DICTRAG: COMPOSABLE CORPUS ACCESS FOR MULTI-HOP QUESTION ANSWERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-hop question answering requires combining constraints to locate evidence scattered across a corpus. Ranked retrieval returns limited passage lists, restricting subsequent filtering to retrieved candidates. We introduce DictRAG, which enables agents to compose corpus-level conditions before selecting evidence to read. A bounded SQL interface exposes documents as field–value records and supports exact lexical filtering, set operations, and reusable relation definitions across turns. An indexed backend executes these operations outside model context, returning only selected observations. In corpus-constrained evaluations on HotpotQA, 2WikiMultiHopQA, MuSiQue, and Bamboogle, DictRAG with Qwen3.5-27B achieves a dataset-macro average of 65.70% exact match (EM) and 75.98% F1. Average gains over both agentic retrievers also hold with Qwen3.5-9B and Gemma4-31B. Progressive ablations on MuSiQue show larger observed gains from exact search and condition composition than from cross-turn relation reuse. Under corpus expansion, DictRAG has higher observed EM and F1 than both retrievers on all three scaling datasets at 100K and 500K documents. With a $533\times$ expansion of the HotpotQA corpus to 5.23 million documents, DictRAG retains 65.50% EM and shows a smaller observed decline than both agentic retrievers: 3.0 percentage points, versus 5.2 for Agentic-BM25 and 4.1 for Agentic-BGE. These results support composable corpus access as a useful capability for multi-hop question answering over large document collections.

1 INTRODUCTION

A multi-hop question may depend on only a few documents within a collection of millions. Retrieval-augmented generation (RAG) grounds language models in external corpora (Lewis et al., 2020), and agentic systems use intermediate evidence to guide subsequent searches (Press et al., 2023; Yao et al., 2023; Trivedi et al., 2023; Li et al., 2025; Du et al., 2026). We study whether letting an agent compose corpus-level conditions before reading evidence improves multi-hop QA and preserves accuracy as the collection expands.

For example, finding the birthplace of a director whose film features actors A and B requires identifying a shared film, verifying its director, and reading the biography. A ranked-search agent can filter observed passages but cannot apply conditions outside its returned pool. Increasing k enlarges that pool without exposing corpus-level composition.

We introduce DICTRAG, a bounded SQL interface for composable corpus access. The agent constructs candidates over the corpus, combines conditions before selecting observations, and saves relation definitions for reuse as intermediate entities emerge. Documents remain field–value records, and an indexed backend executes accepted operations outside model context. Both DictRAG and our agentic retriever baselines retain observations across turns; the main design difference is which corpus operations their tools expose.

Across four datasets, DictRAG achieves mean EM/F1 of 65.70/75.98 versus Agentic-BM25's 63.35/73.07. It leads the evaluated retrievers on all three scaling datasets at 100K and 500K documents and retains 65.50 EM on 5.23 million HotpotQA documents (Figure 1). Progressive MuSiQue comparisons examine exact search, composition, and relation reuse.

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

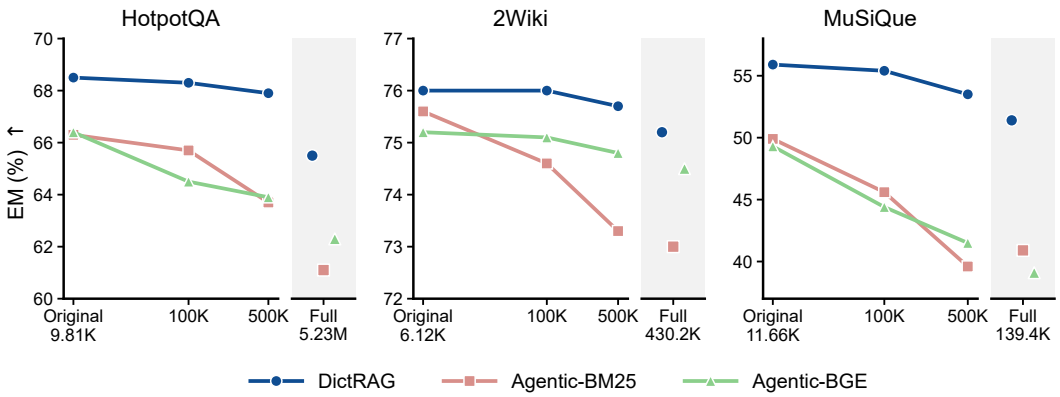


Figure 1: Accuracy under corpus expansion. DictRAG achieves the highest observed EM across all three datasets at 100K and 500K documents. Curves connect Original, 100K, and 500K conditions; shaded strips show separate Full-corpus results. The 1,000 questions per dataset remain fixed.

We address this question through an agent-facing interface, indexed execution, and empirical comparisons across access configurations.

Our contributions are:

- **Composable corpus access.** A bounded SQL interface supports corpus-level candidate operations before evidence exposure, with saved relations for cross-turn reuse.
- **Indexed execution.** An immutable query representation connects the interface to indexed candidate selection and exact residual checks over collections of up to 5.23 million documents.
- **An empirical comparison of access capabilities.** Progressive MuSiQue comparisons show larger observed increments for exact search and composition than for relation saving; four-dataset evaluations characterize original-corpus accuracy, with corpus-expansion experiments on HotpotQA, 2Wiki, and MuSiQue.

2 Related Work

Retrieval and external context. RAG and retrieval pretraining connect language models to corpora (Lewis et al., 2020; Guu et al., 2020), using sparse or dense retrievers such as BM25, DPR, Contriever, and BGE-M3 (Robertson & Zaragoza, 2009; Karpukhin et al., 2020; Izacard et al., 2022; Chen et al., 2024). Full-corpus prompting increases context demands and can be sensitive to evidence position (Liu et al., 2024). DictRAG keeps the corpus outside context and selects evidence through executable conditions. We evaluate retrieval depth, not full-corpus prompting.

Agentic and structured retrieval. Self-Ask, ReAct, IRCoT, FLARE, Self-RAG, and Search-o1 let models control retrieval (Press et al., 2023; Yao et al., 2023; Trivedi et al., 2023; Jiang et al., 2023; Asai et al., 2024; Li et al., 2025); A-RAG exposes keyword search, semantic search, and chunk reading (Du et al., 2026). GraphRAG organizes evidence through graphs and summaries (Edge et al., 2024), while HippoRAG2, HopRAG, and LinearRAG support structured evidence discovery (Gutiérrez et al., 2025; Liu et al., 2025; Zhuang et al., 2026). Interact-RAG exposes retrieval primitives for search fusion, entity matching, and context shaping (Hui et al., 2026). DictRAG focuses on composing corpus-level predicates and set operations through bounded SQL, with reusable saved relation definitions and no precomputed document graph.

Executable reasoning. PAL and Program-of-Thoughts use code for reasoning (Gao et al., 2023; Chen et al., 2023); Recursive Language Models inspect external context programmatically with recursive model calls (Zhang et al., 2025). DictRAG restricts execution to indexed corpus operations and bounded observations, carrying saved query definitions between calls.

3 METHOD

3.1 TASK SETTING

Given a question q and a finite ordered corpus $D = (d_1, \ldots, d_N)$, the system produces an answer $\hat{a}$. Each document exposes `id`, `title`, and `content`; an implicit `doc_index` records its position in the corpus. Document identity and corpus order serve different roles: set operations use identity, while observations follow corpus order. Gold answers, supporting facts, and evidence-chain annotations are not provided in model prompts or tool arguments. Section 4.3 defines the answer metrics.

3.2 COMPOSABLE CORPUS ACCESS

DictRAG exposes dictionary-style field–value records through a bounded SQL interface.

DictRAG supports a four-stage interaction: construct candidates, refine them, observe selected documents, and submit an answer. The immutable `documents` relation exposes the corpus through three native tools:

- `sql_save(name, query, params)` saves a lazy relation definition with snapshots of its parameters and dependencies.
- `sql_query(query, params, max_rows)` observes a bounded result from a supported query over the corpus or a saved relation.
- `final_answer(answer)` submits the answer string. The prompt requests a short supported answer; the tool does not itself verify evidential support.

The agent can combine conditions over the complete candidate relation before choosing which documents to read. For example, it can save documents matching A, refine that relation with B, and then observe a small ordered result. A saved relation retains a query definition rather than placing all matching document IDs or contents in the dialogue; saving alone does not imply result caching. Each question starts a fresh session, and saved relations persist only within that question. Figure 2 illustrates the distinction between a relation and its visible preview.

Supported operations. The SQL subset includes filters, projections, nonrecursive `WITH`, counts, existence checks, ordered slices, and document-identity `UNION`, `INTERSECT`, and `EXCEPT`. Named document relations can be refined across turns. Projected saved relations support observation and counts, but cannot be filtered or projected again. Observations follow ascending `doc_index` order. General joins, grouping, arbitrary ordering, raw FTS calls, and writes are rejected. Appendix A.1 gives an illustrative save–refine–observe sequence.

Dictionary-style access keeps the corpus outside model context. Fixed functions specify required-key access, missing-key defaults, and dictionary-shaped projection (Appendix A.1); these record semantics are not isolated by the empirical comparisons.

3.3 INDEXED EXECUTION AND CONTROLLER

Indexed execution. A parser maps accepted SQL to an immutable intermediate representation. The SQLite backend uses B-tree and trigram indexes to select candidates, followed where needed by fixed residual checks of the requested conditions. Queries require indexed predicates or bounded root access; the runtime does not fall back to unrestricted corpus scans. Unsupported syntax or work beyond the observation or residual budget produces an error rather than silently relaxing predicates or approximating the requested result. These checks bound supported operations, not total accumulated dialogue or end-to-end computation.

At each turn, the model receives the question, preceding tool calls, and observations. Automatic tool choice permits a brief plan alongside native calls. The host validates tool names and JSON arguments, executes supported SQL actions, and returns structured results or errors. A `final_answer` call must be the last call in its turn. Error feedback can suggest lowercasing a parameter or saving an unprojected relation; this assistance is part of the access configuration and never silently changes the SQL.

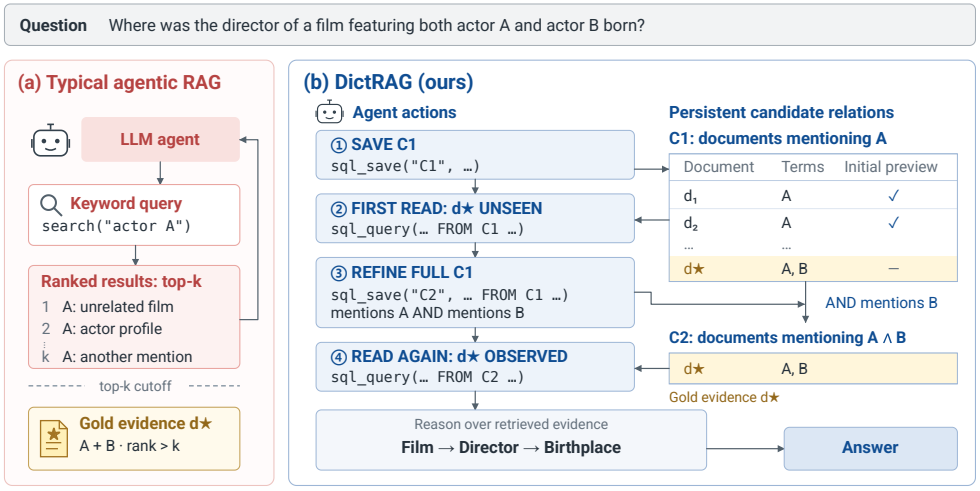


Figure 2: Illustrative evidence exposure under ranked retrieval and corpus-level refinement. (a) Gold evidence $d^\star$ can fall below a typical agentic RAG retriever's top-k cutoff. (b) A DictRAG agent (1) saves a lazy relation C_1 matching A, (2) reads a preview that omits $d^\star$, (3) refines the full C_1 with B to obtain C_2, and (4) reads again, now observing $d^\star$. Gold and preview markings are annotations for the reader, not information available to the agent; tool arguments and rows are schematic.

The tools expose neither filesystem nor network operations, and database connections are read-only. Execution deadlines and error limits are specified in Appendix A.1; these controls do not guarantee a bound on total computation or accumulated context.

4 Experimental Setup

4.1 Datasets and Corpus Conditions

We evaluate four datasets: HotpotQA, 2WikiMultiHopQA, MuSiQue, and Bamboogle. For the first three, we use the HippoRAG2-released 1,000-question evaluation subsets (Yang et al., 2018; Ho et al., 2020; Trivedi et al., 2022; Gutiérrez et al., 2025). Questions remain fixed across corpus conditions. *Original* denotes each released subset corpus. The constructed *100K* and *500K* corpora are shared across the three datasets. We merge their Original corpora and add documents from a pool of approximately 21 million Wikipedia documents to reach 100,000 documents. We then retain the entire 100K corpus and add documents from the same pool to reach 500,000 documents, so the 100K corpus is a subset of the 500K corpus. Document order is shuffled within each constructed corpus. *Full* denotes dataset-specific collections of 5,233,329 HotpotQA, 430,225 2Wiki, and 139,416 MuSiQue documents; their composition and order differ, and Full is not uniformly larger than 500K. Appendix B.3 describes the reported scaling conditions.

Table 1 summarizes the four original corpora. Each question is answered against a shared collection of thousands of documents rather than its gold evidence alone. Bamboogle uses all 125 questions (Press et al., 2023) and a fixed, project-constructed mixed corpus of 13,228 documents, comprising 1,572 candidate passages retrieved from DPR Wikipedia-18/W100 (Karpukhin et al., 2020) and 11,656 MuSiQue documents; this is not an official Bamboogle corpus (Appendix B.2). Like the other benchmarks, it is evaluated as corpus-constrained QA, not live web retrieval, and contributes equally to the four-dataset average. Corpus-expansion experiments cover HotpotQA, 2Wiki, and MuSiQue only.

4.2 Baselines and Comparison Protocol

Baselines include closed-book DirectLLM; single-shot BM25 and BGE-M3; graph or structure-augmented LinearRAG, HopRAG, and HippoRAG2; IRCoT; A-RAG; and Agentic-BM25/BGE.

Table 1: Original-corpus statistics. Token counts use `cl100k_base` over title plus content and are corpus descriptors, not model-specific usage measurements. All four datasets are evaluated under fixed corpus-constrained conditions.

Dataset	# Docs	Total tok.	Avg. doc tok.
MuSiQue	11,656	1,348,035	115.7
2Wiki	6,119	682,480	111.5
HotpotQA	9,811	1,299,765	132.5
Bamboogle	13,228	1,574,265	119.0

Single-shot BM25/BGE retrieve the top five passages once. Agentic-BM25/BGE expose `search(query)` through native tool calling; each search returns ranked top-k passages and deduplicates documents already observed for that question. Their default is $k = 5$ with $K = 30$.

Agentic-BM25/BGE and DictRAG share a backbone, question subsets, nominal retrieval-turn budget, and native tool-calling protocol. Their access-specific prompts, tools, and feedback remain part of the configurations. Retriever agents retain passages across turns; DictRAG additionally saves candidate definitions and selects observations through SQL. A DictRAG turn may contain multiple SQL calls, so equal turn limits do not match tool-call counts, evidence exposure, tokens, or execution time. The comparison therefore estimates configuration-level performance under a common nominal turn budget, not an isolated interface effect at equal cost. Appendix C reports retriever-only depth diagnostics. Appendix A.2 provides additional implementation details for the rerun baselines.

Additional interface diagnostics. All three interface diagnostics use Qwen3.5-27B, the same 1,000-question MuSiQue cohort, and nominal budget $K = 30$. *Decomp-BM25* adds an explicit sub-question plan to Agentic-BM25 and requests separate searches for each sub-question; the realized plan is not fixed across interfaces.

SQL-FTS5 exposes read-only SQLite/FTS5 with Boolean full-text matching, LIKE filters, joins, CTEs, aggregations, and persistent candidate state. It compares general structured access with DictRAG's restricted operation set.

Corpus-as-string exposes one plain `corpus_text` variable to an agent without structured document fields. It tests an unstructured executable-access configuration. Tools and prompts differ across these diagnostics.

4.3 Implementation and Metrics

The primary experiments use Qwen3.5-27B (Qwen Team, 2026), with temperature 0, disabled thinking, and native tool calling for DictRAG and the agentic retrievers. The same native SQL implementation is used across DictRAG evaluations. Additional comparisons use Qwen3.5-9B and Gemma4-31B (Qwen Team, 2026; Gemma Team, 2026). All comparator scores are local reruns on the stated question subsets. Appendix A.1 lists the reported DictRAG execution settings; cross-model settings appear in Appendices A.3 and A.4.

Metrics. We report EM and token-level F1 using SQuAD-style normalization (Rajpurkar et al., 2016), with additional normalization of diacritics and hyphens. The scorer removes punctuation and articles and takes the best score over the answer and dataset-provided aliases.

We report the unweighted mean of the four dataset-level scores, giving HotpotQA, 2WikiMultiHopQA, MuSiQue, and Bamboogle one quarter of the aggregate weight each. This dataset-macro average is not a pooled-question average: the datasets contain 1,000, 1,000, 1,000, and 125 questions, respectively. Means and differences are computed from displayed per-dataset scores, with rounding applied after arithmetic.

Statistical analysis. On MuSiQue, DictRAG's EM gains over both agentic retrievers pass Holm correction in every evaluated corpus condition; the smaller gaps on the other datasets do not meet the adjusted threshold. Full paired McNemar tests, bootstrap confidence intervals, and Holm-adjusted p-values for the primary comparisons and MuSiQue ablation appear in Appendix E.

5 RESULTS

5.1 MAIN RESULTS

Table 2 compares DictRAG with the broader baselines. DictRAG achieves the highest observed EM and F1 on all four datasets among the methods shown, with a dataset-macro average of 65.70 EM and 75.98 F1. Table 3 provides the direct evidence-access comparison with Agentic-BM25/BGE under the shared nominal turn budget.

Table 2: Main results using Qwen3.5-27B on the original corpora. HotpotQA, 2Wiki, and MuSiQue each contain 1,000 fixed questions; Bamboogle contains 125 questions. Scores are EM/F1 (%). Average weights all four datasets equally. Bold and underline mark the best and second-best values per column. The direct Agentic-BM25/BGE and DictRAG comparison appears separately in Table 3.

Method	MuSiQue		2Wiki		HotpotQA		Bamboogle		Average	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
DirectLLM	7.20	16.45	31.40	36.56	25.20	33.53	24.80	35.14	22.15	30.42
BM25	7.00	8.99	20.30	23.99	32.00	40.37	5.60	7.60	16.23	20.24
BGE	13.80	17.83	30.70	35.21	48.10	59.10	12.80	18.92	26.35	32.77
LinearRAG	23.70	33.69	57.50	63.85	53.60	64.38	26.40	37.37	40.30	49.82
HopRAG	31.70	45.50	51.50	55.89	58.20	70.07	41.60	52.07	45.75	55.88
HippoRAG2	42.00	52.78	56.80	61.93	61.30	72.60	38.40	48.91	49.63	59.06
IRCoT	24.90	34.59	56.00	62.04	57.10	69.84	31.20	39.70	42.30	51.54
A-RAG	<u>45.30</u>	<u>55.86</u>	<u>73.80</u>	<u>82.11</u>	<u>65.30</u>	<u>78.55</u>	<u>56.80</u>	<u>64.04</u>	<u>60.30</u>	<u>70.14</u>
DictRAG(Ours)	**55.90**	**67.18**	**76.00**	**83.65**	**68.50**	**81.05**	**62.40**	**72.05**	**65.70**	**75.98**

Evidence-access interface comparison. Table 3 compares Agentic-BGE, Agentic-BM25, and DictRAG on all four datasets. DictRAG has the highest four-dataset mean, 65.70 EM and 75.98 F1. Relative to Agentic-BM25, its mean advantage is 2.35 EM and 2.91 F1 points; relative to Agentic-BGE, it is 3.18 EM and 3.45 F1 points. The gains are largest on MuSiQue. On 2Wiki, DictRAG leads Agentic-BM25 by 0.40 EM and 0.26 F1 points. On Bamboogle, DictRAG reaches 62.40/72.05, Agentic-BM25 61.60/69.64, and Agentic-BGE 59.20/68.79 EM/F1.

Table 3: Direct evidence-access comparison using Qwen3.5-27B on the original corpora. HotpotQA, 2Wiki, and MuSiQue each contain 1,000 fixed questions; Bamboogle contains 125 questions. Scores are EM/F1 (%). Average weights all four datasets equally. Agentic retrievers use $k = 5$ and $K = 30$; DictRAG uses $K = 30$. Bold marks the highest observed value. Paired tests and CIs appear in Table 12.

Method	MuSiQue		2Wiki		HotpotQA		Bamboogle		Average	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Agentic-BGE	49.30	59.91	75.20	82.89	66.40	78.54	59.20	68.79	62.53	72.53
Agentic-BM25	49.90	60.54	75.60	83.39	66.30	78.72	61.60	69.64	63.35	73.07
DictRAG	**55.90**	**67.18**	**76.00**	**83.65**	**68.50**	**81.05**	**62.40**	**72.05**	**65.70**	**75.98**

5.2 ACCURACY UNDER CORPUS EXPANSION

Scaling uses the same 1,000 questions per dataset across conditions (Table 4; Figure 1). HotpotQA, 2Wiki, and MuSiQue share the 100K and 500K corpora defined in Section 4.1; 100K is a subset of 500K. Full corpora are separate collections and are not uniformly larger than 500K. Composition and ordering vary alongside size.

Fivefold corpus expansion. From 100K to 500K documents, DictRAG loses 0.4 EM points on HotpotQA, 1.9 on MuSiQue, and 0.3 on 2Wiki. Agentic-BM25 loses 2.0, 6.0, and 1.3 points on these datasets, respectively. Agentic-BGE loses 0.6 points on HotpotQA, 2.9 on MuSiQue, and 0.3 on 2Wiki. DictRAG has smaller observed declines than both retrievers on HotpotQA and MuSiQue,

matches the Agentic-BGE decline on 2Wiki, and achieves the highest absolute EM among the three methods on all three datasets at both corpus sizes.

Table 4: Accuracy across corpus conditions; 1,000 fixed questions per dataset. Numbers are EM and F1 (%). Full is not uniformly larger than 500K. Paired tests and CIs appear in Table 13.

Corpus	Dataset	Documents	DictRAG		Agentic-BM25		Agentic-BGE	
			EM	F1	EM	F1	EM	F1
100K	MuSiQue	100,000	55.40	66.75	45.60	56.06	44.40	54.73
100K	2Wiki	100,000	76.00	83.69	74.60	82.50	75.10	83.27
100K	HotpotQA	100,000	68.30	80.45	65.70	77.52	64.50	77.11
500K	MuSiQue	500,000	53.50	64.79	39.60	50.18	41.50	51.63
500K	2Wiki	500,000	75.70	83.71	73.30	80.91	74.80	82.51
500K	HotpotQA	500,000	67.90	80.22	63.70	75.73	63.90	75.87
Full	MuSiQue	139,416	51.40	63.13	40.90	50.90	39.10	49.86
Full	2Wiki	430,225	75.20	83.02	73.00	80.49	74.50	82.42
Full	HotpotQA	5,233,329	65.50	77.97	61.10	73.23	62.30	73.96

HotpotQA at 5.23 million documents. The corpus grows from 9,811 to 5,233,329 documents, approximately $533\times$. DictRAG retains 65.50 EM compared with 68.50 on the original corpus, a loss of 3.0 points. Using the original-corpus scores in Table 3, Agentic-BM25 falls from 66.30 to 61.10 (5.2 points), and Agentic-BGE from 66.40 to 62.30 (4.1 points). DictRAG has both the highest observed Full HotpotQA EM and the smallest original-to-Full EM decline among these methods.

Full-corpus conditions and ordering. Full 2Wiki reaches 75.20/83.02 EM/F1, 0.80 EM points below DictRAG's original score and 0.70 above the Full BGE result. Full MuSiQue reaches 51.40/63.13, below its 500K score despite containing fewer documents. Because composition and ordering also differ, this comparison does not isolate corpus size.

Mechanism hypothesis. Applying conditions before evidence exposure may reduce competition from added distractors. This interpretation is consistent with the candidate-pool diagnostic in Section 5.4, but the accuracy results do not directly measure how agents discover complete evidence chains.

5.3 Access-Capability Ablation

DictRAG's MuSiQue EM margins are +6.00 over Agentic-BM25 and +6.60 over Agentic-BGE, compared with +2.20 and +2.10 on HotpotQA. We use MuSiQue to compare progressively richer operations and diagnose candidate coverage. Table 5 examines progressively richer access on MuSiQue. Exact-search provides single-condition title or content matching, document-ID lookup, and pagination, without compound predicates or saved relations. Composable-search adds compound predicates and document-identity set operations, but does not retain named relations across calls; within-query composition can recompute the required candidates. Full DictRAG additionally saves and reuses named relations across turns. Agentic-BM25 provides the ranked-search reference.

Table 5: Progressive evidence-access ablation on MuSiQue. EM and F1 are percentages; Δ gives the percentage-point change from the preceding variant. Paired tests and CIs appear in Table 14.

Variant	Score		Δ	
	EM	F1	EM	F1
Agentic-BM25	49.90	60.54	—	—
Exact-search	53.00	64.01	+3.10	+3.47
Composable-search	55.10	66.32	+2.10	+2.31
Full DictRAG	**55.90**	**67.18**	+0.80	+0.86

In this ordering, replacing ranked search with exact search increases EM/F1 by 3.10/3.47 points; adding composition increases them by another 2.10/2.31. Saving relations adds 0.80/0.86, for a total of

6.00/6.64 over Agentic-BM25. These are conditional increments, not independent component effects. Exact search and composition pass Holm-corrected EM tests ($p = 0.000196$ and $p = 0.002901$, respectively; Table 14). Relation reuse adds a smaller observed increment that does not reach the EM significance threshold ($p = 0.057373$).

The largest observed increments occur at exact search and composition, which motivates our emphasis on corpus-level candidate operations. Composable-search can recompute candidates within a query, while Full DictRAG adds named cross-turn reuse.

5.4 Interface and Candidate-Coverage Diagnostics

Decomposition and structured access. Explicit decomposition prompting reaches 47.00 EM and 58.39 F1, compared with DictRAG's 55.90/67.18, leaving an 8.90 EM and 8.79 F1 gap. This intervention does not close the gap, but does not establish that planning quality is identical across interfaces. General SQL/FTS5 access reaches 49.50 EM and 63.21 F1, 6.40 and 3.97 points below DictRAG at similar input-token usage. Corpus-as-string reaches 49.40/60.85 despite using more input tokens, leaving a 6.50 EM and 6.33 F1 gap. These comparisons support further investigation of structured corpus operations; they do not isolate dictionary semantics or SQL syntax as independent causes.

Table 6: MuSiQue interface diagnostics on 1,000 questions. Scores are percentages; input tokens are per-question means in thousands.

Configuration	EM	F1	Input tok. (K)
Decomp-BM25	47.00	58.39	78.6
SQL-FTS5	49.50	63.21	47.2
Corpus-as-string	49.40	60.85	115.1
DictRAG	55.90	67.18	45.5

Candidate-pool coverage. A hybrid variant first retrieves a BGE top-200 pool and then applies programmable filtering only within that pool. It reaches 45.00 EM and 58.27 F1, compared with full-corpus DictRAG's 55.90/67.18. The pool contains all gold supporting documents for 69.4% of questions, with 86.95% average supporting-document recall. On those same 694 fully-covered questions, the hybrid and full-corpus DictRAG perform similarly, reaching 53.89/68.32 and 54.03/67.55, respectively. These results support candidate-pool coverage as an important bottleneck, but do not establish that recall alone explains the overall gap.

5.5 Cross-Model Generalization

Table 7 repeats the four-dataset comparison with Qwen3.5-9B and Gemma4-31B. DictRAG reaches mean EM/F1 of 58.65/68.65 and 64.03/74.43, respectively, exceeding both agentic retrievers with each backbone. MuSiQue again shows the largest EM margins, whereas several 2Wiki differences are small. On Bamboogle, Gemma DictRAG exceeds Agentic-BGE by 3.20 EM and 5.54 F1 points (59.20/66.66 vs. 56.00/61.12). Appendix A.4 gives the Bamboogle evaluation settings.

5.6 Accuracy–Cost Trade-off

Appendix D, Table 11, reports original MuSiQue costs at concurrency 100. DictRAG uses 45,519 input tokens versus Agentic-BM25's 31,499 (+44.5%) for a 6.00 EM gain. Recorded time is 3.18 versus 2.25 seconds per question (+41.3%). These concurrent-run summaries do not establish isolated-request latency or median/P95 latency. Neither this comparison nor retrieval-depth diagnostics match computational cost.

On MuSiQue, increasing Agentic-BM25 retrieval depth to $k = 10$ yields 57.69 F1 with 46.8K input tokens, versus DictRAG's 67.18 F1 with 45.5K. At $k = 50$, Agentic-BM25/BGE reach 55.00/57.06 F1 with 140.5K/137.5K input tokens (Appendix C). These deeper configurations do not close the accuracy gap, but also change evidence exposure and do not isolate a pure computation-budget effect.

Table 7: Cross-model original-corpus results. HotpotQA, 2Wiki, and MuSiQue each contain 1,000 questions; Bamboogle contains 125 questions. Scores are EM/F1 (%). Average weights all four datasets equally. Bold marks the best value within each backbone.

Method	MuSiQue		2Wiki		HotpotQA		Bamboogle		Average	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Qwen3.5-9B										
Agentic-BM25	40.70	50.77	70.80	79.09	62.80	74.37	48.00	56.24	55.58	65.12
Agentic-BGE	40.30	49.99	72.40	80.01	62.70	74.82	46.40	53.67	55.45	64.62
DictRAG	**46.40**	**57.00**	**72.90**	**80.36**	**64.10**	**76.34**	**51.20**	**60.91**	**58.65**	**68.65**
Gemma4-31B										
Agentic-BM25	49.70	62.25	74.60	82.68	64.20	77.02	55.20	60.53	60.93	70.62
Agentic-BGE	49.80	62.65	74.10	82.40	64.80	77.76	56.00	61.12	61.18	70.98
DictRAG	**55.70**	**68.78**	**74.70**	**82.86**	**66.50**	**79.43**	**59.20**	**66.66**	**64.03**	**74.43**

6 Conclusion

DictRAG enables agents to compose corpus-level conditions before selecting evidence to read through a bounded SQL interface. Across four benchmarks and three backbones, it achieves higher dataset-macro EM and F1 than the evaluated agentic retrievers under a shared nominal turn budget. Progressive ablations on MuSiQue show larger observed gains from exact search and condition composition than from cross-turn relation reuse. Under corpus expansion, DictRAG retains 65.50% EM on 5.23 million HotpotQA documents, a decline of 3.0 percentage points from the original corpus. Together, these results support corpus-level candidate construction and refinement as useful capabilities for multi-hop question answering over large document collections.

Limitations

DictRAG's exact lexical predicates can miss aliases or paraphrases, and its restricted operations limit query expressiveness. Each method–condition pair is evaluated once, so variance across random seeds is not estimated. Bamboogle's small sample ($n = 125$) makes small score differences sensitive to individual outcomes. In the constructed corpora, composition and ordering change alongside size, preventing attribution of performance differences to corpus size alone.

Ethical Considerations

The experiments use public benchmarks without new human-subject data. Corpus access can expose sensitive or biased material; deployments should enforce document permissions, query auditing, read-only execution, and process-level resource limits.

AI Use Statement

We used generative AI tools to assist with drafting and revising manuscript text, translating manuscript content, refining the presentation of the method, and providing feedback on the interpretation of results and experimental limitations. We also used these tools to assist with coding, experiment scripts, and LaTeX formatting. The authors reviewed the AI-assisted text, checked reported results against experimental records, verified references against their original sources, and tested AI-assisted code for correctness. The authors take responsibility for the final manuscript, including all claims, results, references, and code produced with AI assistance.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.137. URL https://aclanthology.org/2024.findings-acl.137/.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=YfZ4ZPt8zd.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Shaohan Wang, Pengyu Wang, Xiaorui Wang, and Zhendong Mao. A-RAG: Scaling agentic retrieval-augmented generation via hierarchical retrieval interfaces, 2026. URL https://arxiv.org/abs/2602.03442.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph RAG approach to query-focused summarization, 2024. URL https://arxiv.org/abs/2404.16130.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10764–10799. PMLR, 2023. URL https://proceedings.mlr.press/v202/gao23f.html.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From RAG to memory: Non-parametric continual learning for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21497–21515. PMLR, 2025. URL https://proceedings.mlr.press/v267/gutierrez25a.html.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3929–3938. PMLR, 2020. URL https://proceedings.mlr.press/v119/guu20a.html.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Yulong Hui, Chao Chen, Zhihang Fu, Yihao Liu, Jieping Ye, and Huanchen Zhang. Interact-RAG: Reason and interact with the corpus, beyond black-box retrieval. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=yHUjWb6eMe.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=jKN1pXi7b0.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, 2023. doi: 10.18653/v1/2023.emnlp-main.495. URL https://aclanthology.org/2023.emnlp-main.495/.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, Suzhou, China, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.276. URL https://aclanthology.org/2025.emnlp-main.276/.

Hao Liu, Zhengren Wang, Xi Chen, Zhiyu Li, Feiyu Xiong, Qinhan Yu, and Wentao Zhang. HopRAG: Multi-hop reasoning for logic-aware retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 1897–1913, 2025. doi: 10.18653/v1/2025.findings-acl.97. URL https://aclanthology.org/2025.findings-acl.97/.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711, 2023. doi: 10.18653/v1/2023.findings-emnlp.378. URL https://aclanthology.org/2023.findings-emnlp.378/.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392. Association for Computational Linguistics, 2016. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264/.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10014–10037, 2023. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Alex L. Zhang, Tim Kraska, and Omar Khattab. Recursive language models, 2025. URL https://arxiv.org/abs/2512.24601.

Luyao Zhuang, Shengyuan Chen, Yilin Xiao, Huachi Zhou, Yujing Zhang, Hao Chen, Qinggang Zhang, and Xiao Huang. LinearRAG: Linear graph retrieval augmented generation on large-scale corpora. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mCtfkypdm6.

A Implementation and Evaluation Protocol

A.1 DictRAG Experimental Settings and Prompt Summary

The primary native SQL DictRAG evaluations use a fixed implementation; settings for the separate Bamboogle cross-model evaluation appear in Appendix A.4. A question starts with a fresh session exposing the immutable `documents` relation and no pre-existing saved relations. The system prompt describes the supported SQL subset, snapshot semantics, and error boundaries. The user message supplies the question, document count, and schema version, without answer or support annotations.

Native tool protocol and prompt. The tools are `sql_save(name, query, params)`, `sql_query(query, params, max_rows)`, and `final_answer(answer)`. Model-facing `params` is an array of JSON scalars, and `max_rows` is an explicit positive integer. Automatic tool choice permits prose planning alongside calls; the prompt requests the answer type, unresolved relation, and next retrieval target.

The prompt recommends selective predicates with bound parameters, followed by small, ordered sets of full evidence documents. If a field uses `lower`, its string parameter should also be lowercase. A representative initial observation uses `params=["entity"]` and `max_rows=5` with:

```
SELECT id, title, content
FROM documents
WHERE contains(lower(title), ?)
ORDER BY doc_index ASC LIMIT 5
```

For reusable candidates, the agent saves `SELECT *` rather than a projection. A later query can filter the saved name, count candidates, or observe selected rows. Full text for a known document is accessed by its observed ID. Prefix previews are treated as discovery aids because they can omit matching sentences or bridge evidence.

Illustrative save–refine–observe sequence. The following tool arguments illustrate the supported operations; they are not a recorded trajectory. Save `C1` with `params=["actor a"]`:

```
SELECT * FROM documents
WHERE contains(lower(content), ?)
```

Then save `C2` with `params=["actor b"]`:

```
SELECT * FROM C1
WHERE contains(lower(content), ?)
```

Finally use `sql_query` with `params=[]` and `max_rows=5`:

```
SELECT id, title, content FROM C2
ORDER BY doc_index ASC LIMIT 5
```

The saved definitions do not include the observation's limit. Later refinement therefore acts on the defined relation, rather than only the five visible rows. Co-occurrence must still be checked by reading the returned evidence.

Dictionary semantics through SQL. The prompt exposes `field`, `get`, and `build` explicitly. For a document d, `field(document, 'title')` corresponds to required-key access d`['title']`, whereas `get(document, 'title', '')` corresponds to access with an empty-string default. The former raises a missing-key error; the latter returns the default when the key is absent. Dictionary projection is expressed, for example, as:

```
SELECT build('id', id, 'title', title) AS value
FROM documents ORDER BY doc_index ASC LIMIT 1
```

Here each returned `value` is a dictionary with `id` and `title` keys. The frontend maps these operators to the shared corpus intermediate representation and fixed evaluator.

Execution controls. Table 8 reports the primary Qwen3.5-27B experimental settings. The retrieval budget counts model turns, each of which may contain multiple SQL calls. Saved bindings are isolated by question; rejected queries do not silently relax predicates. The SQLite deadline does not preempt residual or regex evaluation, and the trajectory deadline is checked between calls.

Table 8: DictRAG experimental settings for the primary Qwen3.5-27B evaluation. Cross-model and Bamboogle settings are described in Appendices A.3 and A.4.

Hyperparameter	Value
Backbone / hardware	Qwen3.5-27B / H200 NVL
Generation	Temperature 0; top-p 1; top-k 0; thinking disabled
Native tool parsing	`qwen3_coder`; automatic tool choice
Retrieval	At most 30 model turns
SQLite deadline	30 seconds

Reproduction scope. This appendix summarizes the native tool protocol, prompt guidance, and reported execution settings; it is not a verbatim prompt or a complete run manifest. Exact prompt text, model and code revisions, observation/residual cap values, decoding limits, and corpus/index manifests are required to reproduce execution precisely. Appendices B.1 and B.3 describe the reported question cohorts.

A.2 Baseline Implementation Details

Baseline runs use the same question subsets and answer aliases. DirectLLM receives only the question. Single-shot BM25 and BGE-M3 retrieve five passages once and pass them to the Qwen3.5-27B answer generator. LinearRAG, HopRAG, HippoRAG2, and A-RAG use method-specific retrieval and answer pipelines. IRCoT interleaves retrieval and intermediate reasoning. These pipeline descriptions do not specify all implementation settings; exact prompts and configurations are needed for reproduction.

Agentic-BM25/BGE invoke `search(query)` through native tool calling, retain prior observations, and reason over ranked passages. Both retriever agents and DictRAG use native tools, but only DictRAG exposes its corpus-level SQL relations. Baseline scores follow Section 4.3. A complete configuration must additionally specify whether deduplication refills the top-k result list, per-request limits, and error feedback; equal nominal turn budgets alone do not specify these behaviors.

Table 9: Evidence-access and controller configuration for the original-corpus comparison.

Row	Corpus access	Controller
DirectLLM	No corpus passages	One answer prompt
BM25/BGE	Top-5 passages from a subset-built index	One answer prompt over retrieved passages
Graph RAG family	Method-specific retrieval structures	Method-specific retrieval and answer pipeline
IRCoT	Iterative retrieval	Interleaved retrieval and reasoning
A-RAG	Hierarchical keyword, semantic, and chunk tools	A-RAG agentic loop
Agentic-BM25/BGE	`search(query)` returning ranked deduplicated passages	Native tool calling
DictRAG	Dictionary-style records and saved relations via bounded SQL	Native tool calling; bounded SQL execution

A.3 Cross-Model Evaluation Settings

Official checkpoint model cards are available for Qwen3.5-27B, Qwen3.5-9B, and Gemma4-31B.

For each of Qwen3.5-9B and Gemma4-31B, HotpotQA, 2Wiki, and MuSiQue contain 1,000 questions per method–dataset cell and Bamboogle contains 125. DictRAG alone contributes 6,250 outcomes over 3,125 distinct questions across the two backbones; this count excludes the agentic retriever rows in Table 7.

These native SQL evaluations use the same implementation, question subsets, and indexes as the Qwen3.5-27B original-corpus evaluation. They use 30 retrieval turns.

A.4 Bamboogle Evaluation Settings

The shared question cohort and corpus are described in Appendix B.2.

The Qwen3.5-9B and Gemma4-31B Bamboogle evaluations use BF16 weights, temperature 0, top-p 1, top-k 0, and disabled thinking. Model requests have a 600-second timeout, trajectories a 3,600-second ceiling, and retrieval a 30-turn budget. Native SQL uses a 30-second SQLite deadline. Retriever agents invoke BM25Okapi or BGE-M3 through native tools and receive top-5 deduplicated results; their reported policy has no fallback answer request.

B Datasets and Corpus Construction

B.1 Original-Corpus Result Provenance

The original-corpus DictRAG evaluation contains 1,000 unique question indices for each of HotpotQA, 2Wiki, and MuSiQue, plus 125 Bamboogle questions. HotpotQA and 2Wiki each use one complete run; MuSiQue combines four predefined, disjoint 250-question ranges. Results are selected by question range, independently of answer correctness.

All newly tabulated means and differences use the displayed per-dataset scores before final rounding. All reported cross-dataset means use the four datasets with equal weight, including Bamboogle. This presentation convention makes the arithmetic reproducible without implying additional precision in the underlying estimates.

B.2 Bamboogle Corpus Construction

The Bamboogle columns in Tables 2, 3, and 7 use the same 125 questions and ordered 13,228-document mixed corpus. They describe corpus-constrained QA rather than live web retrieval. All table averages include Bamboogle with the same dataset-level weight as each 1,000-question benchmark.

The mixed corpus is constructed for this project and is not an official Bamboogle corpus. For the 125 Bamboogle questions, BM25, BGE, and bridge-answer full-text search (FTS) retrieve candidate passages from the 21,015,324-passage DPR Wikipedia-18/W100 collection (Karpukhin et al., 2020). The official DPR release provides the passage file `psgs_w100.tsv.gz`. Deduplicating these retrieved candidates yields 1,572 passages. We combine them with 11,656 documents from the MuSiQue corpus, without cross-source deduplication, giving $1{,}572 + 11{,}656 = 13{,}228$ documents.

B.3 Corpus Scaling Provenance

Each cell in Table 4 contains the same 1,000 questions as the corresponding original-corpus condition. We merge the Original corpora of HotpotQA, 2WikiMultiHopQA, and MuSiQue, then add documents from a pool of approximately 21 million Wikipedia documents to construct one shared corpus of 100,000 documents. We retain all documents in this 100K corpus and add further documents from the same pool to construct a shared corpus of 500,000 documents. All three datasets are evaluated on these same shared corpora; the 100K corpus is therefore a subset of the 500K corpus. Base and added documents are randomly shuffled together within each corpus.

Scaling retrievers. Scaling retrievers use BM25Okapi and BGE-M3 exhaustive ranking.

For each method, the twelve original and expanded corpus conditions reuse 3,000 distinct questions. Outcomes across conditions are paired by question rather than independent samples. The Wikipedia pool version, sampling and shuffle seeds, deduplication rules, and ordering manifests are needed to separate corpus-size effects from composition and order and to reconstruct the collections.

C Retriever Depth Diagnostics

The main agentic retriever comparison uses $k = 5$ for Agentic-BM25/BGE. To characterize the context cost of deeper retrieval, Table 10 reports average observed-document counts and input-token usage at $k \in \{10, 20, 50\}$ on the same 1,000-example benchmark subsets. Increasing k from 10 to 50 substantially increases the amount of observed evidence and input-token usage. This diagnostic does not include a matched cost comparison with DictRAG.

Table 10: Average observed documents and input-token budgets in the full 1,000-example larger-depth sweep. D columns are observed-document counts; Tok columns are input tokens in thousands.

Data	Ret.	D@10	D@20	D@50	Tok@10	Tok@20	Tok@50
HotpotQA	BM25	22.8	45.9	122.2	16.9	26.0	44.8
HotpotQA	BGE	22.0	43.3	117.2	16.4	25.1	44.4
2Wiki	BM25	31.6	62.5	162.8	22.2	34.4	66.4
2Wiki	BGE	30.9	61.3	162.7	23.9	38.9	70.9
MuSiQue	BM25	52.6	104.7	266.8	46.8	79.0	140.5
MuSiQue	BGE	50.7	101.2	262.8	48.1	80.3	137.5

Accuracy at increased retrieval depth. On MuSiQue, Agentic-BM25 reaches 57.69 F1 at $k = 10$ using 46.8K input tokens, and 55.00 at $k = 50$ using 140.5K. Agentic-BGE at $k = 50$ reaches 57.06 F1 using 137.5K input tokens. DictRAG reaches 67.18 F1 using 45.5K. Thus, these larger-depth retriever configurations do not close the gap despite comparable or greater input-token usage. Increasing depth changes the amount and composition of observed evidence; these results do not isolate a pure computation-budget effect. Accuracy is not reported for every cell in the preceding depth table.

D Detailed Execution Cost

Table 11: Original MuSiQue cost summaries at concurrency 100. Token counts are per-question means. Reported seconds per question are concurrent-run summaries, not isolated-request latency measurements.

Method	Input tokens	Output tokens	Reported s/question
DirectLLM	86	106	0.03
BM25	770	108	0.08
BGE	802	112	0.07
Agentic-BM25	31,499	1,371	2.25
Agentic-BGE	32,241	1,311	2.37
Agentic-LexSearch	31,126	291	0.90
SQL-FTS5	47,237	1,931	2.88
DictRAG	45,519	2,172	3.18

Agentic-LexSearch exposes corpus-wide contains/regex search and appears only as a lexical-tool cost reference; no corresponding accuracy is reported here, and it is not identified with the Exact-search ablation. These measurements do not characterize index construction, memory use, or latency scaling on large corpora.

E PAIRED STATISTICAL COMPARISONS

All differences below are in percentage points and compare paired outcomes on the same questions. For a comparison A–B, n_{10} counts questions answered exactly correctly only by A, and n_{01} counts those answered exactly correctly only by B. The two-sided exact McNemar p-value is computed from a binomial null with success probability $1/2$ on these discordant pairs. Holm correction uses separate families of eight original-corpus, eighteen scaling, and three adjacent ablation tests. The CIs are paired bootstrap 95% confidence intervals for differences, without multiplicity correction. Raw and adjusted p-values refer exclusively to EM; no F1 p-values are reported. Values are rounded for presentation; Holm correction is applied before rounding.

Original-corpus comparisons use 1,000 questions per dataset except Bamboogle (125). Scaling and ablation comparisons each use 1,000 questions. BM25 and BGE denote Agentic-BM25 and Agentic-BGE. All comparisons use Qwen3.5-27B. These tests do not establish significance for the broader baseline table, other backbones, or macro averages.

E.1 INTERPRETATION AND SCOPE

Only the MuSiQue EM gains over both agentic retrievers pass Holm correction across the eight original-corpus comparisons (Table 12). The 2Wiki margin over Agentic-BM25 is four additional correct answers out of 1,000 (ΔEM 95% CI: $[-2.40, +3.20]$ points); the Bamboogle margin is one out of 125 ($[-8.00, +9.60]$). Thus, the observed ranking does not establish a significant advantage on every dataset.

Across the eighteen comparisons in Table 13, all six MuSiQue EM gains survive Holm correction. None of the 2Wiki or HotpotQA EM comparisons does. Several pointwise F1 CIs exclude zero, but these are not Holm-adjusted F1 tests.

Relation reuse does not pass the EM test ($p = 0.057373$), and its F1 CI spans zero ($[-1.04, +2.73]$ points). Its EM bootstrap CI ($[+0.10, +1.60]$) excludes zero, but the exact McNemar test does not reject at 0.05; these procedures need not yield identical decisions. We therefore do not claim a statistically established incremental benefit from relation reuse.

The reported confidence intervals are pointwise and conditional on the evaluated runs; they do not quantify variation across seeds or corpus constructions. Statistical tests cover the primary agentic comparisons and progressive ablation, not all broader baselines, cross-model results, or dataset-macro averages. The small Bamboogle cohort yields wide intervals. Failure to reject an EM difference does not establish equivalence.

For the primary Qwen3.5-27B comparisons, we report two-sided exact McNemar tests on paired question-level EM outcomes and paired bootstrap 95% confidence intervals (CIs) for EM and F1 differences. Holm correction is applied separately to three comparison families: the eight original-corpus comparisons, the eighteen expanded-corpus comparisons, and the three adjacent ablation comparisons. We use a Holm-adjusted EM p-value below 0.05 as the significance criterion. Bootstrap CIs are pointwise, not multiplicity-adjusted; the reported p-values test EM, not F1. Appendix E reports all comparisons. These analyses quantify question-level uncertainty for the evaluated runs, not variability across training or inference seeds.

Table 12: Original-corpus paired comparisons for Table 3. Differences are DictRAG minus the comparator. Holm correction covers all eight EM tests.

Condition	Comparator	n_{10}/n_{01}	ΔEM [95% CI]	ΔF1 [95% CI]	Raw p	Holm p
MuSiQue	BM25	157/97	+6.00 [+2.90, +9.10]	+6.64 [+4.09, +9.15]	2.0×10^{-4}	0.001
MuSiQue	BGE	155/89	+6.60 [+3.60, +9.70]	+7.27 [+4.86, +9.72]	2.9×10^{-5}	2.3×10^{-4}
2Wiki	BM25	100/96	+0.40 [-2.40, +3.20]	+0.26 [-1.79, +2.28]	0.830	1.000
2Wiki	BGE	102/94	+0.80 [-2.00, +3.50]	+0.76 [-1.31, +2.81]	0.617	1.000
HotpotQA	BM25	120/98	+2.20 [-0.70, +5.10]	+2.33 [+0.26, +4.38]	0.155	0.929
HotpotQA	BGE	123/102	+2.10 [-0.80, +5.00]	+2.51 [+0.35, +4.67]	0.182	0.929
Bamboogle	BM25	16/15	+0.80 [-8.00, +9.60]	+2.41 [-4.96, +9.68]	1.000	1.000
Bamboogle	BGE	19/15	+3.20 [-5.60, +12.00]	+3.26 [-4.12, +10.69]	0.608	1.000

Table 13: Corpus-scaling paired comparisons for Table 4. Differences are DictRAG minus the comparator within each corpus condition. Holm correction covers all eighteen EM tests.

Condition	Comparator	n_{10}/n_{01}	ΔEM [95% CI]	ΔF1 [95% CI]	Raw p	Holm p
100K MuSiQue	BM25	173/75	+9.80 [+6.80, +12.80]	+10.69 [+8.19, +13.20]	4.4×10^{-10}	5.7×10^{-9}
100K MuSiQue	BGE	187/77	+11.00 [+7.90, +14.10]	+12.02 [+9.48, +14.62]	9.9×10^{-12}	1.5×10^{-10}
100K 2Wiki	BM25	104/90	+1.40 [-1.30, +4.20]	+1.19 [-0.78, +3.18]	0.351	1.000
100K 2Wiki	BGE	103/94	+0.90 [-1.80, +3.70]	+0.42 [-1.55, +2.39]	0.569	1.000
100K HotpotQA	BM25	122/96	+2.60 [-0.30, +5.50]	+2.93 [+0.79, +5.08]	0.090	0.631
100K HotpotQA	BGE	125/87	+3.80 [+1.00, +6.70]	+3.34 [+1.25, +5.40]	0.011	0.098
500K MuSiQue	BM25	200/61	+13.90 [+10.90, +16.90]	+14.61 [+12.12, +17.11]	1.9×10^{-18}	3.5×10^{-17}
500K MuSiQue	BGE	193/73	+12.00 [+8.90, +15.10]	+13.16 [+10.60, +15.69]	1.1×10^{-13}	1.8×10^{-12}
500K 2Wiki	BM25	112/88	+2.40 [-0.40, +5.10]	+2.80 [+0.71, +4.89]	0.104	0.631
500K 2Wiki	BGE	104/95	+0.90 [-1.80, +3.60]	+1.20 [-0.84, +3.22]	0.571	1.000
500K HotpotQA	BM25	141/99	+4.20 [+1.20, +7.30]	+4.49 [+2.25, +6.73]	0.008	0.080
500K HotpotQA	BGE	123/83	+4.00 [+1.20, +6.80]	+4.35 [+2.28, +6.43]	0.006	0.077
Full MuSiQue	BM25	182/77	+10.50 [+7.40, +13.60]	+12.23 [+9.65, +14.80]	5.7×10^{-11}	7.9×10^{-10}
Full MuSiQue	BGE	189/66	+12.30 [+9.20, +15.30]	+13.27 [+10.78, +15.78]	6.4×10^{-15}	1.1×10^{-13}
Full 2Wiki	BM25	117/95	+2.20 [-0.60, +5.00]	+2.53 [+0.36, +4.67]	0.149	0.745
Full 2Wiki	BGE	112/105	+0.70 [-2.20, +3.60]	+0.60 [-1.55, +2.71]	0.684	1.000
Full HotpotQA	BM25	151/107	+4.40 [+1.30, +7.60]	+4.74 [+2.43, +7.05]	0.007	0.080
Full HotpotQA	BGE	131/99	+3.20 [+0.30, +6.20]	+4.01 [+1.88, +6.18]	0.041	0.326

Table 14: Progressive MuSiQue paired comparisons for Table 5. R: ranked-search (Agentic-BM25); E: exact-search; C: composable-search; S: relation reuse (Full DictRAG). Holm correction covers the three adjacent EM comparisons.

Comparison	n_{10}/n_{01}	ΔEM [95% CI]	ΔF1 [95% CI]	Raw p	Holm p
E–R	45/14	+3.10 [+1.70, +4.60]	+3.47 [+1.41, +5.59]	6.53063×10^{-5}	0.000195919
C–E	31/10	+2.10 [+0.90, +3.40]	+2.31 [+0.39, +4.25]	0.00145049	0.00290098
S–C	11/3	+0.80 [+0.10, +1.60]	+0.86 [-1.04, +2.73]	0.057373	0.057373