MAB-DQA: Addressing Query Aspect Importance in Document Question Answering with Multi-Armed Bandits

Anonymous ACL submission

Abstract

Document Question Answering (DQA) involves generating answers from a document based on a user's query, representing a key task in document understanding. This task requires interpreting visual layouts, which has prompted recent studies to adopt multimodal Retrieval-Augmented Generation (RAG) that processes page images for answer generation. However, in multimodal RAG, visual DQA struggles to utilize a large number of images effectively, as the retrieval stage often retains only a few candidate pages (e.g., Top-4), causing informative but less visually salient content to be overlooked in favor of common yet low-information pages. To address this issue, we propose a Multi-Armed Bandit–based DQA framework (MAB-DQA) to explicitly model the varying importance of multiple implicit aspects in a query. Specifically, MAB-DQA decomposes a query into aspect-aware subqueries and retrieves an aspect-specific candidate set for each. It treats each subquery as an arm and uses preliminary reasoning results from a small number of representative pages as reward signals to estimate aspect utility. Guided by an exploration–exploitation policy, MAB-DQA dynamically reallocates retrieval budgets toward high-value aspects. With the most informative pages and their correlations, MAB-DQA generates the expected results. On four benchmarks, MAB-DQA shows an average improvement of 5%-18% over the state-of-the-art method, consistently enhancing document understanding.

1 Introduction

Document Question Answering requires AI to answer user questions about given documents (Tanaka et al., 2023; Lee et al., 2025). DQA proves valuable in real-world applications such as financial forms, medical report interpretation, and academic literature assistance (Ye et al., 2024; Huo et al., 2025). Accomplishing this task relies on visual-language models and retrieval-augmented generation (RAG) to understand long documents with complex layouts (Zhou et al., 2024). Existing advanced approaches, such as Colpali (Faysse et al., 2024) and MoloRAG (Wu et al., 2025), adopt a vision-query late interaction (LI) paradigm for retrieval. They compute the dot product between each query token embedding and all document image patch embeddings, retaining the maximum similarity score between each query token and the most relevant image patch (Santhanam et al., 2022). This operator preserves fine-grained token-patch interactions but merely performs a me-

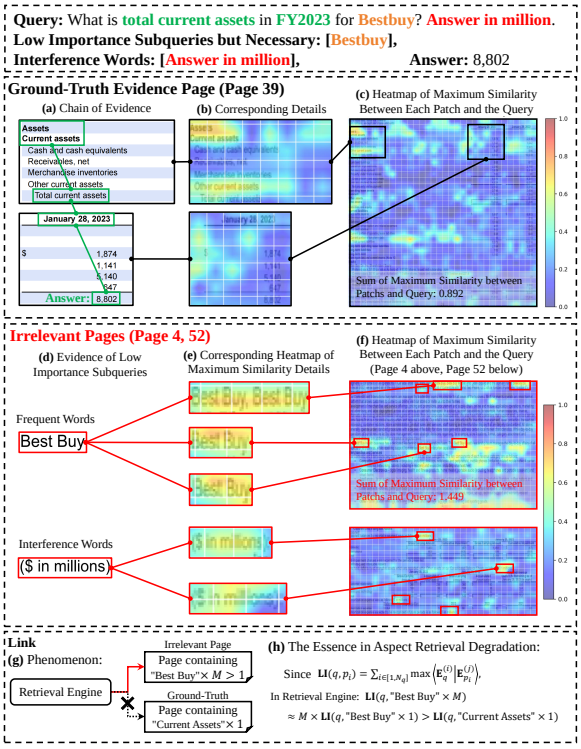


Figure 1: Aspect Retrieval Degradation in DQA. (a) Ground-truth evidence and answer. (b-c) Similarity heatmap between query and patches. (d) Aspects of low importance in the question. (e-f) High spurious similarity on irrelevant pages due to frequent terms. (g-h) Aspects of low importance' aggregated score can exceed crucial evidence.

chanical "max-pooling + summation" operation. As a result, it cannot mimic the human ability to weigh the importance of multiple aspects in a query and selectively focus on them, which limits performance on complex multi-aspect questions.

Fig. 1 illustrates an example from MMLongBench using a financial report. Fig. 1(a)-(c) show the maximum similarity scores obtained by Colpali on the ground-truth evidence page (Page 39). The green box in Fig. 1(a) highlights the correct retrieval chain, while Fig. 1(b)-(c) display interpretability heatmaps. The color in the heatmaps indicates the maximum similarity (dot product) between each image patch embedding and the query, with red representing high similarity and blue indicating irrelevance. Based on the LI paradigm, Colpali assigns Page 39 a score of $\mathbf{LI}(\text{Query}, \text{Page } 39) = 0.892$, as shown in Fig. 1(c). However, as seen in Fig. 1(d), pages unrelated to the answer, such as those containing high-frequency but low-importance words like "Best Buy" or irrelevant terms like "million", can also receive high scores (Fig. 1(e)-(f)), such as $\mathbf{LI}(\text{Query}, \text{Page } 4) = 1.449$.

To address this limitation, we propose the MAB-DQA, a multi-armed bandit–guided DQA framework that explicitly models and exploits the varying importance of multiple implicit aspects in a query. Our core idea is to dynamically decompose a query into aspect-aware subqueries, treat each subquery as an arm in a multi-armed bandit, and use preliminary reasoning feedback as a reward signal to estimate the utility of each aspect. Guided by an exploration–exploitation policy, MAB-DQA reallocates retrieval attention and budget toward high-value aspects, thereby retrieving a more informative and balanced set of evidence pages. The final answer is generated by reasoning over the retrieved pages and their correlations. MAB-DQA outperforms existing methods on four benchmarks, with a 10.38% average gain in answer accuracy over the strongest baseline, and achieves new state-of-the-art retrieval performance. Our contributions are:

- We propose MAB-DQA, a novel multi-armed bandit–based DQA framework that advances multi-aspect query by dynamically discerning the importance of query aspects, thereby guiding retrieval to prioritize evidence containing critical information.
- We design a bandit-guided retrieval strategy that treats each aspect-aware subquery as an arm and uses reasoning feedback as rewards, enabling adaptive exploration–exploitation in page retrieval.
- We conduct extensive experiments on four benchmarks, demonstrating that MAB-DQA consistently enhances document understanding performance and outperforms existing methods significantly.

2 Related Work

Document Question Answering. DQA serves as a key task for evaluating models' document comprehension capabilities (Cao et al., 2025; Zhu et al., 2025). With the advancement of large language models (LLMs) and VLMs, research focus has shifted from short-form unimodal to long-form multimodal understanding (Liu et al., 2025). LLMs such as GPT-4o (OpenAI et al., 2024) and Qwen-VL (Bai et al., 2023) support long document inputs by extending context windows, but suffer from information dilution (Ye et al., 2025; Peng et al., 2025; Cheng et al., 2025; Jiang et al., 2025). To address this, RAG has become a mainstream paradigm, improving generation quality by building external knowledge indexes.

Retrieval-Augmented Generation (RAG). A common strategy in RAG is the "chunk-vectorize" approach, which uses models like BERT (Devlin et al., 2019), Colapli (Faysse et al., 2024), and ColBERT (Santhanam et al., 2022) to generate embeddings, and leverages LLMs for optimization. For instance, MDocAgent (Han et al., 2025) employs multi-agent collaboration for DQA (Zhang et al., 2025a). However, chunk-based RAG fails to represent complex relationships.

Graph-based RAG and Query Decomposition. Graph-based RAG (Edge et al., 2025) has gained attention for enhancing reasoning through knowledge graphs and relational paths (Zhang et al., 2025c). Simultaneously, research on query decomposition and query rewriting has also progressed (Zhang et al., 2023). Some studies advance from the perspectives of GraphRAG or query decomposition. For example, Cog-RAG (Hu et al., 2025) constructs hypergraph indexes to improve topic consistency; Self-RAG (Asai et al., 2023) trains models to autonomously evaluate retrieval and query quality; RA-DIT (Lin et al., 2023) jointly optimizes the retriever and generator; DRAG (Zhang et al., 2025b) deconstructs complex query risks. MBA-

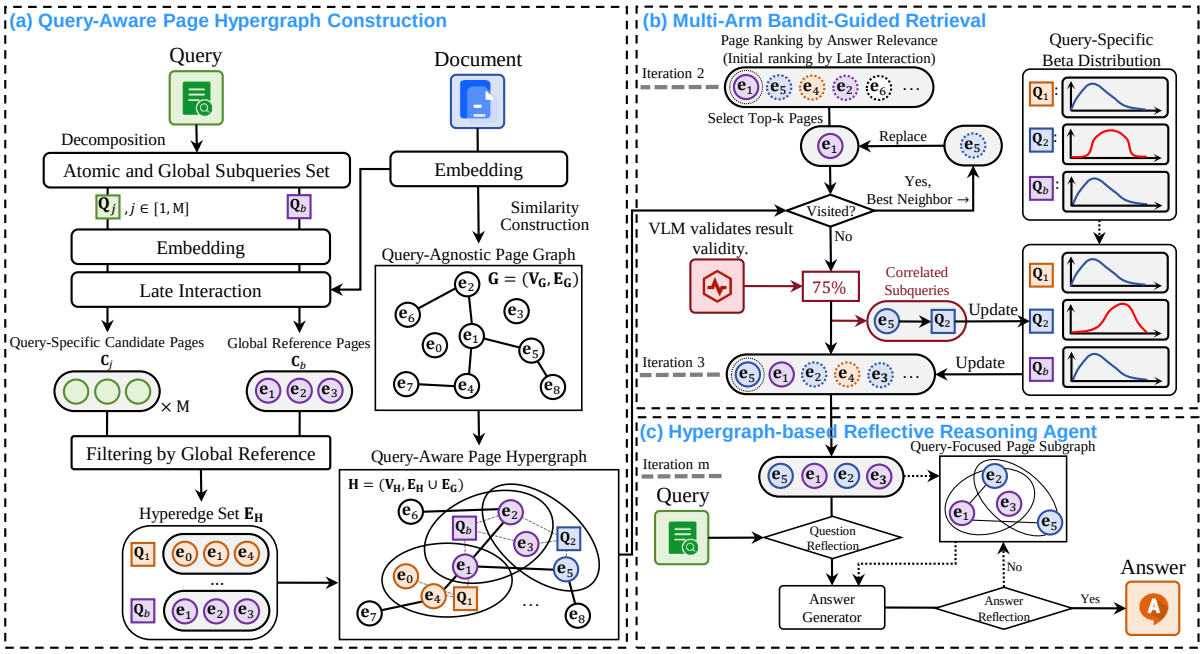


Figure 2: The overview of our proposed framework MAB-DQA. (a) Decompose a query into aspect-aware subqueries and model the relationships using hyperedges to construct a hypergraph; (b) For the hypergraph's jumping retrieval pages, model each subquery as an arm and use the Bandit-based method to select the next page based on VLM feedback; (c) Derive the final answer through reflective reasoning.

RAG (Tang et al., 2025) also employs MAB but differs from MAB-DQA, focusing on unimodal contexts, where MAB selects retrieval strategies for cost control. MoloRAG (Wu et al., 2025) is a state-of-the-art multimodal method that enhances DQA by constructing a graph and applying a VLM for graph traversal. It advances retrieval through modeling page relationships, yet operates under a fixed retrieval budget. A major drawback is its uniform attention to all query aspects, which can overlook less salient but informative content. However, existing methods are unable to replicate the ability of humans to evaluate aspects of a query while visually reading documents.

3 Methodology

We propose a Multi-Armed Bandit–based Document Question Answering framework (MAB-DQA). As illustrated in Fig. 2, the key to our approach is the explicit modeling of aspect-aware subqueries and the dynamic allocation of retrieval effort among them. To achieve this, we first decompose the query and represent the document as a hypergraph (Sec. 3.2, Fig. 2(a)), where hyperedges represent an aspect-specific candidate set relevant to each subquery. Subsequently, a MAB mechanism treats each subquery as an arm and uses preliminary Vision-Language Model (VLM) feedback as rewards to guide an exploration–exploitation policy over the hypergraph (Sec. 3.3, Fig. 2(b)). Finally, the answer is obtained through multi-stage verification via a Hypergraph-based Reflective Reasoning Agent (Sec. 3.4, Fig. 2(c)).

3.1 Preliminary: Late Interaction Retrieval

Given a query q and N document pages p_i, their multi-vector representations in $\mathbb{R}^D$ are $\mathbf{E_q} \in \mathbb{R}^{N_q \times D}$ and, per page, $\mathbf{E_p} \in \mathbb{R}^{N_p \times D}$, with N_q and N_p as vector counts. The Late Interaction (LI) (Santhanam et al., 2022) operator $\mathbf{LI}(q, p)$ computes for each query vector $\mathbf{E_q}^{(k)}$ the maximum dot product with all page vectors $\mathbf{E_p}^{(l)}$, denoted $\max\langle\cdot|\cdot\rangle$, and sums these:

$$\mathbf{LI}(q, p_i) = \sum_{k=1}^{N_q} \max_{l=1}^{N_p} \langle \mathbf{E_q}^{(k)} | \mathbf{E_{p_i}}^{(l)} \rangle. \quad (1)$$

A limitation of this formulation is that it assigns equal weight to every query vector $\mathbf{E}_q^{(k)}$, which may not reflect the varying importance of different semantic aspects in the query.

3.2 Query-Aware Page Hypergraph

To capture the multi-aspect nature of complex queries, we first build a Query-Agnostic Page Graph $\mathbf{G}(\mathbf{V_G}, \mathbf{E_G})$ to represent the relationships

between pages. Each node $p_i \in \mathbf{V_G}$ corresponds to a page. An edge $\mathbf{E_G}$ is added between nodes $\{p_i, p_j\}, i \neq j$ if the similarity between the two pages exceeds the threshold $\theta_{\mathbf{G}}$, expressed as:

$$\mathbf{E_G} = \{\{p_i, p_j\} \mid \text{sim}\langle \mathbf{E}_{p_i}, \mathbf{E}_{p_j} \rangle \geq \theta_{\mathbf{G}}\}, \quad (2)$$

where $\text{sim}\langle \cdot, \cdot \rangle$ denotes the inner product. Based on the graph $\mathbf{G}$, a VLM rewrites the original query q and decomposes it into a set of aspect-aware subqueries $\mathcal{E}_q = \{q_1, q_2, ..., q_M\}$. We then define the Atom-Integral Subqueries Set as:

$$\mathbf{Q} = \{\{\hat{q}\} \mid \hat{q} \in \mathcal{E}_q\} \cup \{\mathcal{E}_q\}, \quad (3)$$

which includes both fine-grained (atomic) subqueries and the global query $\mathcal{E}_q$ (denoted as $\mathcal{E}_{M+1}$). We select the top-$\theta_{\mathbf{H}}$ pages with the highest $\mathbf{LI}(\mathbf{Q}_j, p_i)$ scores to form the Query-Specific Candidate Pages set $\mathbf{C}_j$. We then select pages not in the global reference candidate pages set $\mathbf{C}_b$, $b = M + 1$, or those in both $\mathbf{C}_j$ and $\mathbf{C}_b$, with better ranking under $\mathbf{Q}_j$, to construct a hyperedge:

$$\hat{\mathbf{E}}_j = \{p \in \mathbf{C}_j \mid p \notin \mathbf{C}_b \vee \text{rank}(\mathbf{LI}(\mathbf{Q}_j, p)) \leq \text{rank}(\mathbf{LI}(\mathbf{Q}_b, p))\}, \quad (4)$$

where $\text{rank}(\mathbf{LI}(\cdot, p))$ denotes the descending-order position of page p under the corresponding query based on its $\mathbf{LI}$ score. Finally, the Query-Aware Page Hypergraph is defined as:

$$\mathbf{H}(\mathbf{V_H}, \mathbf{E_H} \cup \mathbf{E_G}) = (\mathbf{V_G}, \{\hat{\mathbf{E}}_j\}_{j=1}^{M+1} \cup \mathbf{E_G}). \quad (5)$$

3.3 Multi-Arm Bandit-Guided Retrieval

We frame the retrieval process over $\mathbf{H}$ as a combinatorial multi-armed bandit problem. Each subquery $\mathbf{Q}_j$ is treated as an arm, and the goal is to sequentially decide which arms (subqueries) to "pull" in order to retrieve the most informative pages.

Reward Model and Thompson Sampling. When the VLM inspects a retrieved page p_i, it produces a relevance score $s_i^{\text{vlm}} \in [0, 1]$ indicating whether the page contains useful evidence for the query (Wu et al., 2025). This score serves as the reward signal for the bandit.

Each arm $\mathbf{Q}_j$ maintains a Beta distribution $\text{Beta}(\alpha_j, \beta_j)$ to model its reward probability. Initially, $\alpha_j = \beta_j = 1$ (uniform prior). The probability density function is:

$$f(x; \alpha_j, \beta_j) = \frac{1}{\text{Beta}(\alpha_j, \beta_j)} x^{\alpha_j - 1}(1 - x)^{\beta_j - 1} \quad (6)$$

$$\text{Beta}(\alpha_j, \beta_j) = \int_0^1 t^{\alpha_j - 1}(1 - t)^{\beta_j - 1} dt = \frac{\Gamma(\alpha_j)\Gamma(\beta_j)}{\Gamma(\alpha_j + \beta_j)}, \quad (7)$$

where $\Gamma(\cdot)$ is the Gamma function, and $x \in [0, 1]$. The initial parameters are set as $\alpha_j = 1$ and $\beta_j = 1$. We employ Thompson sampling (Agrawal and Goyal, 2012; Chapelle and Li, 2011) to balance exploration and exploitation. In each retrieval step, a sample is drawn from each arm's Beta distribution, and the arm with the highest sample value is selected to guide the subsequent page retrieval.

Scoring and Node Expansion. During a jumping retrieval of $\mathbf{H}$, each page node p_i receives a composite score:

$$\text{score}(p_i) = (1 - \alpha) \max_{j \in [1, M+1]} \mathbf{LI}(\mathbf{Q}_j, p_i) + \alpha s_i^{\text{vlm}} + \beta[(1 - \lambda)h_i + \lambda \bar{s}_i^{\text{cb}}], \quad (8)$$

where h_i is the degree of page p_i in $\mathbf{H}$. $\bar{s}_i^{\text{cb}}$ denotes the Thompson Sampling confidence score of the associated subqueries, calculated as:

$$\bar{s}_i^{\text{cb}} = \frac{1}{|\hat{\mathbf{Q}}_i|} \sum_{\mathbf{Q}_j \in \hat{\mathbf{Q}}_i} \mathbb{E}[\text{Beta}(\alpha_j, \beta_j)], \quad (9)$$

where $\hat{\mathbf{Q}}_i$ denotes all subqueries linked to page p_i via hyperedges in $\mathbf{H}$. Generally, h_i emphasizes the page's own contribution, whereas $\bar{s}_i^{\text{cb}}$ focuses more on the contribution of subqueries. A larger $\alpha \in [0, 1]$ gives greater weight to the VLM evaluation results. The β is used to adjust the proportion between hyperparameters. The $\lambda \in [0, 1]$ balances the page degree and arm confidence; Larger λ values favor retrieval toward the highest overall subquery combinations expectation across a subquery rather than a single page.

The top-k nodes with the highest scores are expanded in each round. After the VLM evaluates a retrieved page p_i and returns s_i^{vlm}, all arms $\mathbf{Q}_j$ associated with p_i update their parameters:

$$\forall \mathbf{Q}_j \in \hat{\mathbf{Q}}_i, (\alpha_j, \beta_j) \leftarrow (\alpha_j + s_i^{\text{vlm}}, \beta_j + 1 - s_i^{\text{vlm}}). \quad (10)$$

This update increases α_j if the page is relevant (high VLM score) and increases β_j otherwise, thereby refining the reward estimate for each aspect-specific subquery.

3.4 Hypergraph-based Reflective Reasoning Agent

After the MAB-guided retrieval obtains a set of relevant pages from the hypergraph **H**, the Hypergraph-based Reflective Reasoning Agent (HRRA) synthesizes the final answer through a multi-stage verification process. Employing a "initial response–verification–optimization" pipeline, HRRA first generates an initial answer using the retrieved evidence. If inconsistencies or gaps are detected, the agent re-enters the hypergraph and constructs a Query-Focused Page Subgraph in a reflective loop.

4 Experiment

4.1 Implementation Details

The main experiments employ the QWen-2.5VL-7B-Instruct model (Bai et al., 2023) for both retrieval-augmented generation and question answering. Additional vision-language models are also included in the ablation studies. For generating embeddings, the ColPali (Faysse et al., 2024) model is selected as a vision-language embedding model specifically optimized for documents. All embedding generation, information retrieval, and question answering processes are conducted on a system equipped with four NVIDIA V100 GPUs.

Table 1: Statistics of Datasets. "Issue" refers to the number of samples in the labeled dataset that the Colpali model may be ignoring regarding key query conditions.

Dataset	Document	Question	Issue
MMLongBench	134	1073	**212**(19.8%)
LongDocURL	396	2325	**647**(27.8%)
PaperTab	307	393	-
FetaTab	871	1016	-

Datasets. To validate the effectiveness of the proposed method, experiments were conducted on four benchmark datasets, which are briefly described below. MMLongBench (Ma et al., 2024) comprehensively evaluates model document understanding capabilities. Its questions often require cross-page reasoning (33.7%) and include approximately 20.6% unanswerable questions, specifically designed to detect "hallucination" tendencies in DQA systems. LongDocURL (Deng et al., 2025) is a multimodal dataset focused on long document processing. It contains a large number of cross-modal questions to assess model performance in long-text contexts. PaperTab (Hui et al., 2024) is a specialized dataset for scientific paper comprehension, with particular emphasis on interpreting document structure and tabular data. FetaTab (Nan et al., 2022) is a Wikipedia-based question answering dataset that includes rich tabular and chart information. Key statistics for all datasets are listed in Table 1. Furthermore, we perform a dedicated analysis based on the labeled data. The "Issue" column in Table 1 shows the proportion of retrieval errors in the Colpali model caused by ignoring key query constraints. The judging rule is: an error is counted if the retrieval performance for a subquery $\mathbf{Q}_j$ is better than that for the original query q.

Baselines. We selected three types of frameworks as baselines: A pure VLM-based DQA framework; A multimodal DQA framework based on multi-RAG and multi-agent, represented by MDocAgent (Han et al., 2025); A multimodal RAG-based DQA system. In the pure VLM approach, documents are directly provided as context to the VLM for question answering. In the multimodal RAG approach, M3DocRAG (Cho et al., 2024), and the MoloRAG (Wu et al., 2025), which also employs graph structures and VLM evaluation, were selected. For fair comparison, MoloRAG uses a 7B model instead of the 3B model. The MoloRAG+ model, compared to MoloRAG, employs a fine-tuned model for retrieval.

Metrics. For the DQA evaluation metrics, the experiments are the same as those used in MDocAgent, LongDocURL, and MMLongBench, employing GPT-4o (OpenAI et al., 2024) to assess the outputs. Given a question and its reference answer, GPT-4o compares the DQA system's output and returns a Boolean value indicating whether the answer is correct and complete. For the RAG evaluation metrics, the experimental evaluation metrics align with those in MoloRAG, including Recall, Precision, Normalized Discounted Cumulative Gain (NDCG), and Mean Reciprocal Rank (MRR).

4.2 Main Results

The comparative results between our method and various baselines are presented in Table 2 and Table 3. Our approach consistently outperforms all baseline methods across the four evaluated datasets in terms of question answering accuracy, achieving an average improvement of 10.38% over the strongest baseline. Notably, the performance gain is especially pronounced on the PaperTab dataset (+18.50%), which emphasizes the understanding of document structure and tables.

Table 2: Comparison of our method with DQA approaches. Pure VLM-based, Multi-Agent-based, and RAG-based methods are evaluated using accuracy (%).

Model	RAG Method	MMLongBench	LongDocURL	FetaTab	PaperTab	Avg
Qwen-2.5-VL-7B	Direct	0.204	0.398	0.350	0.112	0.266
LLaVA-1.6-7B	Direct	0.176	0.110	0.301	0.102	0.172
MDocAgent	ColPali + ColBert	0.315	0.527	0.598	0.227	0.417
M3DocRAG	ColPali (Top-4)	0.296	0.503	0.537	0.152	0.372
MoloRAG	ColPali (Top-4)	0.371	0.536	0.554	0.157	0.405
MoloRAG+	MoloRAG+ (Top-4)	0.372	0.528	0.600	0.195	0.424
MAB-DQA (Ours)	**ColPali (Top-4)**	**0.399**	**0.564**	**0.638**	**0.269**	**0.468**
Improvement over second-best (%)		**+7.25%**	**+5.22%**	**+6.33%**	**+18.50%**	**+10.38%**

Table 3: Retrieval Performance Comparison on MMLongBench and LongDocURL Benchmarks under Top-K ($K = 1, 3, 5$) Settings. All values are in %. The best results are in bold.

Top-K	Method	MMLongBench				LongDocURL			
		Recall	Precision	NDCG	MRR	Recall	Precision	NDCG	MRR
1	M3DocRAG	45.31	56.80	56.80	56.80	46.82	64.51	64.45	64.51
	MDocAgent (ColBert)	30.65	40.50	40.50	40.50	40.72	56.31	56.31	56.31
	MDocAgent (Colpali)	46.61	59.86	59.86	59.86	46.90	64.18	64.18	64.18
	MoLoRAG	48.93	64.11	64.11	64.11	49.59	67.84	67.84	67.84
	MoLoRAG+	50.30	64.82	64.82	64.82	48.70	66.90	66.90	66.90
	MAB-DQA (Ours)	**50.97**	**66.35**	**66.35**	**66.35**	**50.60**	**69.95**	**69.95**	**69.95**
3	M3DocRAG	64.69	32.74	38.04	65.47	66.98	33.53	38.79	72.51
	MDocAgent (ColBert)	45.70	21.92	30.84	47.64	56.70	28.35	39.84	63.44
	MDocAgent (Colpali)	65.90	32.47	38.42	67.73	68.14	34.45	40.91	72.92
	MoLoRAG	67.40	33.18	39.70	70.66	69.58	35.27	42.30	75.60
	MoLoRAG+	66.95	33.10	39.97	71.02	69.42	35.18	42.07	74.89
	MAB-DQA (Ours)	**69.53**	**34.32**	**41.05**	**72.94**	**70.46**	**35.73**	**43.07**	**77.78**
5	M3DocRAG	72.43	22.67	30.06	66.92	74.54	23.52	32.07	73.99
	MDocAgent (ColBert)	53.15	16.06	26.43	49.27	64.53	20.09	30.60	64.88
	MDocAgent (Colpali)	73.07	23.07	30.25	69.04	75.49	23.87	31.81	74.30
	MoLoRAG	71.42	21.96	30.13	71.27	73.86	23.20	32.01	76.18
	MoLoRAG+	70.32	21.42	29.99	71.58	73.69	23.15	31.83	75.56
	MAB-DQA (Ours)	**75.86**	**24.13**	**32.14**	**73.85**	**77.02**	**24.30**	**33.13**	**78.73**

Moreover, as shown in Table 3, our method also establishes a new state-of-the-art in retrieval performance on both the MMLongBench and LongDocURL benchmarks, surpassing existing methods across all Top-K settings and all retrieval metrics (Recall, Precision, NDCG, and MRR). Our proposed method outperforms the baseline (Colpali) by an average of 6.55% across all metrics. These results demonstrate the effectiveness of MAB-DQA in enhancing both the precision of retrieval and the accuracy of answer generation in DQA tasks.

4.3 Ablation Studies

This section evaluates the contributions of two core modules: Multi-Arm Bandit-Guided Retrieval (MABR) and Hypergraph-based Reflective Reasoning Agent (HRRA) to the framework performance through ablation experiments. As in Table 4, when only basic retrieval is used (Colpali), the performance is the weakest. This indicates that the lack of differentiation among condition importance leads to retrieval interference from secondary information, severely limiting the accuracy of evidence localization and answer generation. When using MABR (w/ MABR), performance improves (an average gain of 22.8%), yet remains significantly lower than the full model. This demonstrates that the absence of a dynamic path selection mechanism reduces retrieval precision and robustness. When MABR is further retained, but HRRA is removed (w/ HRRA), the model achieves an average improvement of 26.5%, yet still underperforms on complex questions requiring multi-hop reasoning. This suggests that reflective reasoning and multi-stage verification are essential for information validation and error correction. The full model (Ours) achieves the best performance across all datasets, with an average improvement of 33.1%;

Table 4: Ablation Study on the Proposed MAB-DQA Modules. Additional ablation in the Appendix.

Model Variant	Total	MABR	HRRA	MMLongBench	LongDocURL	FetaTab	PaperTab	Avg. Imp.
Colpali (Baseline)	×	×	×	0.296	0.554	0.537	0.152	0.0%
w/ MABR	○	×	×	0.388	0.543	0.609	0.226	22.8%
w/ HRRA	○	○	×	0.395	0.561	0.624	0.236	26.5%
MAB-DQA (Ours)	○	○	○	**0.399**	**0.564**	**0.638**	**0.269**	33.1%

the gain from HRRA is most pronounced on complex datasets. MABR and HRRA progressively build upon and reinforce each other: MABR enables adaptive retrieval focusing on key aspects, and HRRA further integrates and validates information through reflective reasoning.

Meanwhile, we conducted an ablation study on the performance of different VLMs under the MAB-DQA framework, as shown in Table 5. In the table, we used the MoloRAG model as a reference and considered the top-3 retrieval results as the evaluation scope. This experiment was designed to evaluate whether "measuring the importance of aspects" is essential for different VLM backbones in DQA tasks. We selected four models: Qwen2.5-7B (Bai et al., 2023), Llava-13B (Liu et al., 2024), Qwen3-30B (a MoE model), and Qwen3-32B.

4.4 Sensitivity Analysis

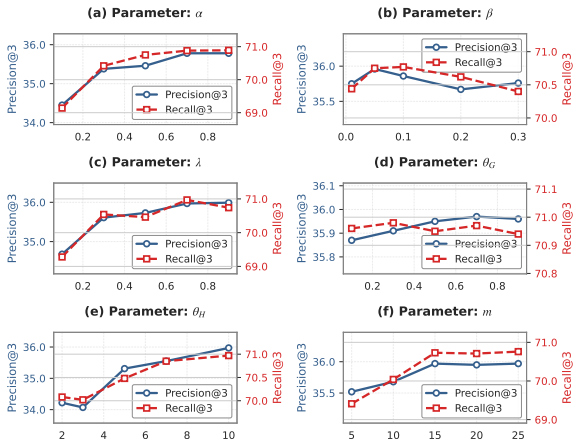


Figure 3: Sensitivity Analysis of Key Hyperparameters of the MAB-DQA Framework under Top-3 Retrieval on LongDocURL. The blue dashed line represents Precision. The red line represents Recall.

To evaluate the robustness of the MAB-DQA framework to key hyperparameters, we conducted a systematic sensitivity analysis on the LongDocURL dataset (Fig. 3). Using a controlled variable approach, we adjusted one target parameter at a time and observed changes in Recall and Precision under the Top-3 retrieval setting, while keeping all other parameters fixed. The results indicate that: (1) Parameter α has a significant positive effect on performance (Fig. 3a), with higher values better leveraging the visual-language model (VLM) to extract effective semantics; (2) Parameter β has a relatively minor impact on performance (Fig. 3b); (3) Parameter λ reflects the model's focus on conditional importance, and increasing it clearly improves performance (Fig. 3c). Additionally, we examined the effects of the edge connection threshold $\theta_{\mathbf{G}}$, hyperedge capacity $\theta_{\mathbf{H}}$, and retrieval iteration number m. Based on the analysis, the hyperparameters are set as follows for subsequent experiments: $\alpha = 0.8$, $\beta = 0.1$, $\lambda = 0.75$, $\theta_{\mathbf{G}} = 0.8$, $\theta_{\mathbf{H}} = 10$, and $m = 20$, ensuring stable and reliable performance across different configurations. Appendix F contains more ablation and sensitivity experiments.

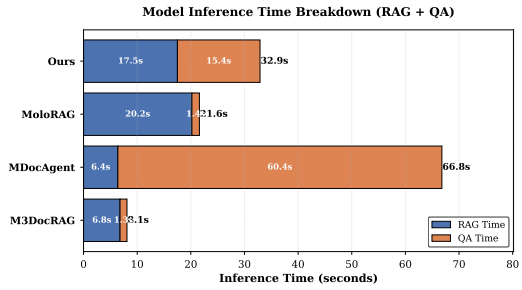


Figure 4: Comparison of Average Inference Time Across Different Models on 4 NVIDIA V100 GPUs, with retrieval time (Top-10) in blue and QA time (Top-4) in orange.

4.5 Mechanism Analysis

We provide the complete algorithm table for MAB-DQA in Appendix D. During the retrieval phase, the time complexity of the MABR algorithm is $O(mT_{\text{VLM}})$, where T_{VLM} denotes the time cost of VLM evaluation. We also tested the DQA efficiency of different frameworks under the same hardware environment, as shown in Fig. 4.

Qualitative cases of the MAB-DQA are provided to analyze the underlying mechanism of the model. In Fig. 5, we selected a representative multi-aspect query. For this query, the MAB-DQA framework

Table 5: Evaluation on MMLongBench and LongDocURL across VLM Backbones and Methods. Retrieval performance was evaluated based on Top-3 settings.

Backbone	Method	MMLongBench				LongDocURL			
		Recall	Precision	NDCG	MRR	Recall	Precision	NDCG	MRR
Qwen2.5-VL-7B	MoloRAG	67.40	33.18	39.70	70.66	69.58	35.27	42.30	75.60
LLaVa-13B		65.46	32.23	38.05	67.00	-	-	-	-
Qwen3-30B-A3B		68.41	34.08	40.13	70.78	69.80	36.02	42.47	75.77
Qwen3-32B		72.67	35.46	42.23	73.34	70.57	35.93	43.40	77.95
Qwen2.5-VL-7B	Ours	69.53	34.32	41.05	72.94	70.46	35.73	43.07	77.78
LLaVa-13B		66.24	32.37	38.21	67.12	-	-	-	-
Qwen3-30B-A3B		72.12	35.60	42.15	74.23	70.97	35.97	43.09	77.52
Qwen3-32B		**73.05**	**36.25**	**43.19**	**76.09**	**73.65**	**36.11**	**43.91**	**80.45**

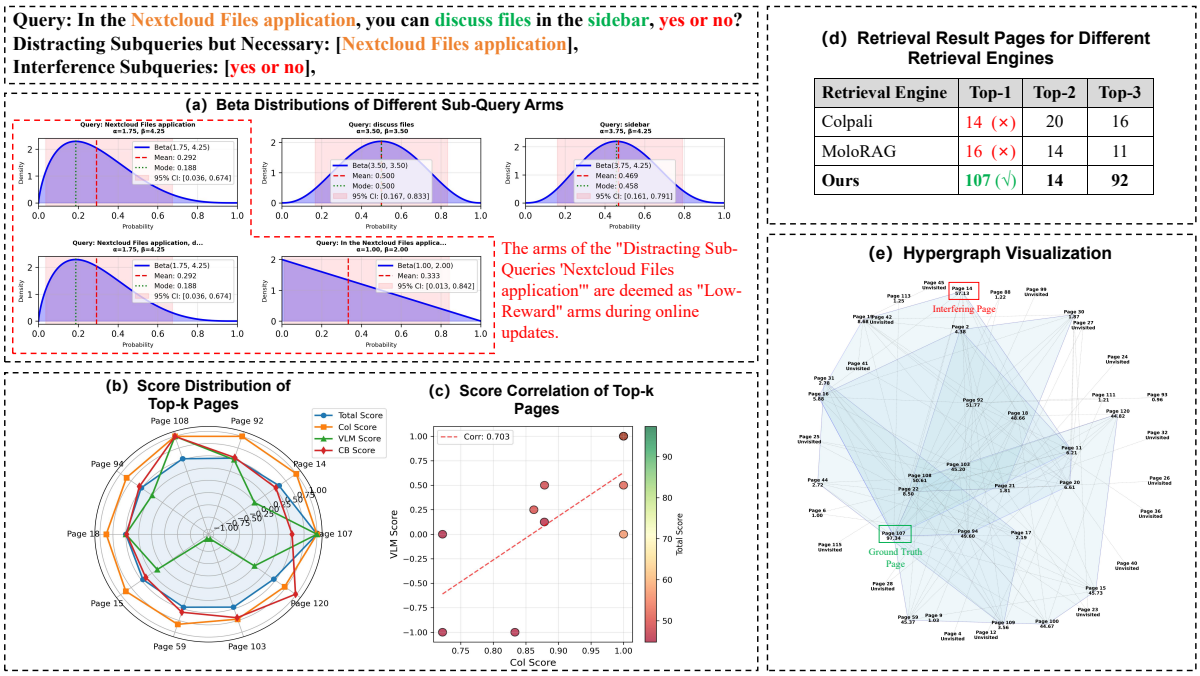


Figure 5: Qualitative Analysis of the MAB-DQA Framework.

correctly retrieved the evidence page (Page 107), while the other two baseline methods failed to do so. In this Fig. 5(a), there exists a low-importance condition, "Nextcloud Files application." The evidence lies in the fact that queries containing this condition (indicated by the red dashed section in Fig. 5(a)) are all assigned low rewards by the VLM (the Beta distribution tends towards 0). After correctly evaluating the importance of conditions, MAB-DQA successfully retrieved the evidence on the hypergraph (Fig. 5(e)).

5 Conclusion

This paper addresses a key challenge in multi-aspect DQA, where the retrieval process is often dominated by less important query aspects, leading to the omission of critical evidence. To tackle this, we propose MAB-DQA, a Multi-Armed Bandit-based DQA framework that explicitly models and dynamically allocates attention to the varying importance of implicit aspects within a query. By decomposing the query into aspect-aware subqueries and treating each as an arm in a bandit setup, MAB-DQA uses preliminary reasoning signals to estimate aspect utility and dynamically redistributes retrieval budget toward high-value aspects.

Extensive experiments on four benchmarks demonstrate that MAB-DQA significantly enhances document understanding performance, achieving an average improvement of 10.38% in answer accuracy over the strongest baseline. Moreover, MAB-DQA establishes new state-of-the-art retrieval performance, outperforming existing methods across all Top-K settings ($K = 1, 3, 5$) on MMLongBench and LongDocURL benchmarks, as shown in the comprehensive evaluation table.

Limitations

While the proposed MAB-DQA framework demonstrates significant improvements in multi-aspect Document Question Answering, several limitations remain to be addressed in future work.

Dependence on Visual-Language Model Performance. The framework heavily relies on the capability of the underlying VLM for both query decomposition and evidence evaluation. If the VLM performs poorly in specific domains, such as technical, legal, or medical documents, or under low-resource scenarios, the retrieval and reasoning performance may degrade accordingly.

Scalability with Document Length and Complexity. The MAB-DQA method we proposed is applied in our study to long documents (ranging from 40 to 500 pages). If a query requires retrieving a larger number of pages (e.g., as many as over 100 pages of evidence, though this is uncommon), it may be necessary to adjust hyperparameters or perform optimization.

Limitations of Hyperparameter Balancing. Our method relies on several hyperparameters (α, β, and λ) to balance different scoring components. Although we selected values via grid search, they may not generalize optimally to all document/query types. Our experiments show that adjustments to these hyperparameters lead to consistent performance fluctuations across multiple datasets. In future work, we plan to develop a version of MAB-DQA that incorporates Bayesian optimization for hyperparameter selection, thereby enhancing its adaptability to diverse DQA scenarios.

Restriction to Thompson Sampling. Our study exclusively employs Thompson Sampling (TS) as the core bandit algorithm, driven by its principled Bayesian approach, which aligns naturally with the probabilistic reward signals from the VLM. The Bernoulli-like feedback (relevant/irrelevant) is well-modeled by the Beta-Bernoulli conjugate prior, facilitating efficient online updates. However, this focus precludes a comparative analysis against other bandit strategies, such as Upper Confidence Bound (UCB) or Epsilon-Greedy. UCB offers stronger theoretical regret bounds and deterministic action selection, which might provide more stable retrieval paths. Epsilon-Greedy, while simpler, could be more effective in highly non-stationary environments where query aspect importance shifts rapidly. The superiority of TS in our specific combinatorial bandit setting has been empirically observed. Future work should include a comprehensive study to explore adaptive mechanisms that dynamically select the most suitable bandit strategy based on query characteristics.

Ethical Statement

In this work, we utilize models and datasets sourced from open-source platforms. All models and datasets are licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0). Their use fully complies with the corresponding license terms and is strictly limited to academic and research purposes. No private, sensitive, or personally identifiable information was employed in this study. Our research adheres to the ACL Code of Ethics and follows the ACL ethics guidelines, ensuring integrity, transparency, and reproducibility throughout the work.

References

Shipra Agrawal and Navin Goyal. 2012. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pages 39–1. JMLR Workshop and Conference Proceedings.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *The Twelfth International Conference on Learning Representations*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, and Fei Huang. 2023. Qwen technical report. *arXiv preprint*.

Ruisheng Cao, Hanchong Zhang, Tiancheng Huang, Zhangyi Kang, Yuxin Zhang, Liangtai Sun, Hanqi Li, Yuxun Miao, Shuai Fan, Lu Chen, and Kai Yu. 2025. NeuSym-RAG: Hybrid neural symbolic retrieval with multiview structuring for PDF question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6211–6239, Vienna, Austria. Association for Computational Linguistics.

Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24.

Rong Cheng, Jinyi Liu, Yan Zheng, Fei Ni, Jiazhen Du, Hangyu Mao, Fuzheng Zhang, Bo Wang, and Jianye Hao. 2025. DualRAG: A dual-process approach to integrate reasoning and retrieval for multi-hop question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31877–31899, Vienna, Austria. Association for Computational Linguistics.

Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. 2024. M3DocRAG: Multi-modal Retrieval Is What You Need for Multi-page Multi-document Understanding. *arXiv preprint*.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and Cheng-Lin Liu. 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, Vienna, Austria. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. *arXiv preprint*.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Celine Hudelot, and Pierre Colombo. 2024. ColPali: Efficient Document Retrieval with Vision Language Models. In *The Thirteenth International Conference on Learning Representations*.

Siwei Han, Peng Xia, Ruiyi Zhang, Tong Sun, Yun Li, Hongtu Zhu, and Huaxiu Yao. 2025. MDocAgent: A Multi-Modal Multi-Agent Framework for Document Understanding. *arXiv preprint*.

Hao Hu, Yifan Feng, Ruoxue Li, Rundong Xue, Xingliang Hou, Zhiqiang Tian, Yue Gao, and Shaoyi Du. 2025. Cog-RAG: Cognitive-Inspired Dual-Hypergraph with Theme Alignment Retrieval-Augmented Generation. *arXiv preprint arXiv:2511.13201*.

Yulong Hui, Yao Lu, and Huanchen Zhang. 2024. UDA: A Benchmark Suite for Retrieval Augmented Generation in Real-World Document Analysis. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Nan Huo, Jinyang Li, Bowen Qin, Ge Qu, Xiaolong Li, Xiaodong Li, Chenhao Ma, and Reynold Cheng. 2025. Micro-act: Mitigate knowledge conflict in question answering via actionable self-reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18550–18574, Vienna, Austria. Association for Computational Linguistics.

Songtao Jiang, Chenyi Zhou, Yan Zhang, Yeying Jin, and Zuozhu Liu. 2025. Fast or slow? integrating fast intuition and deliberate thinking for enhancing visual question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 525–534, Vienna, Austria. Association for Computational Linguistics.

Dosung Lee, Wonjun Oh, Boyoung Kim, Minyoung Kim, Joonsuk Park, and Paul Hongsuck Seo. 2025. ReSCORE: Label-free iterative retriever training for multi-hop question answering with relevance-consistency supervision. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 341–359, Vienna, Austria. Association for Computational Linguistics.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Richard James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, and Mike Lewis. 2023. Ra-dit: Retrieval-augmented dual instruction tuning. In *The Twelfth International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.

Runxuan Liu, Bei Luo, Jiaqi Li, Baoxin Wang, Ming Liu, Dayong Wu, Shijin Wang, and Bing Qin. 2025. Ontology-guided reverse thinking makes large language models stronger on knowledge graph question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15269–15284, Vienna, Austria. Association for Computational Linguistics.

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

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, Dragomir Radev, and Dragomir Radev. 2022. FeTaQA: Free-form Table Question Answering. *Transactions of the Association for Computational Linguistics*, 10:35–49. Place: Cambridge, MA Publisher: MIT Press.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, A. J. Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, and 400 others. 2024. GPT-4o System Card. *arXiv preprint*.

Boci Peng, Yongchao Liu, Xiaohe Bo, Jiaxin Guo, Yun Zhu, Xuanbo Fan, Chuntao Hong, and Yan Zhang. 2025. M³GQA: A multi-entity multi-hop multi-setting graph question answering benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30594–30620, Vienna, Austria. Association for Computational Linguistics.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. 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*, pages 3715–3734, Seattle, United States. Association for Computational Linguistics.

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):13636–13645.

Xiaqiang Tang, Qiang Gao, Jian Li, Nan Du, Qi Li, and Sihong Xie. 2025. MBA-RAG: a bandit approach for adaptive retrieval-augmented generation through question complexity. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3248–3254, Abu Dhabi, UAE. Association for Computational Linguistics.

Xixi Wu, Yanchao Tan, Nan Hou, Ruiyang Zhang, and Hong Cheng. 2025. Molorag: Bootstrapping document understanding via multi-modal logic-aware retrieval. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14035–14056.

Linhao Ye, Zhikai Lei, Jianghao Yin, Qin Chen, Jie Zhou, and Liang He. 2024. Boosting Conversational Question Answering with Fine-Grained Retrieval-Augmentation and Self-Check. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2301–2305, Washington DC USA. ACM.

Linhao Ye, Lang Yu, Zhikai Lei, Qin Chen, Jie Zhou, and Liang He. 2025. Optimizing question semantic space for dynamic retrieval-augmented multi-hop question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17814–17824, Vienna, Austria. Association for Computational Linguistics.

Peitian Zhang, Shitao Xiao, Zheng Liu, Zhicheng Dou, and Jian-Yun Nie. 2023. Retrieve Anything To Augment Large Language Models. *CoRR*.

Taolin Zhang, Dongyang Li, Qizhou Chen, Chengyu Wang, and Xiaofeng He. 2025a. BELLE: A bi-level multi-agent reasoning framework for multi-hop question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4184–4202, Vienna, Austria. Association for Computational Linguistics.

Zhange Zhang, Yuqing Ma, Yulong Wang, Shan He, Tianbo Wang, Siqi He, Jiakai Wang, and Xianglong Liu. 2025b. Lexical Diversity-aware Relevance Assessment for Retrieval-Augmented Generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27758–27781.

Zheyuan Zhang, Yiyang Li, Nhi Ha Lan Le, Zehong Wang, Tianyi Ma, Vincent Galassi, Keerthiram Murugesan, Nuno Moniz, Werner Geyer, Nitesh V Chawla, Chuxu Zhang, and Yanfang Ye. 2025c. NGQA: A nutritional graph question answering benchmark for personalized health-aware nutritional reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5934–5966, Vienna, Austria. Association for Computational Linguistics.

Junjie Zhou, Zheng Liu, Shitao Xiao, Bo Zhao, and Yongping Xiong. 2024. VISTA: Visualized Text Embedding For Universal Multi-Modal Retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3185–3200, Bangkok, Thailand. Association for Computational Linguistics.

Rongzhi Zhu, Xiangyu Liu, Zequn Sun, Yiwei Wang, and Wei Hu. 2025. Mitigating lost-in-retrieval problems in retrieval augmented multi-hop question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22362–22375, Vienna, Austria. Association for Computational Linguistics.

A Details of Evaluation Metrics

This paper employs two categories of evaluation metrics: **Document-based Question Answering accuracy metrics** and **Retrieval-Augmented Generation retrieval metrics**. The following details the definition, calculation, and application of each metric in the experiments.

A.1 Document-based Question Answering Accuracy Metrics

The DQA metrics are used to evaluate the correctness of the model-generated answers. The evaluation is specifically based on automated assessment using GPT-4o. The process is as follows: Given a question and its reference answer, GPT-4o compares the system output with the reference answer and judges whether the answer is "correct and complete." The evaluation results in a binary score (0 or 1). Score 1: The output answer is consistent with

the reference answer in terms of facts and logic (including handling "unanswerable" questions; if the reference answer is "Not answerable" and the model output is the same, it receives a score of 1). Score 0: The answer is incorrect or lacks key information. The final DQA accuracy is defined as the proportion of correctly answered questions to the total number of questions, as shown in Table 2 of the main text. This metric emphasizes the precision of the answer rather than partial correctness.

A.2 Retrieval-Augmented Generation Retrieval Metrics

The retrieval metrics are used to evaluate the performance of the hypergraph retrieval module. They include Recall, Precision, Normalized Discounted Cumulative Gain (NDCG), and Mean Reciprocal Rank (MRR). These metrics are computed under Top-K ($K = 1, 3, 5$) settings. Assuming the retrieval engine obtains a predicted page sequence $\hat{p}^{\text{pred}}$ and a ground-truth evidence page sequence $\hat{p}^{\text{gt}}$, the metrics are defined as follows:

Recall: Measures the proportion of retrieved relevant pages to all relevant pages.

$$\text{Recall@K} = \frac{|\hat{p}^{\text{pred}}@K \cap \hat{p}^{\text{gt}}|}{|\hat{p}^{\text{gt}}|}, \tag{11}$$

where $\hat{p}^{\text{pred}}@K$ denotes the top-K retrieved pages, and $|\cdot|$ denotes the cardinality of a set.

Precision: Measures the proportion of relevant pages among the retrieved results.

$$\text{Precision@K} = \frac{|\hat{p}^{\text{pred}}@K \cap \hat{p}^{\text{gt}}|}{K}. \tag{12}$$

Normalized Discounted Cumulative Gain (NDCG): Evaluates the quality of the retrieval ranking by considering positional weighting for relevant pages. The relevance score is based on the LI score.

$$\text{DCG@K} = \sum_{i=1}^{K} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)}, \tag{13}$$

$$\text{IDCG@K} = \sum_{i=1}^{|\hat{p}^{\text{gt}}|} \frac{2^{\text{rel}_i^{(\text{ideal})}} - 1}{\log_2(i+1)}, \tag{14}$$

$$\text{NDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}}, \tag{15}$$

where rel_i is the relevance score (e.g., LI score) of the i-th retrieved page, and $\text{rel}_i^{(\text{ideal})}$ is the relevance score of the i-th page in the ideal ranking (sorted by relevance in descending order).

Mean Reciprocal Rank (MRR): Computes the average of the reciprocal rank of the first relevant page.

$$\text{MRR} = \frac{1}{Q_D} \sum_{j=1}^{Q_D} \frac{1}{\text{rank}_j}, \tag{16}$$

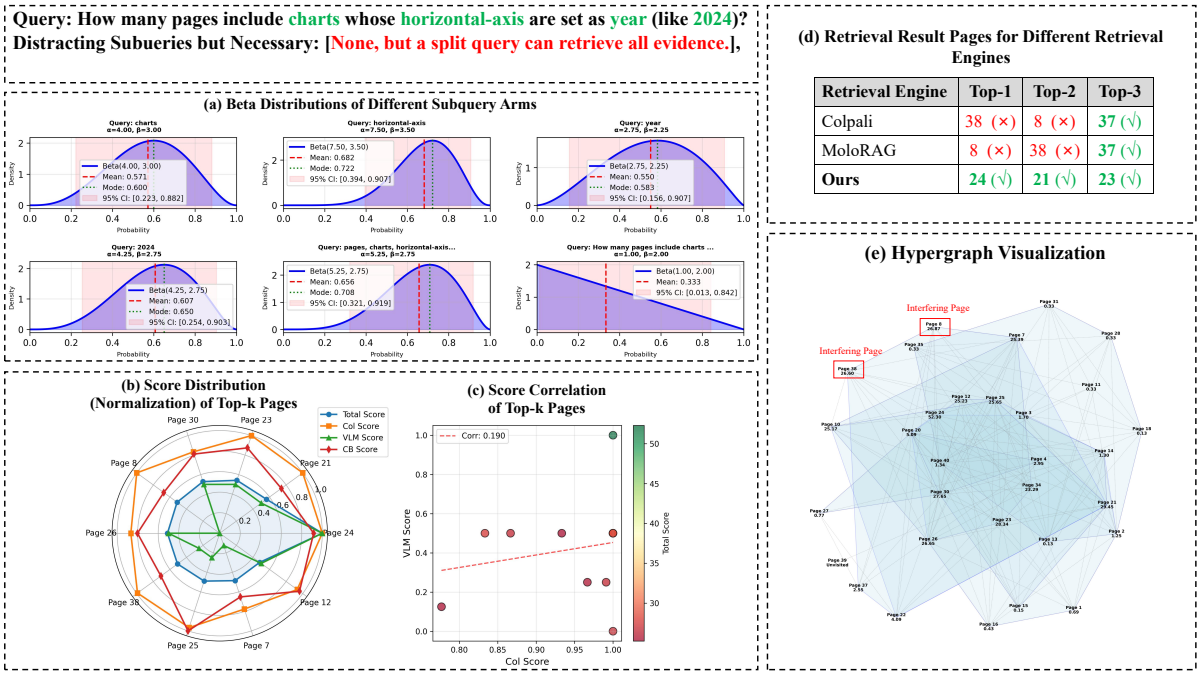
where Q_D is the number of queries, and rank_j is the rank position of the first relevant page for the j-th query.

These metrics collectively provide a comprehensive view of the retrieval process: Recall emphasizes coverage, Precision emphasizes accuracy, and NDCG and MRR emphasize ranking quality.

B Additional Case Studies and Qualitative Analysis

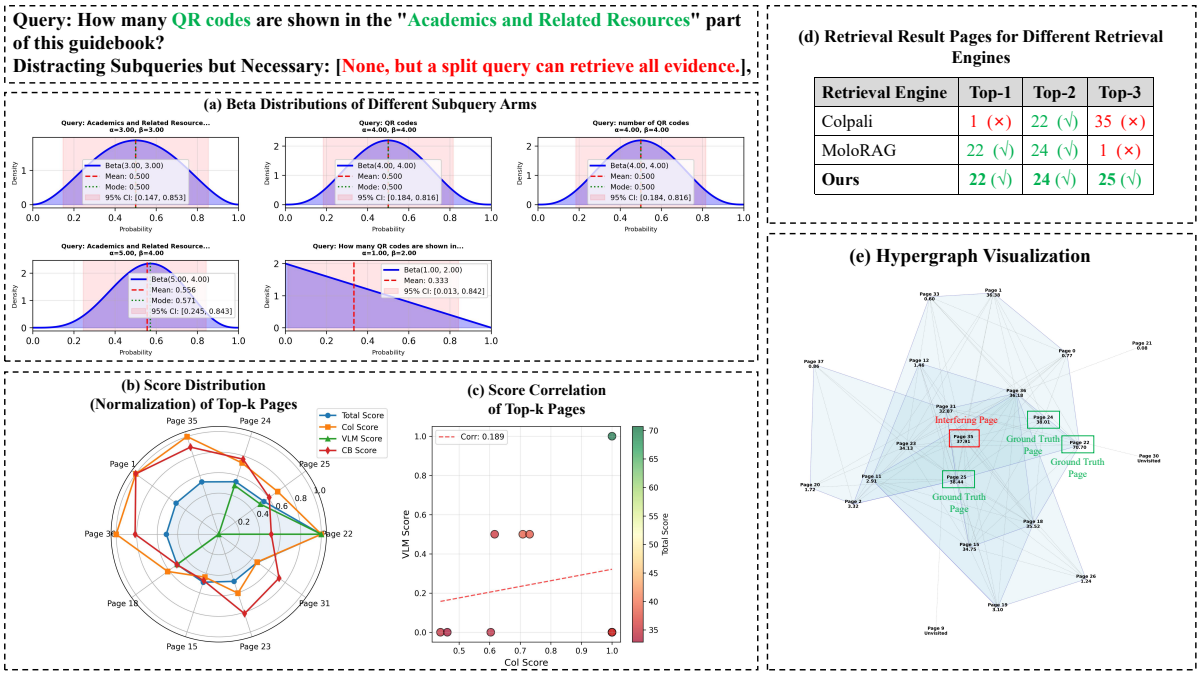
The Appendix B provides detailed qualitative analyses to further illustrate the effectiveness of the proposed MAB-DQA framework in handling diverse multi-aspect DQA scenarios. Through three representative case studies, we demonstrate how MAB-DQA addresses key challenges such as extensive page regression, multi-hop reasoning, and distraction from high-frequency terms. Each case includes a visual breakdown of the retrieval process, highlighting the framework's ability to dynamically weigh query conditions via hypergraph-based retrieval and reflective reasoning. The figures below present real examples from the MMLongBench benchmark, with annotations to clarify the retrieval mechanisms.

In Figure 6, the query involves evidence spread across 13 pages, testing the framework's ability to handle large-scale regression. MAB-DQA's query decomposition and hypergraph structure allow it to prioritize relevant pages while containing distractors within sparse hyperedges, as visualized in Fig. (e). Figure 7 focuses on a multi-hop query, where the initial retrieval by baselines includes irrelevant pages (e.g., page 35). Through iterative VLM feedback, MAB-DQA refines the path to associate all correct evidence. Figure 8 addresses a common pitfall where high-frequency terms like "guide book" mislead retrieval. By evaluating condition importance online, MAB-DQA suppresses such distractions and accurately locates the two evidence pages. Collectively, these cases validate the framework's robustness in complex, real-world DQA settings.



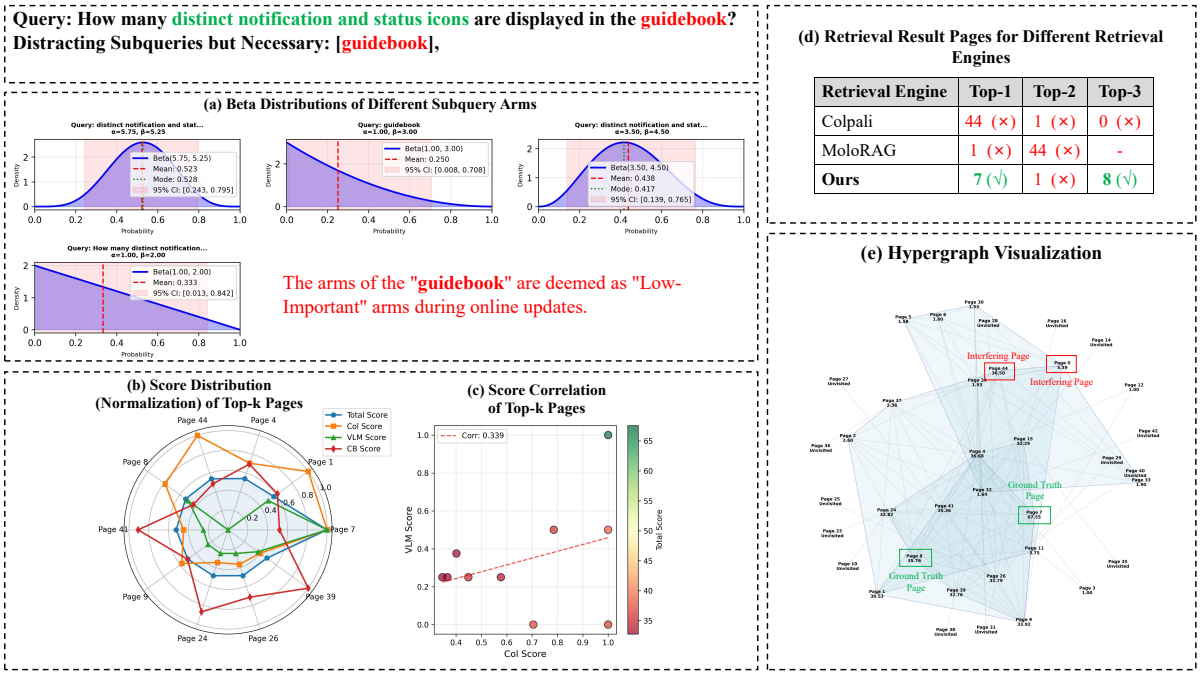
Retrieval Engine	Top-1	Top-2	Top-3
Colpali	38 (×)	8 (×)	37 (√)
MoloRAG	8 (×)	38 (×)	37 (√)
Ours	24 (√)	21 (√)	23 (√)

Figure 6: Qualitative Analysis of the MAB-DQA Framework. We selected a representative query that requires extensive page regression. The evidence for the problem spans 13 pages (specifically required: 10, 12, 14, 15, 20, 21, 22, 23, 24, 25, 26, 30, 37). It is evident that this query contains a large number of correct pages. This example demonstrates that our method, in the presence of positive samples, recalls a broader range of correct pages more comprehensively. As shown in Fig. (a), MAB-DQA, by providing query decomposition, recalls more correct evidence pages. As shown in Fig. (e), when extensive recall is required, distractor pages are usually contained within a small number of hyperedges.



Retrieval Engine	Top-1	Top-2	Top-3
Colpali	1 (×)	22 (√)	35 (×)
MoloRAG	22 (√)	24 (√)	1 (×)
Ours	22 (√)	24 (√)	25 (√)

Figure 7: Qualitative Analysis of the MAB-DQA Framework. We selected a representative query that requires a multi-hop answer. The evidence for the problem spans 3 pages (specifically required: 22, 24, 25). This query requires the QA framework to both accurately identify evidence and completely recall it. As shown in Fig. (b), the initial scores provided by Colpali include distractor pages (35, 1). As shown in Fig. (d), through multiple rounds of online evaluation, MAB-DQA correctly excludes the distractor pages. MAB-DQA provides an online reranking algorithm that correctly associates all evidence.



Retrieval Engine	Top-1	Top-2	Top-3
Colpali	44 (×)	1 (×)	0 (×)
MoloRAG	1 (×)	44 (×)	-
Ours	7 (√)	1 (×)	8 (√)

Figure 8: Qualitative Analysis of the MAB-DQA Framework. We selected a representative query that requires avoiding distracting conditions. The evidence for the problem spans 2 pages (specifically required: 7, 8). "Guide book" is likely a high-frequency term that appears frequently throughout the entire document. This causes the retrieval engine to consider every page as directly relevant to the evidence for this problem. As shown in Fig. (d), both Colpali and MoloRAG retrieve incorrectly. MAB-DQA avoids the unimportant condition "guide book," enabling the recall of the correct evidence pages.

C Prompt Settings

The Appendix C section provides a comprehensive collection of the prompt engineering templates utilized throughout the MAB-DQA framework. These carefully crafted prompts play a crucial role in enabling the framework's multi-aspect importance-aware retrieval and reflective reasoning capabilities. Each prompt serves a specific function in the pipeline, from initial query decomposition to final answer refinement.

C.1 Prompt for Decompose Query

The following prompt template is designed to extract meaningful subqueries from complex user questions, which form the foundation for the hypergraph-based retrieval approach:

> As an AI agent specialized in document retrieval query processing, your primary task is to handle each query by first ignoring any irrelevant information (such as output format requests or non-retrieval instructions).
> Then, extract meaningful entities and key phrases that capture the core intent of the query.
> Finally, output the result as a comma-separated list of key phrases, for example: "key_phrase1, key_phrase2, ...". Ensure clarity and conciseness throughout.

C.2 Prompt for Evaluate the Retrieval Evidence

> # GOAL # You are a Retrieval Expert, and your task is to evaluate how relevant the input document page is to the given query.
> Rate the relevance on a scale of 1 to 5, where:
> - 5: Highly relevant - contains COMPLETE information to fully answer the query (be cautious with this rating)
> - 4: Very relevant - contains most information needed but may lack some details (be cautious with this rating)

- 3: Moderately relevant - contains some useful information but significant gaps remain
- 2: Slightly relevant - has minor connection to the query
- 1: Irrelevant - contains no information related to the query
INSTRUCTION # Based on previous retrieval system judgment, we believe that this document snapshot is at least ' + priori + ' relevant. Please first read the given query, think about what specific information is required to answer that query comprehensively, and then carefully examine the document snapshot.
IMPORTANT # Before giving a score of 4 or 5, verify that the page actually contains the specific facts needed to answer the query, not just related information.
QUERY# + query + Think step by step about the relevance, then provide just a single number (1-5) representing your judgment.

C.3 Prompt For Evaluation:

Question: {question}
Predicted Answer: {answer}
Ground Truth Answer: {gt}
Please evaluate if the predicted answer is correct compared to the ground truth, considering the following criteria:
- If the Ground Truth Answer is "Not answerable":
- And the Predicted Answer indicates that the model cannot answer, then it is considered CORRECT (score 1).
- Otherwise:
- Score based on whether the Predicted Answer is factually and logically consistent with the Ground Truth Answer.
Score the answer on Binary correctness (0-1): 1 if the answer is correct, 0 if it is incorrect Return only a JSON-parsable string in the format: {{"binary_correctness": <score>}}
Output:

C.4 Prompt For Question Answering

The evaluation prompt ensures consistent assessment of answer quality across different benchmarks, maintaining standardization in performance measurement:

Using the provided { num_images } document screenshots, answer this question: "{question}"
Requirements:
- Reply must be extremely concise (as short as possible)
- Use only information visible in the screenshots
- If the answer cannot be clearly found, respond exactly: "Not answerable"
Answer:

C.5 Prompt For Question Reflection

The reflection prompt enables the HRRA component to refine ambiguous queries, improving retrieval precision through iterative clarification:

Based on the provided { num_images } document screenshots, rephrase the following question to make it clearer and more specific.
Original question: "{question}"
Requirements for rewriting:
1. If the question is clear and can be answered using ONLY information in the screenshots, keep it essentially the same
2. If the question is ambiguous or vague, clarify it based on what information appears to be available in the screenshots
3. If the question cannot be answered with the screenshots, note this, but still try to rephrase for clarity
4. The rewritten question should be specific, direct, and answerable using visible document content
5. Keep the core intent of the original question
6. If screenshots show specific entities (names, dates, numbers, terms), use them in the rewritten question
7. Output only the rewritten question, nothing else
Rewritten question:

C.6 Prompt For Answer Reflection

This prompt template facilitates the multi-stage verification process by assessing whether answers adequately address the original query requirements:

You will be given a question and a corresponding answer. Your task is to determine whether the answer addresses the question, regardless of whether the answer is correct or not.
Focus only on whether the answer responds to the question and covers the necessary points

(i.e., no essential content is missing).
If no answer is provided, consider it as not answering.
Question: {question}
Answer: {answer}
Did the answer address the question? (yes/no)

C.7 Prompt For Hypergraph Summary

The hypergraph summary prompt enables structural analysis of complex queries, supporting the framework's ability to handle multi-hop reasoning tasks:

Analyze the following question and identify the core concepts and relationships that need to be understood to answer it properly.
Question: "{question}" {"Key aspects to focus on: " + hypergraph if hypergraph else "Identify the key concepts and relationships in this question."}
Requirements:
- Break down the question into fundamental components
- Identify what specific information is needed to answer each component
- Note any implicit relationships or assumptions in the question
- Be concise but thorough in your analysis
Analysis:

C.8 Prompt For Refined Question Answering

The final refinement prompt implements the critical thinking component of HRRA, enabling iterative improvement of initial answers through evidence synthesis:

Based on the following context, provide a better answer to the question through careful reasoning.
Question: {question}
Initial incomplete answer: {initial_answer}
Relevant information summary: {summary}
CRITICAL THINKING REQUIREMENTS:
1. First, analyze what the question is REALLY asking for
2. Compare the initial answer with the available information
3. Identify gaps or inaccuracies in the initial answer
4. Synthesize information from the summary to fill these gaps
5. Formulate a coherent response that directly addresses the question
DO NOT simply copy phrases from the summary. Instead, use the information to construct a thoughtful answer.
If the summary indicates no relevant information, respond: "Not answerable"
Reasoning process:
- [Analyze the question requirements]
- [Compare initial answer with evidence]
- [Identify what needs to be improved]
- [Synthesize the improved answer]
Improved answer:

D Algorithm Design

This section provides detailed algorithmic descriptions of the two core components in our MABDQA framework: the hypergraph construction process and the MABR algorithm.

D.1 Hypergraph Construction Algorithm

The hypergraph construction algorithm (Algorithm 1) serves as the foundation for our multi-aspect retrieval framework. It transforms the original document and query into a structured hypergraph representation that captures complex conditional associations.

The algorithm begins by computing visual-language embeddings for both the query and all document pages. It then constructs a page similarity graph $\mathbf{G}$ where edges connect pages with similarity scores exceeding the threshold $\theta_{\mathbf{G}}$. This graph captures the intrinsic relationships between document pages based on their semantic content.

A key innovation is the decomposition of the original query q into subqueries $q_1, q_2, \ldots, q_M$ using the VLM, forming an Atomic and Global Subqueries Set that includes both individual subqueries and their complete combination. For each query subset $\mathbf{Q}_j$, the algorithm selects the top-$\theta_{\mathbf{H}}$ pages based on late interaction scores, then filters them against a global reference set $\mathbf{Q}_b$ to ensure that only pages with improved ranking under the specific subquery are included in the hyperedge.

The time complexity of Algorithm 1 is $O(N^2 \cdot D + M \cdot N \cdot D)$, where N is the number of pages, M is the number of subqueries, and D is the embedding dimension. The quadratic term arises from the page similarity graph construction, while the linear term accounts for the hyperedge generation process.

Algorithm 1 Query-Aware Page Hypergraph Construction

Require: Query q, document pages $\{p_1, p_2, \ldots, p_N\}$, VLM $\mathcal{V}$, graph threshold $\theta_{\mathbf{G}}$, hyperedge capacity $\theta_{\mathbf{H}}$
Ensure: Hypergraph $\mathbf{H} = (\mathbf{V_G}, \mathbf{E_H} \cup \mathbf{E_G})$

procedure HYPERGRAPHCONSTRUCTION($q, \{p_i\}_{i=1}^{N}, \theta_{\mathbf{G}}, \theta_{\mathbf{H}}$) ▷ Construct hypergraph structure from documents and query
  $\mathbf{E_q}, \mathbf{E_{p_i}} \leftarrow$ VLM embeddings of q and p_i
  $\mathbf{G}(\mathbf{V_G}, \mathbf{E_G}) \leftarrow$ empty graph
  for $i = 1$ to N **do**
    for $j = i + 1$ to N **do**
      if $\text{sim}(\mathbf{E}_{p_i}, \mathbf{E}_{p_j}) \geq \theta_{\mathbf{G}}$ **then**
        $\mathbf{E_G} \leftarrow \mathbf{E_G} \cup \{\{p_i, p_j\}\}$
      end if
    end for
  end for
  $\mathcal{E}_q \leftarrow \{q_1, q_2, \ldots, q_M\} \leftarrow \mathcal{V}.\text{decompose}(q)$ ▷ VLM decomposes query
  $\mathbf{Q} \leftarrow \{\{\hat{q}\} \mid \hat{q} \in \mathcal{E}_q\} \cup \{\mathcal{E}_q\}$ ▷ Atom-Integral Subqueries Set
  Let $\mathbf{Q}_b = \mathcal{E}_q$ (global reference, $b = M + 1$)
  $\mathbf{H} \leftarrow (\mathbf{V_G}, \mathbf{E_H} \cup \mathbf{E_G})$ ▷ Composite hypergraph
  for $j = 1$ to $M + 1$ **do**
    $\mathbf{C}_j \leftarrow$ Top-$\theta_{\mathbf{H}}$ pages by $\text{LI}(\mathbf{Q}_j, p)$ ▷ Select top pages for each query subset
    $\hat{\mathbf{E}}_j \leftarrow \{p \in \mathbf{C}_j \mid p \notin \mathbf{C}_b \vee \text{rank}(\text{LI}(\mathbf{Q}_j, p)) \leq \text{rank}(\text{LI}(\mathbf{Q}_b, p))\}$ ▷ Filter pages based on global reference
    $\mathbf{E_H} \leftarrow \mathbf{E_H} \cup \{\hat{\mathbf{E}}_j\}$
  end for
  return $\mathbf{H}$
end procedure

D.2 Multi-Armed Bandit–based Retrieval

Algorithm 2 implements our novel retrieval strategy that formulates the retrieval process as a combinatorial multi-armed bandit problem. Each subquery $\mathbf{Q}_j$ is treated as an "Arm" with a Beta distribution $\text{Beta}(\alpha_j, \beta_j)$ modeling its reward distribution.

The algorithm maintains a dynamic scoring function that combines three components: (1) the maximum late interaction score between the page and any subquery, (2) the direct VLM relevance assessment s_i^{vlm}, and (3) a hypergraph-based term balancing page connectivity h_i and bandit confidence scoress_i^{vlm}.

The retrieval proceeds iteratively, with the beam focusing on the most promising pages based on the composite score. Thompson sampling ensures a balance between exploration (trying less certain subqueries) and exploitation (focusing on combinations that have yielded high rewards). After each VLM evaluation, the algorithm updates the Beta parameters for all subqueries associated with the evaluated page, enabling online learning of condition importance.

The time complexity of Algorithm 2 is $O(m \cdot T_{\text{VLM}} + m \cdot |E_{\mathbf{H}}|)$, where m is the number of iterations and T_{VLM} is the VLM evaluation time. The algorithm's efficiency stems from its focused evaluation strategy that prioritizes pages with high potential relevance while maintaining theoretical guarantees through the multi-armed bandit formulation.

E Randomness Statement and Control Measures

In the experiments presented in this paper, randomness primarily stems from the following sources: (1) The Visual Language Model (VLM) for query decomposition may generate different subqueries; the VLM temperature is fixed at 0, but there may still be inherent stochastic risks. (2) The random initialization in page similarity computations (e.g., embedding generation) during the hypergraph construction process. (3) Uncertainties arising during hyperedge construction. (4) The random exploration strategy of Thompson Sampling in the multi-armed bandit problem. These stochastic factors may cause slight fluctuations in the results of a single experimental run. To mitigate their impact, we ensure that all key experiments are conducted with multiple independent runs (the specific number is detailed in the experimental setup) and employ a fixed random seed (set to 42) to guarantee reproducibility. For instance, in the hyperparameter sensitivity analysis (Sec. 4.4), we repeat experiments using a controlled variable method to isolate random noise. All experimental results reported in this paper are based on multiple runs (typically 3), and the best performance values are reported to demonstrate the potential of the method. In our

Algorithm 2 Multi-Armed Bandit–based Retrieval

Require: Hypergraph $\mathbf{H} = (\mathbf{V_G}, \mathbf{E_H} \cup \mathbf{E_G})$, retrieval iteration m, hyperparameters α, β, λ, VLM $\mathcal{V}$, query q, original pages $\{p_i\}_{i=1}^{N}$
Ensure: Top-10 retrieved page indices $\mathbf{R}_{\text{top10}}$

procedure MAB-RETRIEVAL($\mathbf{H}, m, \alpha, \beta, \lambda$) ▷ Hypergraph Bandit Thompson Sampling Search
  Initialize bandit arms: $\forall j, \alpha_j = 1, \beta_j = 1$
  visited $\leftarrow \emptyset$, scores $\leftarrow \emptyset$, current_page $\leftarrow \emptyset$
  Initialize score$(p_i) = \mathbf{LI}(q, p_i)$ for all $p_i \in \mathbf{V_H}$
  for iteration $t = 1$ to m **do** ▷ Main search loop
    to_evaluate $\leftarrow \emptyset$ ▷ Set of pages to evaluate in this iteration
    if current_page $= \emptyset$ **then** ▷ No current page, evaluate all pages
      to_evaluate $\leftarrow \mathbf{V_H}$
    else ▷ Start from current page
      to_evaluate $\leftarrow$ {current_page}
    end if
    for each page $p_i \in$ to_evaluate **do**
      $\hat{\mathbf{Q}}_i \leftarrow$ subquery linked to p_i
      $s_i^{\text{cb}} \leftarrow \frac{1}{|\hat{\mathbf{Q}}_i|} \sum_{\mathbf{Q}_j \in \hat{\mathbf{Q}}_i} \mathbb{E}[\text{Beta}(\alpha_j, \beta_j)]$ ▷ Bandit score from Thompson sampling
      $h_i \leftarrow \deg(p_i)$ in $\mathbf{H}$
      if $p_i \in$ visited **then**
        $\text{Adj}_{\mathbf{H}}(p_i) \leftarrow \{p_j \in \mathbf{V_H} \mid \exists e \in \mathbf{E_H}, \{p_i, p_j\} \subseteq e\}$
        Unvisited $\leftarrow \text{Adj}_{\mathbf{H}}(p_i) \setminus$ visited
        if Unvisited $\neq \emptyset$ **then** ▷ Found unvisited neighbors, jump to the best one
          $p^* \leftarrow \arg\max_{p_j \in \text{Unvisited}} \text{score}(p_j)$
          current_page $\leftarrow p^*$ ▷ Update current page for next iteration
        else
          Continue ▷ No unvisited neighbors, reset
        end if
      else
        $s_i^{\text{vlm}} \leftarrow$ VLM evaluates relevance of p_i to q ▷ Direct VLM evaluation for unvisited nodes
        visited $\leftarrow$ visited $\cup \{p_i\}$
        current_page $\leftarrow p_i$ ▷ Stay on this page for potential jump next iteration
      end if
      score$(p_i) \leftarrow (1-\alpha) \max_j \text{LI}(\mathbf{Q}_j, p_i) + \alpha s_i^{\text{vlm}} + \beta[(1-\lambda)h_i + \lambda s_i^{\text{cb}}]$ ▷ Composite scoring function
      scores$[p_i] \leftarrow$ score(p_i)
    end for
    ▷ Update bandit parameters for all visited pages (not just candidates)
    for each $p_i \in$ visited **do**
      $\hat{\mathbf{Q}}_i \leftarrow$ linked subqueries
      if p_i was evaluated by VLM in this iteration **then**
        $\forall \mathbf{Q}_j \in \hat{\mathbf{Q}}_i : (\alpha_j, \beta_j) \leftarrow (\alpha_j + s_i^{\text{vlm}}, \beta_j + 1 - s_i^{\text{vlm}})$
      end if
    end for
  end for
  $\mathbf{R} \leftarrow$ top 10 pages by final scores
  return $\mathbf{R}$
end procedure

experiments, we observe that performance fluctuations due to randomness are approximately $\pm 0.5\%$, which is considered acceptable. This assessment is primarily based on the following reasons: (1) Sensitivity analysis (Sec. 4.4) shows that variations in hyperparameters (such as α, β, λ) exhibit distinct peaks in their impact on the metrics; (2) Results across multiple datasets (e.g., MMLongBench, LongDocURL) are highly consistent, and the observed fluctuations do not significantly alter the conclusions.

F Supplementary Ablation and Sensitivity Experiments

The Appendix F presents additional ablation studies that investigate the individual contributions of key components in the MAB-DQA framework. Unlike the progressive ablation approach in the main text (Sec. 4.3), which sequentially added modules, here we examine the impact of removing single components while keeping others intact. This provides a more granular understanding of each module's role in the overall system performance.

The ablation experiments were conducted on the same four benchmark datasets as in the main exper-

Table 6: Individual Component Ablation Study on MAB-DQA Framework

Model Variant	MMLongBench	LongDocURL	FetaTab	PaperTab
w/o Query-Agnostic Page Graph	0.394	0.560	0.636	0.247
w/o Atomic and Global Subqueries Set	0.317	0.482	0.543	0.198
w/o Question Reflection	**0.401**	0.556	0.631	0.219
w/o Answer Reflection	0.397	0.561	0.625	0.247
Ours (Full Model)	0.399	**0.564**	**0.638**	**0.269**

iments: MMLongBench, LongDocURL, FetaTab, and PaperTab. We evaluated four modified versions of our framework by individually removing specific components:

- **w/o Query-Agnostic Page Graph**: Removes the Query-Agnostic Page Graph construction (Sec. 3.1), disabling the modeling of inter-page relationships.
- **w/o Atomic and Global Subqueries Set** : Uses only the Atomic Set (Eq. 3), limiting the framework's ability to capture global information.
- **w/o Question Reflection**: Disables the question reflection component in HRRA, eliminating the query clarification and refinement process.
- **w/o Answer Reflection**: Removes the answer reflection mechanism in HRRA, disabling the multi-stage verification and refinement of generated answers.

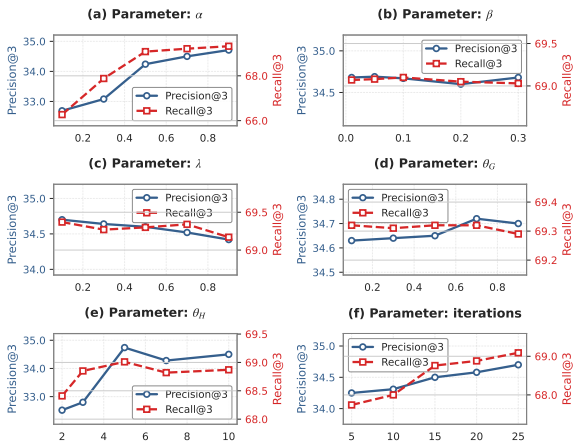


Figure 9: Sensitivity Analysis of Key Hyperparameters of the MAB-DQA Framework under Top-3 Retrieval on MMLongBench. The blue dashed line represents Precision. The red line represents Recall.

All experiments maintained the same evaluation metrics (accuracy) and experimental conditions as described in Sec. 4.1. In Fig. 9, we have supplemented the sensitivity analysis of the MAB-DQA framework on the MMLongBench dataset. Table 6 presents the quantitative results of the individual component ablation study. Several key observations emerge:

Query-Agnostic Page Graph Contribution: The removal of the Query-Agnostic Page Graph Contribution causes performance degradation across all datasets, with the most significant impact on PaperTab (-8.2%), which contains complex tabular structures. This demonstrates that modeling inter-page relationships is particularly crucial for documents with strong structural dependencies.

Atomic and Global Subqueries Set: Removing query decomposition results in the most substantial performance drop overall, particularly on LongDocURL (-14.5%), which contains complex multi-aspect questions. This highlights that capturing both fine-grained and holistic query aspects is essential for handling diverse question types.

Reflection Components: The question and answer reflection mechanisms show complementary effects.

The impact of individual components varies across datasets. PaperTab, focusing on scientific document understanding, benefits most from structural components (similarity graph), while LongDocURL, emphasizing long-document reasoning, relies heavily on query decomposition strategies. These findings confirm that each component in MAB-DQA contributes uniquely to the framework's overall effectiveness, with different modules playing dominant roles depending on the specific document characteristics and question types encountered in the DQA task.