30	0	1	3	0
	Doc-Near	29/30	0	0	1	0
	Doc-Far	25/30	0	1	4	0
	Cross-Doc	44/50	0	3	0	3
Total		746/800	13	20	17	4

Table 12: **Stratified annotation audit.** F: incorrectly formulated or insufficiently identifying query; E: incorrect evidence; U: unnecessary evidence page; N: non-presence query without directly retrievable positive evidence.

A.7 Annotation Validation and Adjudication

Table 12 reports the human audit over all 24 source-by-evidence-structure combinations. There are 50 Cross-Doc entries and 30 entries from every other structure per source. In total, 746/800 entries are fully correct (93.25%), while 54/800 contain an observed issue (6.75%). The issue counts are 13 formulation issues, 20 incorrect evidence sets, 17 sets containing an unnecessary evidence page, and four non-presence queries without directly retrievable positive evidence.

These rates describe the audited entries before corrections. The reported 95% Wilson interval (91.3–94.8%) is a nominal interval for the audited-entry proportion rather than a population-wide or post-correction accuracy estimate. Cross-Doc correctness is 177/200 (88.5%), with the lowest source-specific rate observed for PubMed Cross-Doc (40/50, 80.0%). The Cross-Doc audit contains 200 entries representing 199 unique queries; all counts in Table 12 refer to audit entries. This retained-annotation audit is distinct from the 1,438-example query-cleaning evaluation in Appendix A.5.

Incorrect or unnecessary annotated pages can make RecallAll and nDCGAll mark a retrieval as incomplete even when answer-relevant evidence was retrieved. The lower Cross-Doc quality therefore remains relevant to interpreting its retrieval scores. A separate census covers all 47,013 Cross-Doc queries and 100,504 evidence elements. It finds 39,600 fully consistent queries (84.23%), 5,860 queries with an element spanning rendered pages (12.46%), 839 with an out-of-range stored bounding box (1.78%), and 714 with stored/configured pages differing from the bounding-box-derived page (1.52%). No mappings, element IDs, page images, configurations, or evidence documents are missing in these checks.

After applying the accepted corrections, we re-evaluated the twelve frozen retrieval runs. The complete method order and best method were unchanged for all twelve metric-cutoff combinations under Micro, structure-balanced Macro, source-averaged Macro, and domain-

averaged Macro. This held within the nine-method full-corpus comparison and the separately evaluated three-method.

B Evaluation Result Details

Below, we present more detailed evaluation results on our benchmark. We show especially results grouped by different query categories, such as by query evidence structure type and data source type, or by evidence type. Additionally, we conduct further experiments to measure retrieval performance on our benchmark. We have experimented with the 40GB A100 and 94GB H100-equipped computing nodes. Each retrieval experiment per retrieval model has consumed approximately 100 hours of compute time on A100s and H100s.

B.1 Overall Results

The overall benchmark results refer to the evaluation of our whole benchmark across all query types, data source types, and evidence types. Table 13 presents the complete results for the visual, OCR-based, native-text, and hybrid retrieval methods. The abbreviations Ctx, Rerank, and Hierarchical refer to document-context scoring, reranking, and hierarchical retrieval, respectively. B+Ctx combines the ranked lists from base retrieval and document-context scoring. Q-Split 1–3 denote the three query-splitting variants, with Q-Split 3 corresponding to Q-Split in Table 3. ColNomic (Ensemble) combines the reranked, context, and base retrieval lists. Within each aggregation protocol and metric column, the highest, second-highest, and third-highest scores are shown in bold blue, bold green, and bold black, respectively.

ColNomic (Ensemble) is the best-performing configuration under both aggregation protocols, achieving nDCGAll@10 of 69.28 and RecallAll@10 of 80.64 under Macro (75.64 and 86.23 under Micro, respectively). Under Macro, ColNomic (Ctx) obtains higher nDCGAll@10 and RecallAll@10 than ColNomic (Rerank), while under Micro, ColNomic (Rerank) obtains higher scores for both metrics.

Among the base visual retrievers, ColQwen3 [19] achieves the highest nDCGAll@10 under both Macro and Micro, followed by ColNomic [18]. The Jina V4 [16] multi-vector model outperforms its single-vector counterpart by 11.6 and 10.5 percentage points in Macro nDCGAll@10 and RecallAll@10, respectively (12.9 and 10.0 under Micro). For ColQwen2.5, Q-Split achieves higher RecallAll for all three values of k . Its RecallAll@10 scores are 69.19 under Macro and 78.63 under Micro, compared with 67.23 and 77.04 for the base model, respectively.

OCR-BM25 achieves the highest Macro nDCGAll@10 and RecallAll@10 among the evaluated text baselines, with scores of 50.24 and 61.69, respectively. The corresponding scores are 19.35 and 24.70 for OCR-dense (BGE-M3) [70], and 46.96 and 57.44 for DOM-BM25.

RRF fusion of OCR-BM25 with ColNomic (Ctx) increases RecallAll@10 from 78.19 to 78.67 under Macro and from 82.26 to 84.24 under Micro. The corresponding nDCGAll@10 scores decrease from 66.03 to 65.13 and from 70.07 to 69.75, respectively. For ColNomic (Rerank), fusion with OCR-BM25 decreases Macro RecallAll@10 from 74.62 to 74.39 and increases Micro RecallAll@10 from 83.11 to 83.30, with lower nDCGAll@10 under both aggregation protocols.

B.2 Source- and Domain-Averaged Macro Results

We report source-averaged and domain-averaged Macro for nine full-corpus configurations: the ColNomic base, Ctx, Rerank, and Ensemble settings; OCR-BM25, OCR-dense (BGE-M3), and DOM-BM25; and the two ColNomic-OCR-BM25 RRF hybrids. These aggregations assign equal weight to data sources or domains, while retaining equal weight across evidence structures within each group. Structure-balanced Macro remains the primary aggregation protocol.

For a source or domain g , let $Q_{g,e}$ denote the set of queries with evidence structure e , and let $m(g)$ denote the query-level metric. We compute the structure-balanced mean within

Model	Aggregate	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
		@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColSmol 500	Macro	49.43	53.36	56.26	52.89	61.04	68.62	35.30	42.11	47.33	39.01	49.10	58.15
	Micro	54.98	57.76	59.95	61.35	67.70	73.97	50.63	53.87	56.58	57.03	63.45	69.81
ColModernVBERT	Macro	47.33	50.91	53.72	50.48	57.93	65.30	33.28	39.31	44.29	36.73	45.70	54.35
	Micro	51.69	54.39	56.55	57.73	63.92	70.11	47.45	50.57	53.20	53.51	59.73	65.96
GME Qwen2-VL	Macro	55.27	59.51	62.72	59.35	68.10	76.42	41.08	48.82	55.10	45.33	56.70	67.34
	Micro	60.18	63.16	65.44	67.20	73.94	80.35	55.34	59.01	62.11	62.37	69.38	76.22
Jina V4 SV	Macro	48.31	52.61	55.80	52.40	61.33	69.69	34.70	42.09	48.03	38.80	49.86	60.24
	Micro	52.55	55.75	58.24	59.58	66.87	74.01	48.17	51.93	55.10	55.16	62.60	70.00
Jina V4 MV	Macro	60.92	64.99	67.79	64.43	72.78	79.99	45.69	53.56	59.59	49.61	60.92	70.77
	Micro	67.24	69.83	71.74	73.36	79.17	84.47	61.80	65.15	67.98	68.01	74.16	80.03
ColQwen3	Macro	65.54	69.09	71.36	68.15	75.35	81.13	50.39	57.86	63.20	53.74	64.11	72.42
	Micro	72.54	74.53	75.89	77.44	81.80	85.49	67.13	69.92	72.15	72.21	77.00	81.24
ColPali v1.3	Macro	42.03	45.47	48.26	45.04	52.18	59.52	28.93	34.55	39.41	32.14	40.58	49.13
	Micro	46.10	48.74	50.92	51.76	57.80	64.08	42.21	45.20	47.85	47.87	53.90	60.26
ColPali v1.3 (Ctx)	Macro	47.99	53.19	57.46	52.21	63.01	74.20	33.51	42.55	50.87	37.63	51.13	65.69
	Micro	50.46	54.12	57.22	57.45	65.84	74.76	45.48	49.71	53.48	52.47	60.92	69.90
ColPali v1.3 (B+Ctx)	Macro	47.28	52.03	56.15	51.28	61.14	71.98	32.67	40.65	48.69	36.66	48.59	62.55
	Micro	50.29	53.66	56.69	57.22	64.96	73.68	45.51	49.39	53.14	52.45	60.22	69.12
ColPali v1.3 (Rerank)	Macro	57.25	59.73	60.92	58.96	63.94	66.94	43.30	48.42	51.12	45.88	52.93	57.16
	Micro	62.97	64.26	64.95	66.38	69.19	71.03	58.31	60.18	61.33	61.93	65.05	67.17
ColQwen2.5	Macro	58.68	62.56	65.15	62.08	70.02	76.72	44.15	51.48	56.72	47.93	58.51	67.23
	Micro	64.94	67.41	69.21	70.84	76.37	81.40	59.84	62.97	65.51	65.82	71.63	77.04
ColQwen2.5 (Ctx)	Macro	61.22	65.91	69.30	65.20	74.82	83.56	46.17	55.36	62.80	50.39	63.52	75.78
	Micro	66.02	68.97	71.15	72.52	79.16	85.28	60.13	63.84	66.85	66.77	73.67	80.10
ColQwen2.5 (B+Ctx)	Macro	61.56	66.11	69.40	65.81	75.15	83.63	46.51	55.48	62.74	50.95	63.80	75.68
	Micro	66.66	69.56	71.72	73.58	80.10	86.11	60.97	64.67	67.81	67.96	74.83	81.41
ColQwen2.5 (Q-Split)	Macro	58.13	62.29	65.15	62.30	70.81	78.20	44.04	51.69	57.48	48.21	59.39	69.19
	Micro	63.99	66.72	68.74	70.76	76.87	82.52	59.25	62.73	65.64	65.95	72.44	78.63
ColQwen2.5 (Rerank)	Macro	65.82	69.22	71.15	68.55	75.45	80.31	51.33	58.45	62.88	54.69	64.60	71.62
	Micro	72.61	74.49	75.65	77.26	81.38	84.50	67.33	70.02	71.94	72.14	76.73	80.36
ColNomic	Macro	63.85	67.61	70.05	66.84	74.53	80.78	48.73	56.38	61.82	52.36	63.06	71.76
	Micro	70.46	72.75	74.35	75.95	81.05	85.46	65.06	68.17	70.68	70.71	76.24	81.24
ColNomic (Ctx)	Macro	64.85	69.37	72.57	68.26	77.51	85.72	49.33	58.64	66.03	53.28	66.31	78.19
	Micro	69.75	72.51	74.48	75.70	81.88	87.38	63.50	67.15	70.07	69.69	76.28	82.26
ColNomic (B+Ctx)	Macro	65.69	70.25	73.34	69.63	78.96	86.89	50.31	59.77	67.06	54.63	67.91	79.60
	Micro	71.02	73.77	75.71	77.75	83.88	89.24	65.11	68.87	71.96	71.95	78.66	84.81
ColNomic (Rerank)	Macro	66.82	70.50	72.76	69.77	77.24	83.00	52.25	59.84	65.00	55.77	66.37	74.62
	Micro	73.77	75.87	77.29	78.72	83.33	87.15	68.44	71.42	73.76	73.52	78.63	83.11
ColNomic (Ensemble)	Macro	68.42	72.68	75.45	72.03	80.67	87.73	53.58	62.60	69.28	57.43	70.04	80.64
	Micro	75.00	77.39	79.03	80.59	85.85	90.30	69.43	72.89	75.64	75.02	80.95	86.23
OCR-BM25	Macro	51.30	55.40	58.47	54.99	63.50	71.53	37.56	44.59	50.24	41.45	51.91	61.69
	Micro	56.06	59.16	61.53	62.70	69.76	76.48	51.55	55.20	58.26	58.18	65.36	72.34
OCR-dense (BGE-M3)	Macro	22.90	24.97	26.59	24.58	28.90	33.16	14.14	16.92	19.35	15.92	20.26	24.70
	Micro	23.37	24.95	26.24	26.66	30.28	34.00	20.81	22.45	23.86	24.10	27.55	31.12
DOM-BM25	Macro	47.81	51.49	54.34	51.12	58.75	66.19	35.24	41.67	46.96	38.82	48.37	57.44
	Micro	51.11	53.86	56.00	57.07	63.34	69.41	47.04	50.31	53.07	53.01	59.44	65.72
Ctx + OCR-BM25 (RRF)	Macro	62.88	67.70	71.23	67.11	77.01	86.10	47.87	57.14	65.13	52.30	65.60	78.67
	Micro	67.74	70.97	73.26	75.00	82.26	88.62	62.27	66.34	69.75	69.53	77.13	84.24
Rerank + OCR-BM25 (RRF)	Macro	61.38	65.84	69.10	65.41	74.60	82.96	46.78	54.95	61.86	51.07	62.90	74.39
	Micro	67.30	70.39	72.61	74.29	81.25	87.37	62.16	66.07	69.34	69.16	76.47	83.30

Table 13: Overall retrieval benchmark results with the nDCG@k [67], Recall@k, nDCGAll@k [67], and RecallAll@k as evaluation metrics. The best models are highlighted with **blue**, the second best with **green**, and the third best with **bold** text. The numbers below the evaluation metrics represent the values of parameter k.

each group as

$$M_g(m) = \frac{1}{6} \sum_{e=1}^6 \frac{1}{|Q_{g,e}|} \sum_{q \in Q_{g,e}} m(q), \quad (9)$$

where all four sources and all 23 domains contain queries from each of the six evidence structures. The two aggregate scores are then defined as

$$\text{SourceMacro}(m) = \frac{100}{4} \sum_{s=1}^4 M_s(m), \quad (10)$$

$$\text{DomainMacro}(m) = \frac{100}{23} \sum_{d=1}^{23} M_d(m). \quad (11)$$

Model / Setting	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
	@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColNomic	64.68	68.47	70.89	67.65	75.36	81.58	49.33	57.13	62.63	52.99	63.85	72.56
ColNomic (Ctx)	65.11	69.58	72.74	68.45	77.58	85.71	49.55	58.77	66.07	53.51	66.34	78.01
ColNomic (Rerank)	67.61	71.33	73.59	70.44	77.97	83.72	52.82	60.59	65.82	56.31	67.09	75.39
ColNomic (Ensemble)	69.02	73.22	75.97	72.43	80.95	87.93	54.10	63.08	69.78	57.88	70.37	80.85
OCR-BM25	52.04	56.18	59.29	55.69	64.28	72.39	38.00	45.14	50.95	41.92	52.49	62.45
OCR-dense (BGE-M3)	23.39	25.54	27.22	25.14	29.61	34.06	14.50	17.33	19.84	16.36	20.79	25.40
DOM-BM25	48.80	52.52	55.44	52.21	59.94	67.54	35.91	42.39	47.86	39.61	49.23	58.54
Ctx + OCR-BM25 (RRF)	63.28	68.00	71.53	67.42	77.13	86.19	48.21	57.36	65.40	52.63	65.70	78.75
Rerank + OCR-BM25 (RRF)	62.07	66.51	69.79	66.06	75.20	83.61	47.22	55.46	62.58	51.53	63.40	75.13

Table 14: **Source-averaged Macro retrieval results (%)**. Each score is the equal-weight mean of the four source-specific six-structure Macro scores. Bold indicates the highest score in each column.

Model / Setting	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
	@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColNomic	70.06	74.15	76.58	73.02	81.32	87.51	54.93	63.96	69.97	58.58	70.98	80.31
ColNomic (Ctx)	67.87	72.58	75.65	70.71	80.28	88.15	52.05	62.50	70.11	55.70	69.86	81.76
ColNomic (Rerank)	72.22	76.21	78.55	74.95	83.00	88.91	57.74	66.77	72.47	61.15	73.40	82.34
ColNomic (Ensemble)	72.83	77.18	79.82	75.83	84.58	91.22	58.28	68.50	75.26	61.85	75.56	85.93
OCR-BM25	58.06	62.23	65.51	61.86	70.47	78.91	43.78	51.68	58.38	47.87	59.16	70.42
OCR-dense (BGE-M3)	24.09	26.25	28.04	25.76	30.25	34.94	15.07	18.09	20.82	16.89	21.59	26.58
DOM-BM25	56.75	60.81	63.87	60.57	69.00	76.93	42.61	50.38	56.73	46.75	58.10	68.63
Ctx + OCR-BM25 (RRF)	67.80	72.22	75.59	71.62	80.67	89.29	53.60	62.70	70.95	57.88	70.49	83.51
Rerank + OCR-BM25 (RRF)	67.43	71.84	75.08	71.15	80.16	88.43	53.20	62.06	69.46	57.44	69.81	81.78

Table 15: **Domain-averaged Macro retrieval results (%)**. Each score is the equal-weight mean of the 23 domain-specific six-structure Macro scores. Bold indicates the highest score in each column.

Tables 14 and 15 report nDCG, Recall, nDCGAll, and RecallAll for $k \in \{3, 5, 10\}$. ColNomic (Ensemble) achieves the highest scores among the nine configurations for all metrics and values of k under both aggregations. Its nDCGAll@10 and RecallAll@10 are 69.78 and 80.85 under source-averaged Macro, and 75.26 and 85.93 under domain-averaged Macro, respectively. The relative performance of Ctx and Rerank differs between the two aggregations: ColNomic (Ctx) obtains higher RecallAll@10 under source-averaged Macro, whereas ColNomic (Rerank) obtains higher RecallAll@10 under domain-averaged Macro.

B.3 Results by Base Model

In Figure 8, we show further evaluation results in micro-average grouped by the three base models ColPali v1.3 [2], ColQwen2.5 [2], and ColNomic [18]. For ColPali v1.3 [2], the base model has the lowest performance. Our assumed faster hierarchical approach shows a slightly higher performance across all k parameter values. However, the difference is more visible for our other approaches. For $k = 3$, ColPali v1.3 (Rerank) is the best performing approach by RecallAll@k, even surpassing the RecallAll@10 of the base model. In general, ColPali v1.3 (Ctx) and ColPali v1.3 (Normal&Ctx) display similar RecallAll@k scores. The relative performance, the performance gains for increased retrieved pages are stronger for our document context approach than for reranking. While RecallAll@5 is still the highest for ColPali v1.3 (Rerank), for RecallAll@10, ColPali v1.3 (Ctx) surpasses it.

The results are similar for the ColQwen2.5 [2] variants. One difference to the ColPali v1.3 [2] variants is that ColQwen2.5 (Normal&Ctx) improves upon ColQwen2.5 (Ctx), while the ColPali v1.3 ensemble performs slightly worse than the pure document context approach. One possible explanation for this might be that the ColQwen2.5 base model is relatively closer to the ColQwen2.5 (Ctx) model than ColPali v1.3 is to ColPali v1.3 (Ctx). Furthermore, we test for ColQwen2.5 query splitting approaches. Both ColQwen2.5 (Q-Split 1) and ColQwen2.5 (Q-Split 2) do overall perform worse in terms of RecallAll@k than the base ColQwen2.5 model [2], whereas ColQwen2.5 (Q-Split 3) has a higher RecallAll@k. As for ColPali v1.3 [2], the RecallAll@k increases the strongest for the approaches that include document context embeddings.

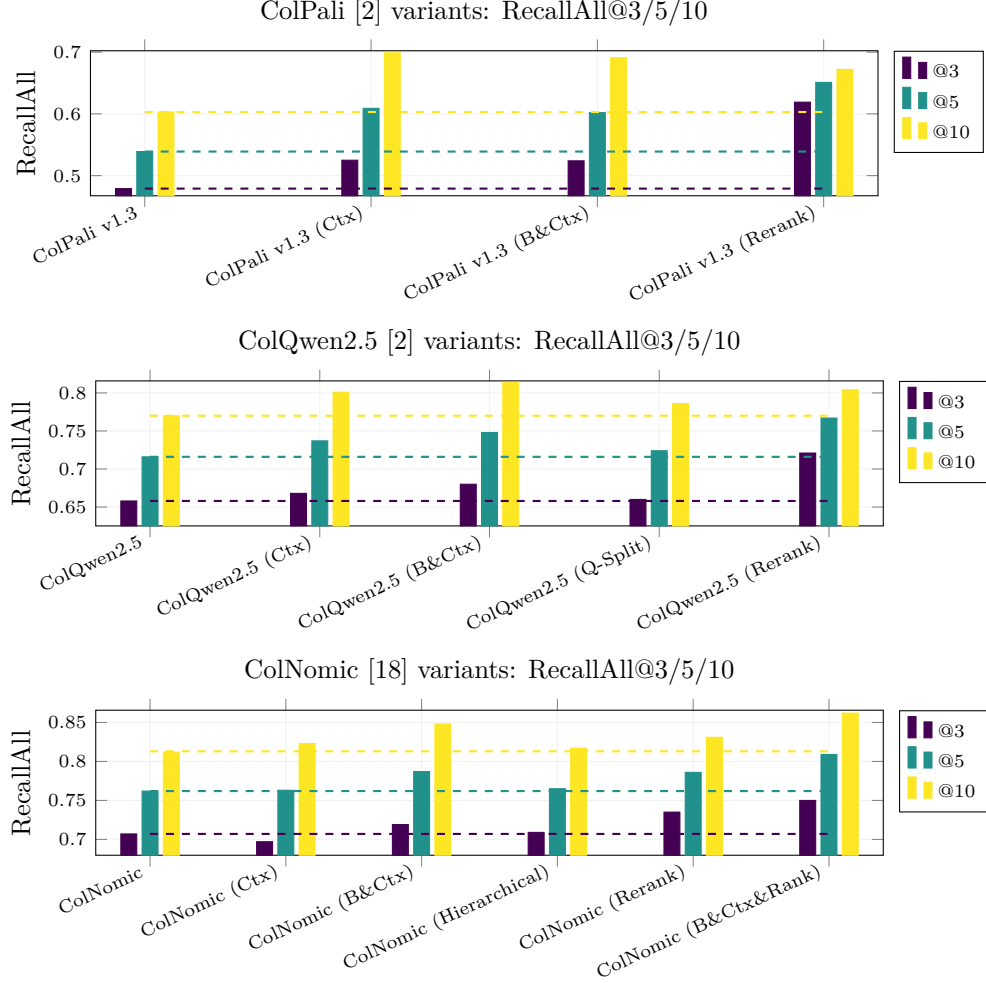


Figure 8: Overall RecallAll@k results grouped by ColPali [2], ColQwen2.5 [2], and ColNomic [18] variants for $k \in \{3, 5, 10\}$. B denotes base model and Ctx stands for document context embeddings

In the case of ColNomic [18] variants, the document context approach ColNomic (Ctx) has a slightly lower performance for $k = 3$. However, as for the other two base models, ColPali v1.3 [2] and ColQwen2.5 [2], rising k values improve the performance stronger. As for ColQwen2.5 [2], ColNomic (Ctx) profits from the usage of an ensemble with the retrieval results from the base model. ColNomic (Hierarchical) also exhibits the same performance as the base model, although it requires fewer computing resources during page retrieval. In contrast to the other two base models, reranking displays even for $k = 5$ no better RecallAll@k values than ColNomic (Normal&Ctx). Additionally, we include in this analysis the ensemble combination of the ColNomic base model with ColNomic (Ctx) and ColNomic (Rerank), which shows the best RecallAll@k for all parameter k values.

B.4 Results by Query Evidence Structure Type and Data Source Type

Furthermore, we evaluate the benchmark results in micro-average grouped by query evidence structure and data source type to determine whether the query evidence structure type performance differences are consistent for different data sources. Table 16 displays the RecallAll@5 metric for all pairs of query evidence structure types and data source types.

Across all data sources and models, we observe that multi-page queries are more challenging than single-page queries. Retrieval on Cross-Doc queries exhibits an even lower RecallAll@5

indicating that Cross-Doc queries are fundamentally more challenging than queries about cohesive document pages. Additionally, Table 16 reveals a slight tendency of Doc-Near queries being more challenging than Doc-Adjacent queries, and Doc-Far being more challenging than Doc-Near queries.

For single-page queries, the best performing model is consistently ColNomic (Normal&Ctx&Rank). In contrast to this, the second-best model depends on the data source. For Wikipedia, arXiv, and PubMed, ColNomic (Rerank) is the second-best model, whereas for SEC Proxy Reports, ColNomic (Ctx) is the second-best. In the case of ColQwen2.5 [2], we observe the same behavior for the reranking and document context embeddings approach, with the reranking model performing worse on the SEC Proxy Reports data source. A potential cause for this might be the missing context on pages closer to the end of a document in SEC Proxy Reports caused by missing company names on pages.

We observe a similar model behavior for multi-page queries. Document context embeddings lead to a higher RecallAll@5 for Wikipedia and SEC Proxy Reports, while for arXiv and PubMed reranking is more favorable. A possible explanation for this approach behavior could be that the scientific article data source queries are more clearly specified and reference articles more directly, which might lower the positive impact of document context embeddings.

Additionally, Table 16 shows that the RecallAll@5 differs strongly between different data sources for Cross-Doc queries. Whereas the best-performing model achieves a RecallAll@5 of 50.9% for Wikipedia, the best metric values for SEC Proxy Reports are 26.6%, for arXiv 18.0%, and for PubMed 28.0%. One possible explanation for the low RecallAll@5 scores might be that the queries are not well specified. This hypothesis could be supported by the significant drop in the RecallAll@5 from multi-page to Cross-Doc queries. Another explanation could be that cross-document retrieval is fundamentally harder, especially as document page counts grow, such as in SEC Proxy Reports and scientific articles. A third option might be that different query parts average themselves out in the late interaction max-similarity scoring. This is partly supported by the RecallAll@5 improvement when using the Q-Split 3 approach on ColQwen2.5 [2], where the top k retrieved pages are an ensemble over the sub-queries. However, the still low score by absolute terms contradicts that this is the sole explanation. We conduct further analysis in the Appendix Subsections B.8 and B.9. Another observation for Cross-Doc queries is that document context embeddings are affecting the retrieval performance negatively in contrast to multi-page queries. The use of reranking improves over the base models ColQwen2.5 [2] and ColNomic [18], however, does exceed the ColQwen2.5 (Q-Split) RecallAll@5 only for the Wikipedia data source.

B.5 Results by Evidence Type

Furthermore, we compare the benchmark results grouped by evidence type as shown in Table 17. We choose as evidence groups text, table, and image which refer to the cases that only this evidence is the ground truth. We exclude combinations of evidence types, such as tables with text, to remove the impact of inter-dependencies. Overall, nDCGAll@k and RecallAll@k are similar for the text and image evidence types, with a slight tendency for image evidence being harder to find. In contrast to those, the retrieval performance is consistently lower for table-only evidence. Across all evidence type and model pairs, ColNomic (Normal&Ctx&Rank) exhibits the highest nDCGAll@k and RecallAll@k values. In case of text-only evidence, the second-best model depends on the metric and the number of retrieved pages. For lower k values, ColQwen3 [19] ranks as the second-best model, while for higher k values, ColNomic (Rerank) is the second-best model. For both table-only and image-only evidence, ColNomic (Rerank) is the second-best model except for RecallAll@k, where ColNomic (Ctx) scores higher.

B.6 Results by Domain Type

In Figure 9, we show further benchmark evaluation results in micro-average grouped by domain type. We select all domains with at least 100 Cross-Doc queries and take the maximum RecallAll@5 of the following six models: ColQwen3, ColQwen2.5, ColQwen2.5 (Ctx), ColQwen2.5 (Q-Split), ColQwen2.5 (Reranking), ColNomic, ColNomic (Ctx), and ColNomic (Reranking). The evaluation results show that there is a large difference between

Model		Page-Single	Page-Multi	Doc-Adjacent	Doc-Near	Doc-Far	Cross-Doc
Wikipedia	ColQwen3 [19]	90.5	90.3	57.1	49.7	49.8	45.2
	ColQwen2.5 [2]	84.9	85.6	48.9	44.3	46.7	41.0
	ColQwen2.5 (Ctx)	83.8	85.0	62.0	59.9	57.3	32.9
	ColQwen2.5 (Q-Split)	84.3	84.5	48.2	45.1	47.5	45.8
	ColQwen2.5 (Rerank)	89.9	90.4	56.0	49.8	58.4	50.9
	ColNomic [18]	89.4	89.7	54.3	50.0	53.3	48.5
	ColNomic (Ctx)	86.6	87.9	67.2	64.4	67.1	34.2
	ColNomic (Rerank)	91.1	91.3	58.1	51.4	59.6	50.8
ColNomic (Normal&Ctx&Rank)		92.5	93.0	63.5	59.5	62.4	49.8
SEC Proxy Reports	ColQwen3 [19]	83.4	85.4	52.5	48.5	43.7	15.3
	ColQwen2.5 [2]	71.0	73.8	36.3	37.4	35.3	10.0
	ColQwen2.5 (Ctx)	81.7	83.7	57.3	54.9	51.5	13.9
	ColQwen2.5 (Q-Split)	70.8	73.6	37.5	40.6	36.0	17.5
	ColQwen2.5 (Rerank)	78.3	80.6	47.3	45.9	42.9	15.3
	ColNomic [18]	79.5	82.1	43.7	45.4	37.5	13.8
	ColNomic (Ctx)	87.1	88.8	62.1	57.5	57.3	17.9
	ColNomic (Rerank)	83.4	85.7	50.4	50.3	45.9	17.0
ColNomic (Normal&Ctx&Rank)		89.0	90.2	59.8	59.7	53.7	26.6
arXiv	ColQwen3 [19]	93.2	94.6	74.3	77.1	71.1	11.5
	ColQwen2.5 [2]	90.5	93.5	68.5	68.8	65.5	7.4
	ColQwen2.5 (Ctx)	90.1	93.2	74.4	71.9	63.3	3.5
	ColQwen2.5 (Q-Split)	89.4	92.3	65.2	66.2	63.6	18.0
	ColQwen2.5 (Rerank)	94.1	96.1	74.8	76.0	72.2	12.4
	ColNomic [18]	93.0	95.5	73.4	72.1	69.4	11.2
	ColNomic (Ctx)	91.5	94.4	74.7	70.5	65.7	4.2
	ColNomic (Rerank)	95.2	96.9	77.6	75.4	73.1	15.1
ColNomic (Normal&Ctx&Rank)		95.8	97.1	79.5	77.3	75.5	14.1
PubMed	ColQwen3 [19]	93.4	94.8	75.4	69.2	67.7	18.7
	ColQwen2.5 [2]	92.7	94.7	73.9	71.8	66.9	14.3
	ColQwen2.5 (Ctx)	92.4	94.6	77.0	76.0	63.3	5.1
	ColQwen2.5 (Q-Split)	91.4	93.7	71.8	72.4	62.5	28.0
	ColQwen2.5 (Rerank)	95.3	96.7	79.3	75.3	72.4	19.6
	ColNomic [18]	94.3	96.1	77.1	74.3	70.0	18.5
	ColNomic (Ctx)	92.9	95.2	78.0	74.7	66.7	5.6
	ColNomic (Rerank)	95.7	97.1	79.8	76.1	72.9	20.0
ColNomic (Normal&Ctx&Rank)		96.3	97.4	82.9	79.6	73.7	20.0

Table 16: Retrieval results by data source and query evidence structure type with the RecallAll@5 evaluation metric. The best models are highlighted with **blue**, the second best with **green**, and the third best has **bold** text.

Evidence Type	Model	nDCGAll@k [67]			RecallAll@k		
		3	5	10	3	5	10
Text	ColQwen3 [19]	68.5	71.1	73.3	73.5	78.1	82.3
	ColQwen2.5 [2]	59.3	62.4	65.0	65.6	71.6	77.4
	ColQwen2.5 (Ctx)	59.0	62.4	65.3	65.8	72.4	78.9
	ColQwen2.5 (Q-Split)	58.3	61.8	64.8	65.4	72.2	78.9
	ColQwen2.5 (Rerank)	67.1	69.7	71.8	72.1	76.8	80.8
	ColNomic [18]	64.8	67.7	70.3	70.8	76.3	81.6
	ColNomic (Ctx)	62.6	65.9	68.6	68.8	75.1	81.0
	ColNomic (Rerank)	68.2	71.0	73.5	73.5	78.6	83.5
	ColNomic (Normal&Ctx&Rank)	68.9	72.1	74.8	74.7	80.4	85.7
Table	ColQwen3 [19]	58.2	61.5	64.1	62.9	68.3	73.4
	ColQwen2.5 [2]	51.9	55.4	58.3	57.1	63.1	69.0
	ColQwen2.5 (Ctx)	53.7	58.0	61.7	59.5	67.0	74.2
	ColQwen2.5 (Q-Split)	53.1	56.7	59.8	58.6	65.0	71.5
	ColQwen2.5 (Rerank)	59.5	62.6	64.7	64.0	69.1	73.0
	ColNomic [18]	57.5	61.1	64.0	62.5	68.6	74.1
	ColNomic (Ctx)	57.0	61.5	65.3	62.6	70.0	77.0
	ColNomic (Rerank)	60.7	64.1	66.8	65.4	71.1	76.3
	ColNomic (Normal&Ctx&Rank)	62.6	67.0	70.4	68.1	75.3	81.7
Image	ColQwen3 [19]	64.3	67.5	69.8	71.3	77.3	82.0
	ColQwen2.5 [2]	54.7	58.0	60.6	62.1	68.6	74.3
	ColQwen2.5 (Ctx)	56.6	60.9	64.0	65.3	74.1	81.7
	ColQwen2.5 (Q-Split)	54.3	58.0	60.9	62.3	69.6	75.9
	ColQwen2.5 (Rerank)	63.7	66.4	68.2	69.7	74.6	77.8
	ColNomic [18]	61.7	65.0	67.6	68.7	74.8	80.0
	ColNomic (Ctx)	61.2	65.4	68.4	69.4	77.9	84.9
	ColNomic (Rerank)	66.1	69.2	71.3	72.7	78.3	82.4
	ColNomic (Normal&Ctx&Rank)	66.8	70.7	73.4	74.1	81.1	86.9

Table 17: Retrieval results by evidence type with the nDCGAll@k [67] and RecallAll@k as evaluation metrics. The best models are highlighted with **blue**, the second best with **green**, and the third best has **bold** text. The numbers below the evaluation metrics represent the values of parameter k. The three evidence groups text, table, and image refer to the cases that only this evidence is the ground truth. Combinations are not shown to remove the impact of inter-dependencies.

the domains. While retrieval on the *Science* domain, sourced from Wikipedia, achieves a RecallAll@5 of 75.6%, retrieval on the *Financial Reports* and *Astrophysics* domain less than 20%.

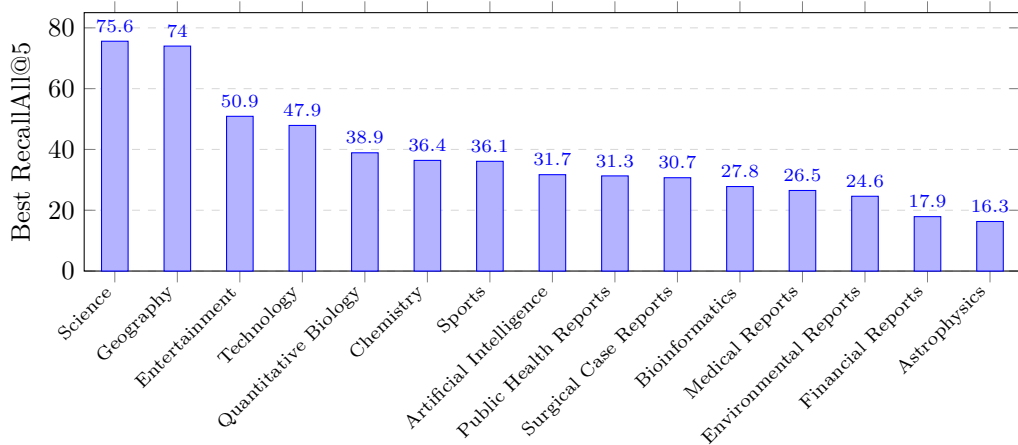


Figure 9: Best cross-document retrieval RecallAll@5 for domains with at least 100 Cross-Doc queries. For each domain, the value is the maximum over all evaluated models.

B.7 Document Context Embeddings Vector Count Impact

To find out the relation between document context embedding vectors count n and retrieval performance, we analyze the retrieval performance for $n \in \{32, 128, 256, 512, 1024\}$. We display the results of this analysis in Figure 10, where each line contains the results for all n values for a specific query evidence structure type. As the evaluation metric, we choose RecallAll@10. The results show that the optimal value for n depends on the query evidence structure type. For the single-page queries Page-Single and Page-Multi, ColQwen2.5 reaches the optimal performance between 128 and 256 document context embedding vectors. Whereas for Doc-Adjacent and Doc-Near queries, the optimum is between 256 and 512 vectors. The optimum for Doc-Distant queries is around 256 document context embedding vectors. In contrast to single-page and multi-page queries, we observe the optimum for Cross-Doc queries to be around $n = 32$. One explanation for the lower optimal n for Cross-Doc queries might be that document context embeddings favor the pages of the closest document, especially for queries that mention another document only with a few tokens. However, we observe that document context embeddings improve the RecallAll@10 across all query evidence structure types for a certain n in case of our benchmark.

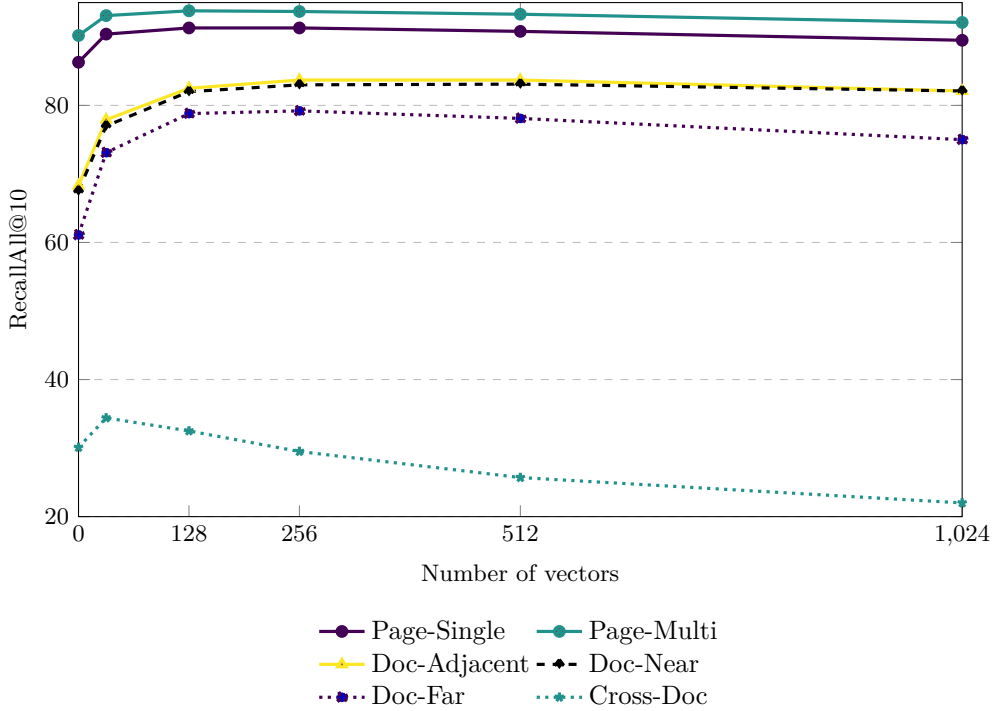


Figure 10: Analysis of vectors count impact in document context embeddings measured by RecallAll@10 for each query evidence structure type for ColQwen2.5 [2] as base model. We omit the cross-document line for reasons of clarity since the low RecallAll@10 values would distort the axis.

B.8 Page Count Reduction Analysis

To identify potential causes for our benchmark dataset being challenging, we evaluate the base models ColPali v1.3 [2], ColQwen2.5 [2], and ColNomic [18] on the reduced document dataset with micro-average results. We reduce the corpus size from 388,548 document page images to 38,551 by removing all pages that are not part of the evidence for at least one query. Table 18 displays the evaluation results for the combination of query evidence structure type and data source. In general, we observe an increase in RecallAll@5 across all models for every query evidence structure and data source type pair.

The RecallAll@5 is in the case of single-page queries for arXiv and PubMed close to perfect retrieval. Since those RecallAll@5 are already for the full dataset, this might indicate that the large dataset and in especially many pages per document cause the missing percentage points. For Wikipedia and SEC Proxy Reports, the RecallAll@5 gap to perfect retrieval still remains, however, still leaves room for improvement. Thus, we hypothesize that, especially for SEC Proxy Reports, there are factors other than high page count which prohibit perfect retrieval for single-page queries.

Evidence Structure	Model	Wikipedia	SEC Proxy Reports	arXiv	PubMed
Page-Single	ColPali	63.9	41.5	70.4	79.4
	ColQwen	84.9	71.0	90.5	92.7
	ColNomic	89.4	79.5	93.0	94.3
	ColPali Reduced	78.8	53.6	85.0	89.4
	ColQwen Reduced	92.7	81.1	96.9	97.2
	ColNomic Reduced	95.5	86.8	98.1	97.9
Page-Multi	ColPali	67.3	44.6	78.3	85.3
	ColQwen	85.6	73.8	93.5	94.7
	ColNomic	89.7	82.1	95.5	96.1
	ColPali Reduced	80.6	56.9	89.6	92.4
	ColQwen Reduced	92.6	83.4	98.0	97.9
	ColNomic Reduced	95.1	88.7	99.0	98.6
Doc-Adjacent	ColPali	30.3	13.8	43.1	57.2
	ColQwen	48.9	36.3	68.5	73.9
	ColNomic	54.3	43.7	73.4	77.1
	ColPali Reduced	51.3	26.9	66.9	77.3
	ColQwen Reduced	67.6	57.3	87.8	89.0
	ColNomic Reduced	73.2	61.7	91.4	91.3
Doc-Near	ColPali	29.2	15.0	44.3	53.5
	ColQwen	44.3	37.4	68.8	71.8
	ColNomic	50.0	45.4	72.1	74.3
	ColPali Reduced	51.9	31.7	67.7	75.5
	ColQwen Reduced	61.8	58.2	90.1	89.3
	ColNomic Reduced	69.4	63.3	90.9	90.5
Doc-Far	ColPali	30.6	11.3	43.2	52.2
	ColQwen	46.7	35.3	65.5	66.9
	ColNomic	53.3	37.5	69.4	70.0
	ColPali Reduced	57.6	25.0	69.0	74.2
	ColQwen Reduced	70.2	53.4	91.8	87.4
	ColNomic Reduced	78.4	57.4	94.6	89.1
Cross-Doc	ColPali	32.4	1.6	3.6	10.4
	ColQwen	41.0	10.0	7.4	14.3
	ColNomic	48.5	13.8	11.2	18.5
	ColPali Reduced	39.0	3.5	8.5	20.4
	ColQwen Reduced	46.7	16.1	15.2	25.5
	ColNomic Reduced	54.8	19.7	20.8	31.4

Table 18: RecallAll@5 by data source and query evidence structure type comparing the the retrieval on the full and reduced dataset. The model names with suffix 'Reduced' are the models which are evaluated on the reduced dataset, whereas the other models are evaluated on the whole document dataset. We evaluate the three base models ColPali [2], ColQwen2.5 [2], and ColNomic [18]. The best models are highlighted with **blue**, the second best with **green**, and the third best has **bold** text.

In the case of multi-page queries, we observe a strong increase in RecallAll@5 for all models. We observe across all four data sources room for improvement also with retrieval on the reduced dataset. This does apply especially for the Wikipedia and SEC Proxy Reports queries. For example, the best base model ColNomic [18] has still an absolute RecallAll@5 gap of 38.3% for Doc-Adjacent. Therefore, this strongly indicates that there are other factors than the large page count in the dataset that make the benchmark challenging.

For Wikipedia Cross-Doc queries, the results show a similar pattern to that for multi-page queries, but with a lower RecallAll@5 increase in the reduced dataset retrieval. This could indicate that Cross-Doc query retrieval is harder than multi-page queries, not only because of the larger number of similar pages to a query, but also because of other factors such as the lack of document diversity in the top k retrieval results. For the other three datasets, the RecallAll@5 values are in absolute terms very low for the full dataset with no model crossing 19%. After our page count reduction, this changes for the three data sources. The best base

model ColNomic [18] achieves on arXiv queries 20.8% and on PubMed queries 31.4% after the reductions. In absolute terms, those values are still low, however it shows that one cause of the very low RecallAll@5 values for the whole dataset can be attributed to the large page count. The missing RecallAll@5 percentage points might be caused by a fundamental problem in the late-interaction max-similarity calculation, still a large number of possible pages, or wrongly formulated Cross-Doc queries. One argument for the problem in the late-interaction max-similarity calculation is that the formula sums over the maximum-similarity of each token. When a query primarily consists of tokens that reference one specific document, the pages from the other documents might receive a much lower score. This is supported by the difference of the RecallAll@5 to the much higher Recall@5. For example, ColNomic [18] has for PubMed queries on the reduced dataset a RecallAll@5 of 31.4% and a Recall@5 of 57.7%. For SEC Proxy reports, this difference is relatively even larger with a RecallAll@5 of 19.7% and a Recall@5 of 45.6%. However, at the same time this might also indicate that some Cross-Doc queries are incorrectly created or categorized.

B.9 Cross-Document Retrieval Error Analysis

We analyze in this subsection what potential causes might be for the low performance in micro-average on Cross-Doc queries. Figure 11 shows a detailed cross-document retrieval performance breakdown for each data source for the models ColQwen2.5 [2], ColQwen2.5 (Doc Ctx), ColQwen2.5 (Q-Split), and ColQwen2.5 (Rerank). The retrieval performance breakdown is not just binary, but also considers the retrieval on a page-level (filled bars) and document-level (dotted bars). By page-level, we mean in this case whether the evidence pages are correctly retrieved, and by document-level, whether the documents of the pages are correctly retrieved. Additionally, we measure for both levels whether all, partial, or no evidence could be retrieved. The aim of this analysis is to find out whether the retrieval models even struggle to find the correct documents. As Figure 11 shows, all models have both for the page-level and document-level retrieval a high percentage of retrieving at least one correct evidence item. This speaks against the possibility that there might be multiple other documents which are evidence for a query. As already noted in previous sections, Cross-Doc query retrieval on the Wikipedia data source exhibits a relatively high performance in contrast to the three other data sources SEC Proxy Statements, arXiv, and PubMed. We observe that the breakdown distribution heavily depends on the retrieval model, while the breakdown distribution is consistent across the three other data sources for a given retrieval model. The base model ColQwen2.5 [2] and ColQwen2.5 (Rerank) display a similar performance for SEC Proxy Statements and arXiv articles with roughly the same percentage of complete and partial document-level retrieval. For page-level retrieval, this proportion shifts towards partial evidence retrieval. ColQwen2.5 (Doc Ctx) has even a higher percentage of partial retrieval both for page-level and document-level retrieval. A likely explanation for this behavior is that the document context embeddings bias the page retrieval to the pages of the most similar document to the query. ColQwen2.5 (Q-Split) has a high document-level retrieval success percentage, whereas at the page-level the retrieval percentage of all ground truth evidence is about half of the document-level retrieval percentage. The relatively high page-level partial evidence retrieval might be due to long documents. The lowest successful complete evidence retrieval for SEC Proxy Statements, which have the highest document page count supports this. However, this might also indicate that there are alternative document pages containing the evidence for a query. Overall, the query splitting achieves the best retrieval performance for the ColQwen2.5 [2] base model.

B.10 Difference between Query Generators

We compare queries generated by GPT-4.1-mini (59,983 queries) and Qwen3-VL-32B-Instruct [51] (211,777 queries) to examine the effect of query generation on retrieval performance. All methods retrieve from the complete 388,548-page corpus. Within each generator subset, Macro assigns equal weight to the six evidence structures, while Micro averages over all queries. Both protocols retain each generator’s original query distribution.

Figure 12 compares Macro RecallAll@10 across the visual retrieval configurations. Most configurations obtain higher scores on GPT-4.1-mini queries, with differences ranging from -0.1 to 7.1 percentage points. For ColPali, ColQwen2.5, and ColNomic, document-context

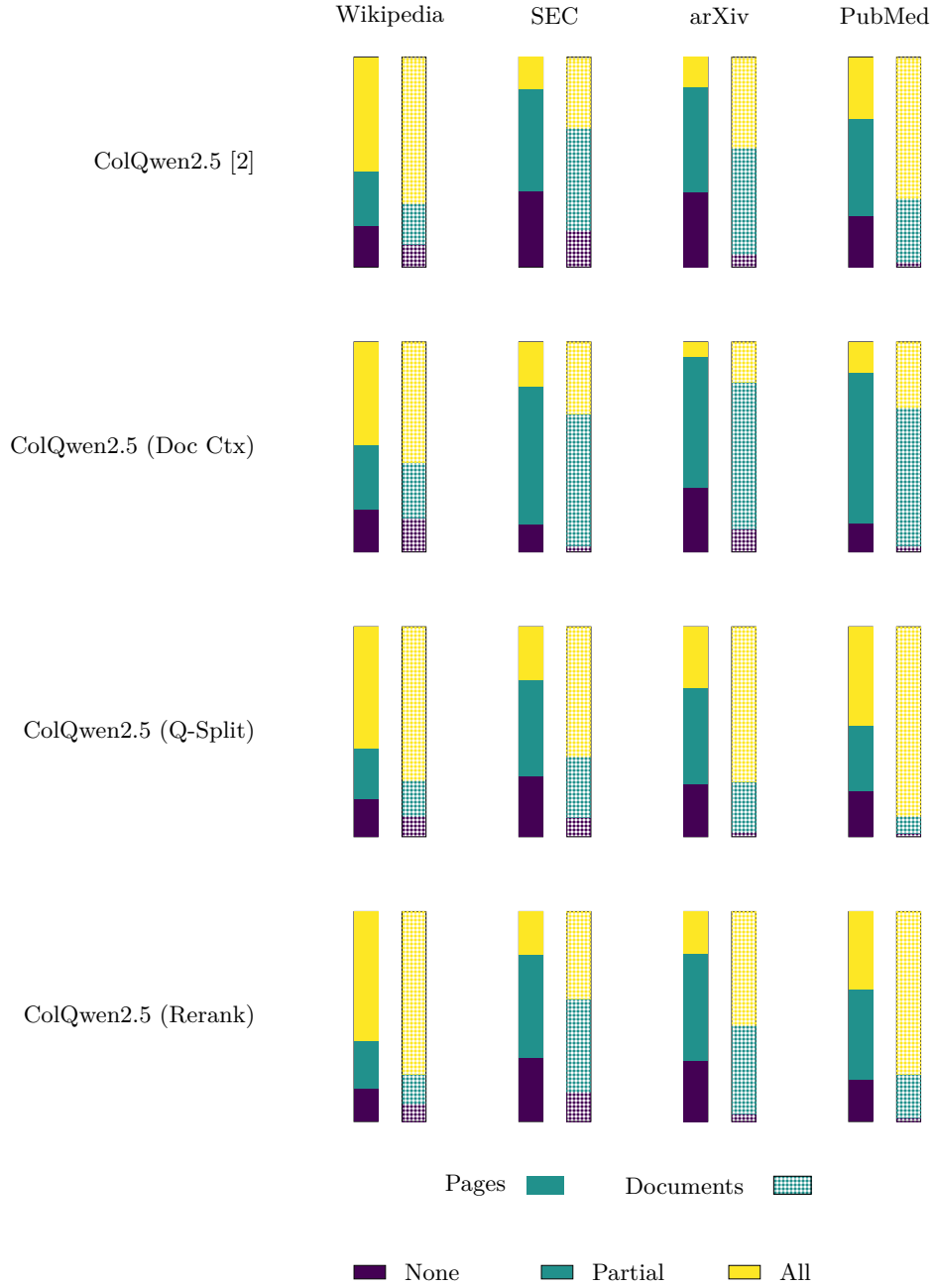


Figure 11: Analysis of Cross-Doc query retrieval results for each data source and the models ColQwen2.5 [2], ColQwen2.5 (Doc Ctx), ColQwen2.5 (Q-Split), and ColQwen2.5 (Rerank). Instead of just deciding whether a retrieval is successful, we breakdown the results into page-level (filled bars) and document-level (dotted bars) evaluation. Additionally, each level is divided into the three classes that all evidences could be retrieved, partially, or none which is indicated by the colors.

retrieval exhibits smaller differences between generators than the corresponding base and reranking settings. For example, the ColNomic difference decreases from 5.3 percentage points for base retrieval and 4.8 for reranking to 1.2 for document-context retrieval.

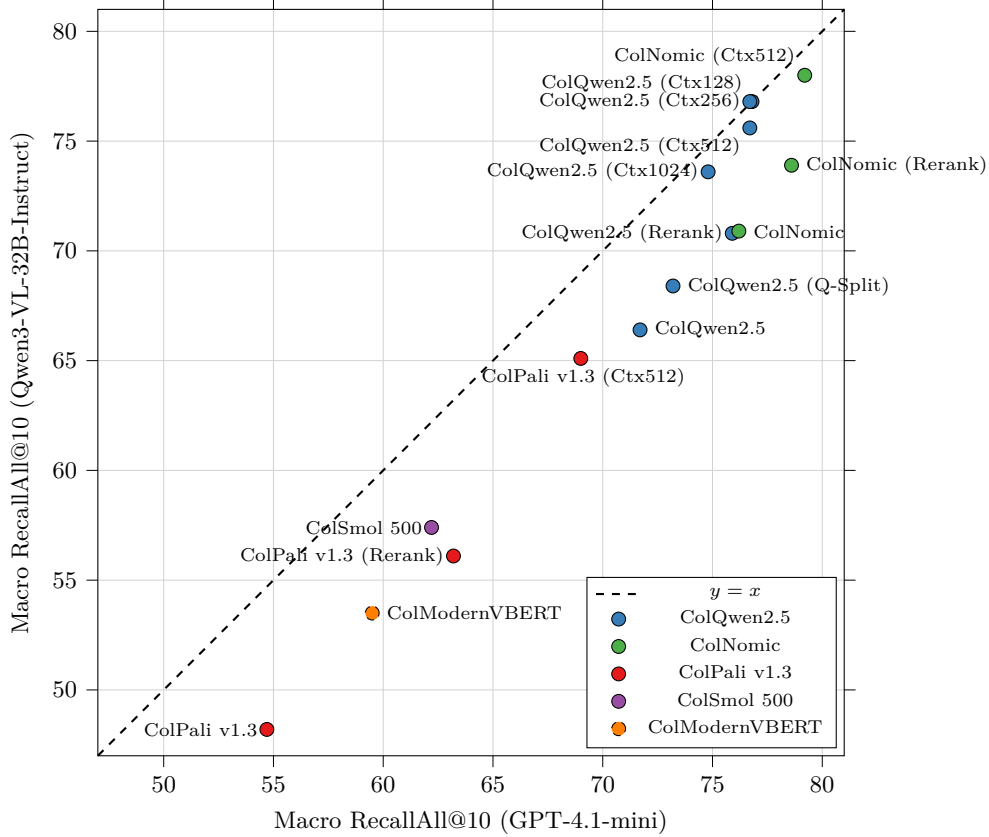


Figure 12: Comparison of structure-balanced Macro RecallAll@10 between queries generated by GPT-4.1-mini and Qwen3-VL-32B-Instruct. Each point represents a retrieval configuration evaluated on the same full corpus. Colors indicate the model family, and the dashed line denotes equal scores.

Tables 19 and 20 report nDCG, Recall, nDCGAll, and RecallAll for $k \in \{3, 5, 10\}$ for the same nine visual, text, and hybrid configurations under both aggregation protocols. At $k = 10$, all nine configurations obtain higher nDCGAll and RecallAll on GPT-4.1-mini queries than on Qwen3-VL-32B-Instruct queries under both Macro and Micro. ColNomic (Ensemble) achieves the highest scores among the nine configurations for both generators at all evaluated values of k . Its Macro nDCGAll@10 and RecallAll@10 are 71.51 and 81.74 on GPT-4.1-mini queries, compared with 68.85 and 80.46 on Qwen3-VL-32B-Instruct queries. The corresponding Micro scores are 78.05 and 87.80, and 74.96 and 85.78, respectively. OCR-BM25 achieves the highest nDCGAll@10 and RecallAll@10 among the three text baselines under both aggregation protocols for each generator. Its Macro RecallAll@10 is 64.19 on GPT-4.1-mini queries and 61.28 on Qwen3-VL-32B-Instruct queries. For ColNomic (Ctx), RRF fusion with OCR-BM25 reduces Macro RecallAll@10 from 79.20 to 79.14 on GPT-4.1-mini queries, but increases it from 78.03 to 78.61 on Qwen3-VL-32B-Instruct queries. Under Micro, the same fusion increases RecallAll@10 from 84.21 to 85.70 and from 81.70 to 83.82, respectively. These comparisons retain the source and evidence-structure composition of each generator subset. The 4,152-query subset balanced across generators and evidence structures is evaluated separately to examine the effect of query composition. Differences in retrieval performance may reflect both query formulation and dataset composition; they do not by themselves establish differences in annotation correctness.

Model / Setting	Aggregate	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
		@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColNomic (Ensemble)	Macro	70.58	74.42	76.89	74.01	81.76	88.02	57.31	65.67	71.51	61.16	72.66	81.74
	Micro	77.58	79.53	80.87	83.24	87.57	91.22	73.31	75.94	78.05	79.02	83.67	87.80
ColNomic	Macro	67.41	70.93	73.39	70.64	77.83	84.12	53.60	60.91	66.34	57.39	67.55	76.19
	Micro	73.27	75.35	76.85	78.92	83.60	87.78	68.95	71.61	73.85	74.68	79.63	84.23
ColNomic (Ctx)	Macro	65.72	70.13	73.18	68.99	77.99	85.78	51.46	60.75	67.74	55.26	68.07	79.20
	Micro	72.01	74.40	76.08	78.07	83.50	88.21	67.59	70.52	72.81	73.75	79.30	84.21
ColNomic (Rerank)	Macro	69.89	73.46	75.72	72.92	80.16	85.88	56.14	63.91	69.16	59.75	70.38	78.63
	Micro	76.86	78.72	79.93	82.04	86.15	89.42	72.64	75.15	77.06	77.89	82.30	86.02
OCR-BM25	Macro	51.75	56.13	59.29	55.44	64.52	72.77	38.54	46.26	52.18	42.46	53.83	64.19
	Micro	57.04	60.02	62.27	63.94	70.79	77.28	53.58	56.98	59.79	60.45	67.35	74.06
OCR-dense (BGE-M3)	Macro	25.75	28.31	30.38	27.72	33.02	38.44	16.35	19.88	23.27	18.27	23.72	29.85
	Micro	24.76	26.45	27.95	28.23	32.16	36.50	22.56	24.31	25.92	25.99	29.75	33.92
DOM-BM25	Macro	49.41	53.56	56.64	52.85	61.43	69.47	36.58	43.78	49.53	40.29	50.84	60.79
	Micro	52.75	55.46	57.60	59.10	65.31	71.47	49.33	52.41	55.11	55.65	61.90	68.24
ColNomic (Ctx) + OCR-BM25 (RRF)	Macro	63.55	68.21	71.42	67.87	77.44	85.69	49.81	59.19	66.28	54.30	67.67	79.14
	Micro	69.89	72.74	74.68	77.31	83.80	89.25	65.89	69.27	71.94	73.27	79.90	85.70
ColNomic (Rerank) + OCR-BM25 (RRF)	Macro	62.82	67.33	70.69	67.25	76.50	85.09	49.32	57.54	64.57	53.87	65.72	77.54
	Micro	69.52	72.50	74.59	76.91	83.68	89.48	65.58	69.15	72.04	72.93	79.85	86.10

Table 19: **Retrieval results on GPT-4.1-mini queries** (%; 59,983 queries). Macro gives equal weight to the six evidence structures within this subset, and Micro averages over its queries. All nine configurations retrieve from the same 388,548-page corpus.

Model / Setting	Aggregate	nDCG@k			Recall@k			nDCGAll@k			RecallAll@k		
		@3	@5	@10	@3	@5	@10	@3	@5	@10	@3	@5	@10
ColNomic (Ensemble)	Macro	67.97	72.32	75.15	71.63	80.46	87.68	52.85	62.02	68.85	56.71	69.57	80.46
	Micro	74.26	76.79	78.51	79.84	85.37	90.04	68.32	72.03	74.96	73.88	80.17	85.78
ColNomic	Macro	63.15	66.96	69.39	66.12	73.90	80.13	47.82	55.53	60.95	51.43	62.24	70.93
	Micro	69.67	72.02	73.64	75.11	80.32	84.80	63.96	67.19	69.78	69.59	75.28	80.39
ColNomic (Ctx)	Macro	64.64	69.19	72.42	68.08	77.40	85.73	48.87	58.20	65.69	52.87	65.97	78.03
	Micro	69.11	71.97	74.03	75.03	81.42	87.14	62.34	66.19	69.29	68.53	75.42	81.70
ColNomic (Rerank)	Macro	66.18	69.88	72.16	69.12	76.64	82.43	51.47	59.03	64.19	54.98	65.58	73.86
	Micro	72.89	75.07	76.54	77.78	82.54	86.51	67.25	70.36	72.83	72.28	77.59	82.28
OCR-BM25	Macro	51.25	55.30	58.36	54.95	63.36	71.36	37.44	44.33	49.93	41.33	51.61	61.28
	Micro	55.78	58.91	61.32	62.35	69.46	76.25	50.97	54.70	57.83	57.53	64.80	71.85
OCR-dense (BGE-M3)	Macro	22.34	24.32	25.84	23.96	28.11	32.11	13.71	16.34	18.59	15.47	19.59	23.69
	Micro	22.98	24.53	25.75	26.21	29.75	33.29	20.31	21.92	23.28	23.57	26.92	30.32
DOM-BM25	Macro	47.55	51.14	53.95	50.84	58.29	65.62	35.06	41.34	46.53	38.62	47.96	56.86
	Micro	50.65	53.41	55.55	56.50	62.79	68.83	46.39	49.72	52.50	52.26	58.74	65.01
ColNomic (Ctx) + OCR-BM25 (RRF)	Macro	62.71	67.57	71.17	66.94	76.93	86.19	47.47	56.74	64.89	51.90	65.24	78.61
	Micro	67.13	70.47	72.86	74.34	81.82	88.44	61.24	65.51	69.13	68.47	76.35	83.82
ColNomic (Rerank) + OCR-BM25 (RRF)	Macro	61.08	65.53	68.77	65.04	74.21	82.53	46.29	54.44	61.32	50.54	62.35	73.78
	Micro	66.67	69.80	72.05	73.54	80.56	86.77	61.19	65.20	68.58	68.09	75.51	82.51

Table 20: **Retrieval results on Qwen3-VL-32B-Instruct queries** (%; 211,777 queries). Macro gives equal weight to the six evidence structures within this subset, and Micro averages over its queries. All nine configurations retrieve from the same 388,548-page corpus.