Constrained Dominant Sets for Multimodal Document Question Answering

Anonymous ACL submission

Abstract

Long multimodal document question answering is limited by which evidence reaches the reader, rather than by the quantity retrieved. In lengthy documents, findings often recur across figures, captions, and introductory sentences, causing similarity-based retrievers in modern multimodal retrieval-augmented generation (RAG) systems to allocate resources to near-duplicates while overlooking complementary evidence. This work introduces a retriever that selects evidence as a Constrained Dominant Set (CDS) on a query-augmented affinity graph, offering three advantages that similarity ranking does not. First, the query is encoded as a hard structural constraint, ensuring that every selected element is directly connected to the question through the cluster anchor. Second, the relevance–redundancy balance is determined automatically by a spectral bound, eliminating the need for manually tuned trade-offs required by diversity-aware selectors. Third, the selection process achieves a global equilibrium via replicator dynamics, thereby avoiding the distortions introduced by greedy heuristics. The method is inherently graph-based and does not require training. Using a Qwen3-VL-32B reader, CDS establishes a new state of the art on VisDoMBench (66.99 average) and improves over the no-retrieval baseline by 37.1 points on VisDoMBench and 4.8 on MMLongBench-Doc.[1]

1 Introduction

Vision–language models (VLM) encounter persistent challenges in processing long multimodal documents. On MMLongBench-Doc (Ma et al., 2024b), which consists of 135 PDFs with an average length of 47.5 pages, GPT-4o achieves an F1 score of only 44.9. Liu et al. (2024) identify a U-shaped attention curve, indicating that evidence situated in the middle of a long context is systematically under-utilized. This phenomenon can result in inferior multi-document question-answering performance compared to the closed-book baseline. Therefore, the principal limitation lies not in the quantity of accessible information, but in the selection process.

Most current multimodal RAG systems provide a flat top-B list ranked by query–candidate similarity (Faysse et al., 2024; Suri et al., 2025; Wang et al., 2025; Wan and Yu, 2025; Guo et al., 2024, 2025; Nguyen et al., 2025; Du et al., 2026). For queries related to a chart or table in a lengthy report, these retrievers frequently return the visual element, its caption, the introductory sentence, and the paragraph reiterating the finding. This method allocates multiple retrieval slots to a single fact, thereby limiting the inclusion of methodology, related results, or qualifying footnotes. Such redundancy is by design: long multimodal documents are structured to restate findings across modalities to support diverse reading patterns, and similarity-based retrieval further amplifies this engineered redundancy.

The information-retrieval community has long acknowledged that relevance alone is insufficient. Maximal Marginal Relevance (MMR) (Carbonell and Goldstein, 1998) incorporates a redundancy penalty into the relevance score, while Determinantal Point Processes (DPPs) (Kulesza and Taskar, 2012; Chen et al., 2018) favor subsets with greater feature diversity. However, both approaches are heuristic in the budget-constrained maximum a posteriori (MAP) regime: MMR employs a hand-tuned linear scalarization $\lambda \operatorname{sim}(q, d) - (1 - \lambda) \max_{j \in S} \operatorname{sim}(d, d_j)$ optimized greedily, causing the initial selection to constrain subsequent choices. DPP MAP is NP-hard, and standard pipelines revert to the same greedy approximation (Gillenwater et al., 2012). Importantly, neither method provides a native mechanism to enforce coherence between the selected set and the query. The query serves as

[1] Code is available at https://anonymous.4open.science/r/CDS-C939/README.mdhere

a scoring signal rather than a constraint. While this distinction may be negligible for flat candidate lists, it becomes significant for graph-structured candidate sets in long multimodal documents, where captions, figures, and passages often form near-identical clusters.

We reformulate evidence selection as a *constrained dominant-set* (CDS) problem (Bulò and Pelillo, 2017) on a query-augmented affinity graph. Dominant sets (Pavan and Pelillo, 2007; Bulò and Pelillo, 2017) generalize maximal cliques by representing strict local maxima of a quadratic program on the simplex, while constrained dominant sets (Zemene and Pelillo, 2016; Zemene et al., 2018b) additionally require the cluster to include a user-specified seed. In this framework, the query and any pinned memory notes constitute the constraint set S. Query–note edges are encoded as similarity, and note–note edges as dissimilarity. The resulting program is solved using replicator dynamics from evolutionary game theory (Pelillo, 1998; Bomze, 1997). The fixed point yields a soft membership vector whose support defines a cluster anchored on the query, characterized by high mutual relevance and low mutual redundancy. This approach requires no hyperparameter tuning, kernel design, or greedy locking; the selection emerges as the dynamics' equilibrium.

CDS methods have demonstrated strong empirical performance in computer vision, including image segmentation (Zemene et al., 2018b), image retrieval (Alemu and Pelillo, 2020), person re-identification (Alemu et al., 2019), multi-target tracking (Tesfaye et al., 2019), and geo-localization (Zemene et al., 2018a). However, these methods have not yet been applied to natural language processing (NLP) retrieval tasks. Long multimodal RAG presents a suitable context for this approach: candidate sets are sufficiently small for rapid convergence of the dynamics, the link structure is complex enough to benefit from graph-native methods, and the evidence budget is sufficiently constrained ($B = 10$, consistent with modern multimodal RAG systems (Du et al., 2026; Wang et al., 2025; Suri et al., 2025; Cho et al., 2024)) to make the relevance and diversity trade-off critical. The impact of varying B is analyzed in Appendix D.3.

Contributions.

- A **query-as-constraint** formulation is introduced, in which the query is incorporated into the constraint diagonal of the CDS objective. This approach enables query coherence to emerge as an equilibrium property rather than as a result of tuning trade-offs. The formulation generalizes to multi-query and pinned-memory settings without modification.
- The **first application of constrained dominant sets to multimodal RAG** is demonstrated, instantiated on an agentic-memory-style note graph (Xu et al., 2026) that incorporates note–note dissimilarity edges, query–note similarity edges, and employs replicator dynamics as the solver.
- **State-of-the-art (SOTA) results on VisDoMBench (Suri et al., 2025) and competitive performance on MMLongBench-Doc (Ma et al., 2024b).** Using a Qwen3-VL-32B reader, the method attains 66.99 average accuracy on VisDoMBench, outperforming G^2-Reader, ViDoRAG, MMGraphRAG, LightRAG, RAG-Anything, and a standalone GPT-5.

2 Related Work

Multimodal RAG. Recent research extends RAG (Lewis et al., 2020) to visually rich documents using several complementary strategies. These include page-image retrieval with late-interaction matching (ColPali (Faysse et al., 2024), M3DocRAG (Cho et al., 2024)) and single-vector visual-document encoding (DSE (Ma et al., 2024a)), text-visual fusion with consistency-constrained or iterative reasoning (VisDoM-RAG (Suri et al., 2025), ViDoRAG (Wang et al., 2025)), entity- or knowledge-graph traversal (GraphRAG (Edge et al., 2024), LightRAG (Guo et al., 2024), MMGraphRAG (Wan and Yu, 2025), RAG-Anything (Guo et al., 2025)), multi-agent chain-of-thought (MA-RAG (Nguyen et al., 2025)), dual evolving graphs (G^2-Reader (Du et al., 2026), previously the state of the art on VisDoMBench), and retrieval-aware tuning for long documents (M-LongDoc (Chia et al., 2025)). In most approaches, evidence is selected by ranking candidates against the query and selecting the top-B using cosine or learned similarity. Even graph-based systems primarily use the graph to produce or enrich candidates, rather than to select a budget-limited subset, with the final selection still based on a flat top-B ranking. The proposed method adopts this candidate-production pipeline but replaces the final

selection step with constrained graph clustering.

Long-document and multimodal QA benchmarks. Document question answering (QA) has evolved from single-page extraction (Mathew et al., 2021) to multi-page reasoning (Tito et al., 2023) and, more recently, to long-context multimodal benchmarks that address distinct modalities. These modalities include slides (SlideVQA (Tanaka et al., 2023)), scientific figures and tables (SPIQA (Pramanick et al., 2024), PaperTab (Hui et al., 2024) and FetaTab (Nan et al., 2022)), charts (SciGraphQA (Li and Tajbakhsh, 2023)), and multi-document settings (VisDoMBench (Suri et al., 2025)). MMLongBench-Doc (Ma et al., 2024b) emphasizes single-document long-context comprehension, while M-LongDoc (Chia et al., 2025) extends the average document length beyond 200 pages, and LongDocURL (Deng et al., 2025) integrates understanding, reasoning, and locating over multimodal long documents.

Memory-augmented LLM systems. The candidate set is constructed following the approach of agentic memory systems, such as MemGPT (Packer et al., 2023), MemoryBank (Zhong et al., 2024), and A-MEM (Xu et al., 2026), where each document element is transformed into a memory note enriched with summaries, keywords, tags, and explicit links. These links provide the affinity signal that drives the CDS dynamics. While previous work has focused on constructing and evolving the memory store, this work addresses the complementary problem of selecting a query-conditioned, budget-respecting, and mutually coherent subset from it.

3 Methodology

We propose a document-grounded multimodal question answering pipeline for long documents containing text, figures, tables, and other visually grounded content. The method represents each document as a multimodal graph, retrieves a compact and coherent evidence subgraph using CDS optimization, and generates the final answer with a VLM conditioned only on the retrieved evidence.

3.1 Problem Formulation

Let $D = \{e_1, \ldots, e_N\}$ be a long multimodal document decomposed into N elements, where each e_i is a self-contained unit of evidence such as text passage, figure, table or caption. Given a question q, the objective is to produce an answer $\hat{a}$ that is grounded in a subset of these elements. In this work, we adopt a *retrieve-then-read* formulation: a retriever selects an evidence set $R \subseteq D$, and a reader VLM generates an answer based on the question and the selected evidence, $\hat{a} = f_{\text{VLM}}(q, R)$.

The primary challenge is selecting R. Relevance alone is insufficient because the most relevant elements are often near-duplicates, such as a passage and its caption or repeated content, which results in redundant information. Effective evidence must be both *relevant* to the question and mutually *non-redundant/diverse*, ensuring that the selected elements collectively cover complementary information. Retrieval is therefore formalized as follows:

$$R^\star = \arg\max_{R \subseteq D} \ \mathrm{Rel}(R, q) \ + \ \mathrm{Div}(R), \quad (1)$$

$\mathrm{Rel}(R, q)$ quantifies query relevance, $\mathrm{Div}(R)$ rewards diversity by minimizing mutual redundancy among selected elements. Both $\mathrm{Rel}(R, q)$ and $\mathrm{Div}(R)$ are quadratic terms as they relate sets, the first considers the relations between $\mathcal{D}$ and $\mathcal{Q}$ while the latter considers $\mathcal{D}$ with $\mathcal{D}$. Equation (1) looks for a subset $R \subseteq D$ and is formulated as a constrained weighted maximal clique extraction problem (CDS) on a query-augmented graph constructed from the document elements. In this graph, edge weights represent query–element relevance and element–element dissimilarity, with the query incorporated as a constraint vertex to anchor the selected cluster. The reader receives only q and R and is instructed to answer strictly on the basis of the provided evidence, to preserve exact figure or table identifiers and numerical values, and to abstain when support is lacking.

3.2 CDS Graph Construction

We use CDS as the retrieval operator over a query-augmented multimodal graph. The task is to extract a (possibly tight) partition of the nodes of a graph that contains a designated node: in our setting, the graph's nodes are the document's elements and the designated node is the query. CDS model addresses exactly this task by extracting a subset of nodes that contains a given seed; it requires a similarity graph and a set of query nodes, and returns the partition. We therefore work with two sets: the document set $\mathcal{D}$ and the query set $\mathcal{Q} = \{q\}$.

Document–Document graph. For each document, a multimodal graph $G_D = (V_D, E_D, \omega_D)$

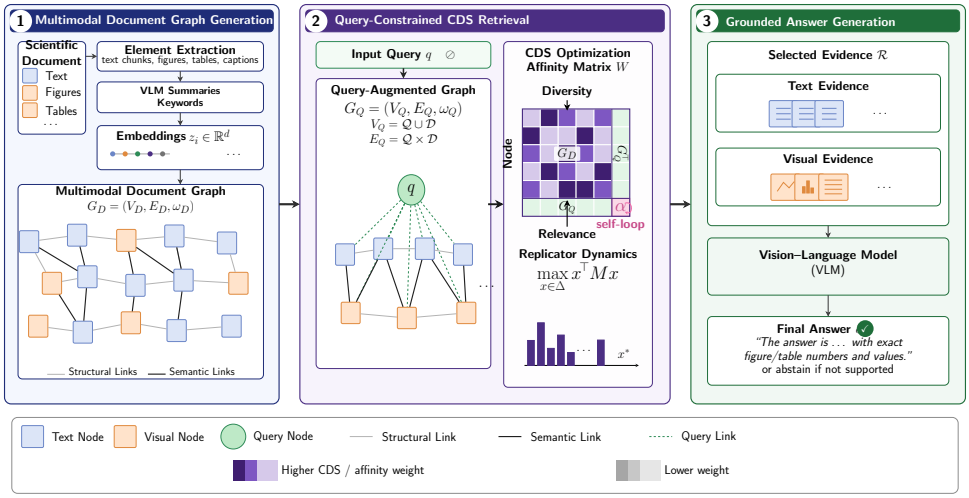

Figure 1: Overview of the proposed pipeline. **(1)** The document is parsed into atomic nodes (text, figures, tables), each embedded as $\mathbf{z}_i \in \mathbb{R}^d$ and connected into a multimodal graph G_D via structural and VLM-verified semantic links. **(2)** The query q is added as a constraint vertex with a spectral self-loop α, forming the query-augmented affinity matrix M that combines note–note dissimilarity (diversity) with query–note similarity (relevance). Replicator dynamics solve $\max_{\mathbf{x} \in \Delta} \mathbf{x}^\top M \mathbf{x}$, yielding a soft membership $\mathbf{x}^\star$ anchored on q. **(3)** The top-B nodes by $x_i^\star$ are passed to a VLM reader that generates the final answer grounded strictly in the retrieved evidence.

is constructed over its elements. The source document is processed by a multimodal parser (e.g., MinerU (Wang et al., 2024) or DeepSeek-OCR (Wei et al., 2025)) that extracts text passages and visual elements while preserving layout and reading order. These elements are segmented into atomic units, forming the node set $V_D = \mathcal{D}$. Each node $v_i \in V_D$ stores its raw content c_i (a text passage or a figure, table, or image crop with its caption and surrounding context), a VLM-generated summary s_i, and a keyword set k_i. To integrate heterogeneous text and visual elements into a shared semantic space, each node is represented by a textual surrogate $m_i = s_i \,\|\, \text{“keywords:”} \,\|\, k_i$, where $\|$ denotes concatenation, and is encoded as $\mathbf{z}_i = f_{\text{emb}}(m_i) \in \mathbb{R}^d$. Edge candidates $E_D \subseteq V_D \times V_D$ are initially proposed based on document structure (such as adjacent text chunks, caption–figure and caption–table pairs, explicit “Figure” or “Table” references, and page proximity) and are retained only if a VLM verifies them as genuine relations (including direct reference, elaboration, part–whole, causal, or contextual dependency). Following G^2-Reader (Du et al., 2026), this construction is iterated to progressively refine the graph, with three evolution iterations employed. Edge weights between node i and j is defined as $\omega_D(i,j) = 1 - \cos(\mathbf{z}_i, \mathbf{z}_j)$[2]. Two properties are notable: (i) this is a dissimilarity graph, where more diverse nodes are more strongly connected, and (ii) it is static, computed once and reused for all queries. The graph is stored as a pair-wise dissimilarity matrix $W \in \mathbb{R}^{|\mathcal{D}| \times |\mathcal{D}|}$. The graph G_D captures the term $\mathrm{Div}(R)$.

Query–Document graph. Given the query set $\mathcal{Q}$ and the document set $\mathcal{D}$, we construct a bipartite graph $G_Q = (V_Q, E_Q, \omega_Q)$ where $V_Q = \mathcal{Q} \cup \mathcal{D}$, $E_Q = \mathcal{Q} \times \mathcal{D}$, and

$$\omega_Q(i,j) = \begin{cases} 0, & i,j \in \mathcal{Q}, \\ \cos(\mathbf{z}_i, \mathbf{z}_j), & \text{otherwise}, \end{cases}$$

with the query encoded using the same encoder, $\mathbf{z}_q = f_{\text{emb}}(q)$. This is a *similarity* graph: the more similar a query $q \in \mathcal{Q}$ is to a document node $d \in \mathcal{D}$, the more strongly they are connected. The graph is stored in a similarity matrix $S \in \mathbb{R}^{|\mathcal{Q}| \times |\mathcal{D}|}$; since $\mathcal{Q} = \{q\}$, this reduces to a row vector $S \in \mathbb{R}^{1 \times |\mathcal{D}|}$. A sparsification threshold τ may be applied to prune weak query–node edges, specifically those for which $\cos(\mathbf{z}_q, \mathbf{z}_i) < \tau$. The default setting is $\tau = 0$, and its effect is analyzed in Appendix D.1. The graph G_Q captures the quadratic term $\mathrm{Rel}(R, q)$.

The next step is to combine the two graphs such that the CDS can extract the relevant nodes from $\mathcal{D}$ given $\mathcal{Q}$. Remember that G_D captures the most

[2]The note block can be optionally scaled by a diversification strength β, where $\beta = 1$ restores the default. This parameter is systematically varied in Appendix D.3 (Table 11)

diverse nodes, while G_Q models the most relevant nodes to the query. We combine the two graphs G_D and G_Q into $G = (V, E, \omega)$, where $V = V_D \cup V_Q$, $E = E_D \cup E_Q$ and ω is defined as:

$$\omega_{i,j} = \begin{cases} \omega_D(i,j) = 1 - \cos(\mathbf{z}_i, \mathbf{z}_j), & i \in \mathcal{D} \text{ and } j \in \mathcal{D}, \\ 0, & i = j \text{ and } i \in \mathcal{D}, \\ \alpha, & i = j \text{ and } i \in \mathcal{Q}, \\ \omega_Q(i,j) = \cos(\mathbf{z}_i, \mathbf{z}_j), & \text{otherwise.} \end{cases} \tag{2}$$

where $\alpha = \lambda_{\max}(W) + \varepsilon$, $\lambda_{\max}(\cdot)$ is the largest eigenvalue of the matrix W, and $\varepsilon > 0$ is a small margin. We use $\varepsilon = 10^{-3}$. The CDS extracts sequentially the maximal cliques from the graph G until all the query nodes in $\mathcal{Q}$ get partitioned. The value α on the self-loop of the node q ensures, by construction, that q will be extracted as part of a maximal clique. The graph G is then represented as a pairwise matrix $M \in \mathbb{R}^{|V|\times|V|}$. The matrix M captures both the $\mathrm{Div}(R)$ and the $\mathrm{Rel}(R, q)$ (see figure 1 for a graphical interpretation).

To extract the subset R that is both diverse and relevant in G, we optimize the following quadratic assignment problem using the replicator dynamics (see (Zemene et al., 2018b) for proofs and details):

$$\mathbf{x}^\star = \arg\max_{\mathbf{x}\in\Delta} \mathbf{x}^\top M \mathbf{x}. \tag{3}$$

where Δ is the standard $|V|-$dimensional simplex (i.e. $\sum_i x_i = 1$). The solution $\mathbf{x}^\star$ assigns a soft membership weight to each document and query vertex to the maximal clique. Its support is $\sigma(\mathbf{x}^\star) = \{i : x_i^\star > 0\}$. At convergence, the query vertex is removed from the support, and document nodes are ranked by $\mathrm{score}(v_i) = x_i^\star$.

Selection and generation. Solving Eq. (3) via replicator dynamics yields a soft membership weight $\mathbf{x}^\star$ to each vertex. We discard the query vertex and rank documents vertex by $\mathrm{score}(v_i) = x_i^\star$, forming the evidence set from the top-B:

$$R = \text{Top-}B_{v_i\in\mathcal{V}}\big(\mathrm{score}(v_i)\big).$$

The VLM then receives *only* the question q and R: for text nodes, the passage (truncated if long); for visual nodes, the image with its caption or generated summary. The answer is thus grounded in the retrieved multimodal evidence rather than in auxiliary reasoning chains.

4 Dataset and Baselines

4.1 Datasets

We conduct evaluations on two multimodal long-document QA benchmarks, adhering to the VisDoM (Suri et al., 2025) and G^2-Reader (Du et al., 2026) protocols to ensure direct comparability with prior state-of-the-art methods. **VisDoMBench** comprises a heterogeneous suite that encompasses the principal modalities of real-world long-document QA: **FetaTab** and **PaperTab** (table-grounded QA over web and scientific tables), **SciGraphQA** (chart-grounded QA over scientific graphs), **SlideVQA** (slide decks containing mixed text, figures, and tables), and **SPIQA** (figure- and table-grounded QA over scientific papers). Each query is associated with a candidate set limited to five documents (one gold and up to four distractors), ensuring that retrieval occurs within a realistic multi-document context. This setting intensifies the redundancy challenge, as a single fact may appear in a passage, its caption, and a downstream summary across the five parsed documents.

We also report results on **MMLongBench-Doc** (Ma et al., 2024b), where each query is grounded in a single, very long document. This benchmark is evaluated separately because the primary retrieval challenge is within-document redundancy, such as repeated passages, caption and passage duplicates, and recurring boilerplate, rather than cross-document overlap. This setting provides a rigorous test of CDS performance under an extended per-query note set.

For both benchmarks, every source PDF is parsed offline with MinerU (Wang et al., 2024) into a structured stream of text chunks (recursive splitting, chunk size 3000, overlap 300) and visual elements (figures, tables, charts) with captions and a $\pm$1000-word local context window. Each element becomes a memory note enriched by a VLM with a content summary, keywords, tags, and links to related notes (Section 3.2). The per-query note set is the union of notes from all candidate documents and forms the vertex set of the affinity graph that every retrieval method below operates on.

4.2 Baselines

To isolate the contribution of CDS-based selection, our *selection-rule* Baselines and the no-retrieval control are executed using an identical pipeline, ensuring consistency across experimental conditions. same parser (MinerU), note-construction and graph-evolution, text encoder (`nomic-embed-text-v1.5`, 768-dim, L_2-normalized) (Nussbaum et al., 2024), evidence budget (B=10), reader VLM, and greedy decoding (τ=0), ensuring that the only variable is the

Table 1: Main results on the full VisDoMBench benchmark. All results are averaged over three runs, with "±" denoting standard deviation. † indicates systems using OpenAI `text-embedding-3-small` as the retrieval backbone; ‡ indicates our method using the open `nomic-embed-text-v1.5` encoder. **Bold** marks the best per-column score.

Model	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	Average
GPT-5	55.22 ± 0.09	63.94 ± 0.31	37.08 ± 0.12	64.08 ± 0.32	45.06 ± 0.10	53.08 ± 0.10
Qwen3-VL-32B†	29.86 ± 0.08	37.39 ± 0.36	34.32 ± 0.27	23.06 ± 0.22	24.87 ± 0.24	29.90 ± 0.11
Deepseek-OCR	63.60 ± 0.40	70.32 ± 0.12	51.58 ± 0.24	61.91 ± 0.40	65.69 ± 0.12	62.62 ± 0.13
RAGAnything	67.69 ± 0.96	57.76 ± 0.24	42.02 ± 1.35	41.60 ± 2.60	52.18 ± 0.49	52.25 ± 0.63
MA-RAG	45.52 ± 0.22	27.70 ± 0.19	33.43 ± 0.45	29.32 ± 0.25	29.40 ± 0.21	33.07 ± 0.13
GraphRAG	62.65 ± 0.20	61.35 ± 0.19	42.90 ± 0.00	65.76 ± 0.38	21.68 ± 0.00	50.87 ± 0.09
LightRAG	73.88 ± 0.00	64.71 ± 0.38	51.02 ± 0.04	**75.00 ± 0.01**	29.63 ± 0.01	58.85 ± 0.08
MMGraphRAG	69.91 ± 0.23	**72.40 ± 0.55**	56.36 ± 0.58	64.11 ± 0.25	54.20 ± 0.15	63.40 ± 0.18
VisDoMRAG	75.44 ± 0.00	61.02 ± 0.50	56.21 ± 0.15	63.36 ± 0.14	69.03 ± 0.36	65.01 ± 0.13
ViDoRAG	68.18 ± 0.46	58.74 ± 0.38	43.67 ± 0.15	37.86 ± 0.14	71.71 ± 0.11	56.03 ± 0.13
G^2-Reader†	73.19 ± 0.21	66.89 ± 0.11	57.10 ± 0.21	61.56 ± 0.11	**72.31 ± 0.00**	66.21 ± 0.07
Qwen3-VL-32B + CDS‡	**78.85 ± 0.00**	70.90 ± 0.00	**65.59 ± 0.01**	57.45 ± 0.31	62.17 ± 0.38	**66.99 ± 0.10**

Table 2: Results on MMLongBench-Doc. Accuracy (%); "+ CDS" is our retriever, reported as mean ± std over 3 runs. The plain rows are the Single-VLM (no-retrieval) baseline.

Model	Params	Accuracy
Qwen3-VL-32B	32B	40.19
Qwen3-VL-32B + CDS	32B	**45.01 ± 0.39**
Qwen2.5-VL-7B	7B	28.36
Qwen2.5-VL-7B + CDS	7B	**32.30 ± 0.10**
GLM-4.1V	9B	**41.04**
GLM-4.1V + CDS	9B	39.15 ± 0.20

Table 3: Cosine vs. CDS on the subset (50 queries/-dataset), the same 250 samples used in G^2-Reader's ablation. Accuracy (%). G^2-Reader reports 68.0 on this subset (Du et al., 2026).

Method	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	Avg.
Cosine (top-B)	88.0	73.5	66.0	48.0	52.0	65.5
CDS	**94.0**	**77.6**	66.0	**50.0**	**60.0**	**69.5**

selection mechanism. Published systems are not re-implemented; instead, their scores are cited from G^2-Reader (Du et al., 2026). These systems are evaluated under their original configurations, which utilize a more advanced retrieval backbone[3] but the same candidate-set protocol, evidence budget, and LLM-as-judge pipeline adopted here.

No-retrieval control: Consistent with G^2-Reader, ten text chunks and ten visual elements are deterministically sampled from the candidate documents and interleaved with the question in a chain-of-thought prompt (Appendix B.6).

Published systems: The primary baseline is G^2-Reader (Du et al., 2026), which represents the previous state of the art on VisDoMBench. Additional comparisons are made with recent multimodal RAG and document question answering systems: GPT-5 (Singh et al., 2025), Deepseek-OCR (Wei et al., 2025), RAGAnything (Guo et al., 2025), MA-RAG (Nguyen et al., 2025), GraphRAG (Edge et al., 2024), LightRAG (Guo et al., 2024), MMGraphRAG (Wan and Yu, 2025), VisDoMRAG (Suri et al., 2025), and ViDoRAG (Wang et al., 2025). Their scores are cited from the G^2-Reader paper, which evaluates all systems under the same candidate-set protocol, evidence budget, and LLM-as-judge pipeline adopted in this study.

Selection-rule baselines. CDS is evaluated against three selection rules using the same per-query graph and encoder, ensuring that only the selection mechanism differs. Full algorithmic details, including the exact MMR, k-DPP, and PPR updates, are deferred to Appendix B. **MMR** (Carbonell and Goldstein, 1998) greedily maximizes $\lambda \cos(z_q, z_i) - (1-\lambda) \max_{j \in S} \cos(z_i, z_j)$; λ is swept over $\{0.3, 0.5, 0.7, 0.9\}$, and the optimal per-dataset setting is reported. **k-DPP** (Kulesza and Taskar, 2011) selects a size-B subset by greedy MAP on $L_{ij} = q_i q_j K_{ij}$, where $q_i = \max(\cos(z_q, z_i), 0)$ and $K_{ij} = \cos(z_i, z_j)$, thereby promoting diversity through determinantal volume rather than query-anchored repulsion. Personalized PageRank (PPR) as a retriever returns the top-B notes by Personalized PageRank from a query-seeded teleport, evaluating whether graph structure alone is sufficient. These baselines isolate the relevance/diversity trade-off, the diversification mechanism, and the graph structure. Any remaining performance gap can thus be attributed to CDS's joint use of similarity, repulsion, and the query-as-constraint formulation. For direct comparison with

[3] OpenAI text-embedding-3-small

Table 4: Comparison with alternative selection strategies. Accuracy (%), mean ± std over 3 runs; Qwen3-VL-32B reader. MMR uses λ=0.5, DPP uses θ=1. **Avg.** is the VisDoMBench average; MMLongBench-Doc is a separate benchmark. CDS is best on every dataset.

Method	FetaTab	PaperTab	SciGraphQA	SlideVQA	SPIQA	**Avg.**	MMLongBench
MMR (λ=0.5)	57.0±0.0	55.5±0.7	44.7±0.2	46.3±0.4	69.9±0.5	54.7±0.2	42.8±0.2
DPP	44.1±0.2	45.6±0.5	34.4±0.3	35.6±0.6	62.5±0.4	44.4±0.3	41.2±0.1
PPR	59.8±0.2	52.4±0.9	49.1±0.7	54.8±0.2	55.9±0.4	54.4±0.1	40.4±0.3
CDS	**71.0±0.6**	**65.6±0.4**	**57.4±0.4**	**62.2±0.4**	**78.8±0.2**	**67.0±0.2**	**45.0±0.4**

G^2-Reader, plain cosine top-B retrieval is reported on a 250-sample subset (Table 3) as the relevance-only reference on identical data.

5 Experimental Setup

We evaluate on **VisDoMBench** (Suri et al., 2025) and **MMLongBench-Doc** (Ma et al., 2024b), following the G^2-Reader protocol (Du et al., 2026): each query is answered against a five-document candidate pool (gold + four distractors), and answers are scored by a GPT-4o-mini judge (Zheng et al., 2023). Unless noted, CDS uses an evidence budget B=10, no query-row sparsification, and uniform initialization; the reader is **Qwen3-VL-32B** (Bai et al., 2025a), with **Qwen2.5-VL-7B** (Bai et al., 2025b) and **GLM-4.1V-9B** (Hong et al., 2025) reported for scaling.

5.1 Main Results

Table 1 presents results for the full VisDoMBench. The proposed method, **Qwen3-VL-32B+CDS**, achieves the highest average accuracy (**66.99**), surpassing all previous systems, including the strongest graph- and agent-based approaches: G^2-Reader (66.21) and VisDoMRAG (65.01). This performance is attained using a single replicator-dynamics retrieval step rather than multi-agent planning or explicit document-graph construction. CDS ranks first on SPIQA (78.85) and PaperTab (65.59), and second on FetaTab (70.90). However, it is less competitive on SciGraphQA and SlideVQA, where chart- and slide-specialized methods such as LightRAG and ViDoRAG perform better. The impact of retrieval is considerable: compared to the no-retrieval Qwen3-VL-32B baseline (29.90), CDS provides an average improvement of +37.1 points. This result suggests that evidence selection, rather than reader capacity, constitutes the primary bottleneck in these multi-document tasks.

Table 2 presents MMLongBench-Doc results for three readers of increasing scale. CDS outperforms the no-retrieval baseline for Qwen3-VL-32B (40.19 → 45.01, +4.82) and Qwen2.5-VL-7B (28.36 → 32.30, +3.94), while performing slightly below the baseline for GLM-4.1V (41.04 → 39.15). The observed gains are smaller than those on VisDoMBench, as MMLongBench-Doc questions are based on a single long document, where directly providing sampled content already constitutes a strong baseline. Nevertheless, CDS benefits both Qwen readers and remains competitive with GLM, demonstrating that the approach generalizes to the long-single-document setting without modification. Importantly, these results are achieved using a lightweight open text encoder (`nomic-embed-text-v1.5`, 768-d), whereas competing systems employ larger embeddings (e.g., OpenAI `text-embedding-3-small`). Achieving comparable or superior performance with a less powerful retrieval backbone highlights that the improvements are attributable to the selection formulation rather than the embedding model.

On the identical subset used in G^2-Reader's ablation, evaluated with the same reader and judge and differing only in the selection rule, CDS outperforms a plain cosine top-B retriever by an average of +4.0 points (69.5 vs. 65.5, Table 3). CDS achieves gains on every dataset except PaperTab, where performance is tied, and surpasses G^2-Reader's reported 68.0 on the same data (Du et al., 2026), despite replacing agentic planning with a single, training-free retrieval step. Figure 2 illustrates this distinction. The cosine method concentrates selections within a single embedding region, whereas CDS distributes selections across the gold-relevant subspace, thereby capturing significantly more gold notes per query.

Comparison with alternative selection strategies. The necessity of the CDS formulation is evaluated against simpler relevance-plus-diversity selectors. Table 4 presents a comparison of CDS with MMR ($\lambda = 0.5$), k-DPP, and PPR-as-retriever, as defined in Section 4.2. CDS achieves the highest performance across all datasets and, on average (67.0), outperforms MMR (54.7), DPP (44.4), and PPR

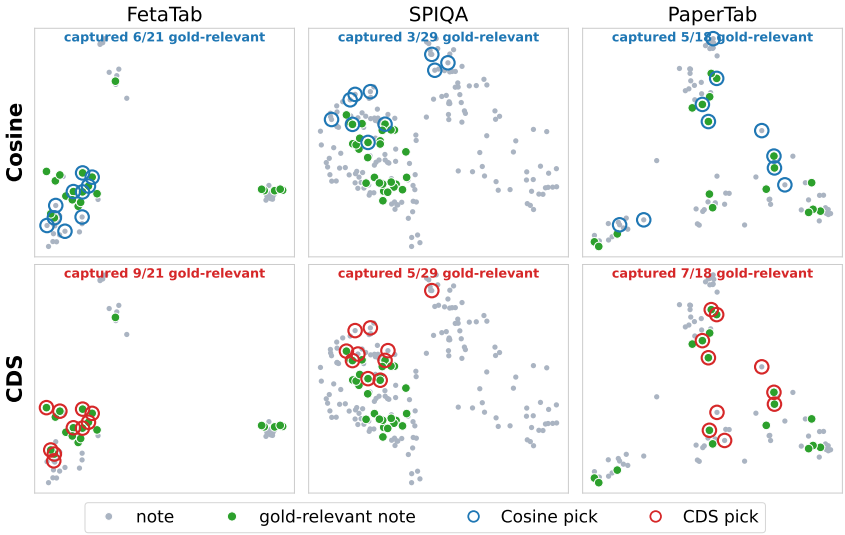

Figure 2: Cosine vs. CDS evidence selection on the per-query note map (2-D Principal Component Analysis, PCA); grey = note, green = gold-relevant). Top: cosine top-B; bottom: CDS. Cosine concentrates its picks in a single region and misses much of the gold evidence, whereas CDS spreads its selection and captures more gold-relevant notes (counts shown per panel). Panels are queries where CDS most outperforms cosine in coverage.

(54.4). The two diversity-based selectors underperform because, at a balanced relevance and diversity weighting, they tend to include off-topic but dissimilar notes in the evidence set, and performance improves only when diversity is entirely suppressed. PPR does not perform better; relying solely on graph structure provides a weaker signal than embedding relevance. CDS avoids this limitation because the spectral bound on its constraint parameter automatically balances query relevance with inter-note dissimilarity, enabling productive diversification without manual trade-off tuning or per-dataset adjustment.

Additional Experiments: Appendix D presents additional ablation studies on Qwen3-VL-32B. The initialization prior (Table 6) and graph-evolution iterations (Table 7) demonstrate that CDS is robust to both seeding and refinement choices. Query-row sparsification (Appendix D.1, Table 9) indicates that performance remains consistent for $\tau \in [0, 0.3]$ and declines only with aggressive thresholding. The diversification-block sign ablation (Appendix D.2, Table 8) reveals that converting note–note edges from dissimilarity to similarity reduces the VisDoMBench average by 4.0 points, confirming that dissimilarity coupling underpins CDS's advantage. Finally, the evidence-budget sweep (Appendix D.3, Table 10) and diversification-strength sweep (Table 11) demonstrate that accuracy remains stable across $B \in [10, 20]$ and $\beta \in [0.5, 1]$

6 Conclusion

This work formulates multimodal evidence selection for document question answering as a CDS problem on a query-augmented graph. A single query (constraint) vertex contributes relevance via its cosine edges, while a node-to-node dissimilarity block encourages non-redundancy. Replicator dynamics yield a soft membership whose support constitutes a relevant and diverse evidence set. The method is training-free, requires no tuning of a relevance-diversity trade-off parameter, and employs a parameter-free uniform initialization. On the VisDoMBench benchmark, the approach achieves the highest average accuracy (66.99), outperforming advanced graph- and agent-based systems such as G^2-Reader and VisDoMRAG, despite its simplicity. It also surpasses a no-retrieval baseline by a substantial margin, indicating that evidence selection, rather than reader capacity, is the primary bottleneck in these multi-document tasks. Extensive ablations (Appendix D) demonstrate the robustness of the approach. These findings suggest that constraint-driven graph clustering warrants further investigation as a general retrieval primitive. Future directions include open-corpus retrieval with larger candidate pools, multi-query and pinned-memory scenarios already supported by the formulation, and closer integration between memory-graph evolution and the selection objective.

Limitations

The proposed method is evaluated using the same five-document retrieval protocol (gold plus four distractors) as G^2-Reader and previous VisDoMBench studies, which ensures a fair and controlled comparison across methods. Nevertheless, this protocol constitutes a relatively constrained retrieval setting. The advantages of diversity-aware selection are expected to become more evident under more stringent conditions, such as when candidate pools are larger or when distractors are more challenging. Stress tests involving larger candidate pools, more difficult distractors, or open-corpus retrieval are reserved for future research. Performance improvements are also less substantial in the single-long-document setting (MMLongBench-Doc), where selection is less critical than in the multi-document scenario for which CDS is primarily designed. The current evaluation includes two benchmarks, three open VLM readers, and an LLM-based judge. Expanding to broader domains, incorporating additional readers, and conducting human evaluations are identified as promising directions for future investigation. Furthermore, CDS does not uniformly benefit all reader VLMs: on MMLongBench-Doc, GLM-4.1V-9B performs marginally below its no-retrieval baseline (-1.89 points), suggesting that the interaction between retrieved-evidence formatting and reader-specific instruction-following warrants further investigation.

References

Leulseged Tesfaye Alemu and Marcello Pelillo. 2020. Multi-feature fusion for image retrieval using constrained dominant sets. *Image and Vision Computing*, 94:103862.

Leulseged Tesfaye Alemu, Marcello Pelillo, and Mubarak Shah. 2019. Deep constrained dominant sets for person re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9855–9864.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, and 1 others. 2025a. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025b. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Immanuel M Bomze. 1997. Evolution towards the maximum clique. *Journal of Global Optimization*, 10(2):143–164.

Samuel Rota Bulò and Marcello Pelillo. 2017. Dominant-set clustering: A review. *European Journal of Operational Research*, 262(1):1–13.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

Laming Chen, Guoxin Zhang, and Eric Zhou. 2018. Fast greedy map inference for determinantal point process to improve recommendation diversity. *Advances in neural information processing systems*, 31.

Yew Ken Chia, Liying Cheng, Hou Pong Chan, Maojia Song, Chaoqun Liu, Mahani Aljunied, Soujanya Poria, and Lidong Bing. 2025. M-longdoc: A benchmark for multimodal super-long document understanding and a retrieval-aware tuning framework. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9244–9261.

Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. 2024. M3docrag: Multimodal retrieval is what you need for multi-page multi-document understanding. *arXiv preprint arXiv:2411.04952*.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and 1 others. 2025. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1135–1159.

Yaxin Du, Junru Song, Yifan Zhou, Cheng Wang, Jiahao Gu, Zimeng Chen, Menglan Chen, Wen Yao, Yang Yang, Ying Wen, and 1 others. 2026. g^2-Reader: Dual Evolving Graphs for Multimodal Document Comprehension. *arXiv preprint arXiv:2601.22055*.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2024. Colpali: Efficient document retrieval with vision language models. *arXiv preprint arXiv:2407.01449*.

Jennifer Gillenwater, Alex Kulesza, and Ben Taskar. 2012. Near-optimal map inference for determinantal point processes. *Advances in Neural Information Processing Systems*, 25.

Zirui Guo, Xubin Ren, Lingrui Xu, Jiahao Zhang, and Chao Huang. 2025. Rag-anything: All-in-one rag framework. *arXiv preprint arXiv:2510.12323*.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tian Ao, and Chao Huang. 2024. Lightrag: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*, 2(3).

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, and 1 others. 2025. Glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv e-prints*, pages arXiv–2507.

Yulong Hui, Yao Lu, and Huanchen Zhang. 2024. Uda: A benchmark suite for retrieval augmented generation in real-world document analysis. *Advances in Neural Information Processing Systems*, 37:67200–67217.

Alex Kulesza and Ben Taskar. 2011. k-dpps: Fixed-size determinantal point processes. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 1193–1200.

Alex Kulesza and Ben Taskar. 2012. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2-3):123–286.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.

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

Shengzhi Li and Nima Tajbakhsh. 2023. Scigraphqa: A large-scale synthetic multi-turn question-answering dataset for scientific graphs. *arXiv preprint arXiv:2308.03349*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173.

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

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, and 1 others. 2024b. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, and 1 others. 2022. Fetaqa: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

Thang Nguyen, Peter Chin, and Yu-Wing Tai. 2025. Ma-rag: Multi-agent retrieval-augmented generation via collaborative chain-of-thought reasoning. *arXiv preprint arXiv:2505.20096*.

Zach Nussbaum, John X Morris, Brandon Duderstadt, and Andriy Mulyar. 2024. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*.

Charles Packer, Vivian Fang, Shishir G Patil, Kevin Lin, Sarah Wooders, and Joseph E Gonzalez. 2023. Memgpt: towards llms as operating systems.

Massimiliano Pavan and Marcello Pelillo. 2007. Dominant sets and pairwise clustering. *IEEE transactions on pattern analysis and machine intelligence*, 29(1):167–172.

Marcello Pelillo. 1998. Replicator equations, maximal cliques, and graph isomorphism. *Advances in Neural Information Processing Systems*, 11.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. 2024. Spiqa: A dataset for multimodal question answering on scientific papers. *Advances in Neural Information Processing Systems*, 37:118807–118833.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Manan Suri, Puneet Mathur, Franck Dernoncourt, Kanika Goswami, Ryan A Rossi, and Dinesh Manocha. 2025. Visdom: Multi-document qa with visually rich elements using multimodal retrieval-augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6088–6109.

Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. 2023. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13636–13645.

Yonatan Tariku Tesfaye, Eyasu Zemene, Andrea Prati, Marcello Pelillo, and Mubarak Shah. 2019. Multi-target tracking in multiple non-overlapping cameras using fast-constrained dominant sets. *International Journal of Computer Vision*, 127(9):1303–1320.

Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. 2023. Hierarchical multimodal transformers for multipage docvqa. *Pattern Recognition*, 144:109834.

Xueyao Wan and Hang Yu. 2025. Mmgraphrag: Bridging vision and language with interpretable multimodal knowledge graphs. *arXiv preprint arXiv:2507.20804*.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, and 1 others. 2024. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*.

Qiuchen Wang, Ruixue Ding, Zehui Chen, Weiqi Wu, Shihang Wang, Pengjun Xie, and Feng Zhao. 2025. Vidorag: Visual document retrieval-augmented generation via dynamic iterative reasoning agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9124–9145.

Haoran Wei, Yaofeng Sun, and Yukun Li. 2025. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2026. A-mem: Agentic memory for llm agents. *Advances in Neural Information Processing Systems*, 38:17577–17604.

Eyasu Zemene and Marcello Pelillo. 2016. Interactive image segmentation using constrained dominant sets. In *European Conference on Computer Vision*, pages 278–294. Springer.

Eyasu Zemene, Yonatan Tariku Tesfaye, Haroon Idrees, Andrea Prati, Marcello Pelillo, and Mubarak Shah. 2018a. Large-scale image geo-localization using dominant sets. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):148–161.

Eyasu Zemene Zemene, Leulseged Tesfaye Alemu, and Marcello Pelillo. 2018b. Dominant sets for "constrained" image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(10):2438–2451.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 19724–19731.

The Use of Large Language Models

We employed a Large Language Model to assist with reducing wordy paragraphs to help the paper fit within the page limit.

A Algorithm: Pseudocode

Algorithm 1 summarises the end-to-end CDS retrieval procedure described in Section 3.2, from raw query and document nodes to the final reader output. The algorithm has three logical stages. Lines 1–2 construct the query-augmented affinity matrix A: note–note entries encode dissimilarity ($1 - \mathbf{z}_i^\top \mathbf{z}_j$) so that redundant evidence repels, while query–note entries encode similarity ($\max(\mathbf{z}_q^\top \mathbf{z}_i, 0)$) so that the query attracts evidence aligned with the question. Line 3–5 optionally apply the sparsification threshold τ to prune weak query edges (Appendix D.1). Line 6 forms the CDS payoff matrix M by subtracting $\alpha \mathbf{I}_{\bar{\mathcal{S}}}$ from A on the complement of the constraint set $\mathcal{S} = \{q\}$, where $\alpha = \lambda_{\max}(A) + \varepsilon$ is the spectral bound that guarantees the query vertex appears in the dominant cluster. Line 7 runs the replicator dynamics from the initial prior π (uniform by default; see Appendix D) until convergence to the local maximiser $\mathbf{x}^\star$ of $\mathbf{x}^\top M \mathbf{x}$ on the simplex. The shift constant c ensures the multiplicative update is well-defined when M has negative entries. Line 8 reads off the evidence set R as the top-B document vertices by $x_i^\star$ above the support threshold θ, discarding the query vertex. Line 9 hands (q, R) to the VLM reader for grounded answer generation. The procedure is deterministic given the inputs and requires no training, no relevance–diversity trade-off parameter, and no greedy locking; selection is the equilibrium of a single quadratic program on the query-augmented graph.

B Baseline Implementation Details

All retrieval baselines operate on the same per-query memory graph and the same text encoder as CDS, so that any difference is attributable to the selection rule alone. Let the graph contain N notes with ℓ_2-normalized embeddings $\mathbf{z}_1, \ldots, \mathbf{z}_N \in \mathbb{R}^d$ (d=768, `nomic-embed-text-v1.5`) and let $\mathbf{z}_q$ be the normalized query embedding. We define the query relevance of note i and the note–note similarity as:

$$\mathrm{rel}_i = \mathbf{z}_q^\top \mathbf{z}_i, \qquad S_{ij} = \mathbf{z}_i^\top \mathbf{z}_j, \tag{4}$$

Algorithm 1 CDS: Multimodal evidence retrieval

Input: document nodes $\mathcal{V} = \{1, \ldots, N\}$ with ℓ_2-normalised embeddings $\{\mathbf{z}_i\}$; question q; evidence budget B; sparsity threshold τ; init prior $\boldsymbol{\pi}$ (uniform by default); embedder f_{emb}; VLM f_{VLM}.
Output: answer $\hat{a}$.

1: $\mathbf{z}_q \leftarrow \mathrm{norm}\big(f_{\text{emb}}(q)\big)$
Vertices: N notes and one query (anchor) vertex; constraint set $S = \{q\}$, complement $\bar{S} = \mathcal{V}$.
2: **Build affinity** $A \in \mathbb{R}^{(N+1)\times(N+1)}$ from embeddings: $A_{ij} = 1 - \mathbf{z}_i^\top \mathbf{z}_j$, $A_{qi} = A_{iq} = \max(\mathbf{z}_q^\top \mathbf{z}_i, 0)$, $\mathrm{diag}(A) = 0$
▷ notes repel one another (dissimilarity); query attracts similar notes
3: **if** $\tau > 0$ **then**
4: zero query edges with $\mathbf{z}_q^\top \mathbf{z}_i < \tau$ (keep the single strongest if none clear τ)
5: **end if**
6: **CDS payoff:** $\alpha \leftarrow \lambda_{\max}(A) + \epsilon$, $M \leftarrow A - \alpha\, I_{\bar{S}}$
▷ $M_{qq} = 0$; $M_{ii} = -\alpha$ for notes $\Rightarrow$ support must contain S
7: **Solve** by replicator dynamics ($\mathbf{x}^{(0)} = \boldsymbol{\pi}$, shift $c = \max(0, -\min M) + \varepsilon$): $\mathbf{x}^{(t+1)} = \dfrac{\mathbf{x}^{(t)} \odot \big(M\mathbf{x}^{(t)} + c\big)}{\big(\mathbf{x}^{(t)}\big)^\top M \mathbf{x}^{(t)} + c}$, $\mathbf{x}^* = \lim_{t\to\infty} \mathbf{x}^{(t)} \in \Delta$
▷ local maximiser of $\mathbf{x}^\top M \mathbf{x}$ on the simplex (constrained dominant set)
8: **Select evidence:** $R \leftarrow \text{Top-}B\{\, x_i^* : i \in \mathcal{V},\ x_i^* > \theta \,\}$
▷ notes only, ranked by x_i^*; anchor excluded
9: **Generate:** $\hat{a} \leftarrow f_{\text{VLM}}(q, R)$
10: **return** $\hat{a}$

where $S = EE^\top$ (with E the stacked embeddings) is symmetric positive-semidefinite and $S_{ii} = 1$. Every method returns an evidence set R of size B (the budget; B=10 unless stated), which is passed to the reader.

B.1 Cosine Retrieval

The simplest baseline ranks notes by query relevance and skips all graph dynamics:

$$R = \text{Top-}B_{i \in \{1,\ldots,N\}}\ \mathrm{rel}_i. \tag{5}$$

This isolates whether the replicator dynamics of CDS add value over plain similarity ranking.

B.2 Random Selection (control)

A no-retrieval control that keeps the reader and budget fixed but removes any notion of relevance: R is drawn uniformly without replacement from the N notes, using a per-query seed $\sigma_q = \mathrm{md5}(q)$ for reproducibility.

B.3 MMR (Maximal Marginal Relevance)

MMR greedily trades off relevance against redundancy. Starting from $R \leftarrow \varnothing$, it first selects

$\arg\max_i \mathrm{rel}_i$ and then repeatedly adds

$$i^\star = \arg\max_{i \notin R} \Big[\lambda\, \mathrm{rel}_i - (1-\lambda) \max_{j \in R} S_{ij} \Big], \qquad R \leftarrow R \cup \{i^\star\}. \tag{6}$$

until $|R| = B$. The trade-off $\lambda \in [0,1]$ interpolates between pure relevance (λ=1) and pure diversity (λ=0); we sweep $\lambda \in \{0.3, 0.5, 0.7, 0.9\}$ and report the best.

B.4 DPP (Determinantal Point Process)

A k-DPP models (Kulesza and Taskar, 2012) a probability over subsets proportional to a subdeterminant of a kernel, jointly rewarding quality and diversity. We use the quality–similarity kernel

$$L = \mathrm{diag}(\mathbf{q})\, S\, \mathrm{diag}(\mathbf{q}), \qquad q_i = \exp\big(\theta\, \mathrm{rel}_i\big), \tag{7}$$

where $q_i > 0$ is the relevance-driven quality (θ a temperature, default 1) and S from Eq. (4) is PSD, so $L \succeq 0$. We seek the MAP configuration

$$R = \arg\max_{|R|=B} \log\det\big(L_R\big), \tag{8}$$

which is NP-hard, and approximate it with the fast greedy algorithm of (Chen et al., 2018). Maintaining incremental Cholesky factors $\mathbf{c}_i$ and residual gains d_i^2 (initialized $d_i^2 = L_{ii} = q_i^2$), each step selects

$$j = \arg\max_{i \notin R} d_i^2, \qquad R \leftarrow R \cup \{j\}, \tag{9}$$

and then updates every remaining $i \notin R$ via

$$e_i = \frac{L_{ji} - \langle \mathbf{c}_j, \mathbf{c}_i \rangle}{d_j}, \qquad \mathbf{c}_i \leftarrow [\,\mathbf{c}_i;\, e_i\,], \qquad d_i^2 \leftarrow d_i^2 - e_i^2. \tag{10}$$

with $d_j = \sqrt{d_j^2}$. The recursion terminates at $|R| = B$ or when no remaining note yields a positive gain ($d_i^2 \le 0$, i.e. the selected set already spans the relevant subspace).

B.5 PPR-as-Retriever

This baseline retrieves using the note *link graph* (the VLM-induced relatedness edges) rather than the embedding affinity, testing whether graph structure alone suffices. Let A^{lnk} be the symmetrized binary adjacency of the note links. We form the row-stochastic transition

$$P_{ij} = \begin{cases} A^{\mathrm{lnk}}_{ij} / \sum_k A^{\mathrm{lnk}}_{ik}, & \sum_k A^{\mathrm{lnk}}_{ik} > 0, \\ 1/N, & \text{otherwise (dangling node)}, \end{cases} \tag{11}$$

and a query-personalized teleport (seed) distribution from the relevances,

$$s_i = \frac{\exp(\mathrm{rel}_i / T)}{\sum_{k=1}^{N} \exp(\mathrm{rel}_k / T)}. \tag{12}$$

The PPR scores are the fixed point of the power iteration

$$\mathbf{r}^{(t+1)} = (1-\beta)\,\mathbf{s} + \beta\, P^\top \mathbf{r}^{(t)}, \qquad \mathbf{r}^{(t+1)} \leftarrow \mathbf{r}^{(t+1)} / \|\mathbf{r}^{(t+1)}\|_1, \tag{13}$$

with damping β=0.85 and temperature T=0.1, iterated to convergence ($\|\mathbf{r}^{(t+1)} - \mathbf{r}^{(t)}\|_\infty < 10^{-9}$). The evidence set is $R = \text{Top-}\, B_i\, r_i^\star$.

B.6 Single-VLM (no retrieval)

This baseline removes retrieval entirely and feeds document content directly to the reader, following the G2-Reader protocol. For query q we take its first M=5 candidate documents and extract, from their MinerU parse, the set of text chunks $\mathcal{C}_q$ (recursive splitting, chunk size 3000, overlap 300) and the set of visual elements $\mathcal{I}_q$ (figures/tables/charts with their crops). To meet the context budget we deterministically subsample

$$\tilde{\mathcal{C}}_q = \mathrm{Sample}\big(\mathcal{C}_q,\ \min(K_c, |\mathcal{C}_q|);\ \sigma_q\big), \qquad \tilde{\mathcal{I}}_q = \mathrm{Sample}\big(\mathcal{I}_q,\ \min(K_i, |\mathcal{I}_q|);\ \sigma_q\big), \tag{14}$$

with K_c=K_i=10 and a per-query seed $\sigma_q = \mathrm{md5}(\mathrm{seed}: q_{\mathrm{id}})$ so sampling is fixed and reproducible across runs; we report mean±std over seed $\in \{42, 43, 44\}$. The sampled chunks are concatenated into a context $c_q = \mathrm{concat}(\tilde{\mathcal{C}}_q)$, interleaved with $\tilde{\mathcal{I}}_q$ and the question in a chain-of-thought prompt (Appendix E), and the answer is decoded greedily,

$$\hat{a} = f_{\mathrm{VLM}}\big(q,\, c_q,\, \tilde{\mathcal{I}}_q\big), \qquad \tau = 0. \tag{15}$$

To avoid context overflow, the generation length is clamped to fit the server window $L_{\max}$, using a ~ 3 characters/token text estimate and a fixed 1500-token allowance per image:

Table 5: Hyperparameters used across all experiments. Values are the defaults used throughout; parentheses denote the ranges explored in ablations.

Hyperparameter	Value
Document processing & graph	
Candidate documents per query	5 (1 gold + 4 distractors)
Text chunk size / overlap	3000 / 300 chars
Image local-context window	$\pm$1000 words
Graph evolution iterations	3
Text encoder	
Model	nomic-embed-text-v1.5
Embedding dimension	768
Normalization	ℓ_2
CDS retrieval	
Evidence budget B (top-B)	10 ({3,5,10,15,20})
Note–note affinity	$1 - \cos$ (dissimilarity)
Diversification strength β	1.0 ({0.5,1,2,5})
Query sparsity τ	0 ({0,0.3,0.5,0.6})
Constraint margin ε	10^{-3}
Support threshold	10^{-6}
Initialization prior	uniform
Replicator max. iterations	10,000
Replicator tolerance δ	10^{-7}
Reader (generation)	
Readers	Qwen3-VL-32B; Qwen2.5-VL-7B; GLM-4.1V-9B
Decoding	greedy (temperature 0)
Max new tokens	1024
Max text chars per node	1500
vLLM serving (Kwon et al., 2023)	
Tensor-parallel size	4 (Qwen3-VL-32B)
Max model length	32768 (Qwen); 65536 (GLM)
GPU memory utilization	0.80
Data type	bfloat16
MM processor cache	disabled
Evaluation	
Judge model	GPT-4o-mini
Runs per configuration	3 (report mean $\pm$ std)
Baseline selectors (ablations)	
MMR trade-off λ	0.5 ({0.3,0.5,0.7,0.9})
DPP quality temperature θ	1.0
Prior temperature T (query-softmax/diffusion)	0.1 ({0.05,0.1,0.2})
PPR damping γ	0.85

$$T_{\text{out}} = \max\Big(256,\ \min\big(8192,\ L_{\max} - \tfrac{|c_q|}{3} - 1500\,|\tilde{\mathcal{I}}_q| - 1500\big)\Big) \tag{16}$$

C Hyperparameters

Table 5 lists all hyperparameters used in our experiments. Values are the defaults used throughout unless a specific ablation sweeps them (sweep ranges noted in parentheses).

D Additional Experiments

We report two further ablations on Qwen3-VL-32B: the replicator-dynamics initialization prior, and the effect of iterative graph evolution.

Initialization prior. CDS solves the quadratic program in Eq. (3) via the replicator dynamics (Eq. (17)).

$$x_i^{(t+1)} = x_i^{(t)} \frac{(M\mathbf{x}^{(t)})_i + c}{(\mathbf{x}^{(t)})^\top M\mathbf{x}^{(t)} + c}. \tag{17}$$

The vector $\mathbf{x}$ can be initialized in different ways. From an initial distribution $\mathbf{x}^{(0)}$, which only determines *which* local optimum the dynamics converge to, not the affinity matrix itself. We compare three choices. The first is a *uniform* prior, $x_i^{(0)} = 1/|\mathcal{U}|$, which is our default. The second is a *query-softmax* prior that seeds each document node

Table 6: Initialization-prior ablation (CDS, Qwen3-VL-32B). Accuracy (%), mean ± std over 3 runs.

Prior	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	**Avg.**	MMLongBench
Uniform (default)	78.8 ± 0.2	71.0 ± 0.6	65.6 ± 0.4	57.4 ± 0.4	62.2 ± 0.4	67.0 ± 0.2	45.0 ± 0.4
Query-softmax (T=0.05)	78.4 ± 0.1	70.0 ± 0.3	66.0 ± 0.0	58.8 ± 0.3	61.6 ± 0.8	67.0 ± 0.1	44.8 ± 0.2
Query-softmax (T=0.1)	78.8 ± 0.3	71.6 ± 0.4	64.6 ± 0.3	59.2 ± 0.5	61.4 ± 0.4	67.1 ± 0.2	44.9 ± 0.1
Query-softmax (T=0.2)	78.8 ± 0.6	71.7 ± 0.3	65.2 ± 0.2	59.1 ± 0.8	61.6 ± 0.2	$\mathbf{67.3 \pm 0.3}$	45.0 ± 0.3
Graph-diffusion	78.0 ± 0.4	71.4 ± 0.2	65.6 ± 0.6	59.1 ± 0.1	61.6 ± 0.1	67.2 ± 0.2	44.4 ± 0.3

Table 7: Graph-evolution ablation (CDS, Qwen3-VL-32B). Accuracy (%) at each evolution iteration, computed on queries answerable in all iterations.

Iter.	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	Avg.
0 (initial)	80.2	70.4	67.0	57.8	60.3	67.1
1	79.5	72.0	64.3	58.3	62.5	67.3
2	77.6	72.3	64.1	57.0	63.2	66.9
3 (final)	78.1	71.7	65.4	58.5	61.2	67.0

by its temperature-scaled relevance to the query,

$$p_i = \frac{\exp(\cos(v_i, q)/T)}{\sum_j \exp(\cos(v_j, q)/T)},$$

which we evaluate at $T \in \{0.05, 0.1, 0.2\}$. The third is a *graph-diffusion* prior that propagates the query-softmax seed $\mathbf{p}$ over the note link graph via Personalized PageRank,

$$\mathbf{r}^{(t+1)} = (1-\gamma)\,\mathbf{p} + \gamma\, P^\top \mathbf{r}^{(t)},$$

where P is the row-stochastic transition matrix of the link graph and γ the damping factor; the converged $\mathbf{r}$ is used as $\mathbf{x}^{(0)}$. We stress that the graph-diffusion prior uses PageRank *only to seed* the dynamics; the evidence is still selected by CDS. This is distinct from the PPR *retriever* in Table 4, where PageRank *replaces* CDS and notes are ranked by their PageRank scores directly. The contrast is informative: as a standalone retriever PPR is weak (54.4 average, Table 4), yet as an initialization it is indistinguishable from any other prior (67.2 average, Table 6). More generally, all priors fall within 0.3 points on the VisDoMBench average (67.0–67.3) and overlap within standard deviation on every dataset (Table 6). CDS is thus robust to initialization—its quality is driven by the affinity, not the seed or the link-graph structure—so we adopt the parameter-free uniform prior by default.

Graph evolution. The memory graph undergoes refinement over up to three evolution iterations, during which the VLM updates links and summaries. Table 7 presents accuracy at each iteration for queries answerable in all iterations, ensuring that the observed trend reflects the effect of evolution alone. The overall impact is minimal; the VisDoMBench average changes from 67.1 to 67.0 across iterations, with only minor, dataset-dependent fluctuations. This suggests that evolution primarily enhances robustness rather than improving accuracy. The final graph is used in all subsequent experiments.

D.1 Query-Row Sparsification

The query vertex connects to every document node weighted by non-negative cosine relevance. The threshold τ optionally prunes weak query–document edges (those with $\cos(q, v_i) < \tau$), concentrating the query anchor on its most relevant neighbours. Table 9 sweeps $\tau \in \{0, 0.3, 0.5, 0.6\}$. Performance is flat for mild sparsification: τ=0 and τ=0.3 are statistically indistinguishable on the VisDoMBench average (66.8 vs. 67.2), with τ=0.3 marginally ahead, indicating that removing the weakest query edges neither helps nor hurts meaningfully. Beyond this point accuracy degrades: τ=0.5 drops to 66.6 and τ=0.6 to 64.5, because an aggressive threshold starts discarding genuinely relevant evidence—the decline is sharpest on SciGraphQA (59.2 → 53.7) and SlideVQA, whose answers draw on more, individually weaker query–node edges. Since the mild-sparsity gain is within noise and adds a hyperparameter, we use no sparsification (τ=0) by default; the result confirms CDS is robust to this choice over a wide range.

D.2 Sign of the Inter-Node Coupling

CDS encodes diversity through the note–note block of the affinity, set to the *dissimilarity* $1 - \cos(v_i, v_j)$ so that similar evidence repels. To test whether this

Table 8: Diversification-block ablation (CDS, Qwen3-VL-32B). The note–note affinity is set to no coupling, similarity, or dissimilarity (ours). Accuracy (%), mean±std over 3 runs.

note–note	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	**Avg.**	MMLongBench
sim ($+\cos$)	75.9 ± 0.3	64.0 ± 0.5	58.9 ± 0.4	56.7 ± 0.3	59.6 ± 0.2	63.0 ± 0.3	45.4 ± 0.2
dissim ($1-\cos$, **CDS**)	$\mathbf{78.8 \pm 0.2}$	$\mathbf{71.0 \pm 0.6}$	$\mathbf{65.6 \pm 0.4}$	$\mathbf{57.4 \pm 0.4}$	$\mathbf{62.2 \pm 0.4}$	$\mathbf{67.0 \pm 0.2}$	45.0 ± 0.4

Table 9: Query-row sparsification ablation (CDS, Qwen3-VL-32B). Accuracy (%), mean±std over 3 runs.

Config	SPIQA	FetaTab	PaperTab	SciGraphQA	SlideVQA	**Avg.**	MMLongBench
CDS (τ=0)	78.1 ± 0.2	71.1 ± 0.4	64.5 ± 0.4	58.3 ± 0.4	61.8 ± 0.4	66.8 ± 0.1	45.1 ± 0.4
CDS (τ=0.3)	78.5 ± 0.2	71.2 ± 0.8	65.1 ± 0.6	59.2 ± 0.6	61.9 ± 0.3	$\mathbf{67.2 \pm 0.1}$	44.7 ± 0.3
CDS (τ=0.5)	77.3 ± 0.3	71.2 ± 0.5	64.0 ± 0.8	58.6 ± 0.1	62.1 ± 0.1	66.6 ± 0.2	44.4 ± 0.1
CDS (τ=0.6)	76.6 ± 0.4	67.3 ± 0.2	65.4 ± 0.8	53.7 ± 0.9	59.6 ± 0.4	64.5 ± 0.1	43.5 ± 0.3

design choice—rather than merely the presence of inter-node edges—is responsible for the gains, we replace it with the opposite sign, a *similarity* coupling $\max(\cos(v_i, v_j), 0)$ that instead rewards selecting mutually similar notes. Table 8 shows the sign is decisive: flipping the block to similarity drops the VisDoMBench average from 67.0 to 63.0, a 4.0-point loss, and degrades every dataset (e.g., FetaTab $71.0 \rightarrow 64.0$, PaperTab $65.6 \rightarrow 58.9$). The similarity coupling drives the dominant set toward a tight cluster of near-duplicate notes, wasting the budget on redundant evidence, whereas the dissimilarity coupling spreads selection across complementary evidence while the query anchor preserves relevance. This confirms that the diversification term must *repel* similar notes: it is the source of CDS's advantage, not an incidental component.

D.3 Evidence budget:

We vary the number of evidence nodes B passed to the reader from 3 to 20 (Table 10). Accuracy increases monotonically with B: it rises steeply up to $B = 10$ ($54.0 \rightarrow 67.0$ on the VisDoMBench average), gains a further $+2.6$ points at $B = 15$ (69.6), and then plateaus on VisDoMBench at $B = 20$ (69.6). On MMLongBench-Doc the upward trend persists slightly longer ($45.0 \rightarrow 46.4 \rightarrow 47.7$), as the single long document can absorb more evidence before saturating. The run-to-run standard deviation is small throughout (≤ 0.9 per dataset), so these differences are well outside noise. We use $B = 10$ in all main experiments to match the evidence budget of prior work (Du et al., 2026) for a controlled comparison; the practical optimum on VisDoMBench is $B = 15$. Crucially, accuracy is stable across a wide range ($B \in [10, 20]$ varies by ≤ 2.6 points), indicating that CDS is robust to the evidence-set size and does not require careful budget tuning.

Diversification strength. The note–note block of our affinity, $\beta(1 - \cos)$, controls how strongly the retriever favors mutually dissimilar (non-redundant) evidence relative to raw query relevance: $\beta \rightarrow 0$ removes the inter-note coupling and reduces CDS to relevance-only selection, $\beta = 1$ is the default, and larger β amplifies the repulsion between selected notes. Table 11 sweeps this knob on the full VisDoMBench. We find a broad, flat optimum for *light* diversification—$\beta = 0.5$ is marginally best (67.4 avg) and the default $\beta = 1$ is statistically indistinguishable (67.0)—indicating that a modest diversity pressure is beneficial and that CDS is insensitive to the exact value in this regime. Pushing diversification further, however, degrades accuracy sharply and monotonically ($\beta = 2 \rightarrow 61.6$, $\beta = 5 \rightarrow 37.7$), as the selected set drifts toward dissimilar but off-topic evidence (the collapse is most severe on SciGraphQA, $58.8 \rightarrow 14.7$). In other words, diversity helps only up to the point where it begins to override relevance.

E Prompts

We list every prompt used in our pipeline and evaluation. Placeholders in braces (e.g. `{question}`) are filled at run time.

Table 10: Evidence-budget (CDS, Qwen3-VL-32B). Accuracy (%), mean $\pm$ std over 3 runs. B is the number of evidence nodes given to the reader.

B	FetaTab	PaperTab	SciGraphQA	SlideVQA	SPIQA	Average	MMLongBench
3	54.8 ± 0.2	50.6 ± 0.2	45.0 ± 0.4	50.5 ± 0.5	69.1 ± 0.3	54.0 ± 0.2	40.2 ± 0.1
5	65.7 ± 0.5	57.2 ± 0.6	48.3 ± 0.6	58.6 ± 0.3	72.4 ± 0.3	60.4 ± 0.2	42.1 ± 0.2
10	71.0 ± 0.6	65.6 ± 0.4	57.4 ± 0.4	62.2 ± 0.4	78.8 ± 0.2	67.0 ± 0.2	45.0 ± 0.4
15	74.5 ± 0.2	67.6 ± 0.5	61.5 ± 0.9	65.3 ± 0.3	79.3 ± 0.4	$\mathbf{69.6 \pm 0.1}$	46.4 ± 0.6
20	73.6 ± 0.3	66.4 ± 0.9	61.6 ± 0.1	66.0 ± 0.1	80.2 ± 0.3	$\mathbf{69.6 \pm 0.2}$	47.7 ± 0.1

Table 11: Diversification-strength. The note–note affinity block is scaled by β (β=1 recovers the default CDS). Accuracy (%), mean $\pm$ std over 3 runs; Qwen3-VL-32B reader.

β	FetaTab	PaperTab	SciGraphQA	SlideVQA	SPIQA	**Avg.**	MMLongBench
0.5	$\mathbf{72.2 \pm 0.2}$	65.0 ± 0.2	$\mathbf{58.8 \pm 0.5}$	$\mathbf{62.8 \pm 0.6}$	78.5 ± 0.0	$\mathbf{67.4 \pm 0.2}$	$\mathbf{45.4 \pm 0.1}$
1.0 (CDS)	71.0 ± 0.6	$\mathbf{65.6 \pm 0.4}$	57.4 ± 0.4	62.2 ± 0.4	$\mathbf{78.8 \pm 0.2}$	67.0 ± 0.2	45.0 ± 0.4
2.0	65.3 ± 1.0	59.4 ± 0.8	50.0 ± 1.2	57.1 ± 0.4	76.2 ± 0.1	61.6 ± 0.1	42.2 ± 0.1
5.0	42.6 ± 0.2	46.1 ± 0.3	14.7 ± 0.1	34.0 ± 0.1	51.1 ± 0.4	37.7 ± 0.1	35.0 ± 0.1

Prompt: Text Element Analysis (memory-note construction)

```
Generate a structured analysis of the following content by:
1. Identifying the most salient keywords (focus on nouns, verbs, and key concepts)
2. Extracting core themes, concepts and arguments
3. Creating relevant categorical tags

Format the response as a JSON object:
{
  "keywords": [  // specific, distinct keywords, ordered most->least important;
                 // at least three, avoid redundancy
  ],
  "summary":     // one sentence: main topic/domain + key points; concise
  ,
  "tags": [      // broad categories/themes (domain, format, type); >=3, non-redundant
  ]
}

Content for analysis: {content}
```

Prompt: Visual Element Analysis (figures/tables, from MinerU crops)

```
Generate a structured analysis of the visual elements in the provided image.
You are also given the image's surrounding text context (before/after) and its caption.

Instructions:
1) Use the context ONLY to aid understanding of the image's role; do not quote or
   rely on it unless it aligns with what is visible or stated in the caption. Base
   the summary primarily on the visual evidence and the caption.
2) Keywords MUST include exact in-image terms: labels, legends, axis titles,
   category names, and domain-specific terms; preserve their exact wording.
3) If the caption has an index (e.g., "Figure 1", "Table 2"), begin the summary by
   formalizing it ("Figure X -- ...", "Table Y -- ...") then describe concisely.

Format the response as a JSON object:
{
  "keywords": [ // exact in-image labels/legends/axis titles/terms; >=3, non-redundant ],
  "summary":    // start with "Figure X -- ..."/"Table Y -- ..." if indexed;
                // describe the visual elements specifically
  ,
  "tags": [ // broad domain/format/type categories; >=3, non-redundant ]
}

(Two few-shot examples are provided: a labeled map and an ablation table.)
```

```
This is context: {context} and caption: {caption}
```

Prompt: Memory Evolution (graph link induction + note refinement)

```
You are an AI memory-evolution agent managing a knowledge base. Given a memory
note (content, summary, keywords) and its neighboring notes, decide its evolution.

The memory note content: {content}
Summary: {context}
Keywords: {keywords}
The {neighbor_number} neighboring notes:
{neighbors}

Determine:
1. Which neighboring notes should be linked to this note?
2. Should this note's summary/keywords be updated given those relationships?
3. If so, what are the new summary and keywords?

Connect two notes ONLY for a specific logical relationship: direct reference,
causal, part-whole, conceptual elaboration, temporal sequence,
contrastive/comparative, hierarchical, or contextual dependency. DO NOT connect
notes that merely share keywords, domain, surface similarity, document adjacency,
or generic concepts. Order connections by relevance (most relevant first).

When updating: make the note more precise and distinctive; describe ONLY this
note's content (no comparative language about other notes); prefer specific,
distinctive keywords; keep the summary self-contained and <= 30 words. If the note
already captures its unique content, do not update.

Return JSON:
{
  "suggested_connections": ["neighbor_memory_ids"],
  "should_update": true or false,
  "new_summary": "new summary",
  "new_keywords": ["keyword_1", ..., "keyword_n"]
}
```

Prompt: Answer Generation (reader VLM, CDS pipeline)

```
[System]
You are answering a question using the provided document context. Each context
item comes from a single document. Some items are text passages; others are
figures or tables sent as images (each accompanied by a short caption/summary).

Rules:
 1. Use ONLY the provided context. Do not invent facts.
 2. Quote table numbers, figure numbers, named entities, and numeric values exactly.
 3. If the answer is not in the context, reply exactly:
    "Not found in the provided context."
 4. Answer concisely. No preamble.

[User]
Question:
{question}

Context:
--- Note 1  (id=.., weight=.., type=text|image) ---
{text passage  OR  image caption/summary + <image>}
--- Note 2  ... ---
...
Answer:
```

Prompt: Single-VLM Baseline (no retrieval, chain-of-thought)

```
Please read the following text and the attached images and answer the question below.

<text>
{context}
```

```
</text>

What is the correct answer to this question: {question}

Format your response as follows:
"<reason>detailed reason for your answer here</reason>
 <answer>the correct answer here</answer>".
Make sure your answer is comprehensive and covers all important information related
to the question. If directly relevant information is not provided, give your best
answer based on the available context (without mentioning that you lack information).
```

Prompt: Answer Evaluation (LLM-as-judge, GPT-4o-mini)

```
You are an expert evaluator assessing answers from a RAG system.
Task: judge whether the generated answer correctly responds to the question,
given the expected answer.

Question:        {question}
Expected Answer: {gold_answers}
Generated Answer:{assistant_answer}

Accuracy (0 or 1): 1 if factually correct and aligned with the expected answer; 0 otherwise.

Apply uniformly:
- Judge factual correctness, not style/formatting. Ignore capitalization,
  punctuation, whitespace, quotes, articles, markup, abbreviation vs full form,
  synonyms, singular/plural.
- Partial-match: correct if the answer contains ALL key facts of the expected
  answer and adds only non-contradictory context.
- Concise-match: a terser label/phrase/number that conveys the same key assertion
  is correct (unless the gold has multiple distinct claims the prediction omits).
- Abstractive-elaboration: paraphrase preserving subject-predicate-object is correct.
- Role-preservation: if entities are kept but the relation direction is reversed,
  mark incorrect.
- Numeric: equivalent up to unit/format (0.82 == 82%); signed deltas must match.
- List: correct if all key items present (any order).
- Multi-gold: correct if it matches ANY one acceptable answer.
- Entity: must refer to the same subject as the question/expected answer.

"Not answerable": correct only if BOTH say not answerable; otherwise incorrect.

Output (JSON only):
{
  "accuracy": 0 or 1,
  "reasoning": "brief explanation"
}
```