RETRIEVAL-AWARE REPRESENTATION COMPRESSION FOR VISUAL DOCUMENT RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-vector visual document retrieval matches queries against patch-level embeddings to capture fine-grained evidence, but storing and scoring many vectors per page is costly. An effective compressor must reduce the vector count while preserving the query-specific evidence that supports retrieval accuracy. This emerges as a critical challenge due to the fact that semantically similar patches may match different queries, so merging them can weaken query-specific matches. Patch importance alone is also insufficient, as it does not indicate whether that evidence is already represented by semantically similar patches. In light of this, we present a training-free framework termed Retrieval-Aware Representation Compression (RARC), which jointly considers retrieval utility and local role distinctiveness under a fixed vector budget. RARC uses page-conditioned synthetic queries to construct retrieval-response fields from MaxSim assignments. These fields characterize local score sensitivity, with their magnitudes estimating patch utility and their directions describing the supported query directions. RARC protects useful patches whose response directions differ from those of their semantic neighbors and merges the remaining patches using utility-based weights. All operations are performed offline, without retraining the encoders or modifying online scoring. Experiments with two retrievers and three clustering backends on ViDoRe V2 and V3 show improvements of 1.27–3.19 nDCG@5 points in every benchmark average over the corresponding clustering baseline at a 10% vector budget. With H-Pool, indexes are approximately 90% smaller and MaxSim scoring is 7.17–8.20$\times$ faster than with uncompressed indexes. These results demonstrate the value of preserving locally distinctive retrieval evidence for accurate retrieval under aggressive compression.

1 INTRODUCTION

Multi-vector visual document retrievers represent each page with patch-level embeddings, allowing queries to match text, tables, and figures directly from the page image (Faysse et al., 2025). MaxSim selects the best-matching document vector for each query token (Khattab & Zaharia, 2020). This fine-grained interaction improves retrieval but makes index storage and scoring expensive. Reducing the number of document vectors therefore requires deciding which patches need individual representations and which can share a representative without losing useful retrieval evidence.

Semantic pooling and clustering reduce vector counts by grouping similar embeddings (Clavié et al., 2024; Ma et al., 2025). However, semantic similarity does not imply redundancy for retrieval. Nearby embeddings can support different query directions, and merging them can weaken matches that their neighbors cannot adequately represent. Recent methods use structural salience, attention, query-relevant coverage, or retrieval-demand weighting to guide compression (Liu et al., 2026; Qin et al., 2026; Mahdizadeh et al., 2026; Yuan et al., 2026). Yet patch importance alone cannot determine whether a useful retrieval role is already represented by neighboring patches. A limited vector budget must therefore account for both retrieval utility and local role distinctiveness when deciding what to preserve.

In light of this, we present a training-free compression framework termed *Retrieval-Aware Representation Compression* (RARC) that jointly considers these two properties (Figure 1). RARC describes a patch's retrieval role through a *retrieval-response field*, formed by summing the query-token vec-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

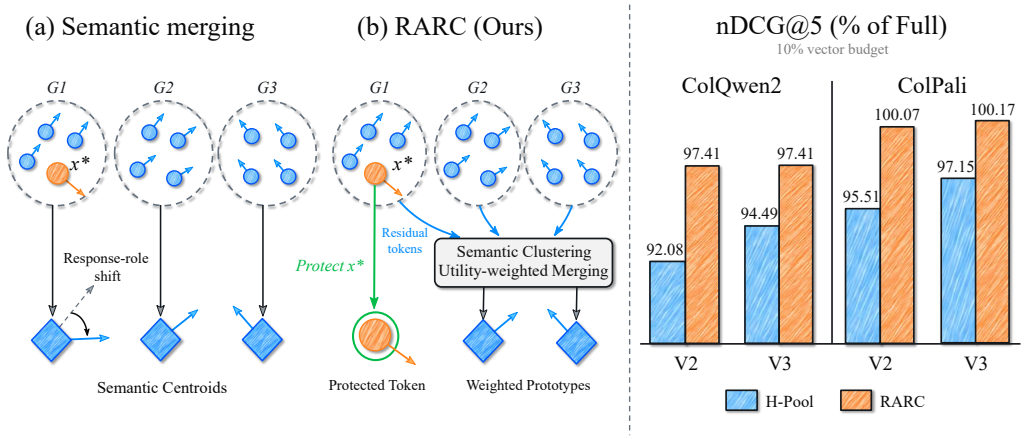


Figure 1: Retrieval-aware compression with RARC. Useful patches with locally distinctive retrieval roles are retained, and the remaining patches are merged using utility-based weights. Right: H-Pool and RARC at the same 10% vector budget, compared with the uncompressed index.

tors for which that patch wins MaxSim. Its magnitude estimates retrieval utility, while its direction characterizes the supported query directions. With locally unchanged, unique MaxSim winners, the field is the gradient of the aggregate probe score with respect to the patch embedding (Section 3). This connects compression decisions to the sensitivity of the retrieval scorer.

These response fields allow us to examine whether semantic similarity reflects similarity in retrieval roles. Using page-conditioned synthetic queries, we estimate the fields and compare semantic neighborhoods with neighborhoods defined by response directions. On ViDoRe V3, semantic and response-direction top-20 neighborhoods disagree by 45.00% and 41.88% for ColQwen2 and ColPali among eligible nonzero-response tokens. Greater disagreement correlates with larger representation errors after semantic merging. These observations provide empirical support for distinguishing retrieval roles within semantic neighborhoods, beyond judging patches by embedding similarity alone.

To account for differences between semantic similarity and retrieval roles, RARC uses the response fields in two complementary operations. Magnitude-gated directional protection retains useful patches whose response directions differ from those of their semantic neighbors. Utility-weighted merging gives greater weight to useful patches when constructing representatives for the remaining semantic clusters. Protection thus preserves distinctive roles in individual vectors, while weighted merging retains relevant content in shared representations. Semantic grouping remains important because approximately 88–89% of patches have zero probe-response fields in our measurements. Combining document geometry with retrieval responses allows RARC to group all patches while using observed retrieval roles to guide protection and merging. Both operations run offline with frozen encoders and standard clustering backends, without requiring evaluation queries or relevance labels for the indexed pages. Online retrieval uses the original query encoder and MaxSim scorer.

Experiments with ColQwen2 and ColPali on ViDoRe V2 and V3 show improvements of 1.27–3.19 nDCG@5 points in every benchmark average over the corresponding clustering baseline at 10% retention. With H-Pool, indexes are approximately 90% smaller and MaxSim scoring is 7.17–8.20× faster than with uncompressed indexes. Ablations confirm the benefits of both modules, and analysis shows greater fidelity recovery on pages with larger semantic–retrieval mismatch. The main contributions are summarized as follows.

- We formulate retrieval-response fields that connect patch utility and supported query directions to local MaxSim sensitivity, and quantify the mismatch between semantic neighborhoods and retrieval roles.
- We develop a training-free framework that combines magnitude-gated directional protection with utility-weighted semantic merging, jointly deciding which patches retain individual vectors and how the remaining patches share a fixed vector budget.

- We demonstrate consistent gains in benchmark averages across two retrievers and three clustering backends evaluated on twelve splits, and show how preserving local retrieval roles improves compression quality with the original retrieval scorer.

2 Related Work

2.1 Visual Retrieval and Efficient Late Interaction

Visual document retrieval encodes page images to represent their text, images, and layout. DSE represents each page with a single embedding (Ma et al., 2024), whereas ColPali combines vision-language representations with ColBERT-style late interaction (Faysse et al., 2025; Khattab & Zaharia, 2020). Several methods improve the efficiency of multi-vector retrieval. ColBERTv2 compresses individual token vectors through residual encoding (Santhanam et al., 2022), and MUVERA maps multiple vectors to fixed-dimensional encodings (Dhulipala et al., 2024). RARC reduces the number of document vectors while retaining the original MaxSim scorer.

2.2 Document-Side Vector Compression

Pruning methods remove vectors according to token-level signals (Lassance et al., 2021; Acquavia et al., 2023; Yan et al., 2025). SAP selects structural anchors from attention graphs after score-retention calibration (Liu et al., 2026). Merging methods combine multiple tokens into shared representations. Token Pooling studies semantic aggregation (Clavié et al., 2024), Light-ColPali compares spatial and semantic reduction (Ma et al., 2025), and Prune-then-Merge combines pruning with hierarchical aggregation (Yan et al., 2026b). ColChunk and ColParse use spatial or layout information (Yan et al., 2026a;c), while AGC uses attention-guided anchors and aggregation (Qin et al., 2026). RARC uses the magnitude and direction of retrieval responses to determine which patches should be retained and how the remaining patches should be merged.

2.3 Query-Guided Compression

Promptagator uses synthetic queries to train dense retrievers (Dai et al., 2023). RARC uses them to measure the retrieval behavior of frozen embeddings. Voronoi pruning measures token influence through the regions of query space in which each token wins (Kankanampati et al., 2026). Margin-Merge combines training-query prototypes with learned representative synthesis and ranking-margin distillation (Mahdizadeh et al., 2026). MAGIC performs training-free compression using retrieval-demand and balanced target marginals in an optimal-transport formulation (Yuan et al., 2026). RARC considers how a useful patch's response direction differs from those of nearby patches in semantic space. This local comparison determines which patches are protected, while response magnitude determines the weights of the remaining patches during semantic merging. RARC thus uses both patch utility and local differences in retrieval roles to guide compression under a fixed budget.

3 From Semantic Similarity to Retrieval Roles

Semantic merging groups document tokens by embedding similarity under a limited vector budget, but similar tokens may support different query directions. We describe each token's retrieval role using a response field whose magnitude and direction guide token protection and merging.

3.1 Retrieval-Response Fields and Local Sensitivity

Frozen encoders f_q and f_d map query q and page d to $Q = f_q(q) = \{q_s\}_{s=1}^{m}$ and $X = f_d(d) = \{x_i\}_{i=1}^{n}$, with vectors in $\mathbb{R}^h$. At indexing time, a compressor produces $\widetilde{X} = \pi_K(X)$ under a budget $K < n$. We examine which properties compression should preserve using the late-interaction score:

$$S(Q, X) = \sum_{s=1}^{m} \max_{i \in \{1,\dots,n\}} q_s^\top x_i. \tag{1}$$

The **MaxSim winner** of q_s is $a_s = \arg\max_i q_s^\top x_i$, breaking ties by original token order. To estimate token roles during indexing, consider probes $\mathcal{P} = \{Q^{(r)}\}_{r=1}^{M}$, where $Q^{(r)} = \{q_s^{(r)}\}_{s=1}^{m_r}$ (Section 4.1). Let $a_{r,s}$ be the winner of $q_s^{(r)}$. The **retrieval-response field** aggregates the probe vectors assigned to x_i:

$$R_i = \sum_{r=1}^{M} \sum_{s=1}^{m_r} \mathbf{1}[a_{r,s} = i]\, q_s^{(r)}. \tag{2}$$

Here, $\mathbf{1}[\cdot]$ is the indicator function. Aligned probe vectors reinforce one another, while opposing vectors cancel. Thus, $\|R_i\|_2$ reflects both the magnitude and directional agreement of assigned query vectors, rather than only the number of MaxSim wins.

Local MaxSim sensitivity. Consider the aggregate probe score $S_{\mathcal{P}}(X) = \sum_{r=1}^{M} S(Q^{(r)}, X)$, with the probe vectors held fixed. In a local region where all winners are unique and unchanged, grouping the query terms by their winning document tokens gives

$$S_{\mathcal{P}}(X) = \sum_{i=1}^{n} R_i^\top x_i, \qquad \nabla_{x_i} S_{\mathcal{P}}(X) = R_i. \tag{3}$$

For nonzero R_i, a perturbation Δx_i that preserves the winners therefore changes the score by

$$\Delta S_{\mathcal{P}} = R_i^\top \Delta x_i = \|R_i\|_2 \left\langle \frac{R_i}{\|R_i\|_2}, \Delta x_i \right\rangle. \tag{4}$$

The magnitude measures local sensitivity, while the direction determines how a perturbation affects the score. We use this magnitude as a proxy for **token retrieval utility**, $\mathrm{TRU}_i = \|R_i\|_2$, with log-scaled utility and unit response direction

$$u_i = \log(1 + \mathrm{TRU}_i), \qquad v_i = \frac{R_i}{\mathrm{TRU}_i}. \tag{5}$$

The direction v_i is defined for $\mathrm{TRU}_i > 0$. With fixed probes and winners, these quantities describe score sensitivity in the ambient space. After merging, MaxSim recomputes winners over the compressed representatives. Appendix A.2 gives the sensitivity and winner-stability analysis.

3.2 Semantic–Retrieval Misalignment and Its Implications

Similar document embeddings can have different retrieval roles. Embedding similarity compares document vectors, while response similarity compares the query directions they support. Document vectors with cosine similarity arbitrarily close to one can have opposite response directions (Appendix A.3). Embedding similarity alone is therefore insufficient to determine whether two tokens have interchangeable retrieval roles.

We study this difference by comparing local neighborhoods of active tokens, defined as tokens with $\mathrm{TRU}_i > 0$. For each active token, let $\mathcal{N}_k^{\mathrm{sem}}(i)$ denote its k nearest other active tokens under document-embedding cosine similarity, and let $\mathcal{N}_k^{\mathrm{ret}}(i)$ denote its k nearest other active tokens under response-direction similarity $v_i^\top v_j$. For pages with at least $k+1$ active tokens, their neighborhood disagreement is

$$\mu_k(i) = 1 - \frac{|\mathcal{N}_k^{\mathrm{sem}}(i) \cap \mathcal{N}_k^{\mathrm{ret}}(i)|}{k}. \tag{6}$$

Larger $\mu_k(i)$ indicates stronger misalignment, with zero indicating identical neighborhoods. On ViDoRe V3, mean top-20 non-overlap reaches **45.00% for ColQwen2** and **41.88% for ColPali**.

Under hierarchical merging at a 10% budget, token-level Spearman correlations between μ_{20} and cosine-fidelity loss are **0.327 for ColQwen2** and **0.314 for ColPali**. This loss is the cosine distance

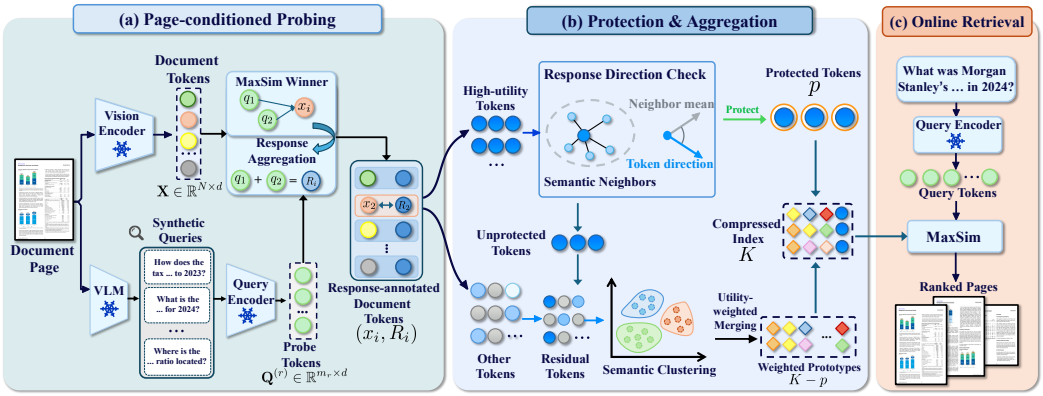


Figure 2: Overview of RARC. Page-conditioned probing estimates token responses. Directional protection and utility-weighted merging then construct a compressed index for standard MaxSim retrieval.

between an original token and its compressed representative. Tokens with greater neighborhood disagreement tend to incur larger compression errors.

Implication for compression. Response fields reveal local retrieval roles, while document embeddings are available even for tokens not reached by the probes. Clustering response fields directly performs worse than combining these signals in our experiments (Appendix B.4). RARC uses document embeddings for grouping, local response directions for token protection, and response magnitudes for weighted merging. We use μ_k to measure neighborhood disagreement and the directional comparison in Section 4.2 to define the protection rule.

4 Retrieval-Aware Representation Compression

RARC uses token responses to select tokens for protection and merge the others (Figure 2). Magnitude-gated directional protection preserves tokens with strong responses and locally distinctive directions. Utility-weighted merging emphasizes tokens with stronger responses within each remaining semantic group. Both steps run offline with frozen encoders, and the compressed index uses the standard MaxSim scorer.

4.1 Page-Conditioned Query Probing

For each page, a vision-language model generates natural-language retrieval queries from the page image. The frozen query encoder f_q encodes these queries into $\mathcal{P}$. MaxSim assignments on the original document vectors yield R_i through Eq.2, and u_i, v_i through Eq.5. These synthetic probes estimate page-specific retrieval roles without access to evaluation queries. We probe each page independently during index construction and reuse the resulting index for online retrieval.

4.2 Magnitude-Gated Directional Protection

RARC preserves tokens whose responses are strong and whose directions differ from those of their semantic neighbors. Let $\mathcal{A}_k(i)$ contain the $\min(k, n-1)$ nearest other document tokens under cosine similarity, and $\mathcal{V}_i = \{j \in \mathcal{A}_k(i) : \mathrm{TRU}_j > \epsilon_R\}$ retain valid-response neighbors, with $\epsilon_R = 10^{-8}$. We select semantic neighbors before filtering them for valid responses. For a nonempty $\mathcal{V}_i$ with nonzero mean, define

$$b_i = \frac{1}{|\mathcal{V}_i|} \sum_{j \in \mathcal{V}_i} v_j, \qquad m_i = \frac{b_i}{\|b_i\|_2}. \tag{7}$$

Write $\mathrm{valid}_i = 1$ when $\mathrm{TRU}_i > \epsilon_R$, $|\mathcal{V}_i| \geq 3$, and $\|b_i\|_2 > \epsilon_R$. For valid tokens, define the directional deviation as $\delta_i = 1 - v_i^\top m_i$. The utility threshold prevents weak responses from triggering

protection solely because their directions are unusual. With thresholds τ and γ_{dir}, the candidate set is

$$\mathcal{I}_{\text{cand}} = \{i : \text{valid}_i = 1,\ u_i \geq \tau,\ \delta_i \geq \gamma_{\text{dir}}\}. \tag{8}$$

For retention ratio $\beta \in (0, 1]$, the budget is $K = \max(1, \lceil \beta n \rceil)$. RARC protects all candidates up to K, selecting those with the largest δ_i when needed and breaking ties by original token order. The resulting set $\mathcal{I}_{\text{prot}}$ occupies $p = |\mathcal{I}_{\text{prot}}|$ slots, leaving $K - p$ for residual compression.

4.3 Utility-Weighted Merging

When $p < K$, a semantic backend groups the unprotected tokens using their original embeddings within the remaining $K - p$ slots. If all vectors are nonzero, this gives $K - p$ nonempty clusters $\mathcal{C}_{\text{res}}$. With weights $w_i = u_i^{\alpha} + \epsilon_w$, each cluster C receives a utility-weighted representative:

$$c_C = \frac{\sum_{i \in C} w_i x_i}{\left\| \sum_{i \in C} w_i x_i \right\|_2}. \tag{9}$$

The exponent $\alpha > 0$ controls the relative weight of high-utility tokens, and $\epsilon_w > 0$ ensures that tokens with zero response also contribute to merging. For fixed clusters and a nonzero weighted sum, c_C minimizes utility-weighted squared representation error over unit vectors (Appendix A.4).

Let Z_0 contain any zero rows reserved by the backend adapter, with $|\mathcal{C}_{\text{res}}| + |Z_0| = K - p$. The final index is

$$\widetilde{X} = \left[(\text{norm}_{\epsilon_x}(x_i))_{i \in \mathcal{I}_{\text{prot}}}, (c_C)_{C \in \mathcal{C}_{\text{res}}}, Z_0 \right]. \tag{10}$$

The compressed index contains exactly K rows. When $p = K$, residual clustering is skipped. For residual compression, the vMFMM adapter reserves one slot for explicit zero padding when present. Otherwise Z_0 is empty. Appendix A.5 describes the safeguarded normalization norm_{ϵ_x}, including its treatment of zero weighted sums, and the backend implementations.

5 Experiments

5.1 Experimental Setup

Retrievers and configuration. We evaluate frozen ColQwen2-2B and ColPali-3B retrievers (Faysse et al., 2025). Qwen3-VL-8B (Bai et al., 2025) generates up to ten synthetic queries from each page. We fix $k = 20$ and $\alpha = 1.5$. Thresholds are calibrated on ViDoRe V1 `tatdqa_text` and transferred to V2 and V3 without retuning, giving $(\tau, \gamma_{\text{dir}}) \approx (1.93, 0.10)$ for ColQwen2 and $(1.82, 0.03)$ for ColPali. Appendix B.1 reports calibration and sensitivity results.

Evaluation. ViDoRe V2 and V3 contain four and eight domain splits, respectively (Macé et al., 2025; Loison et al., 2026). Each query is scored against its own split's document pool. We report nDCG@5 multiplied by 100, with equal split weights within each benchmark and across all twelve splits in the final average. At 10% retention, each page has $K = \max(1, \lceil 0.10n \rceil)$ output vectors, including protected singletons.

Comparisons. We evaluate the uncompressed index (Full), Random Merging (five-seed means), Prune-then-Merge (Yan et al., 2026b), SAP (Liu et al., 2026), Attention Pooling, vMFMM, K-Means, and H-Pool (Clavié et al., 2024). Attention Pooling selects anchors by attention scores, assigns tokens by cosine similarity, and averages each group. RARC uses the same vector budget as its matched merging backend. Appendix B.1 details the clustering configurations.

5.2 Main Results

RARC improves the average retrieval performance in all twelve combinations of two retrievers, three merging backends, and two benchmarks (Table 1). Gains over matched backends are 2.42–

Table 1: Retrieval performance on ViDoRe V2 and V3. Results are nDCG@5 multiplied by 100. Compressed methods retain 10% of the document vectors, including protected vectors. V2 and V3 average over their four and eight splits, respectively, and All averages over all twelve splits, with equal weights. The best and second-best compressed results are bold and underlined, respectively.

Method	ViDoRe V2				ViDoRe V3								Average		
	Econ.	ESG	Bio.	ESG-H	C.S.	Ener.	Fin. EN	Fin. FR	H.R.	Ind.	Phar.	Phys.	V2	V3	All
ColQwen2-2B (original patch number: 747)															
Uncompressed	52.01	54.23	56.69	57.83	66.88	48.80	39.83	22.79	43.70	38.47	52.66	39.45	55.19	44.07	47.78
Random Merging	44.40	37.63	51.79	38.34	57.97	35.19	26.25	12.45	32.66	26.80	44.69	34.64	43.04	33.83	36.90
Prune-then-Merge	43.12	38.48	51.63	37.34	61.28	39.41	30.97	15.62	36.62	30.80	49.00	36.79	42.64	37.56	39.26
SAP	43.42	44.92	51.25	40.71	60.63	40.61	29.24	16.13	38.14	29.19	47.60	37.92	45.08	37.43	39.98
Attention Pooling	45.86	47.47	52.66	43.69	63.15	42.62	32.00	18.32	39.61	32.60	48.32	36.51	47.42	39.14	41.90
vMFMM	**50.07**	48.46	53.93	50.18	64.75	44.59	36.95	19.50	40.56	34.93	51.13	38.18	50.66	41.32	44.44
K-Means	47.80	48.01	54.22	50.59	64.88	45.12	36.83	19.08	40.92	34.20	51.02	37.23	50.15	41.16	44.16
H-Pool	48.33	50.44	53.65	50.85	64.88	45.55	37.84	19.87	40.42	34.74	50.75	39.06	50.82	41.64	44.70
RARC(vMFMM)	49.52	**51.70**	**55.80**	57.65	65.06	47.54	37.53	21.41	43.02	35.51	52.11	**39.81**	53.67	42.75	46.39
RARC(K-Means)	47.49	50.81	54.54	**59.64**	**65.57**	**47.67**	38.25	**21.81**	**44.01**	35.51	52.28	39.24	53.12	**43.04**	46.40
RARC(H-Pool)	48.95	51.17	55.74	59.20	64.99	47.43	**38.95**	20.62	43.84	**35.64**	**52.32**	39.68	**53.76**	42.93	**46.54**
ColPali-3B (original patch number: 1031)															
Uncompressed	48.29	54.59	55.89	57.83	63.76	45.54	34.82	21.78	43.64	35.00	53.00	39.27	54.15	42.10	46.12
Random Merging	41.91	41.65	51.37	45.30	57.58	35.97	26.37	14.75	36.18	26.80	47.03	37.78	45.06	35.31	38.56
Prune-then-Merge	40.75	40.08	52.41	44.70	61.17	39.65	31.46	17.23	39.15	32.49	51.08	38.61	44.48	38.86	40.73
SAP	45.97	44.45	50.65	50.88	57.83	40.28	30.44	18.76	37.06	27.78	48.79	35.88	47.99	37.10	40.73
Attention Pooling	44.85	46.99	51.89	47.19	58.57	40.07	31.38	17.58	39.44	30.21	47.99	37.97	47.73	37.90	41.18
vMFMM	45.43	51.35	54.70	51.41	62.96	42.45	33.76	20.76	41.51	32.06	52.32	39.29	50.72	40.64	44.00
K-Means	46.70	51.50	54.76	53.79	62.60	42.81	33.94	20.38	41.90	32.94	52.13	39.48	51.69	40.77	44.41
H-Pool	46.65	51.80	54.17	54.37	62.64	42.48	34.50	20.49	42.22	33.43	52.07	39.36	51.75	40.90	44.51
RARC(vMFMM)	47.51	**54.74**	56.15	57.27	63.44	**45.39**	34.66	22.23	43.41	33.88	52.47	40.70	53.91	42.02	45.99
RARC(K-Means)	**48.82**	54.70	**56.17**	58.59	63.25	44.91	35.08	**22.86**	43.95	34.50	52.39	**40.80**	**54.57**	**42.22**	**46.34**
RARC(H-Pool)	48.23	53.87	55.73	**58.84**	**63.48**	44.67	**35.38**	21.93	**44.21**	**34.51**	**52.69**	40.52	54.17	42.17	46.17

3.19 points on V2 and 1.27–1.88 on V3, with a maximum relative gain of 6.29% for ColPali with vMFMM on V2. Some individual domains show exceptions. ColPali's RARC(K-Means) also exceeds Full on both benchmarks, while RARC(H-Pool) slightly exceeds Full in the twelve-split average.

On V3, RARC improves H-Pool and K-Means at every tested retention ratio from 5% to 30% (Figure 3). H-Pool gains decrease from 1.74/1.62 points at 5% to 0.83/0.57 at 30% for ColQwen2/ColPali. Gains are generally larger at lower retention ratios, with some exceptions and non-monotonic trends on V2 (Appendix B.2).

RARC(H-Pool) reduces index storage by approximately 90% and achieves 8.20×/7.17× pure MaxSim speedups over Full for ColQwen2/ColPali (Table 3). Timing uses inputs already on the GPU and excludes encoding. Online costs are comparable to H-Pool at the same budget. Appendix B.2 reports offline probing and compression costs.

5.3 Ablation Study

Protection preserves selected tokens individually, while weighting emphasizes useful tokens when aggregating the remaining groups. Each component improves V2 retrieval with H-Pool, and using both gives the best results (Table 2a). ColPali V3 also shows positive, partly overlapping contributions from both components (Appendix B.3). On V3, varying the neighborhood size or weight exponent changes scores by at most 0.14/0.07 points from the default for ColQwen2/ColPali (Table 2b,c). All tested threshold configurations outperform H-Pool in the benchmark averages, although the scores vary more on V2 (Appendix B.1).

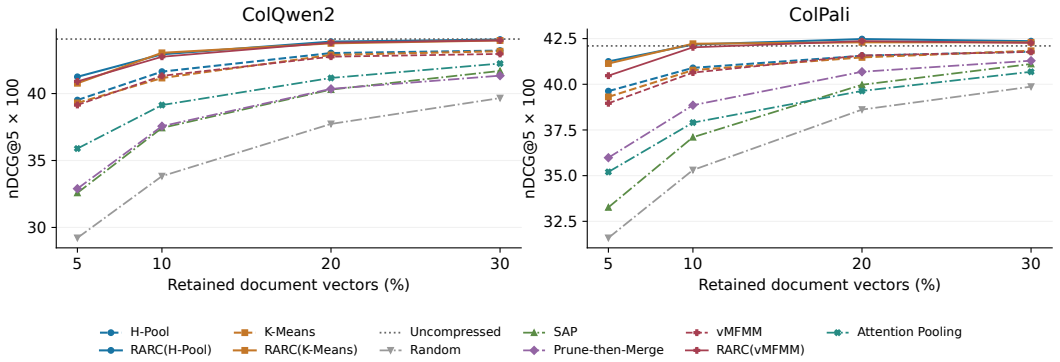


Figure 3: Retrieval performance at different vector budgets on ViDoRe V3. Solid and dashed curves show RARC and the corresponding backends, respectively. Dotted lines indicate Full, and dash-dotted curves show additional baselines.

Table 2: Ablations with H-Pool at 10% retention (nDCG@5 $\times 100$). P/W denote protection/weighting. Each parameter sweep varies one setting, with the other fixed at $k = 20$ or $\alpha = 1.5$.

(a) Components, V2

P	W	ColQwen2	ColPali
–	–	50.82	51.75
✓	–	53.06	53.77
–	✓	53.48	53.53
✓	✓	**53.77**	**54.17**

(b) Neighborhood, V3

k	ColQwen2	ColPali
10	42.90	42.16
20	42.93	42.17
25	42.83	42.19
30	42.80	42.18

(c) Weight exponent, V3

α	ColQwen2	ColPali
1.0	42.90	42.20
1.5	42.93	42.17
2.0	42.90	42.16
2.5	42.86	42.10

Table 3: Retrieval cost on V2 at batch size 1. RARC uses H-Pool at 10% retention. Latency covers pure MaxSim with GPU-resident inputs. GPU reports peak retrieval memory.

Method	Index (GiB)↓	Latency (ms)↓	Peak GPU (GiB)↓
ColQwen2			
Uncompressed	0.8105	3.918	0.793
RARC	0.0816	0.478	0.093
ColPali			
Uncompressed	1.1114	3.692	1.044
RARC	0.1122	0.515	0.118

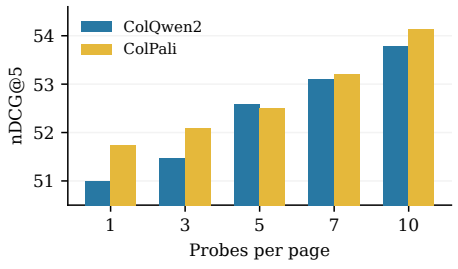


Figure 4: Probe-count sensitivity on V2 with H-Pool at 10% retention.

Table 4: Response and ranking changes on V3 with H-Pool at 10% retention. (a) ColPali means with equal split weights, using fixed relevant–competing page pairs selected by Full, where $\Delta M = E + D$. (b) RARC minus vanilla. Negative first-relevant rank changes indicate improvement. Positive changes in mean absolute relevant-page rank distance indicate greater deviation from Full.

(a) Response changes

Reference	E	D	ΔM
Vanilla	0.0765	0.0323	0.1089
Full	0.0034	0.1213	0.1247

(b) Ranking changes

Model	First relevant rank ↓	Rank distance to Full
ColQwen2	−1.514	+4.213
ColPali	−1.995	+3.400

More probes improve retrieval, while replacing them with queries from other pages reduces the gain (Figure 4 and Appendix B.3). RARC performs best among selectors with the same protection counts, including utility-only selection and perturbed directions. Utility-only selection is competitive, and replacements with similar utility have different effects on the two retrievers (Appendix B.3). These results support the joint selection rule and the use of probes covering the page content.

5.4 Analysis of Response Guidance

RARC improves token-to-representative fidelity more on pages with high mismatch (Figure 5). We define the quartiles using the original representations and measure recovery as the reduction in cosine distance relative to H-Pool. Reapplying probes to preselected high-risk pages also reduces response-direction loss by 0.0745/0.0619 for ColQwen2/ColPali, even though the remaining clusters have greater directional variation. RARC thus better preserves representations on pages where the mismatch in Section 3 is more severe. Appendix B.4 analyzes the effects on scores and rankings.

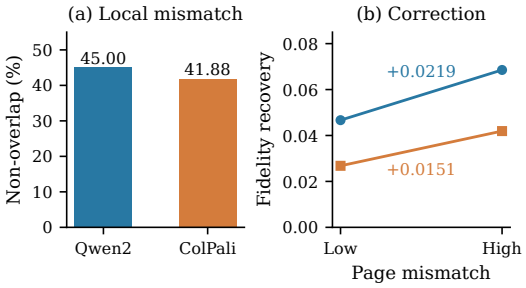


Figure 5: Local mismatch and representation recovery on V3. Blue and orange indicate ColQwen2 and ColPali, respectively. The right panel shows fidelity recovery over H-Pool in the low and high mismatch quartiles.

On V3, 87.81%/89.09% of tokens have zero probe-response fields. Direct response-field clustering reaches 50.85/52.40 on V2, compared with 53.76/54.17 for RARC. These results support using document geometry to maintain semantic coverage and sparse responses to guide token protection and weighted aggregation (Appendix B.4).

For ColPali on V3 with H-Pool, we select each query's highest-scoring judged relevant page and highest-scoring unjudged competitor under Full, then keep these pairs fixed. Let G/L sum positive/absolute negative query-token MaxSim changes from a reference index, with superscripts $+/-$ denoting relevant/competing pages. Their margin change is

$$\Delta M = \underbrace{G^{+} - G^{-}}_{E\text{ : net enhancement}} + \underbrace{L^{-} - L^{+}}_{D\text{ : net suppression}} . \tag{11}$$

Net enhancement dominates relative to vanilla, while net suppression dominates relative to Full (Table 4a). We call the greater suppression of competing responses on these fixed pairs *functional denoising*. The sign of net suppression depends on pair selection, and suppression alone does not determine nDCG@5 (Appendix B.4).

On V3, RARC(H-Pool) improves mean first-relevant rank by 1.514/1.995 positions while moving judged relevant-page ranks farther from Full (Table 4b). K-Means shows the same pattern. Better retrieval therefore does not require reproducing the uncompressed ranking exactly.

6 Conclusion

In this paper, we present a training-free framework termed RARC for multi-vector visual document retrieval, which addresses the mismatch between semantic similarity and retrieval redundancy by jointly considering patch utility and local role distinctiveness. Its retrieval-response fields connect these properties to local MaxSim sensitivity and guide patch protection and weighted merging under a fixed vector budget. Experiments with two retrievers and three clustering backends show consistent gains in benchmark averages over the corresponding compression baselines, with substantially smaller indexes and faster MaxSim scoring than uncompressed retrieval. These findings demonstrate the value of considering retrieval utility and local role differences when compressing document representations without changing the original encoders or retrieval scorer.

AI USE STATEMENT

In this work, we used generative AI tools to draft sections of the paper, polish writing primarily for grammar and clarity, search and review related literature to check for potential overlap or plagiarism in our ideas, prove some mathematical claims, and generate portions of the code. We have not used generative AI tools for idea conception, experiment execution, or verification of experimental results. We have reviewed all AI-assisted work. All AI-assisted writing was manually revised, all AI-suggested references were manually checked, and generated code and proofs were cross-checked by two authors and verified for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we describe the datasets, models, and evaluation protocol in Section 5.1, with additional experimental details and configurations in Appendix B. The assumptions and proofs supporting our theoretical claims are provided in Appendix A. Our code and synthetic queries will be included in the supplementary materials, and the query-generation prompts are provided in Appendix E.

REFERENCES

Antonio Acquavia, Craig Macdonald, and Nicola Tonellotto. Static pruning for multi-representation dense retrieval. In *Proceedings of the ACM Symposium on Document Engineering 2023*, pp. 1–10, 2023.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Benjamin Clavié, Antoine Chaffin, and Griffin Adams. Reducing the footprint of multi-vector retrieval with minimal performance impact via token pooling. *arXiv preprint arXiv:2409.14683*, 2024.

Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. Promptagator: Few-shot dense retrieval from 8 examples. In *International Conference on Learning Representations*, 2023.

Laxman Dhulipala, Majid Hadian, Rajesh Jayaram, Jason Lee, and Vahab Mirrokni. MUVERA: Multi-vector retrieval via fixed dimensional encoding. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. ColPali: Efficient document retrieval with vision language models. In *International Conference on Learning Representations*, 2025.

Yash Kankanampati, Yuxuan Zong, Nadi Tomeh, Benjamin Piwowarski, and Joseph Le Roux. A Voronoi cell formulation for principled token pruning in late-interaction retrieval models. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2026.

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*, pp. 39–48, 2020.

Carlos Lassance, Maroua Maachou, Joohee Park, and Stéphane Clinchant. A study on token pruning for ColBERT. *arXiv preprint arXiv:2112.06540*, 2021.

Zhuchenyang Liu, Ziyu Hu, Yao Zhang, and Yu Xiao. Structural anchor pruning: Training-free multi-vector compression for visual document retrieval. *arXiv preprint arXiv:2601.20107*, 2026. Accepted to EMNLP 2026 (Main Track).

António Loison, Quentin Macé, Antoine Edy, Victor Xing, Tom Balough, Gabriel de Souza P. Moreira, Bo Liu, Manuel Faysse, Céline Hudelot, and Gautier Viaud. ViDoRe V3: A comprehensive evaluation of retrieval augmented generation in complex real-world scenarios. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16570–16600, 2026.

Xueguang Ma, Sheng-Chieh Lin, Minghan Li, Wenhu Chen, and Jimmy Lin. Unifying multimodal retrieval via document screenshot embedding. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6492–6505, 2024.

Yubo Ma, Jinsong Li, Yuhang Zang, Xiaobao Wu, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Haodong Duan, Jiaqi Wang, Yixin Cao, and Aixin Sun. Towards storage-efficient visual document retrieval: An empirical study on reducing patch-level embeddings. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19568–19580, 2025.

Quentin Macé, António Loison, and Manuel Faysse. ViDoRe benchmark V2: Raising the bar for visual retrieval. *arXiv preprint arXiv:2505.17166*, 2025.

Ailar Mahdizadeh, Aria Salari, Sohail Rajabi, Shahriar Mirabbasi, Panos Nasiopoulos, and Alireza Morsali. Coverage matters: MarginMerge for compressing multi-vector visual document retrievers. *arXiv preprint arXiv:2608.02969*, 2026.

Hanxiang Qin, Alexander Martin, Rohan Jha, Chunsheng Zuo, Reno Kriz, and Benjamin Van Durme. Multi-vector index compression in any modality. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1519–1530, 2026.

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: Human Language Technologies*, pp. 3715–3734, 2022.

Yibo Yan, Guangwei Xu, Xin Zou, Shuliang Liu, James Kwok, and Xuming Hu. DocPruner: A storage-efficient framework for multi-vector visual document retrieval via adaptive patch-level embedding pruning. *arXiv preprint arXiv:2509.23883*, 2025.

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. *arXiv preprint arXiv:2604.10167*, 2026a.

Yibo Yan, Mingdong Ou, Yi Cao, Xin Zou, Jiahao Huo, Shuliang Liu, James Kwok, and Xuming Hu. Sculpting the vector space: Towards efficient multi-vector visual document retrieval via prune-then-merge framework. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 24883–24925, 2026b.

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. *arXiv preprint arXiv:2603.01666*, 2026c.

Xu Yuan, Hua Liu, Wenqi Fan, and Qing Li. MAGIC: Marginal-guided compression with optimal transport for efficient visual document retrieval. *arXiv preprint arXiv:2609.21018*, 2026.

A THEORETICAL ANALYSIS

This appendix analyzes the response field and merging rule in Sections 3 and 4 using the piecewise-linear MaxSim score and weighted least squares. Unless stated otherwise, the analysis assumes fixed document embeddings and probes. The results characterize probe scores and do not directly establish guarantees for nDCG or unseen queries.

A.1 RESPONSE FIELDS, SCORES, AND GRADIENTS

Let $X = (x_1, \dots, x_n)$ with $x_i \in \mathbb{R}^h$. Flatten the probe queries $\mathcal{P}$ into a list $(q_1, \dots, q_L)$, where $L = \sum_{r=1}^{M} m_r$. Repeated query vectors are kept as separate terms. Define

$$F(X) = S_{\mathcal{P}}(X) = \sum_{\ell=1}^{L} \max_i q_\ell^\top x_i, \qquad a_\ell = \min \operatorname*{argmax}_i q_\ell^\top x_i. \tag{12}$$

The definition of a_ℓ resolves ties by original token order. Let $\mathcal{I}_i = \{\ell : a_\ell = i\}$, $R_i = \sum_{\ell \in \mathcal{I}_i} q_\ell$, and $r_i = \|R_i\|_2 = \mathrm{TRU}_i$. This is the response field in Eq.2, written with a single query-vector index.

Score decomposition. For any finite vectors, including tied winners,

$$F(X) = \sum_{\ell=1}^{L} q_\ell^\top x_{a_\ell} = \sum_{i=1}^{n} \sum_{\ell \in \mathcal{I}_i} q_\ell^\top x_i = \sum_{i=1}^{n} R_i^\top x_i. \tag{13}$$

The first equality selects a maximizing term, the second regroups the finite sum, and the last uses the definition of R_i. Ties do not affect the equality because every maximizing term has the same value. The individual contributions $R_i^\top x_i$ may be negative.

Gradient at unique winners. If every nonzero probe vector has a unique winner, there is a neighborhood in which all winners remain unchanged because the competing linear functions are continuous and finite in number. In this neighborhood, Eq.13 is linear with fixed R_i, and hence $\nabla_{x_i} F(X) = R_i$. Zero probe vectors contribute zero everywhere. The derivative treats the probes as fixed and excludes differentiation through probe generation or query encoding.

Ties and directional derivatives. Let $\mathcal{T}_\ell(X) = \operatorname{argmax}_i q_\ell^\top x_i$ contain all maximizing indices. For a block perturbation $E = (e_1, \dots, e_n)$, the right directional derivative is

$$F'(X; E) = \sum_{\ell=1}^{L} \max_{i \in \mathcal{T}_\ell(X)} q_\ell^\top e_i. \tag{14}$$

For a sufficiently small positive step, inactive terms remain below the active terms, and the active term with the largest slope determines the new maximum. Summing these slopes proves Eq.14. The deterministic response field also gives a valid subgradient: for every $Y = (y_i)_i$,

$$F(Y) \geq \sum_{\ell} q_\ell^\top y_{a_\ell} = F(X) + \sum_i R_i^\top (y_i - x_i). \tag{15}$$

At a tied point, the inner product of this selected subgradient with E may differ from the directional derivative.

Response magnitude and coherence. Define $T_i = \sum_{\ell \in \mathcal{I}_i} \|q_\ell\|_2$ and, when $T_i > 0$, $\kappa_i = r_i / T_i$. The triangle inequality gives $0 \leq \kappa_i \leq 1$ and $r_i = T_i \kappa_i$. If $T_i = 0$, set $\kappa_i = 0$. Expanding the squared norm also gives

$$r_i^2 = \sum_{\ell \in \mathcal{I}_i} \|q_\ell\|_2^2 + 2 \sum_{\substack{\ell < t \\ \ell, t \in \mathcal{I}_i}} q_\ell^\top q_t. \tag{16}$$

Thus, the magnitude depends on both the total magnitude of assigned query vectors and their directional agreement. A zero field may result from having no assigned probes or from cancellation among them. It does not imply that the token is irrelevant to future queries.

A.2 Local Sensitivity and Winner Stability

A sufficient stability radius. Suppose $n \geq 2$ and every nonzero q_ℓ has a unique winner. Define its positive gap

$$g_\ell = q_\ell^\top x_{a_\ell} - \max_{j \neq a_\ell} q_\ell^\top x_j. \tag{17}$$

If $\max_i \|e_i\|_2 \leq \varepsilon$ and

$$\varepsilon < \min_{\ell : \|q_\ell\|_2 > 0} \frac{g_\ell}{2\|q_\ell\|_2}, \tag{18}$$

then the winners of all nonzero probes remain unchanged at $X + E$ and along the segment from X to $X + E$. Indeed, for every competitor $j \neq a_\ell$,

$$\begin{aligned} q_\ell^\top (x_{a_\ell} + e_{a_\ell} - x_j - e_j) &\geq g_\ell - \|q_\ell\|_2(\|e_{a_\ell}\|_2 + \|e_j\|_2) \\ &\geq g_\ell - 2\varepsilon\|q_\ell\|_2 > 0. \end{aligned} \tag{19}$$

Replacing E by tE, $0 \leq t \leq 1$, preserves the inequality. If all probes are zero, the score is identically zero and no gap condition is needed. With one document vector, there is no competing winner.

Exact local change. More generally, if every selected winner remains a maximizing term after perturbation, regrouping the scores before and after perturbation gives

$$F(X + E) - F(X) = \sum_i R_i^\top e_i. \tag{20}$$

Applying the inner-product inequality termwise and then Cauchy–Schwarz to the scalar sequences gives

$$|F(X + E) - F(X)| \leq \sum_i r_i \|e_i\|_2 \leq \left(\sum_i r_i^2 \right)^{1/2} \left(\sum_i \|e_i\|_2^2 \right)^{1/2}. \tag{21}$$

For one token and $\varepsilon > 0$,

$$\sup_{\|e\|_2 \leq \varepsilon} |R_i^\top e| = \varepsilon r_i, \qquad \max_{\|e\|_2 \leq \varepsilon} R_i^\top e = \varepsilon r_i. \tag{22}$$

The upper bound follows from $|R_i^\top e| \leq r_i \|e\|_2$. For $r_i > 0$, the positive maximum is attained uniquely at $e = \varepsilon R_i / r_i$. For $r_i = 0$, every perturbation has zero linear response. This gives the magnitude and direction interpretation in Eq.4 when the winners are preserved.

Unit-vector constraints. The preceding gradient is in the ambient vector space. For $x = z/\|z\|_2$ with $z \neq 0$, differentiating normalization gives

$$D(z/\|z\|_2)[e] = \frac{1}{\|z\|_2}(I - xx^\top)e. \tag{23}$$

Consequently, at unique winners the gradient with respect to z is $(I - xx^\top)R/\|z\|_2$. On the unit sphere, the tangent gradient is $(I - xx^\top)R$, whose squared norm is $\|R\|_2^2 - (x^\top R)^2$. RARC computes the response field in the ambient space and applies normalization when forming the output representatives.

When winners change. For arbitrary lists X and Y of the same size, the following bound holds even when the winners change:

$$|F(Y) - F(X)| \leq \sum_\ell \|q_\ell\|_2 \max_i \|y_i - x_i\|_2. \tag{24}$$

For real lists $(a_i)_i, (b_i)_i$, $\max_i a_i \leq \max_i b_i + \max_i |a_i - b_i|$. Swapping the lists gives a bound on the absolute difference of their maxima. Applying this bound to $a_i = q_\ell^\top y_i$ and $b_i = q_\ell^\top x_i$, using Cauchy–Schwarz, and summing over ℓ gives the result. This bound is less precise than the local bound for individual tokens, but it also applies when compression exceeds the radius in Eq.18.

A.3 Semantic Proximity with Opposite Retrieval Roles

For every $\eta > 0$, there exist two unit document vectors with cosine similarity greater than $1 - \eta$ whose nonzero response directions have inner product -1.

Construction and proof. In $\mathbb{R}^2$, choose $0 < \theta < \pi/2$ and set

$$x_1 = (\cos\theta, \sin\theta), \quad x_2 = (\cos\theta, -\sin\theta), \qquad q_1 = (0, 1), \quad q_2 = (0, -1). \tag{25}$$

All four vectors have unit norm. Since $q_1^\top x_1 = \sin\theta > -\sin\theta = q_1^\top x_2$, the first probe has the unique winner x_1. Similarly, $q_2^\top x_2 = \sin\theta > -\sin\theta = q_2^\top x_1$, so the second has the unique winner x_2. Therefore $R_1 = q_1$, $R_2 = q_2$, and $v_1^\top v_2 = -1$. Meanwhile,

$$x_1^\top x_2 = \cos^2\theta - \sin^2\theta = \cos(2\theta) \longrightarrow 1 \quad \text{as } \theta \longrightarrow 0. \tag{26}$$

Choosing a sufficiently small positive θ establishes the claim. The example embeds in any higher-dimensional space by appending zeros.

This construction shows that semantic proximity does not ensure similar response directions. This does not imply a nonzero limiting score loss as θ tends to zero. The winner gaps are $2\sin\theta$ and decrease with θ, in agreement with Eq.24. Replacing both document vectors by their normalized mean $(1, 0)$ changes the total probe score from $2\sin\theta$ to zero. Thus, opposite response directions can occur even when the absolute score change is small.

A.4 Weighted Representatives and the Vector Budget

Fix a nonempty residual cluster C, positive weights $w_i = u_i^\alpha + \epsilon_w$, and define $W_C = \sum_{i \in C} w_i > 0$ and $s_C = \sum_{i \in C} w_i x_i$. The weighted squared representation error is

$$J_C(c) = \sum_{i \in C} w_i \|x_i - c\|_2^2 = \sum_{i \in C} w_i \|x_i\|_2^2 + W_C \|c\|_2^2 - 2c^\top s_C. \tag{27}$$

Without a norm constraint, the derivative is $2W_C c - 2s_C$ and the Hessian is $2W_C I \succ 0$. Hence $a_C = s_C / W_C$ is the unique minimizer. Under $\|c\|_2 = 1$, the first two terms in Eq.27 are constant. Minimizing J_C is therefore equivalent to maximizing $c^\top s_C \le \|s_C\|_2$. If $s_C \neq 0$, equality holds uniquely at

$$c_C = s_C / \|s_C\|_2, \tag{28}$$

which proves the claim following Eq.9. If $s_C = 0$, every unit vector attains the same objective, so the optimal direction is not unique. This result assumes fixed clusters and weights. It does not establish optimality of the clustering, utility transform, protection rule, or retrieval ranking.

Relation to local compression error. Suppose each original token is assigned to an output representative z_i, and the list $(z_i)_{i=1}^n$ covers all output rows. Repeated representatives do not change MaxSim, so this expanded list has the same score as the compressed index. Set $e_i = z_i - x_i$. If the original selected winners remain maximizing terms, Eq.20 and weighted Cauchy–Schwarz give

$$\begin{aligned} |F((z_i)_i) - F(X)| &\le \sum_i r_i \|e_i\|_2 \\ &\le \left(\sum_i \frac{r_i^2}{w_i} \right)^{1/2} \left(\sum_i w_i \|e_i\|_2^2 \right)^{1/2}. \end{aligned} \tag{29}$$

The last step writes each summand as $(r_i/\sqrt{w_i})(\sqrt{w_i}\|e_i\|_2)$. Holding the clusters, response fields, and weights fixed, Eq.9 minimizes the residual contribution to the second factor among unit representatives. This bound assumes the stated correspondence between tokens and representatives, with unchanged winners. It does not provide an unconditional guarantee for scores or nDCG.

Exact budget and protection. For $n \ge 1$ and $0 < \beta \le 1$, $K = \max(1, \lceil \beta n \rceil)$ satisfies $1 \le K \le n$. Limiting the number of candidates ensures $p \le K$. If $p < K$, a valid backend adapter returns nonempty membership groups $\mathcal{C}_{\text{res}}$ and any reserved zero rows Z_0 satisfying $|\mathcal{C}_{\text{res}}| + |Z_0| = K - p$. The output then contains exactly $p + |\mathcal{C}_{\text{res}}| + |Z_0| = K$ rows. Without reserved zero rows, the output contains $K - p$ residual clusters. If $p = K$, only the protected rows are output. Equal rows still

count as separate stored vectors. When $p = K < n$, unprotected tokens have no actual assigned residual representative, so the expanded-list argument above does not apply to them.

For a protected nonzero x_i, exact normalization preserves its direction. Writing $x_i = \|x_i\|_2 c_i$ gives $\|c_i - x_i\|_2 = |1 - \|x_i\|_2|$. Thus protection preserves the vector exactly when its input norm is one, and preserves its direction otherwise. The following subsection describes the effects of finite precision.

A.5 Numerical Conventions and Backend Details

Valid responses and neighborhoods. Semantic neighbors are selected from the other document vectors and then filtered for valid responses. Response and neighborhood-mean norms must exceed $\epsilon_R = 10^{-8}$, and at least three selected neighbors must have valid responses. A token with zero response has no unit response direction and fails this validity test. It still participates in residual grouping with a positive weight. The diagnostic μ_k in Eq.6 constructs both neighborhoods within the same active-token set. This diagnostic is separate from the protection criterion.

Normalization and zero vectors. The numerical normalization operation is

$$\mathrm{norm}_{\epsilon_x}(z) = \frac{z}{\max(\|z\|_2, \epsilon_x)}, \qquad \epsilon_x = 10^{-8}. \tag{30}$$

It returns a unit vector when $\|z\|_2 \geq \epsilon_x$, a vector with norm below one when $0 < \|z\|_2 < \epsilon_x$, and zero when $z = 0$. The implementation computes the weighted mean $a_C = s_C / W_C$ before normalization. Near the tolerance, normalizing s_C instead need not give the same numerical output. A zero weighted sum therefore returns a zero vector instead of selecting an arbitrary unit minimizer of Eq.27. Protected rows use the same normalization convention.

Response vectors and utility scalars can be cached separately. With finite precision, the cached utility may differ from the norm of the cached response. The response norm is used to check direction validity, and the nonnegative cached scalar is used to compute utility. The mathematical identities above describe the exact response sum. Representative accumulation uses float32 and the resulting index is stored in float16. For storage error d_i in a representative, $|q^\top d_i| \leq \|q\|_2 \|d_i\|_2$, so exact unit-norm and optimality statements apply before storage rounding.

Residual grouping. H-Pool applies Euclidean Ward linkage directly to the original document embeddings of the unprotected tokens, forming $K - p$ nonempty clusters. The K-Means backend clusters normalized residual vectors with k-means++ initialization, ten initializations, at most 300 iterations, tolerance 10^{-4}, and random seed zero. Both backends require the requested number of nonempty groups before forming representatives. The vMFMM backend fits a von Mises–Fisher mixture using soft EM on normalized nonzero vectors and converts responsibilities to hard assignments, with explicit handling of empty groups and a reserved slot for zero padding when present. For all three backends, RARC groups document embeddings and then merges the original vectors within each group using the same utility weights. Response fields provide the weights and protection decisions.

Budget accounting for degenerate rows. Explicit zero-padding rows in a backend adapter are stored vectors and count toward K. Removing a zero row can change the score because it sets the MaxSim value to zero when all other inner products are negative. If one zero row is reserved, the number of residual-cluster representatives is $K-p-1$, not $K-p$. The budget is checked against the entire stored output, including protected rows, residual representatives, and explicit zero rows. The backend adapter must repair or reject empty groups before counting the output clusters. The unit-sphere optimum above does not apply to a zero-padding row. Experimental settings and diagnostics are in Appendix B.1.

B Additional Experimental Details

B.1 Evaluation and Configuration

Hardware. Experiments were conducted on a system with four NVIDIA L40 GPUs.

Benchmarks and metrics. We evaluate on ViDoRe V2 and V3 (Macé et al., 2025; Loison et al., 2026), using their four and eight domain splits, respectively. The evaluation pools contain 4,544 pages and 1,152 queries for V2, and 19,252 pages and 14,514 queries for V3. Each query is evaluated against its split's document pool using the benchmark relevance judgments. We report nDCG@5 multiplied by 100 and average equally across splits within each benchmark. The final average in Table 1 gives equal weight to all twelve splits.

RARC configuration. We use up to ten synthetic queries generated from each page, semantic neighborhoods of size $k = 20$, and utility exponent $\alpha = 1.5$, with $\epsilon_w = 10^{-6}$. The utility and directional thresholds are approximately $(\tau, \gamma_{\text{dir}}) = (1.93, 0.10)$ for ColQwen2 and $(1.82, 0.03)$ for ColPali. We calibrate the thresholds on the ViDoRe V1 `tatdqa_text` subset. We set τ to the 90th percentile of nonzero u_i and choose γ_{dir} so that the number of protected tokens per page averages 10% of the budget. We use the resulting settings for each retriever on V2 and V3 with all merging backends. All probing and representation correction take place during offline indexing.

H-Pool applies Euclidean Ward linkage directly to document embeddings, using only unprotected tokens for RARC. vMFMM uses hard assignments from a von Mises–Fisher mixture. Query encoding and online MaxSim scoring follow the original retriever.

vMFMM configuration. We fit mixtures to normalized nonzero document vectors using soft EM in float64. We initialize component directions with k-means++ using seeds 0, 1, and 42. Mixture weights start uniformly, and concentrations start at $\kappa = 100$ with an upper bound of 10^5. Each fit stops when the absolute change in mean log likelihood is below 10^{-3}, with at most 1,000 iterations. We retain the valid fit with the highest mean log likelihood. These runs serve as internal restarts for mixture fitting rather than independent repetitions of the experiment. Hard assignments use maximum posterior probability. To repair an empty component, we move a token from a component with more than one token, choosing the move with the smallest log-posterior loss. Ties are resolved by component order and then original token order. At 10% retention, repairs may affect at most 1% of fitting tokens. Explicit zero padding is excluded from mixture fitting and represented by one reserved output slot when present. RARC aggregates the resulting groups with utility weights, while the vanilla counterpart uses uniform weights.

For the 20% and 30% budget runs, the empty-component limit is $\max(\lfloor 0.01m \rfloor, \lceil 0.30J \rceil)$ for m fitting tokens and J requested components. If fitting fails because of a dead component, we retry deterministically with initial concentrations 80, 60, then 50. Successful fits retain the original initialization. Other settings remain unchanged.

Threshold sensitivity after transfer. We vary both thresholds around their model-specific calibrated centers (τ_0, γ_0), where γ denotes γ_{dir}. Each multiplier in $(\tau/\tau_0, \gamma/\gamma_0)$ ranges over $\{0.6, 0.8, 1.0, 1.2, 1.4\}$, giving 25 configurations per model. We evaluate H-Pool at 10% retention on the four V2 and eight V3 domain pools, fixing $k = 20$ and $\alpha = 1.5$. Cached embeddings, probe-response statistics, and scoring are unchanged. Retrieval metrics are multiplied by 100 and averaged equally across the domain pools within each benchmark. Both scans use the original calibrated centers and assess sensitivity without selecting new thresholds.

V2 retrieval performance. Figures 6 and 7 report the complete grids. ColQwen2 obtains nDCG@5 of 52.5913–54.0809, compared with 53.7648 at the center, corresponding to changes of -1.1736 to $+0.3160$ points. ColPali obtains 53.2864–54.3904, compared with 54.1678 at the center, with changes of -0.8814 to $+0.2226$ points. All 25 configurations for each model outperform vanilla H-Pool in the four-pool average. The highest scores occur at multipliers $(1.0, 0.8)$ for ColQwen2 and $(0.8, 1.2)$ for ColPali. We retain the calibrated settings in all other experiments. The gains over H-Pool hold across the tested range, but larger utility thresholds tend to reduce retrieval performance.

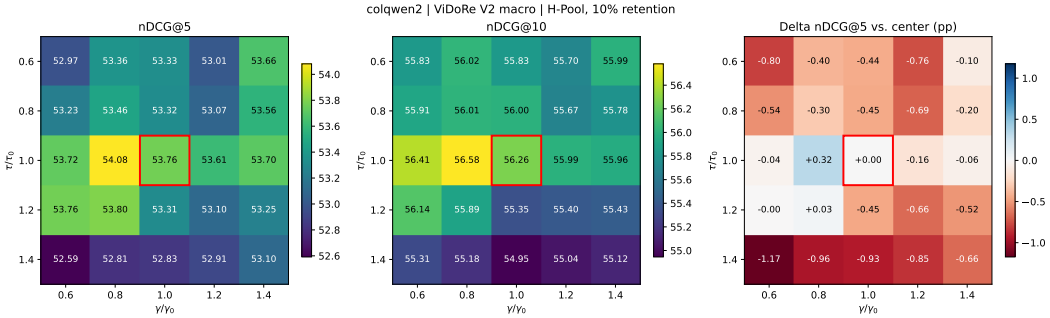


Figure 6: Threshold sensitivity for ColQwen2 on ViDoRe V2 with H-Pool at 10% retention. Panels show nDCG@5, nDCG@10, and the nDCG@5 change from the calibrated center (points). Scores are averaged equally across four pools and multiplied by 100. Red boxes mark $(\tau_0, \gamma_0) = (1.930600, 0.100000)$, where γ denotes γ_{dir}.

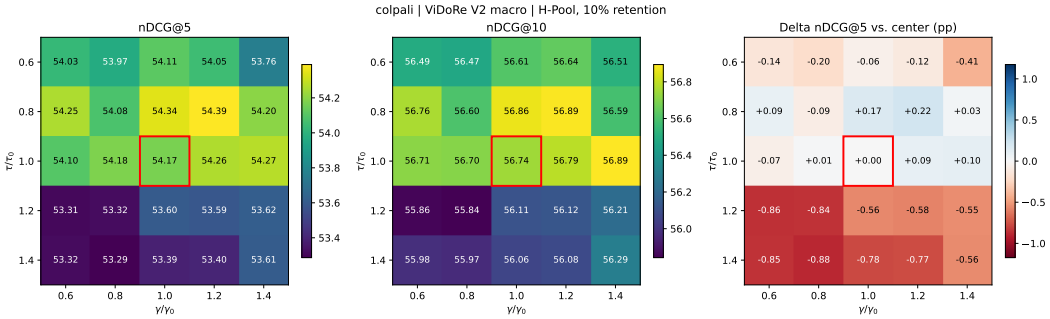


Figure 7: Threshold sensitivity for ColPali under the same protocol as Figure 6. Red boxes mark the calibrated center $(\tau_0, \gamma_0) \approx (1.820572, 0.030009)$. The models use different color scales for absolute scores and the same scale for score changes.

V3 retrieval performance and comparison. We repeat the same grid on all eight V3 subsets without retuning (Figures 8 and 9). ColQwen2 obtains nDCG@5 of 42.7667–43.1154, compared with 42.9341 at the center, corresponding to changes of -0.1674 to $+0.1813$ points. ColPali obtains 42.0765–42.3015, compared with 42.1738 at the center, with changes of -0.0973 to $+0.1277$ points. All 25 configurations per model outperform vanilla H-Pool in the eight-subset average. The best multipliers are $(0.6, 1.2)$ for ColQwen2 and $(0.6, 0.6)$ for ColPali, but the original centers are retained. V3 is less sensitive than V2 over the same grid. The maximum absolute deviation from the center decreases from 1.1736 to 0.1813 points for ColQwen2 and from 0.8814 to 0.1277 points for ColPali. The full score ranges also narrow from 1.4896/1.1040 points on V2 to 0.3487/0.2250 on V3. The transferred settings are therefore more stable on V3 within the tested range.

V3 evaluation consistency. The V3 scan uses the same full-precision threshold centers as V2. Earlier ColPali V3 results used centers rounded to six decimal places. Before reusing these scores, we verified that both precisions select the same protected tokens on every page. For this sensitivity comparison, H-Pool and RARC resolve tied scores by ascending numeric document ID. Earlier H-Pool results used a different rule for tied scores. We independently checked the rankings under the common rule with `pytrec_eval` and use these rankings for the H-Pool comparisons above.

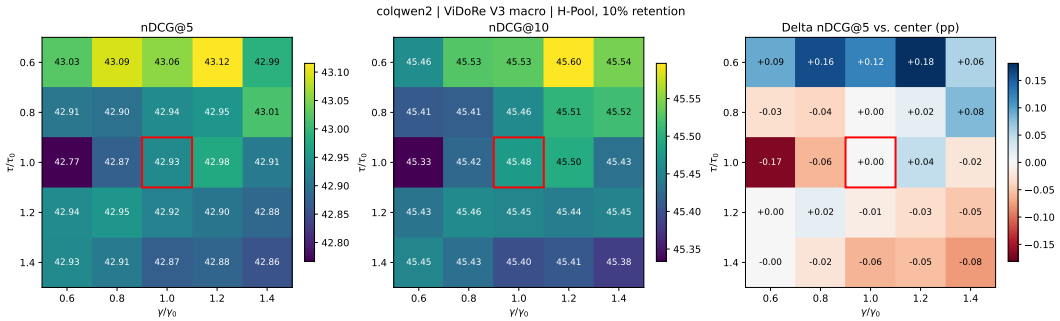


Figure 8: Threshold sensitivity for ColQwen2 on ViDoRe V3 with H-Pool at 10% retention, using the same center and grid as Figure 6. Panels show nDCG@5, nDCG@10, and the nDCG@5 change from the center (points), averaged equally across eight subsets. Red boxes mark the unchanged calibrated center.

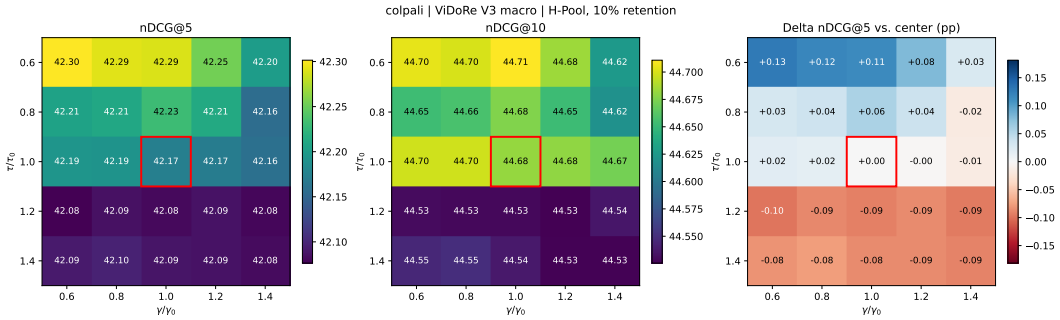


Figure 9: Threshold sensitivity for ColPali on ViDoRe V3 under the same protocol as Figure 8, using its unchanged calibrated center. The two V3 change panels share a color scale that is narrower than the V2 scale. Comparisons between benchmarks should therefore use the numerical changes.

B.2 Budget and Offline Cost

Table 5: Retrieval cost on ViDoRe V2 at 10% retention. Retrieval includes query-vector upload, MaxSim, Top-10 selection, and result transfer, but excludes query encoding and index initialization. GPU memory is the peak allocated memory at batch size 1. Speedups are measured relative to Full under the same conditions.

Backbone	Method	Index (GiB)	Retrieval (ms)	GPU (GiB)	Speedup
ColQwen2	Uncompressed	0.8105	4.075	0.793	1.00×
	H-Pool	0.0816	0.550	0.093	7.40×
	RARC(H-Pool)	0.0816	0.543	0.093	7.51×
ColPali	Uncompressed	1.1114	3.874	1.044	1.00×
	H-Pool	0.1122	0.560	0.118	6.91×
	RARC(H-Pool)	0.1122	0.580	0.118	6.67×

Retrieval measurement protocol. Table 5 uses all 1,152 queries across the four V2 domain pools, with 20 warm-up calls followed by five full measurement passes per configuration. We average latency at batch size one equally across splits, sum serialized index sizes across splits, and report the maximum GPU memory usage across splits. Retrieval timing includes uploading prepared query vectors from the CPU, MaxSim scoring, Top-10 selection, and result transfer. It excludes query encoding and index initialization. The FP16 index includes vectors, offsets, and document identifiers, while scoring uses FP32. Peak allocated GPU memory includes the packed computation index, query cache, and intermediate tensors, but no encoder model. All methods share an optimized scorer, with document packing, masks, and precision conversion prepared before timing. Full

and compressed indexes are measured under the same protocol, with GPU synchronization at each timed call boundary.

Online efficiency. RARC(H-Pool) reduces storage by 89.93%/89.91% and achieves 7.51×/6.67× retrieval speedups for ColQwen2/ColPali. H-Pool and RARC have similar online costs at the same budget, with only small timing differences. With queries and indexes already on the GPU, pure MaxSim takes 3.918/0.478 ms for Full/RARC on ColQwen2 and 3.692/0.515 ms on ColPali, yielding 8.20×/7.17× speedups. At batch size 32, RARC reaches 4,442.62/4,531.31 queries per second versus 664.13/569.26 for Full. All reported timings are averaged equally across the separate domain pools. Compared with the original scorer, the maximum absolute score error is 3.82×10^{-6}, and the Top-10 positions are identical on the checked queries.

Varying the vector budget. Figure 10 provides the complete V2 curves for H-Pool, K-Means, and vMFMM. The H-Pool and K-Means budget evaluation covers both benchmarks at retention ratios of 5%, 10%, 20%, and 30%. At 5% on V2, RARC raises H-Pool from 48.28 to 50.80 for ColQwen2 and from 50.25 to 53.19 for ColPali. The gains generally decrease as more vectors are retained. On V3, both retrievers with both backends show larger gains at 5% than at 30%. Some curves are non-monotonic. On ColQwen2 V2, RARC(H-Pool) scores 54.69 at 20% and 53.96 at 30%, with the latter below H-Pool's 54.21. Each budget produces a new set of representatives and may change the responses and page rankings.

RARC introduces additional costs during offline indexing. We measure index construction and query encoding separately. Compared with H-Pool, constructing the RARC index from cached inputs adds 17.75 s for ColQwen2 and 24.17 s for ColPali. Encoding the 44,516 valid synthetic queries is estimated at 2,851 and 1,570 s from a 200-page timing sample. These estimates cover encoding before index construction. Query generation is a separate offline step, and its total time was not recorded. Once constructed, the index is reused across incoming queries.

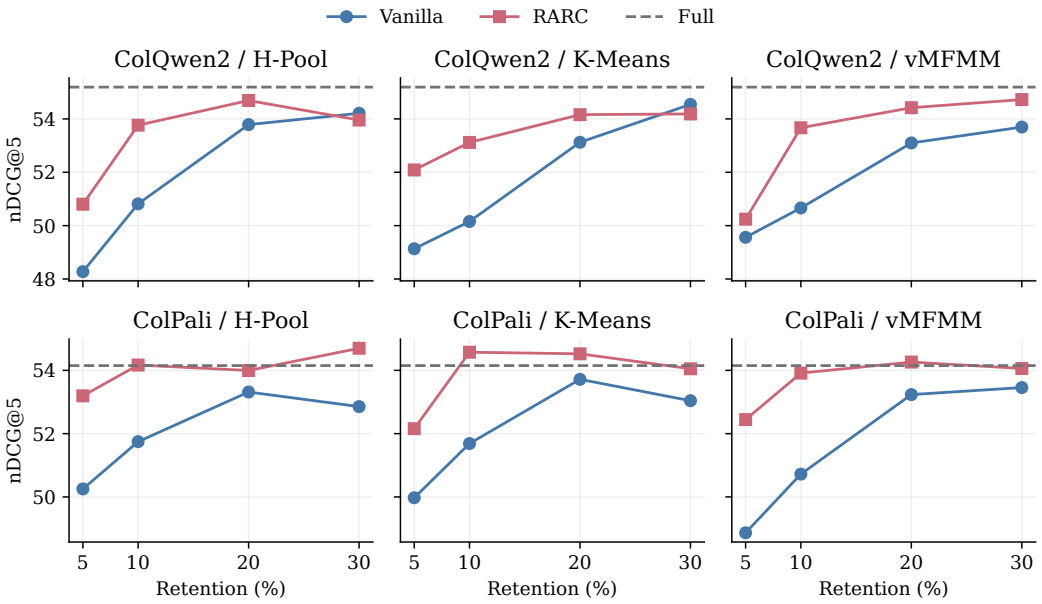


Figure 10: Complete ViDoRe V2 budget curves for both models and all three merging backends. Scores are nDCG@5 ×100, averaged equally across four domain splits. Markers show the measured 5%, 10%, 20%, and 30% retention settings, and dashed lines denote Full.

B.3 Component Attribution and Probe Controls

To quantify component overlap, let Y_{ab} denote the score with protection state a and weighting state b. We compute $\phi_P = \frac{1}{2}[(Y_{10} - Y_{00}) + (Y_{11} - Y_{01})]$ and define ϕ_W analogously. Their sum equals the total improvement. On ColPali V3, both contributions are positive, while the interaction $I = Y_{11} - Y_{10} - Y_{01} + Y_{00}$ is negative for both backends (Table 6(b)). Weighting contributes more

Table 6: Protection (P) and weighting (W) at a 10% budget. (a) nDCG@5 $\times 100$. (b) Average marginal contributions ϕ_P, ϕ_W and interaction I, in nDCG@5 points. The effect of protection includes changes in residual clustering and budget allocation. See Appendix B.3 for definitions.

(a) Components: V2, H-Pool

P	W	ColQwen2	ColPali
—	—	50.817	51.747
✓	—	53.062	53.767
—	✓	53.479	53.527
✓	✓	**53.765**	**54.168**

(b) Attribution: ColPali, V3

Backend	Gain	ϕ_P	ϕ_W	I
H-Pool	+1.2678	+0.4533	+0.8145	−0.4701
K-Means	+1.4360	+0.5931	+0.8429	−0.5411

Table 7: Component ablation on V2 with H-Pool at 10% retention: equal-split macro nDCG@10 $\times 100$. Protection and weighting match Table 6(a).

Method	ColQwen2	ColPali
H-Pool	53.547	54.827
Protection only	55.703	56.548
Weighting only	55.169	56.316
RARC	**56.258**	**56.742**

to the improvement, and protection provides a further gain. The negative interaction shows that their benefits partly overlap. Since protection changes both the singleton set and the clustering budget for the remaining tokens, its measured contribution includes both effects.

Alternative protection signals. We compare RARC with alternative selectors that protect the same number of tokens on each page, while keeping aggregation fixed. These include top-utility, shuffled-direction, inverted-direction, and random selection. RARC has the highest mean among these selectors, but utility-based controls are close. Replacing protected tokens with tokens of similar utility on the same page yields 53.68–53.86 for ColQwen2 and 53.79–53.87 for ColPali, versus RARC's 53.76 and 54.17. The average fraction of protected tokens replaced per page is only 28.7% and 9.5%, respectively. Some replacements slightly exceed RARC on ColQwen2, whereas all reported replacements remain below it on ColPali. These results support the joint selection rule, while showing that the additional benefit of direction varies between retrievers.

Probe coverage and contamination. We vary the number of synthetic queries and replace a fraction with queries from other pages, averaging over five seeds. On V2 with H-Pool, increasing the probe count from one to seven improves nDCG@5 from 50.98 to 53.09 for ColQwen2 and from 51.74 to 53.20 for ColPali. The ten-query scores, 53.78 and 54.13, use the reference configuration of the probe experiment and differ from the main-table values. Replacing 10% of probes gives 53.50 and 53.51, while replacing 70% reduces these to 50.47 and 51.41. These results show that covering more of the page content improves the response signal, while queries from other pages reduce its usefulness. The experiment tests probe coverage and page relevance, leaving other types of generation errors untested.

B.4 Functional Preservation and Ranking Diagnostics

Table 8 metrics. Token-level Spearman correlations are aggregated across splits in Fisher space. Fidelity recovery compares the highest and lowest page-mismatch quartiles. The last three columns use the preselected intersection of high-mismatch and high-direction-spread pages. Prototype loss measures token-to-representative cosine distance. Functional loss measures the cosine distance between original and re-probed response directions on each method's valid-response tokens. The set of valid tokens may differ between methods, so functional loss is measured on each method's own valid set. Dispersion measures directional spread within residual clusters, excluding protected singletons.

Response preservation. Reapplying the probes to the compressed representations gives response-direction losses lower by 0.07452 and 0.06187 on the high-risk pages. Prototype loss decreases as well. Meanwhile, residual-cluster directional dispersion increases by 0.01452 and 0.00699. Re-

Table 8: Misalignment and functional preservation on V3 with H-Pool. V/R denote vanilla/RARC, and positive V−R indicates a reduction. ρ measures mismatch–loss correlation. High−low compares fidelity recovery between the highest and lowest mismatch quartiles. The last three columns describe high-risk pages. Metric definitions are in Appendix B.4.

Backbone	Mismatch–loss ρ	Fidelity recovery: high−low	Prototype loss V−R	Functional loss V−R	Dispersion V−R
ColQwen2	0.3265	+0.02186	+0.10391	+0.07452	−0.01452
ColPali	0.3140	+0.01513	+0.06418	+0.06187	−0.00699

Table 9: (a) Mean MaxSim changes on relevant–competing pairs selected by Full, with equal split weights. Positive E, D favor the relevant page. (b) RARC minus vanilla. Negative rank changes indicate improvement, and positive changes in absolute rank distance indicate greater deviation from Full. Qwen2 denotes ColQwen2. Same-page ranks are averaged over relevant pages, queries, and then splits. Score ties are broken by numeric corpus ID.

(a) Responses: ColPali, H-Pool

Data	Ref.	E	D	ΔM
V2	Full	+0.00597	+0.06444	+0.07041
V2	Vanilla	+0.12457	+0.02069	+0.14526
V3	Full	+0.00337	+0.12129	+0.12466
V3	Vanilla	+0.07654	+0.03235	+0.10888

(b) Relevant-page ranks: H-Pool

Data	Model	First rank	Same-page rank	Distance to Full
V2	Qwen2	−1.331	−1.800	−1.903
V2	ColPali	+0.002	−2.457	+1.329
V3	Qwen2	−1.514	−4.325	+4.213
V3	ColPali	−1.995	−5.869	+3.400

sponse loss therefore decreases even as the remaining clusters show greater directional variation. This agrees with the design of preserving selected tokens individually and weighting the remaining groups. Changes in scores and rankings do not always follow changes in representation fidelity. ColQwen2's high-minus-low mismatch contrast in score-error recovery is −0.09324 despite its positive fidelity-recovery contrast. The difference in margin recovery between high- and low-mismatch pages is not statistically significant for either retriever under the reported bootstrap protocol (Table 21).

Semantic grouping. The probe responses on V3 are sparse. 87.81% and 89.09% of tokens have zero response fields, while the top 10% carry 93.98% and 95.73% of response mass for ColQwen2 and ColPali. A zero field may result from missing probe coverage or vector cancellation. Such tokens may still be relevant to future queries. Directly clustering response fields and then averaging the original document vectors reaches 50.85 and 52.40 on V2 with the hierarchical backend, compared with 53.76 and 54.17 for RARC. The results support semantic grouping to maintain broad coverage, with the response field guiding token protection and aggregation weights.

Response-space geometry. Table 10 complements the coverage and response-mass statistics above. Pair distance is the within-page median pairwise cosine distance, and 20-NN radius is the median cosine distance to the twentieth neighbor. Effective rank is computed from the variance spectrum of centered vectors. Response directions have larger pair distances and local radii, and their effective rank does not decrease. The response field thus has sparse coverage without a lower effective dimension.

Normalization and changes in MaxSim winners. MaxSim evaluates the support function $h_X(q_s) = \max_i q_s^\top x_i$ for each query token. A raw convex average stays within the original support envelope, whereas normalization can move a representative outside the original convex hull. The compressed MaxSim winner can also differ from the representative of the original winner. Across the pages selected by Full, vanilla, and RARC in the V3 H-Pool winner analysis, aggregation reduces the relevant-page response by 3.3913 on average. Normalization restores 1.1616, and changes in the winning representatives restore a further 0.8711. Both effects partly compensate for the loss from aggregation. Their effect on the margin also depends on the competing page. With protection

Table 10: Geometry on V3, averaged equally across eight splits. S and R use the same nonzero-response tokens on each page.

Model	Metric	S	R	$R - S$
ColQwen2	Pair distance	0.8707	0.9361	+0.0654
	20-NN radius	0.7391	0.8226	+0.0835
	Effective rank	33.04	33.18	+0.14
ColPali	Pair distance	0.8018	0.8749	+0.0731
	20-NN radius	0.5906	0.6964	+0.1059
	Effective rank	28.17	29.07	+0.90

and weighting fixed, enabling normalization changes the margin of Full-selected pairs by -0.08577 in the V3 H-Pool analysis.

Ranking changes. On V3, RARC(H-Pool) improves the average rank of the same judged relevant pages by 4.325 positions for ColQwen2 and 5.869 for ColPali, while their average absolute distance from Full increases by 4.213 and 3.400 positions (Table 9(b)). K-Means shows the same pattern. Relevant-page ranks improve by 5.185 and 7.229 positions while the distances grow by 3.487 and 1.986, and the first-relevant rank changes by -1.898 and -3.147, respectively. These benchmark averages show that compression can improve retrieval while changing the uncompressed ranking. Results on V2 are less consistent. ColPali H-Pool improves the same-page average while leaving the first-relevant rank nearly unchanged.

The page-pair analysis shows how response suppression relates to ranking quality. The competitor selected under Full is the highest-scoring unjudged page, whose relevance is unknown. Selecting pairs with RARC instead of Full changes the mean net suppression relative to Full to -0.11200 on V2 and -0.11916 on V3 for H-Pool. Moreover, approximately 70% of queries with positive net suppression show no nDCG@5 increase. Functional denoising describes the response changes for the selected page pairs. Its contribution to retrieval performance depends on rankings across the full document pool and the positions emphasized by the metric.

B.5 Parameter Sensitivity

We vary $k \in \{10, 15, 20, 25, 30\}$ with $\alpha = 1.5$, and $\alpha \in \{1, 1.5, 2, 2.5\}$ with $k = 20$. Across both models and benchmarks, the largest absolute macro nDCG@5 deviation from the default is below 0.47 points. Increasing k increases protection without yielding monotonic retrieval gains. On V3, the protected fractions rise from 0.7200% to 1.2918% for ColQwen2 and from 0.5925% to 0.9583% for ColPali as k increases from 10 to 30. Increasing α beyond 1.5 reduces the macro score in all four model–benchmark combinations. The $\alpha = 1$ condition still uses utility weights, whereas protection-only aggregation uses uniform weights.

Table 11: Complete shared parameter sweep with H-Pool at a 10% budget.

k	α	Qwen2 V2	Qwen2 V3	ColPali V2	ColPali V3
10	1.5	53.9189	42.9034	53.7772	42.1560
15	1.5	54.0209	42.9364	54.1311	42.1857
20	1.5	53.7648	42.9341	54.1678	42.1738
25	1.5	53.5519	42.8290	54.2689	42.1891
30	1.5	53.4561	42.7968	54.3129	42.1821
20	1.0	53.9922	42.8996	54.1515	42.1984
20	2.0	53.3608	42.8956	54.0801	42.1564
20	2.5	53.2999	42.8628	53.9272	42.1049

B.6 Selection and Probe Diagnostics

Increasing probes from one to seven raises utility-rank agreement with the ten-probe reference from 0.4784 to 0.9237 for ColQwen2 and from 0.5360 to 0.9200 for ColPali. Protected-set Jaccard rises

Table 12: Count-matched protection strategies on V2 with H-Pool. All rows match per-page protection counts and the aggregation rule.

Selection strategy	ColQwen2	ColPali
RARC	53.765	54.168
Top-TRU	53.628	54.078
Scalar support	53.440	53.975
Inverted direction	53.140	54.041
Shuffled direction	53.600	54.091
Random	53.117	53.178
Direction only	53.400	53.364

from 0.0252/0.0270 to 0.5642/0.5896. Direction cosine agreement on valid tokens is already high with one probe (0.9520/0.9662), but average protected counts are only 0.152/0.177 per page, compared with 3.536/4.236 at seven probes. High directional agreement among valid tokens can therefore coexist with limited probe coverage.

With cross-page replacement increasing from 10% to 70%, ColQwen2's average protected count increases from 6.017 to 6.388, yet protected-set Jaccard falls from 0.7553 to 0.2115 and nDCG@5 falls from 53.50 to 50.47. ColPali's corresponding Jaccard falls from 0.7752 to 0.1544 and nDCG@5 from 53.51 to 51.41. Probe quality changes which roles are preserved, even when the number of query tokens is approximately constant.

Detailed probe statistics. Tables 13–16 report all four perturbation levels, using equal-split averages for each seed followed by the mean and sample standard deviation across seeds 0–4. We measure agreement with the clean ten-probe reference on the same page. Utility Spearman compares u_i rankings, top-decile overlap compares the highest-utility tokens, and protected Jaccard compares the selected sets. Direction cosine uses tokens with valid responses in both conditions. Protection rate uses the original token count as denominator. A direction is invalid if the response is zero, fewer than three neighbors have valid responses, or the mean direction is zero. Cap-hit rate is the fraction of pages whose candidate count exceeds the budget. An SD rounded to zero can still reflect small variations.

Table 13: ColQwen2 probe count diagnostics on V2 with H-Pool at 10% retention. Values are mean $\pm$ sample SD over five seeds, with rates in percent.

Metric	1	3	5	7
Utility Spearman	0.4784 ± 0.0023	0.7562 ± 0.0008	0.8547 ± 0.0010	0.9237 ± 0.0007
Top-decile overlap	0.1104 ± 0.0006	0.2782 ± 0.0021	0.4587 ± 0.0016	0.6513 ± 0.0017
Protected Jaccard	0.0252 ± 0.0014	0.1248 ± 0.0018	0.2734 ± 0.0027	0.5642 ± 0.0012
Direction cosine	0.9520 ± 0.0005	0.9674 ± 0.0002	0.9793 ± 0.0001	0.9887 ± 0.0001
Query tokens/page	35.66 ± 0.00	106.94 ± 0.00	178.12 ± 0.00	249.03 ± 0.00
Protected tokens/page	0.152 ± 0.007	0.836 ± 0.006	1.817 ± 0.014	3.536 ± 0.012
Protection rate (%)	0.020 ± 0.001	0.111 ± 0.001	0.241 ± 0.002	0.470 ± 0.002
Invalid direction (%)	99.39 ± 0.00	97.54 ± 0.01	95.43 ± 0.02	93.29 ± 0.01
Cap-hit rate (%)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

Table 14: ColQwen2 cross-page replacement diagnostics on V2 with H-Pool at 10% retention. Values are mean ± sample SD over five seeds, with rates in percent.

Metric	10%	30%	50%	70%
Utility Spearman	0.9609 ± 0.0006	0.8762 ± 0.0006	0.7695 ± 0.0010	0.6292 ± 0.0005
Top-decile overlap	0.7712 ± 0.0020	0.5313 ± 0.0021	0.3619 ± 0.0017	0.2334 ± 0.0012
Protected Jaccard	0.7553 ± 0.0023	0.4805 ± 0.0028	0.3180 ± 0.0021	0.2115 ± 0.0011
Direction cosine	0.9895 ± 0.0002	0.9596 ± 0.0003	0.9092 ± 0.0005	0.8236 ± 0.0009
Query tokens/page	351.99 ± 0.08	352.00 ± 0.16	351.90 ± 0.16	351.79 ± 0.09
Protected tokens/page	6.017 ± 0.020	6.160 ± 0.034	6.273 ± 0.017	6.388 ± 0.034
Protection rate (%)	0.799 ± 0.003	0.818 ± 0.004	0.833 ± 0.002	0.848 ± 0.005
Invalid direction (%)	90.24 ± 0.01	90.14 ± 0.02	90.14 ± 0.02	90.29 ± 0.02
Cap-hit rate (%)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

Table 15: ColPali probe count diagnostics on V2 with H-Pool at 10% retention. Values are mean ± sample SD over five seeds, with rates in percent.

Metric	1	3	5	7
Utility Spearman	0.5360 ± 0.0019	0.7778 ± 0.0017	0.8561 ± 0.0005	0.9200 ± 0.0005
Top-decile overlap	0.1071 ± 0.0005	0.2968 ± 0.0009	0.4870 ± 0.0013	0.6602 ± 0.0019
Protected Jaccard	0.0270 ± 0.0006	0.1499 ± 0.0027	0.3378 ± 0.0023	0.5896 ± 0.0029
Direction cosine	0.9662 ± 0.0003	0.9743 ± 0.0001	0.9825 ± 0.0001	0.9901 ± 0.0001
Query tokens/page	28.95 ± 0.11	86.88 ± 0.12	144.67 ± 0.14	202.25 ± 0.15
Protected tokens/page	0.177 ± 0.005	1.076 ± 0.022	2.440 ± 0.020	4.236 ± 0.019
Protection rate (%)	0.017 ± 0.001	0.104 ± 0.002	0.237 ± 0.002	0.411 ± 0.002
Invalid direction (%)	99.49 ± 0.01	98.03 ± 0.01	96.34 ± 0.01	94.58 ± 0.01
Cap-hit rate (%)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

Table 16: ColPali cross-page replacement diagnostics on V2 with H-Pool at 10% retention. Values are mean ± sample SD over five seeds, with rates in percent.

Metric	10%	30%	50%	70%
Utility Spearman	0.9582 ± 0.0004	0.8686 ± 0.0009	0.7585 ± 0.0015	0.6147 ± 0.0022
Top-decile overlap	0.7686 ± 0.0023	0.5211 ± 0.0012	0.3449 ± 0.0017	0.2123 ± 0.0018
Protected Jaccard	0.7752 ± 0.0022	0.4814 ± 0.0022	0.2861 ± 0.0015	0.1544 ± 0.0022
Direction cosine	0.9906 ± 0.0001	0.9636 ± 0.0001	0.9174 ± 0.0001	0.8344 ± 0.0005
Query tokens/page	285.81 ± 0.07	285.78 ± 0.12	285.77 ± 0.13	285.71 ± 0.11
Protected tokens/page	6.810 ± 0.017	6.365 ± 0.016	6.056 ± 0.017	5.874 ± 0.008
Protection rate (%)	0.660 ± 0.002	0.617 ± 0.002	0.587 ± 0.002	0.570 ± 0.001
Invalid direction (%)	92.07 ± 0.01	92.17 ± 0.02	92.35 ± 0.02	92.67 ± 0.02
Cap-hit rate (%)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

B.7 Protected Tokens and Extended Mechanism Results

The tables below report the average number of protected tokens per page and their fraction of the compressed budget. Their fraction of the original representation is smaller because the overall retention rate is 10%.

Neighborhood mismatch and compression loss. Semantic and response-direction neighborhoods are defined on the same eligible nonzero-response tokens. In the V3 H-Pool analysis, within-cluster retention is 0.1220/0.1443 for semantic edges and 0.1052/0.1257 for response-direction edges (ColQwen2/ColPali). Semantic grouping thus retains fewer response-direction edges. The differences in absolute score error between high- and low-mismatch pages are 0.2044/0.2019. Table 8 also reports the correlations between mismatch and fidelity loss.

Figure 5 aggregation. For each page, the correction panel uses the median token-level fidelity recovery. We then average across pages within each mismatch quartile and split, and give equal

Table 17: Protected-token diagnostics on ViDoRe V2.

Model	Split	Protected/page	p/K	Zero-protection pages
ColQwen2	Economics	6.403	0.08404	0
ColQwen2	ESG	6.324	0.08429	2
ColQwen2	Biomedical	4.644	0.06037	8
ColQwen2	ESG-Human	6.370	0.08487	2
ColPali	Economics	7.920	0.07616	0
ColPali	ESG	7.177	0.06901	1
ColPali	Biomedical	5.919	0.05692	7
ColPali	ESG-Human	7.261	0.06982	0

Table 18: Probe-count results underlying Figure 4. Values are nDCG@5 $\times 100$, mean $\pm$ sample SD across seeds 0–4 after equal-split averaging. The ten-probe row uses the reference configuration of this experiment.

Probes/page	Qwen2	ColPali
1	50.980 ± 0.996	51.738 ± 0.398
3	51.457 ± 0.846	52.089 ± 0.664
5	52.571 ± 0.955	52.492 ± 0.393
7	53.090 ± 0.737	53.198 ± 0.471
10	53.776	54.133

Table 19: Probe-noise sensitivity on V2 with H-Pool at 10% retention. Values are nDCG@5 $\times 100$, reported as mean $\pm$ sample SD across seeds 0–4 after averaging equally across splits.

Cross-page replacement	Qwen2	ColPali
10%	53.496 ± 0.481	53.510 ± 0.489
30%	52.429 ± 0.561	52.923 ± 0.735
50%	51.888 ± 0.616	52.561 ± 0.788
70%	50.474 ± 0.538	51.408 ± 0.550

weight to the eight V3 splits. The low/high means are 0.046668/0.068533 for ColQwen2 and 0.026792/0.041919 for ColPali. Quartiles are fixed from the original mismatch diagnostic. The left panel shows the original non-overlap between token neighborhoods, and the right panel shows the recovery of token-to-representative fidelity.

High-risk page analysis. Before examining the improvements, we select pages that fall in both the highest mismatch quartile and the highest response-direction-spread quartile within each split. Functional error compares response directions before and after compression by reapplying the probes to each method's valid nonzero-response tokens. The pages are fixed, but the valid tokens can differ between methods. Residual dispersion excludes protected singletons. These measures assess response preservation on each method's valid tokens and directional variation in the remaining groups.

Statistical protocol. Tables 20–22 use 20,000 split-stratified bootstrap replicates, percentile 95% CIs, and equal split weights. We resample pages for fidelity and process measures, and queries for retrieval measures. Two-sided bootstrap p values are Holm-adjusted per model and analysis (four association/correction endpoints and three preservation endpoints). The Positive column counts splits with effects above zero, regardless of their individual significance. These tests apply only to the reported diagnostics and do not test the main-table nDCG differences.

Metric definitions. Association fidelity is the Spearman correlation between mismatch and loss, aggregated in Fisher space. Other association effects are high-minus-low mismatch contrasts in absolute score error, Full-minus-vanilla margin, and vanilla-minus-Full first-relevant rank. The correction analysis uses the same fixed quartiles. Fidelity recovery is the reduction in token-to-

representative cosine distance from vanilla to RARC, equivalently an increase in cosine similarity. The other recovery endpoints are RARC-minus-vanilla margin and vanilla-minus-RARC absolute score error and first-relevant rank. Preservation is measured as vanilla minus RARC on the preselected high-risk pages. Negative dispersion values indicate greater directional variation within the remaining clusters.

Table 20: Mismatch–compression-loss associations on V3 with H-Pool at 10% retention. CI: 95% confidence interval; positive: positive-effect splits out of eight.

Model	Endpoint	Effect	95% CI	Positive	Holm p
ColQwen2	fidelity	0.326524	$[0.324007, 0.329065]$	8/8	0.00039998
ColQwen2	score	0.204433	$[0.174304, 0.234488]$	8/8	0.00039998
ColQwen2	margin	0.021517	$[-0.017145, 0.059530]$	6/8	0.545573
ColQwen2	rank	0.110955	$[-4.101598, 4.114570]$	4/8	0.937453
ColPali	fidelity	0.314002	$[0.311145, 0.316804]$	8/8	0.00039998
ColPali	score	0.201926	$[0.181850, 0.221969]$	8/8	0.00039998
ColPali	margin	0.039614	$[0.015336, 0.063753]$	6/8	0.00159992
ColPali	rank	3.266342	$[1.489516, 5.226448]$	5/8	0.00119994

Table 21: High-minus-low mismatch contrasts in RARC recovery on V3 with H-Pool at 10% retention. CI: 95% confidence interval; positive: positive-effect splits out of eight.

Model	Endpoint	Effect	95% CI	Positive	Holm p
ColQwen2	fidelity	0.021864	$[0.020935, 0.022808]$	8/8	0.00039998
ColQwen2	score	-0.093241	$[-0.112266, -0.074115]$	0/8	0.00039998
ColQwen2	margin	0.009533	$[-0.032201, 0.051328]$	5/8	0.652667
ColQwen2	rank	1.937920	$[-0.464199, 4.384290]$	5/8	0.226589
ColPali	fidelity	0.015127	$[0.014557, 0.015701]$	8/8	0.00039998
ColPali	score	0.098410	$[0.084788, 0.111529]$	7/8	0.00039998
ColPali	margin	0.007830	$[-0.019537, 0.035517]$	4/8	0.583671
ColPali	rank	2.273159	$[0.361140, 4.254558]$	5/8	0.0419979

Table 22: High-risk structure preservation on V3 with H-Pool at 10% retention. CI: 95% confidence interval; positive: positive-effect splits out of eight.

Model	Endpoint	Effect	95% CI	Positive	Holm p
ColQwen2	dispersion	-0.014519	$[-0.015017, -0.014040]$	0/8	0.000299985
ColQwen2	prototype	0.103913	$[0.102793, 0.105067]$	8/8	0.000299985
ColQwen2	functional	0.074522	$[0.072922, 0.076101]$	8/8	0.000299985
ColPali	dispersion	-0.006993	$[-0.007336, -0.006649]$	0/8	0.000299985
ColPali	prototype	0.064182	$[0.063546, 0.064822]$	8/8	0.000299985
ColPali	functional	0.061870	$[0.060557, 0.063200]$	8/8	0.000299985

C Additional Visualizations

C.1 Retrieval Roles and Compression Examples

We examine four ColPali examples on ViDoRe V2 at 10% retention. The examples were selected using numerical criteria before inspecting the page images. C03/C04 use K-Means and C05/C07 use H-Pool (labeled `standard_ward` in the panels). They illustrate how compression changes retrieval responses, but do not show how often these effects occur. The competing pages have no relevance judgments.

In the panels, F denotes Full, and Y_{ab} denotes protection state a and weighting state b. Thus, Y_{00} is vanilla and Y_{11} is RARC. Scores and margins are computed for the relevant–competing page pair

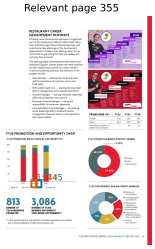
selected by Full. Results for the pair selected by RARC are reported separately. The displayed d denotes directional deviation δ_i, which is undefined when the validity conditions are not met. Retrieval metrics in the panels use the 0–1 scale.

Effects of weighting (C03/C04). In C03, the two displayed tokens have only two valid neighbors and do not meet the protection conditions, but still contribute to weighted residual representatives. With the protection partition fixed, weighting increases the pair margin by 2.2931 and raises nDCG@5 from 0 to 0.6309 relative to protection alone. In C04, normalizing the weighted representatives changes the margin by -1.1078, while nDCG@5 remains 1.0. A newly formed representative can win a MaxSim match without improving the ranking.

Response suppression and similar retrieval roles (C05/C07). In C05, suppression of the competing-page response accounts for most of the margin change on the pair selected by Full ($D = 1.8720$, $E = 0.0418$). nDCG@5 increases from 0.8539 for Full to 1.0 for RARC. In C07, the displayed patches are close in both semantic and response space (cosines 0.9063 and 0.9906), belong to the same vanilla cluster, and are not protected. Here, semantic merging groups tokens with similar retrieval roles, and RARC and vanilla both obtain nDCG@5 of 0.7366.

C03 | Beneficial weighting / synthesis | esg_human | kmeans
Who are the most promoted people at TacoBell?

Relevant page 355



Unjudged competitor 354



Patch 1: token 745



d = N/A (neighbor gate)
TRU = 5.8636
utility = 1.9262
protected = False

Patch 2: token 744



d = N/A (neighbor gate)
TRU = 1.5551
utility = 0.9381
protected = False

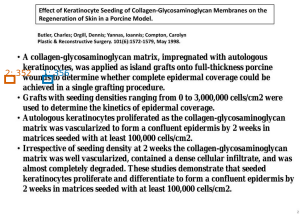
Method	Relevant S	Competitor S	Margin	Page rank	First rel.	nDCG@5	nDCG@10
F	14.29249	14.56485	-0.27236	2	2	0.63093	0.63093
Y00	11.60711	13.05305	-1.44594	10	10	0.00000	0.28906
Y10	11.62493	13.95151	-2.32658	10	10	0.00000	0.28906
Y01	14.22470	13.16868	1.05601	1	1	1.00000	1.00000
Y11	14.00892	14.04243	-0.03351	2	2	0.63093	0.63093

Full-selected pair: E=+0.14780, D=+0.09105, sum=+0.23885
RARC-selected pair: E=+0.14780, D=+0.09105, sum=+0.23885
RARC pair IDs: 355 / 354
Patch semantic cosine: 0.80577
Patch R cosine: 0.55589

A residual synthesized winner accompanies a beneficial fixed-partition weighting change. Main scores use the Full-selected pair; the competitor is unjudged. This is a mechanism example, not a population or causal estimate.

C04 | Synthesis without benefit | biomedical | kmeans
What was the outcome of using collagen-glycosaminoglycan membranes seeded with autologous keratinocytes on porcine wounds?

Relevant page 437



Effect of Keratinocyte Seeding of Collagen-Glycosaminoglycan Membranes on the Regeneration of Skin in a Porcine Model.

- A collagen-glycosaminoglycan matrix, impregnated with autologous keratinocytes, was applied as island grafts onto full-thickness porcine wounds to determine whether complete epidermal coverage could be achieved in a single grafting procedure.
- Grafts with seeding densities ranging from 0 to 3,000,000 cells/cm2 were used to determine the kinetics of epidermal coverage.
- Autologous keratinocytes proliferated as the collagen-glycosaminoglycan matrix was vascularized to form a confluent epidermis by 2 weeks in matrices seeded with at least 100,000 cells/cm2.
- Irrespective of seeding density at 2 weeks the collagen-glycosaminoglycan matrix was well vascularized, contained a dense cellular infiltrate, and was almost completely degraded. These studies demonstrate that seeded keratinocytes proliferate and differentiate to form a confluent epidermis by 2 weeks in matrices seeded with at least 100,000 cells/cm2.

Unjudged competitor 327



Patch 1: token 356



d = 0.20738
TRU = 2.9478
utility = 1.3732
protected = False

Patch 2: token 352

d = 0.14657
TRU = 1.8058
utility = 1.0317
protected = False

Method	Relevant S	Competitor S	Margin	Page rank	First rel.	nDCG@5	nDCG@10
F	26.33944	21.21886	5.12058	1	1	1.00000	1.00000
Y00	25.19978	19.34179	5.85799	1	1	1.00000	1.00000
Y10	25.23357	19.77914	5.45443	1	1	1.00000	1.00000
Y01	26.00111	19.31512	6.68599	1	1	1.00000	1.00000
Y11	26.03722	19.94503	6.09219	1	1	1.00000	1.00000

Full-selected pair: E=+0.17289, D=+0.79872, sum=+0.97161
RARC-selected pair: E=+0.14965, D=+0.28281, sum=+0.43246
RARC pair IDs: 437 / 634
Patch semantic cosine: 0.74649
Patch R cosine: 0.81134

A synthesized winner coexists with a harmful normalization margin change and no nDCG@5 gain over protection-only. Main scores use the Full-selected pair; the competitor is unjudged. This is a mechanism example, not a population or causal estimate.

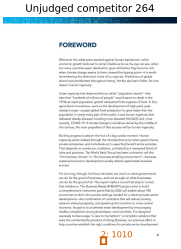
Figure 11: Real-page examples C03 and C04. Appendix C.1 explains the notation and results.

C05 | Attenuation with ranking gain | economics | standard_ward
Quels sont les risques clés pour la croissance économique mondiale identifiés dans le rapport ?

Relevant page 431



Unjudged competitor 264



Patch 1: token 77



d = 0.50858
TRU = 3.1268
utility = 1.4175
protected = False

Patch 2: token 1010

d = N/A (neighbor gate)
TRU = 0.0000
utility = 0.0000
protected = False

Method	Relevant S	Competitor S	Margin	Page rank	First rel.	nDCG@5	nDCG@10
F	15.25697	14.81345	0.44353	1	1	0.85393	0.90521
Y00	14.19952	11.96276	2.23676	1	1	0.69921	0.80481
Y10	14.19952	11.96276	2.23676	1	1	0.86879	0.84149
Y01	14.18147	11.82410	2.35737	1	1	1.00000	1.00000
Y11	14.18147	11.82410	2.35737	1	1	1.00000	1.00000

Full-selected pair: E=+0.04185, D=+1.87200, sum=+1.91384
RARC-selected pair: E=+0.04081, D=-0.40139, sum=-0.36057
RARC pair IDs: 431 / 4
Patch semantic cosine: 0.84726
Patch R cosine: N/A

Net attenuation dominates the Full-pair margin gain and accompanies improved nDCG@5. Main scores use the Full-selected pair; the competitor is unjudged. This is a mechanism example, not a population or causal estimate.

C07 | Semantic / response alignment control | esg_human | standard_ward
How did Chick fil A help people in need?

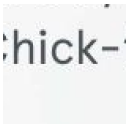
Relevant page 193



Unjudged competitor 200



Patch 1: token 211



d = 0.09362
TRU = 3.9421
utility = 1.5978
protected = False

Patch 2: token 335



d = 0.10920
TRU = 1.9730
utility = 1.0896
protected = False

Method	Relevant S	Competitor S	Margin	Page rank	First rel.	nDCG@5	nDCG@10
F	15.40944	15.27592	0.13352	1	1	0.75370	0.88382
Y00	14.00716	14.29467	-0.28752	5	1	0.73659	0.86672
Y10	14.08445	14.20410	-0.11965	5	1	0.78770	0.91783
Y01	14.11833	14.39717	-0.27884	5	1	0.78770	0.91783
Y11	14.05241	14.36387	-0.31145	5	1	0.73659	0.86672

Full-selected pair: E=-0.03939, D=-0.40558, sum=-0.44498
RARC-selected pair: E=+0.00725, D=+0.56821, sum=+0.57546
RARC pair IDs: 195 / 200
Patch semantic cosine: 0.90634
Patch R cosine: 0.99063

Semantically close patches have aligned retrieval responses and share a vanilla cluster. Main scores use the Full-selected pair; the competitor is unjudged. This is a mechanism example, not a population or causal estimate.

Figure 12: Real-page examples C05 and C07. Appendix C.1 explains the notation and results.

D ALGORITHM PSEUDOCODE

D.1 OFFLINE RARC INDEX CONSTRUCTION

Algorithm 1 summarizes RARC index construction for a single page. We apply it independently to each document page and use the compressed representations with the original query encoder and MaxSim scorer. The backend adapter B returns nonempty membership groups and any reserved zero rows within its assigned budget, as detailed in Appendix A.5. We use $\epsilon_R = \epsilon_x = 10^{-8}$ and $\epsilon_w = 10^{-6}$.

Algorithm 1: Retrieval-Aware Representation Compression

Input: Page image I; document vectors $X = (x_i)_{i=1}^n$; frozen query encoder f_q; probe generator G; semantic backend B; $\beta, k, \tau, \gamma_{\text{dir}}, \alpha$.
Output: Compressed page representation $\widetilde{X}$ with K rows.

/* Step 1: Probe the original document representation */
$K \leftarrow \max(1, \lceil \beta n \rceil)$; $\mathcal{P} \leftarrow f_q(G(I))$; $R_i \leftarrow 0$ for all i
for each query token q in $\mathcal{P}$ **do**
  $a \leftarrow \min \operatorname{argmax}_i q^\top x_i$; $R_a \leftarrow R_a + q$
end
$r_i \leftarrow \|R_i\|_2$; $u_i \leftarrow \log(1 + r_i)$ for all i
$v_i \leftarrow R_i / r_i$ for $r_i > \epsilon_R$
/* Step 2: Select high-utility, directionally distinct tokens */
$\mathcal{I}_{\text{cand}} \leftarrow \varnothing$
for $i = 1, \ldots, n$ **do**
  $\mathcal{A}_k(i) \leftarrow \min(k, n-1)$ nearest other tokens by cosine similarity
  $\mathcal{V}_i \leftarrow \{j \in \mathcal{A}_k(i) : r_j > \epsilon_R\}$
  if $r_i > \epsilon_R$ and $|\mathcal{V}_i| \geq 3$ **then**
    $b_i \leftarrow |\mathcal{V}_i|^{-1} \sum_{j \in \mathcal{V}_i} v_j$
    if $\|b_i\|_2 > \epsilon_R$ **then**
      $\delta_i \leftarrow 1 - v_i^\top b_i / \|b_i\|_2$
      Add i to $\mathcal{I}_{\text{cand}}$ if $u_i \geq \tau$ and $\delta_i \geq \gamma_{\text{dir}}$
    end
  end
end
$\mathcal{I}_{\text{prot}} \leftarrow$ at most K candidates with largest δ_i
  Break ties by original token order.
/* Step 3: Merge residual tokens under the remaining budget */
$p \leftarrow |\mathcal{I}_{\text{prot}}|$; $\widetilde{X} \leftarrow [\text{norm}_{\epsilon_x}(x_i)]_{i \in \mathcal{I}_{\text{prot}}}$
if $p < K$ **then**
  $(\mathcal{C}_{\text{res}}, Z_0) \leftarrow B(X \setminus X_{\mathcal{I}_{\text{prot}}}, K - p)$
  Require $|\mathcal{C}_{\text{res}}| + |Z_0| = K - p$.
  for each nonempty cluster $C \in \mathcal{C}_{\text{res}}$ **do**
    $w_i \leftarrow u_i^\alpha + \epsilon_w$ for $i \in C$
    $a_C \leftarrow (\sum_{i \in C} w_i x_i) / (\sum_{i \in C} w_i)$
    Append $\text{norm}_{\epsilon_x}(a_C)$ to $\widetilde{X}$
  end
  Append the reserved rows Z_0 to $\widetilde{X}$
end
return $\widetilde{X}$ with exactly K stored rows

E PAGE-CONDITIONED QUERY GENERATION

E.1 GENERATION CONFIGURATION

We use `model=qwen3-vl-8B`, `temperature=0.7`, and `thinking=False` to generate page-conditioned synthetic queries.

E.2 PROMPTS

The prompt below specifies ten retrieval intents and the structured JSON output.

System prompt

You are helping a retrieval research project. You are given ONE page of a visual document. Your task is NOT captioning. Your task is to generate retrieval-demanding user queries that could be used to retrieve this page in a document retrieval system. Output JSON only. Generate EXACTLY 10 queries. Each query must target a DIFFERENT retrieval intent, not merely a different phrasing.

Use the following fixed intent set exactly once each:

1. entity 2. numeric 3. relation 4. temporal 5. local-detail
6. semantic-summary 7. evidence 8. comparison 9. anomaly 10. table-lookup

For each query, output:

• "query": the user query text

• "intent": one intent label from the fixed set

• "evidence": a short grounded description of the exact page content that justifies this query

• "page_anchor": a concrete anchor such as title, section header, table cell, chart label, percentage, date, footnote, caption, small text, or named entity on the page

STRICT RULES:

• ALL 10 intents must appear exactly once.

• Queries must be plausible real user search queries for retrieval — not captions, not descriptions.

• Do NOT describe what the page looks like.

• Each query must be grounded in specific page content (cite the evidence).

Output a JSON object with key "queries" containing an array of 10 objects. Example output: {"queries": [{"query": "...", "intent": "entity", "evidence": "...", "page_anchor": "..."}, ...]}

E.3 EXAMPLE OUTPUT

This example uses a financial-results page from the `finance_en` subset of ViDoRe V3 (corpus ID 2225). The page contains deposit balances, borrowing maturities, credit-rating tables, and accompanying analysis. The ten generated questions cover the ten specified intents, with one question for each intent. The following panels show the page and the complete generated JSON.

Input page | ViDoRe V3, finance_en



Table of Contents

Management's Discussion and Analysis

Morgan Stanley

deposit carried at fair value, which are primarily composed of: instruments whose payments and redemption values are linked to the performance of a specific index, a basket of stocks, a specific equity security, a commodity, a credit exposure or basket of credit exposures; and instruments with various interest rate-related features, including step-ups, step-downs and zero coupons. Also included are unsecured contracts that are not classified as OTC derivatives because they fail initial net investment criteria. As part of our asset/liability management strategy, when appropriate, we use derivatives to make adjustments to the interest rate risk profile of our borrowings (see Notes 6 and 13 to the financial statements).

Deposits

$ in millions	At December 31, 2024	At December 31, 2023
Savings and demand deposits:		
Brokerage sweep deposits[1]	$ 142,550	$ 148,274
Savings and other	157,348	139,978
Total Savings and demand deposits	299,898	288,252
Time deposits[2]	76,109	63,552
Total[3]	**$ 376,007**	$ 351,804

1. Amounts represent balances swept from client brokerage accounts.
2. Our Time deposits are predominantly brokered certificates of deposit.
3. Our deposits are primarily held in U.S. offices.

Deposits are primarily sourced from our Wealth Management clients and are considered to have stable, low-cost funding characteristics relative to other sources of funding. Each category of deposits presented above has a different cost profile and clients may respond differently to changes in interest rates and other macroeconomic conditions. Total deposits in 2024 increased primarily due to increases in Savings and Time deposits, partially offset by a reduction in Brokerage sweep deposits, largely due to net outflows to alternative cash equivalent and other investment products.

Borrowings by Maturity at December 31, 2024[1]

$ in millions	Parent Company	Subsidiaries	Total
Original maturities of one year or less	$ —	$ 4,512	$ 4,512
Original maturities greater than one year			
2025	$ 7,544	$ 14,377	$ 21,921
2026	24,738	13,231	37,969
2027	20,716	13,334	34,050
2028	13,844	14,875	28,719
2029	16,318	9,841	26,159
Thereafter	98,886	36,603	135,489
Total greater than one year	$ 182,046	$ 102,261	$ 284,307
Total	**$ 182,046**	**$ 106,773**	**$ 288,819**

1. Original maturity in the table is generally based on contractual final maturity. For borrowings with put options, maturity represents the earliest put date.

Borrowings of $289 billion at December 31, 2024 increased when compared with $264 billion at December 31, 2023, primarily due to issuances net of maturities and redemptions.

We believe that accessing debt investors through multiple distribution channels helps provide consistent access to the unsecured markets. In addition, the issuance of borrowings with original maturities greater than one year allows us to reduce reliance on short-term credit-sensitive instruments. Borrowings with original maturities greater than one year are generally managed to achieve staggered maturities, thereby mitigating refinancing risk, and to maximize investor diversification through sales to global institutional and retail clients across regions, currencies and product types.

The availability and cost of financing to us can vary depending on market conditions, the volume of certain trading and lending activities, our credit ratings and the overall availability of credit. We also engage in, and may continue to engage in, repurchases of our borrowings as part of our market-making activities.

For further information on Borrowings, see Note 13 to the financial statements.

Credit Ratings

We rely on external sources to finance a significant portion of our daily operations. Our credit ratings are one of the factors in the cost and availability of financing and can have an impact on certain trading revenues, particularly in those businesses where longer-term counterparty performance is a key consideration, such as certain OTC derivative transactions. When determining credit ratings, rating agencies consider both company-specific and industry-wide factors. See also "Risk Factors—Liquidity Risk."

Parent Company and U.S. Bank Subsidiaries Issuer Ratings at February 14, 2025

	Parent Company		
	Short-Term Debt	Long-Term Debt	Rating Outlook
DBRS, Inc.	**R-1 (middle)**	**A (high)**	**Positive**
Fitch Ratings, Inc.	**F1**	**A+**	**Stable**
Moody's Investors Service, Inc.	**P-1**	**A1**	**Stable**
Rating and Investment Information, Inc.	**a-1**	**A+**	**Stable**
S&P Global Ratings	**A-2**	**A-**	**Stable**

	MSBNA		
	Short-Term Debt	Long-Term Debt	Rating Outlook
Fitch Ratings, Inc.	**F1+**	**AA-**	**Stable**
Moody's Investors Service, Inc.	**P-1**	**Aa3**	**Stable**
S&P Global Ratings	**A-1**	**A+**	**Stable**

	MSPBNA		
	Short-Term Debt	Long-Term Debt	Rating Outlook
Moody's Investors Service, Inc.	**P-1**	**Aa3**	**Stable**
S&P Global Ratings	**A-1**	**A+**	**Stable**

Incremental Collateral or Terminating Payments

In connection with certain OTC derivatives and certain other agreements where we are a liquidity provider to certain financing vehicles associated with the Institutional Securities

December 2024 Form 10-K

48

Generated JSON

{"queries": [

{
"**query":** "What is the total amount of deposits at Morgan Stanley as of December 31, 2024?",
"**intent":** "numeric",
"**evidence":** "Total deposits at December 31, 2024, are $376,007 million.",
"**page_anchor":** "Deposits table, Total row, At December 31, 2024 column"
},

{
"**query":** "Which credit rating agency assigned a 'Positive' outlook to the Parent Company's short-term debt?",
"**intent":** "entity",
"**evidence":** "DBRS, Inc. assigned a 'Positive' rating outlook to the Parent Company's short-term debt.",
"**page_anchor":** "Parent Company and U.S. Bank Subsidiaries Issuer Ratings table, DBRS, Inc. row, Rating Outlook column"
},

{
"**query":** "What is the relationship between the Parent Company's short-term debt rating and its long-term debt rating as of February 14, 2025?",
"**intent":** "relation",
"**evidence":** "The Parent Company's short-term debt rating is R-1 (middle) and its long-term debt rating is A (high).",
"**page_anchor":** "Parent Company and U.S. Bank Subsidiaries Issuer Ratings table, Parent Company section"
},

{
"**query":** "What date was the credit ratings information last updated?",
"**intent":** "temporal",
"**evidence":** "The credit ratings are dated February 14, 2025.",
"**page_anchor":** "Parent Company and U.S. Bank Subsidiaries Issuer Ratings at February 14, 2025"
},

{
"**query":** "What is the exact footnote reference for the definition of 'Brokerage sweep deposits'?",
"**intent":** "local-detail",
"**evidence":** "Footnote 1 states: 'Amounts represent balances swept from client brokerage accounts.'",
"**page_anchor":** "Footnote 1 under the Deposits table"
},

Generated JSON (continued)

{
"**query":** "What is the semantic summary of Morgan Stanley's strategy for managing interest rate risk through derivatives?",
"**intent":** "semantic-summary",
"**evidence":** "The text states: 'As part of our asset/liability management strategy, when appropriate, we use derivatives to make adjustments to the interest rate risk profile of our borrowings.'",
"**page_anchor":** "First paragraph under 'Management's Discussion and Analysis'"
},

{
"**query":** "What evidence supports the claim that Morgan Stanley's borrowings increased in 2024?",
"**intent":** "evidence",
"**evidence":** "The text states: 'Borrowings of $289 billion at December 31, 2024 increased when compared with $264 billion at December 31, 2023.'",
"**page_anchor":** "Text below the 'Borrowings by Maturity at December 31, 2024' table"
},

{
"**query":** "How does the amount of time deposits compare to savings and demand deposits in 2024?",
"**intent":** "comparison",
"**evidence":** "In 2024, time deposits were $76,109 million, while savings and demand deposits totaled $299,898 million.",
"**page_anchor":** "Deposits table, Time deposits row and Total Savings and demand deposits row, At December 31, 2024 column"
},

{
"**query":** "What is the anomaly in the deposit composition between 2023 and 2024?",
"**intent":** "anomaly",
"**evidence":** "Brokerage sweep deposits decreased from $148,274 million in 2023 to $142,550 million in 2024, while total deposits increased.",
"**page_anchor":** "Deposits table, Brokerage sweep deposits row and Total row, At December 31, 2023 and 2024 columns"
},

{
"**query":** "What is the total amount of borrowings for the 'Thereafter' category in 2024?",
"**intent":** "table-lookup",
"**evidence":** "The 'Thereafter' category in the Borrowings by Maturity table shows a total of $135,489 million.",
"**page_anchor":** "Borrowings by Maturity at December 31, 2024 table, Thereafter row, Total column"
}

]}