RETRIEVAL GAINS THAT DO NOT REACH THE READER: WHY RECALL MISLEADS IN BOTH DIRECTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieval-augmented generation is often evaluated by retrieval quality (e.g., Recall@k), implicitly assuming that better retrieval yields better answers. We test this assumption directly. We introduce SCOPE (Schema-Checked One-Pass Expansion), which extends SIRA's validated query expansion to long structured documents (financial filings and papers). In a single call, SCOPE proposes additional search terms and likely table locations for the answer, each proposal is validated against the document before one search is issued.

On 539 questions from UDA, SCOPE improves Recall@10 over SIRA by 7.6 points, and by 13.6 points when the answer is in a table. Its retrieval is statistically indistinguishable from a five-turn ReAct agent (-0.003, with interval $[-0.036, 0.027]$) while using over $5\times$ fewer search tokens, and a one-turn agent matches it more cheaply still. In filings, much of the gain comes from a single instruction: preserve every year and fiscal period, since the year is often not the answer but the column it appears in.

Despite delivering correct evidence more often, SCOPE does not measurably change answer accuracy. The net effect is churn, and evidence arrives for 49 questions but disappears for 18 that were previously answered well (0.674), while newly served questions are answered worse (0.502). Aggregate recall hides these transitions and can mislead in the other direction. At the same time, a re-ranker that reduces SCOPE by 5.4 Recall@10 points causes no measurable accuracy loss. A five-search agent improves accuracy with no measurable recall gain by collecting more varied context. We therefore recommend reporting evidence transitions (which questions gain vs. lose evidence) and accuracy conditional on evidence, alongside Recall@k. We include document-level intervals throughout and report our preregistered targets and negative results in full.

1 INTRODUCTION

Retrieval-augmented generation (Lewis et al., 2020) is evaluated mostly on retrieval metrics, such as Recall@k, on the assumption that locating the right passage is the difficult part and that answering from it follows. These systems now answer questions over financial filings, contracts, and research papers, where a wrong figure is worse than no answer. We test the assumption and find it does not hold in either direction, and testing it needs a retrieval improvement large enough to trace, so we begin by building one.

Retrieval over such documents fails on vocabulary it cannot see. A question about "borrowing costs" must find a row labeled *Interest expense, net*, so the string identifying the evidence is often a column header or row label rather than a content word. Agents attack this by searching repeatedly (Yao et al., 2023; Trivedi et al., 2023; Jiang et al., 2023; Shao et al., 2023), while single-pass approaches improve the query instead (Gao et al., 2023; Wang et al., 2023; Ma et al., 2023), and SIRA (Yang et al., 2026) adds a safeguard to the second family, validating each generated term against the corpus so a hallucinated term cannot pollute the search.

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.

Our instrument is SCOPE, built on SIRA. Different kinds of generated queries need different checks. A search term can be checked by how often it occurs, but the answer in the 2007 column cannot, because "2007" occurs on nearly every page. Yang et al. (2026) check such claims against curated metadata, such as Wikipedia's category graph, but a filing carries none, so SCOPE builds the equivalent from the document's own layout. One call proposes terms and layout locations, each is validated against the index able to detect its errors, and the system searches once.

Tracing the gain separates two steps that can fail independently. The evidence must reach the model's context, and the model must then use it. SCOPE succeeds at the first and fails at the second. Answer accuracy does not measurably change because, per question, the net delivery gain is evidence moving both into and out of the window, and the questions that lose it are the ones the reader was answering best, and what the newly reached questions add is drained away again. A search agent shows the reverse, gaining accuracy from repeated searching with no measurable retrieval gain. The metric therefore misleads in both directions, so methods compared on it alone, ours among them, may be ordered incorrectly.

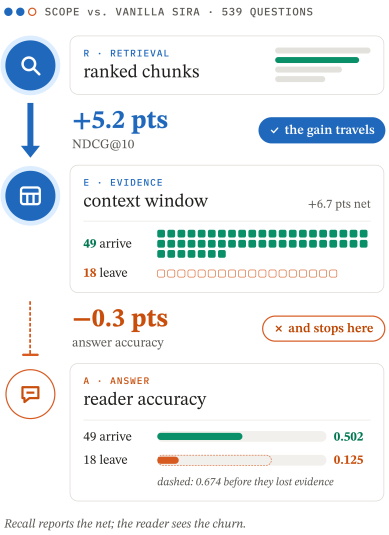

Figure 1: The retrieval gain reaches the reader's context and stops there. While SCOPE delivers better retrieval than SIRA

This paper makes four contributions.

- SCOPE, the single-pass method the rest of the paper measures with, and a strong enough instrument for the measurement to mean something, at $+7.6$ points of Recall@10 over SIRA, not separable from a five-turn ReAct agent (-0.003 [-0.036, 0.027]) at $5.5\times$ fewer search tokens (Section 3).
- A one-line rule for query modification, to keep every year and fiscal period. On financial filings, it is worth 3.7 points on its own, a post hoc finding reported as such (Section 5.2).
- Better retrieval does not mean better answers, and net recall hides why. SCOPE gets evidence to the model 6.7 points more often on net while moving 67 questions in both directions, and answer accuracy does not change (Section 5.3).
- The reverse also holds, since five searches instead of one raise an agent's accuracy by 2.0 points with no measurable retrieval gain. Reporting the delivery transition between two methods, not just net recall, catches both cases (Section 5.4).

2 Related Work

Expanding a query with related terms predates neural IR (Rocchio, 1971; Abdul-Jaleel et al., 2004), and neural variants generate the expansion (Gao et al., 2023; Wang et al., 2023; Jagerman et al., 2023), and document-side expansion (Nogueira et al., 2019) and learned sparse representations (Formal et al., 2021) attack the mismatch from the index. All trust the produced text, while SIRA (Yang et al., 2026) validates it against document frequency and, where the corpus carries curated structure, against that structure. That principle is SIRA's, not ours. What is new is that our corpus has none to compare against, so we *construct* the index from the document's own layout and validate positional keys (`Table:`, `Column:`, `Row:`) rather than topical ones. That index is per-document and derived rather than curated, a difference that bounds how much weight it can carry (Section 5.5). Ma et al. (2023) rewrite the question instead, and a rewrite *replaces* the surface form and can drop misleading words where expansion can only add.

Agentic retrieval interleaves reasoning with tool calls (Yao et al., 2023; Trivedi et al., 2023; Jiang et al., 2023; Shao et al., 2023; Asai et al., 2024). Adaptive-RAG (Jeong et al., 2024) routes on predicted question complexity, the closest antecedent to our uncertainty gate, though it routes among pipelines with a trained classifier, whereas we route on a signal the lexical index already provides.

This literature rarely reports accuracy gained per token spent, which Section 5.4 measures exactly that.

The sharpest case for single-pass over agentic search is Yang et al. (2026): multi-round exploratory search should be compressed into one corpus-discriminative action, and they outperform a browsing agent at every budget over a 25.6M-document Wikipedia corpus. SCOPE adopts that thesis and tests it where it is hardest to win, inside a single already-identified document where lexical retrieval alone reaches 0.700 Recall@10. We reproduce the qualitative result and not the margin. One pass reaches the same Recall@10 as a five-turn agent rather than beating it (-0.003, interval spanning zero) at $5.5\times$ fewer retrieval-stage tokens (Section 5.1).

That comparison is also conducted on the wrong quantity, since both sides report Recall@k, which on this task neither predicts nor bounds end-to-end accuracy in either direction (Section 5.3, Section 5.4). Whatever the right resolution of single-pass versus agentic retrieval, it cannot be read off the metric the debate currently uses.

Structure-aware retrieval usually imposes organization the corpus does not carry: RAPTOR (Sarthi et al., 2024) builds a summary tree, GraphRAG (Edge et al., 2024) an entity graph. Our schema graph records the structure the PDF already has and uses it only to *validate* proposed keys, never to reorganize the corpus.

That retrieval and answer quality can diverge is not unprecedented: Salemi and Zamani (2024) show retrieval metrics correlate poorly with downstream RAG performance, Ru et al. (2024) separate retriever from generator diagnostics including context utilisation, Joren et al. (2025) show models do not succeed even on appropriate context and argue for a sufficiency criterion rather than the presence of any relevant passage, Cuconasu et al. (2024) find the highest-ranked documents are not those that most help the generator, and Liu et al. (2024) show models use long contexts unevenly. We add a decomposition that identifies *which* link breaks, with a method that yields large, statistically solid retrieval gains.

BEIR (Thakur et al., 2021), the standard zero-shot suite, and SIRA are sets of clean passage collections. UDA (Hui et al., 2024) keeps the source PDFs of filings (Chen et al., 2021), hybrid reports (Zhu et al., 2022), and papers (Dasigi et al., 2021), which is what makes the structural axis measurable.

3 METHOD

SCOPE takes a question q over a document D and produces one ranked list in a single pass (Figure 2). One call emits three typed expansions, a validator checks each type against the index to falsify it, and one weighted pass ranks the document's chunks. An uncertainty gate that skips the call applies only to the additive configuration, and is described first. The design is nested, since $\beta = 0$ with $q' = q$ recovers vanilla SIRA and $\alpha = \beta = 0$ recovers BM25, so every ablation later is a parameter setting of one implementation rather than a separate system.

We follow one question from the benchmark throughout: *how much did interest expense change from 2006 to 2007?*, asked of a 187-page annual report split into 891 chunks C. Write $\mathrm{df}(t)$ for the number of chunks containing token t, and $\mathrm{BM25}(x, d)$ for the score of chunk d under query text x.

3.1 UNCERTAINTY-GATED ROUTING

Not every question needs an LLM, and the lexical index tells us which ones don't. If BM25 on the raw question produces one clear winner, the words were enough. Ours does not, because hundreds of chunks mention interest and *2007* appears on nearly every page. We measure that flatness as the normalised entropy of the top-k scores, $\mathcal{H}(q) = -\sum_{i \le k} p_i \log p_i / \log k$ with $p_i \propto \mathrm{BM25}(q, d_i)$, and expand only when $\mathcal{H}(q) > \delta$. At $\delta = 0.570$, chosen on development data, two-fifths of questions skip the model at no cost in recall.

The gate applies to the additive configuration only. Skipping an *added* expansion leaves a well-formed search, while skipping one that *replaces* the query, as in the configuration we report as SCOPE, would gate away the query itself. SCOPE therefore issues one expansion call per question, averaging 2.00 LLM calls including the answer against 1.63 for the ablation that keeps the initial

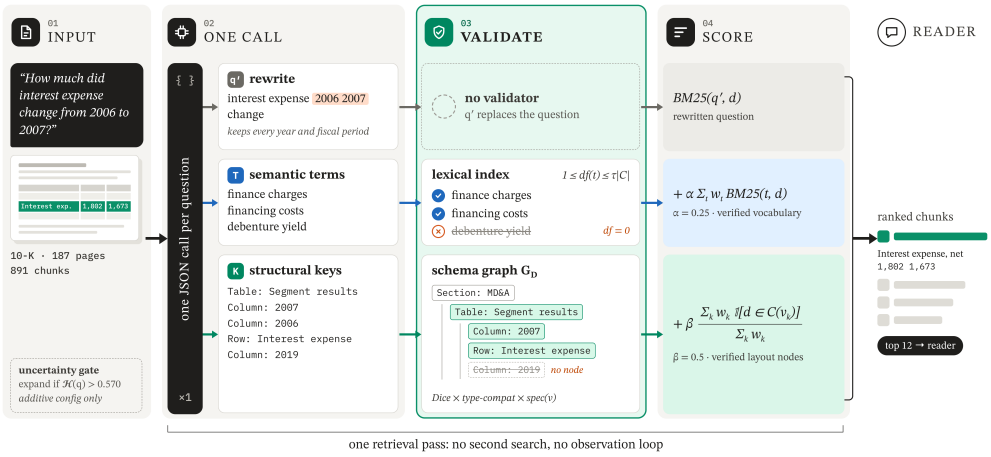

Figure 2: SCOPE system architecture. The single-pass design resolves vocabulary and structural gaps by generating and validating semantic and layout cues directly against the document, avoiding the need for costly iterative search loops.

input (Table 10). We report the gate here as part of the design space, not because it is active in the headline row.

3.2 One Call, Three Axes

A question can be underspecified in three ways: it may lack the document's vocabulary (ours says *borrowing costs*, the filing says *interest expense*), it may not say *where* the evidence sits, and its surface form may mislead. One JSON-mode call addresses all three, returning semantic terms (*finance charges*), typed layout strings (`Table: Segment results`, `Column: 2007`, `Row: Interest expense`) and an updated query q' that must preserve every year, quarter and fiscal period, so *from 2006 to 2007* survives as *2006 2007*, because a year is not the answer but the column it sits in (Section 5.2). The semantic axis is prompt-identical to our SIRA reimplementation, so differences come from the added axes rather than prompt drift, and an unparseable response degrades to plain BM25 (Section H).

3.3 Dual-Index Validation

None of this is trustworthy yet. A model asked for vocabulary it cannot see will invent some, and an invented term that conforms to text elsewhere is worse than none. We keep SIRA's rule, that a term survives when $1 \le \mathrm{df}(t) \le \tau|C|$ with $\tau = 0.3$, so *finance charges* survives while an invented *debenture yield* occurs nowhere and is discarded, as is anything too common to discriminate. Terms already in the question are dropped, and a failing phrase backs off to the tokens that pass, at reduced weight. Survivors form T with weights w_t.

The layout strings cannot be checked this way, and here our setting departs from SIRA's. `Column: 2007` holds a token of unremarkable document frequency, so the lexical index cannot say whether such a column exists, since it indexes words rather than structure. A map of the document's layout can, and we build one while parsing. The schema graph G_D carries one node v per section, table, column header and row label, with token set $\mathrm{tok}(v)$ and covered chunks $C(v)$. Each key keeps its best-matching node v_k under

$$\mathrm{sim}(k, v) = \frac{2, |\mathrm{tok}(k) \cap \mathrm{tok}(v)|}{|\mathrm{tok}(k)| + |\mathrm{tok}(v)|} \cdot \mathrm{compat}\big(\mathrm{type}(k), \mathrm{type}(v)\big), \qquad \mathrm{spec}(v) = \frac{\log!\big(1 + |C|/|C(v)|\big)}{\log\big(1 + |C|\big)}, \tag{1}$$

with compat equal to 1 for an exact type match, 0.6 across incompatible types and 0.9 when untyped. So `Column: 2007` matches each node while a hallucinated `Column: 2019` matches nothing. Specificity sets what a match is worth, so a row covering one chunk approaches 1, a section

spanning the document approaches 0, which stops a key from naming a generic header such as *Total* from promoting half the filing. Survivors form K with weights $w_k = \mathrm{sim}(k, v_k), \mathrm{spec}(v_k)$.

3.4 Single Weighted Pass

What survives validation is searched for once:

$$\mathrm{Score}(d) = \mathrm{BM25}(q', d) + \alpha \sum_{t \in T} w_t \, \mathrm{BM25}(t, d) + \beta \, \frac{\sum_{k \in K} w_k \mathbb{1}[d \in C(v_k)]}{\sum_{k \in K} w_k}. \tag{2}$$

The first term searches the rewritten question, the second the verified vocabulary, and the third raises the chunks that verified layout nodes cover, normalized so its scale does not depend on how many keys survived, and the chunk holding our two figures scores on all three. There is no second round and no observation loop. The scorer carries no dense term, since document frequency has no dense analog, and a row reading *Interest expense, net 1,802* carries little distributional signal. At the same time, its label is an exact lexical match for a validated key. Table 1 reports a dense baseline so the cost is visible.

4 Experimental Setup

We evaluate on three UDA subsets (Hui et al., 2024): **FinHybrid** (FinQA (Chen et al., 2021)), **TatHybrid** (TAT-DQA (Zhu et al., 2022)) and **PaperText** (Qasper (Dasigi et al., 2021)). We sample 160 documents (seed 2027), parse with PyMuPDF, and chunk at 800 characters with 150 overlap rather than UDA's 3000 (Section G reports both). This yields 107 development and 539 test questions, split by document hash, and the answer generator receives the top 12 chunks. We compare against BM25 (Robertson and Zaragoza, 2009), a dense retriever (Wang et al., 2022), HyDE (Gao et al., 2023), chain-of-thought expansion (Wei et al., 2022), vanilla SIRA, and ReAct (Yao et al., 2023) at one, two, and five turns, all reading the same top 12 chunks through the same answer prompt.

UDA supplies evidence spans rather than chunk relevance, so gold labels come from length-scaled containment with a numeric-agreement criterion for table evidence, we exclude questions whose evidence cannot be located from retrieval means and count them (460 of 539 are locatable).

We report unit-level Recall@k, binary NDCG@k (Järvelin and Kekäläinen, 2002), MRR, end-to-end QA accuracy (exact match with 1% tolerance for FinQA, numeracy-aware F1 for TAT-DQA, token F1 for Qasper), and `evidence_in_context`: whether any relevant chunk reached the generator's window. Intervals are 95% document-clustered bootstrap (Efron and Tibshirani, 1993), resampling documents rather than questions, since treating questions as independent would understate them (Fuhr, 2017).

One frozen model serves all methods: Llama-3.3-70B-Instruct (Grattafiori et al., 2024) at temperature 0, with a 131k context window. $\alpha = 0.25$, $\beta = 0.5$ and $\delta = 0.570$ are selected on the development split only and frozen before test scoring (Section B reports the 33-variant sweep from which these came). Section H varies the model.

Our "Vanilla SIRA" row reimplements the query-side validated expansion on this corpus, not the published system or its absolute numbers. It omits the corpus-side enrichment, infeasible at our document count; the metadata-graph validation, which has no counterpart on documents without a curated category structure; and the margin filter of Yang et al. (2026), whose omission may weaken it.

5 Results

5.1 Retrieval, and What It Costs

SCOPE reaches 0.797 Recall@10 (Table 1). Against every non-agentic baseline the margin is decisive: $+0.098$ [0.061, 0.136] over BM25, $+0.076$ [0.039, 0.114] over vanilla SIRA, and $+0.136$ [0.079, 0.193] over SIRA on table-heavy questions. Against the agent, it is not distinguishable, with paired differences against all three ReAct budgets spanning zero.

Table 1: **Retrieval** on 539 UDA test questions (Recall@10 / NDCG@10). "Tokens" is mean retrieval-stage tokens per question, and the answer call adds ≈2800 for every method. Best per column in bold. SCOPE is shown at $\beta = 0.5$, selected on overall development recall, while the pre-registered criterion picked $\beta = 1.0$, which reaches 0.777 on test (Section B).

Method	FinHybrid	TatHybrid	PaperText	R@10	N@10	Tok.
BM25	0.780 / 0.588	0.746 / 0.508	0.543 / 0.318	0.700	0.481	**0**
Dense E5	0.755 / 0.508	0.722 / 0.514	0.611 / 0.394	0.701	0.477	**0**
HyDE	0.708 / 0.497	0.637 / 0.400	0.643 / **0.414**	0.664	0.439	257
CoT	0.803 / 0.565	0.802 / 0.515	**0.678** / **0.414**	0.767	0.504	326
Vanilla SIRA	0.786 / 0.575	0.746 / 0.513	0.610 / 0.366	0.721	0.493	243
SCOPE − rewrite	0.782 / 0.572	0.820 / 0.553	0.588 / 0.343	0.740	0.499	240
ReAct, 1 turn	0.878 / **0.651**	0.848 / 0.594	0.613 / 0.379	0.791	0.552	201
ReAct, 2 turns	0.876 / 0.640	0.848 / 0.594	0.662 / 0.403	**0.804**	0.555	1095
ReAct, 5 turns	**0.886** / 0.641	0.829 / 0.592	0.660 / 0.408	0.800	**0.556**	2753
SCOPE	0.861 / 0.621	**0.865** / **0.602**	0.636 / 0.382	0.797	0.545	501

The remaining comparison is cost, and SCOPE wins on every measure recorded. It spends 501 retrieval-stage tokens per question against the five-turn agent's 2753, makes 2.00 LLM calls against 3.90, and is 1.65× faster at the retrieval stage and 1.19× end-to-end (Table 10, where every ratio divides one column mean by the other). The pre-registered latency gate was written against the no-rewrite configuration, whose router leaves it 0.37 fewer calls (1.63 against 2.00) and widens the retrieval-stage gap to 1.96× [1.78, 2.20] (Section K). The 5.5× is retrieval-stage tokens, which the shared answer call reduces to 1.65× end-to-end, the one-turn agent is cheaper on both tokens and calls while staying level on recall, making it the honest cost comparison.

5.2 The Largest Gain Is One Instruction

Comparing the queries the two methods issued on the same FinHybrid test questions, the agent carried the question's year tokens on 70% of them, and our rewrite, before this clause was added, on 32%. Requiring the rewrite to preserve every year, quarter, and fiscal period, including both endpoints of a range, is worth +0.037 Recall@10 on FinHybrid and +0.060 table-heavy from one prompt clause. We made the diagnosis on test queries after scoring the pre-registered configuration, so the clause and its gain are post hoc and excluded from every pre-registered comparison.

5.3 The Gains Do Not Reach the Answer

This is the paper's central result. Decomposing SCOPE against vanilla SIRA, +0.052 NDCG@10 produces +0.067 evidence delivered to the generator's window and −0.003 answer accuracy. The first link works and the second does not, and the per-question records say why. The net gain is churn.

Of the 460 locatable test questions, SCOPE delivers evidence where vanilla SIRA does not on 49 and loses it on 18, for a net gain of 31 and +0.067. Those four cells decompose the null, and weighted by size they sum to −0.001 against the measured −0.003, the difference coming from the 79 non-locatable questions, which are scored for answers but have no delivery cell (Table 2). First, on the 352 questions both deliver, SCOPE answers 0.025 fewer correctly with an interval spanning zero [-0.059, 0.008], and the table-heavy restriction agrees (−0.024 [-0.079, 0.031], $n = 152$): **given the same delivered evidence we cannot show that SCOPE's context is worse to read**, only that the sign points that way. Second, the 49 questions it newly reaches convert at 0.502 versus 0.567 on the shared set. Hence, the drop in conditional accuracy from 0.596 to 0.559 reflects dilution by harder questions rather than a failure to use what arrives. The reader does use it, since vanilla SIRA scores 0.158 on those same questions. Third, the 18 questions that lose their evidence were being answered at 0.674, and SCOPE answers them at 0.125. It trades questions the reader was converting well for questions it converts less well. Neither drain cancels the gain on its own. The +0.037 the new questions contribute is offset by two of equal size, −0.022 from the 18 lost and −0.019 from the slight decline on the 352 shared, and the 49- and 18-question cells are small enough that their accuracies carry wide uncertainty. A seven-point recall gain moves evidence both into and out of

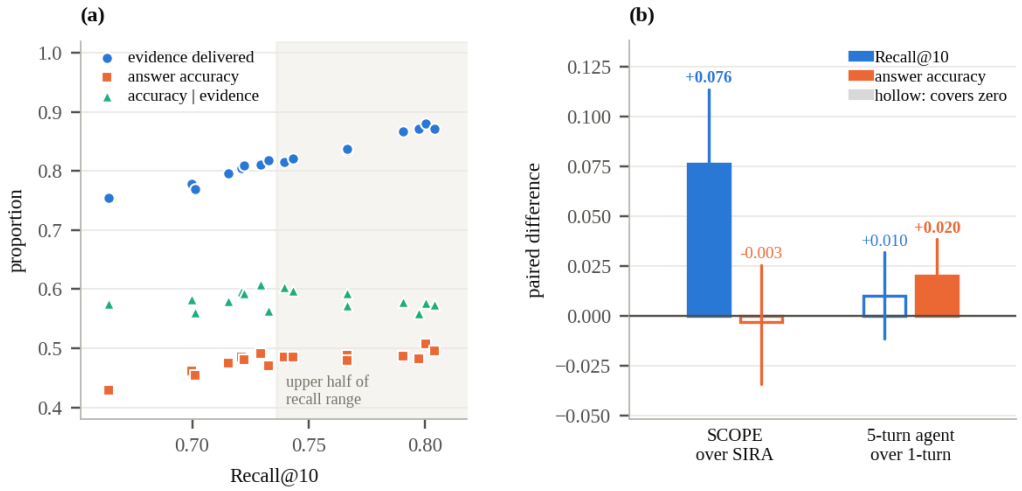

Figure 3: **The proxy fails in both directions.** **(a)** Across sixteen methods, delivery rises with Recall@10 throughout while answer accuracy flattens and conditional accuracy declines. **(b)** Paired differences with 95% document-clustered intervals, hollow markers covering zero.

the window, and aggregate recall reports only the net. Evidence presence is worth $+0.384$ accuracy on average, so the net gain should have been worth $+0.026$ end to end, which the measured interval [-0.035, 0.026] does not exclude. Nor is it a retrieval ceiling: Recall@100 reaches 0.983 against 0.797 at 10 (Section D). Failure is not particular to gains. A reranker that costs SCOPE 0.054 Recall@10 and 0.033 of its delivery does not cost it answers ($+0.013$ [-0.017, 0.043], Section F).

Every retrieval advantage in Table 1 is decisive, and no answer difference is, except one that runs against us: -0.025 [-0.051, -0.001] against the five-turn agent.

The null is an aggregate, not a uniform failure. On table-heavy questions, the no-rewrite configuration gains $+0.033$ [0.009, 0.059] answer accuracy in comparison to vanilla SIRA, an interval clear of zero (G3, Section I). By subset, TatHybrid converts ($+0.119$ recall yields $+0.021$ accuracy) while FinHybrid ($+0.075 \rightarrow -0.011$) and PaperText ($+0.026 \rightarrow -0.023$) do not, and SCOPE's PaperText accuracy of 0.298 is the second lowest of the ten methods we score while its recall there is mid-table. PaperText has no tables, so whatever goes wrong there is not about table rows (Figure 3, Section C).

5.4 Iteration Buys Answers, Not Rankings

The same agent at one, two, and five turns gives Recall@10 of 0.791, 0.804 and 0.800, a spread of 1.3 points across a $13.7\times$ difference in token spend. Paired against one turn, five are worth $+0.010$ [-0.012, 0.032] Recall@10 but $+0.020$ [0.002, 0.038] on answers, an interval clear of zero.

What they buy is visible in the conditional accuracies. When the gold chunk is *missed*, accuracy rises monotonically with the number of searches fused: 0.192 at one turn, 0.255 at two, 0.294 at five. Each turn adds another ranking to the reciprocal-rank fusion (Cormack et al., 2009), so the window diversifies where a single sharpened query fills it with near-duplicates, while SCOPE, issuing one query, sits at 0.208. Diversity does not help you hit the target; it helps you survive a miss, and Recall@10 cannot see the difference.

5.5 What Structure Adds, and What Fails

SCOPE is decisive on the questions the structural axis targets, $+0.136$ table-heavy over vanilla SIRA. Still, that figure is the full method, and the axis alone contributes $+0.057$ (G1), and against a rewrite control it adds nothing, so rewriting subsumes most of what structure was for. Three attempts to make it carry more weight all fail, and fail harder the more weight they are given, and a cross-

encoder reranker over the top 100 fares worse still, costing SCOPE 0.054 Recall@10 concentrated on table rows (Section F). Validated structure is a useful prior and an unreliable constraint, and what separates the two cases is whether it is curated, as Wikipedia's category graph is, or derived from a parser, as ours is.

5.6 Pre-Registered Gates

We fixed five gates before scoring any test question. All five failed (Section I): three measure real effects short of the bar we set, and two promised a $3\times$ latency advantage where we measured $1.96\times$, with none at all against the one-turn agent our results make the relevant comparison. We report them as written.

6 Conclusion

We set out to improve retrieval over real documents, and SCOPE does so. It gains 7.6 points of Recall@10 over SIRA and matches a five-turn agent at a fifth of its retrieval cost. Our delivery analysis then shows exactly how these gains reach the reader. A net delivery gain combines one gain and two losses. The newly reached questions convert at 0.502, the questions that stop being reached convert at 0.674, and the shared questions convert slightly worse. SCOPE reliably delivers more evidence to the reader's context window, and this analysis identifies the reader as the component that determines final accuracy.

Two further results sharpen this picture. First, the structural axis that motivated the method is a genuinely useful prior. On its own, it contributes $+0.057$ on table-heavy questions, and our ablations show clearly how best to use it: raising β, propagating along schema edges, and localizing by region all degrade performance in proportion to the weight they add, so structure works best as a light prior rather than a hard constraint. Second, the pattern holds in both directions. When we removed recall instead of adding it, reranking the top 100 with a cross-encoder caused SCOPE to lose 0.054 Recall@10 but no measurable accuracy. Seeing the same behavior under both interventions makes this our most robust finding.

These results carry two practical lessons. For system builders, retrieval improvements pay off while evidence reaches the context window less than roughly 82% of the time. SCOPE reaches 87%, so it offers a strong, efficient retrieval base, and the reader is the component to improve on top of it. For evaluation, we recommend reporting the delivery transition between two methods alongside net recall. It costs one table, it can be computed from records these papers already keep, and it separates a method that finds new evidence from one that trades some away. On this task, a net gain of thirty-one questions concealed sixty-seven questions of movement, a difference that no standard retrieval metric we report would have revealed.

Use of AI Assistance

Anthropic's Claude wrote the implementation, evaluation, and plotting code, and drafted the revised prose. The authors specified, ran, and inspected all experiments, and read every reported number from result files. Four method variants arose in discussion with the model: schema-graph propagation and structural localization (negative results, Section 5.5), the replace-versus-union rewrite choice, and the locator clause (Section 5.2). We selected all of these on development data per Section 4, except the locator clause, which we diagnosed on test queries and report post hoc. The authors verified every citation and take full responsibility for the paper's contents.

Reproducibility Statement

We release the code, configurations, per-question traces, and the pre-registered gates with this paper. Section J maps each name used in the paper to its name in the code, and Section L lists the contents of the workbook. Each experiment runs by execution ID from a single entry point, and every result directory includes a configuration block that records the code hash, model, calibration values, and library versions. All materials are available at `https://anonymous.4open.science/r/scope-schema-checked-one-pass-expansion-4897/`.

REFERENCES

Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. UMass at TREC 2004: Novelty and HARD. In *Proceedings of the Text REtrieval Conference (TREC)*, 2004.

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 (ICLR)*, 2024.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of EMNLP*, 2021.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of SIGIR*, 2009.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. The power of noise: Redefining retrieval for RAG systems. In *Proceedings of SIGIR*, 2024.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of NAACL-HLT*, 2021.

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

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. SPLADE: Sparse lexical and expansion model for first stage ranking. In *Proceedings of SIGIR*, 2021.

Norbert Fuhr. Some common mistakes in IR evaluation, and how they can be avoided. In *ACM SIGIR Forum*, volume 51, pages 32–41, 2017.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the Association for Computational Linguistics (ACL)*, 2023.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yulong Hui, Yao Lu, and Huanchen Zhang. UDA: A benchmark suite for retrieval augmented generation in real-world document analysis. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. Query expansion by prompting large language models. *arXiv preprint arXiv:2305.03653*, 2023.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of NAACL-HLT*, 2024.

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 EMNLP*, 2023.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval augmented generation systems. In *International Conference on Learning Representations*, 2025.

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 (NeurIPS)*, 2020.

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 (TACL)*, 12:157–173, 2024.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting for retrieval-augmented large language models. In *Proceedings of EMNLP*, 2023.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*, 2019.

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

J. J. Rocchio. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice-Hall, 1971.

Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, Zizhao Zhang, Binjie Wang, Jiarong Jiang, Tong He, Zhiguo Wang, Pengfei Liu, Yue Zhang, and Zheng Zhang. RAGChecker: A fine-grained framework for diagnosing retrieval-augmented generation. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, 2024.

Alireza Salemi and Hamed Zamani. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations (ICLR)*, 2024.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of EMNLP*, 2023.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021.

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 Association for Computational Linguistics (ACL)*, 2023.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Liang Wang, Nan Yang, and Furu Wei. Query2doc: Query expansion with large language models. In *Proceedings of EMNLP*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Zeyu Yang, Xu Han, Qi Ma, Jason Chen, and Anshumali Shrivastava. Superintelligent retrieval agent: The next frontier of agentic retrieval. *arXiv preprint arXiv:2605.06647*, 2026. URL https://arxiv.org/abs/2605.06647.

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 (ICLR)*, 2023.

Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. Towards complex document understanding by discrete reasoning. In *Proceedings of the ACM International Conference on Multimedia (MM)*, 2022.

A LIMITATIONS

Our evaluation covers 160 documents, 539 test questions, and three subsets of one benchmark family. At this scale, effects of roughly three points or more are replicated from development to test, which is why the $+0.136$ table-heavy gain is robust. The smaller $+0.016$ structural effect falls below that resolution and would need a larger evaluation to confirm. We also do not include a long-context baseline, so whether retrieval is necessary at these document lengths remains an open question. Finally, the 82% saturation threshold is a median split across sixteen methods. It characterizes the region our methods occupy rather than a fitted knee, and estimating that knee directly is a natural extension.

The within-question crowding account is consistent with our results but remains unestablished. On the 352 questions where both configurations retrieve the evidence, SCOPE answers 0.025 fewer correctly, with an interval that includes zero. Likewise, for the 18 questions whose evidence is lost after the rewrite, none of the features we measure predicts which questions are affected. Future work will explain these cases.

We derive gold labels rather than annotate them. Because containment-based scoring can credit numerically similar rows, part of the conversion gap may reflect the metric rather than the reader. A human-annotated subset would separate these two explanations.

B DEVELOPMENT SWEEP

Seven of the 33 configurations were swept on the 107 development questions, with α and β swept per configuration: two of the three tied at the top, the union variant and the no-structure control that led the two secondary metrics, the base configuration, and two of the weakest. The rest are pseudo-relevance-feedback, fusion, weighting, and localization settings, which is why the sweep is reported at all rather than presented as a search that found something. **Twenty-six of the thirty-three score identically to three decimals in one of four ties** (11, 10, 3 and 2 configurations): most of this design space does not move the selection metric at all, and what does move it moves it downwards, the sweep running from 0.741 to 0.419. Every configuration, including the omitted ones, is in the released workbook (`dev_variants`). The pre-registered selection criterion is table-heavy Recall@10, on which vanilla SIRA scores 0.6296 on that slice.

Variant	α	β	TH R@10	ALL R@10	ALL N@10
rewrite + structure	0.25	1.0	**0.741**	0.715	0.458
localise $\times 0.5$	0.25	0.25	**0.741**	0.743	0.497
rewrite $\cup$ query + structure	0.25	0.5	0.730	**0.760**	0.521
rewrite only (β=0)	0.25	0.0	0.730	0.732	**0.529**
current (sum / flat)	0.25	0.5	0.685	0.738	0.501
component RRF	0.25	0.25	0.585	0.696	0.421
localise $\times 8$	0.25	0.25	0.419	0.505	0.313

The criterion decides the winner. Selecting on table-heavy recall (the pre-registered objective) picks *rewrite + structure* at $\beta = 1.0$ (0.741 dev table-heavy, 0.777 test overall). The global (α, β) grid, which selects on overall development recall, picks $\beta = 0.5$ (0.797 test overall). Both are development selections, and neither used test data. Their disagreement follows the same pattern the paper reports across: optimizing a narrow target costs performance on everything else. Note also that *rewrite* $\cup$ *query* leads on overall development recall (0.760) while losing on table-heavy, and

loses to replace on test (0.781 vs. 0.797): the ranking under the non-pre-registered criterion does not survive the split.

C WHERE THE GAIN IS LOST

The delivery transition behind Section 5.3, then the same two links for five methods.

Table 2: **Delivery churn, and how it cancels.** The 460 locatable test questions by which configuration put evidence in the reader's window, alongside each cell's accuracy and its contribution to the aggregate answer difference. Two drains of equivalent size offset the gain on the newly reached questions. The two small cells carry wide uncertainty on 49 and 18 questions.

Evidence delivered by	n	SCOPE	Vanilla SIRA	difference	contribution
both	352	0.567	0.592	−0.025 [-0.059, 0.008]	−0.019
SCOPE only	49	0.502	0.158	+0.344	+0.037
vanilla SIRA only	18	0.125	0.674	−0.549	−0.022
neither	41	0.245	0.207	+0.038	+0.003
all	460				−0.001

Five methods are compared, from plain BM25 to the five-turn agent, with both pipeline links side by side. "In context" is the share of the 460 locatable questions where a relevant chunk reached the generator's 12-chunk window, while the two conditional columns split those same 460 locatable questions by whether it happened. The QA column is overall 539, so the conditionals do not average to it: the difference is the 79 non-locatable questions, which are scored for answers but excluded from retrieval means. Reading down the table, retrieval and delivery rise together while accuracy does not, and accuracy *given* evidence runs in the opposite direction: SCOPE retrieves best among the single-pass methods and converts worst, and its no-rewrite ablation runs the reverse. Figure 3 plots all sixteen scored methods, and the released workbook carries the rest.

Table 3: **Both links, five methods.** Sorted by Recall@10.

Method	R@5	R@10	NDCG@10	in context	QA	QA,|,ev.	QA,|,no ev.
BM25	0.597	0.700	0.481	0.778	0.461	0.583	0.171
Vanilla SIRA	0.613	0.721	0.493	0.804	0.485	0.596	0.180
SCOPE − rewrite	0.611	0.740	0.499	0.815	0.485	**0.603**	0.157
SCOPE	0.682	0.797	0.545	0.872	0.482	0.559	0.208
ReAct, 5 turns	**0.695**	**0.800**	**0.556**	**0.880**	**0.507**	0.577	**0.294**

D RETRIEVAL AT DEPTH 100

Recall at every depth we score, so the gap between what a method *finds* and what it *ranks* is visible in one place. Best per column in bold.

Table 5 is the reason this paper treats the remaining gap as a ranking problem.

E WHAT STRUCTURE ADDS, AND WHAT IT DOES NOT

Section 5.5 states the conclusion, and the evidence is here.

Against vanilla SIRA, the full method is decisive on the questions the structural axis targets: table-heavy Recall@10 improves by +0.136 [0.079, 0.193], of which the axis alone contributes +0.057 [0.022, 0.100] (G1). The baseline it improves on is instructive. **Vanilla SIRA is worse than plain BM25 on table-heavy questions** (0.689 vs. 0.694 on test, 0.630 vs. 0.641 on dev), because semantic expansion unvalidated for structure matches prose discussion of a concept rather than the cell holding its value.

Table 4: **Recall at depth.** The same 460 locatable test questions at $k = 5, 10, 20, 100$. The spread at $k = 5$ is 13.7 points and at $k = 100$ it is 9.4; most of the difference between these methods is ordering, not reach.

Method	R@5	R@10	R@20	R@100
BM25	0.597	0.700	0.790	0.949
Dense E5	0.599	0.701	0.800	0.977
HyDE	0.558	0.664	0.773	0.889
CoT	0.625	0.767	0.860	0.962
Vanilla SIRA	0.613	0.721	0.825	0.946
SCOPE − rewrite	0.611	0.740	0.815	0.955
ReAct, 1 turn	0.677	0.791	0.881	0.975
ReAct, 2 turns	0.686	**0.804**	0.882	0.977
ReAct, 5 turns	**0.695**	0.800	**0.886**	0.975
SCOPE	0.682	0.797	0.879	**0.983**

Table 5: **Retrieval at depth 100.** The same 539 questions at $k = 100$. Every method recovers most of the evidence somewhere in the top hundred, so differences in Table 1 reflect *ranking*, not what was found. ALL columns are pooled over questions.

Method	FinHybrid	TatHybrid	PaperText	R@100	NDCG@100
BM25	0.926 / 0.640	0.933 / 0.559	0.999 / 0.443	0.950	0.554
Dense E5	0.959 / 0.577	0.975 / 0.578	**1.000** / 0.501	0.977	0.555
HyDE	0.884 / 0.553	0.805 / 0.442	**1.000** / **0.521**	0.890	0.504
CoT	0.948 / 0.619	0.945 / 0.558	0.999 / 0.508	0.962	0.565
Vanilla SIRA	0.928 / 0.630	0.920 / 0.565	0.999 / 0.476	0.946	0.562
SCOPE − rewrite	0.929 / 0.628	0.945 / 0.595	0.999 / 0.458	0.955	0.567
ReAct, 5 turns	**0.980** / **0.685**	0.957 / **0.630**	0.992 / 0.502	0.975	**0.613**
SCOPE	0.977 / 0.673	**0.982** / 0.640	0.992 / 0.488	**0.983**	0.608

Against a rewrite control, the axis adds nothing. Once the same call emits an updated query, validated structure contributes $+0.016$ [-0.006, 0.039] recall and $+0.009$ [-0.010, 0.027] NDCG, both spanning zero, and only table-heavy recall survives at $+0.059$ [0.022, 0.098]. **Rewriting subsumes most of what structure was for**: asked to rewrite "how much did borrowing costs rise", the model produces *interest expense 2007 2006 change*, which already carries the row label. Explicit structure stays auditable, since the validator rejects a hallucinated table reference a rewrite would smuggle through, but on these benchmarks that does not convert into recall.

Three ways of giving the axis more weight, all failing. Recall@100 is 0.983 against Recall@10 of 0.797 (Table 4), so nearly a fifth of the evidence is fetched and then ranked out of the window. Table 6 is what happened when we tried to reach it. Propagation lifts siblings of a strongly scoring chunk along schema edges. Localisation multiplies every chunk in a matched region by $(1 + \lambda)$ instead of adding to it, which costs no extra call and recovers the scorer exactly when no key validates; it tied for best of the 33 development variants and came out null on test.

Table 6: **Asking the structural axis to carry more weight.** All three degrade monotonically in the weight given, on development for the sweeps and on test for the outcome.

Attempt	Development sweep	Outcome on test
Raise β past 1.0	β=0.5 $\rightarrow$ 2.0: 0.738 $\rightarrow$ 0.600	worse
Propagate along schema edges	not swept	-0.106 recall, -0.14 table-heavy
Localise by region	λ=0.5 to 8: 0.743 $\rightarrow$ 0.505	0.797 against 0.797, null

The axis is directionally right, worth $+0.057$ table-heavy as a gentle additive prior (G1), but too imprecise to define a region, propagate through, or dominate a ranking (Section 5.5). Three variants tied on the pre-registered criterion, and we broke the tie on overall development recall; the tie-break

registered in advance covers the seven extensions named there and localization is not among them, so we do not claim it selects this variant, though it would have at 0.497 against 0.458 and 0.450.

F RERANKING THE TOP 100

The 18.6 points between Recall@10 and Recall@100 are what a reranker exists to recover, so we tried one: `bge-reranker-base` over the top 100 of every stored ranking, scoring each chunk against the original question so no method is favored by its own rewrite. Only the head is reordered, so Recall@100 cannot move.

It makes every strong method worse (Table 7), and the damage lands where this paper says structure lives. **The four methods above** 0.81 **table-heavy recall lose** 0.10 **to** 0.13 **of it, against** 0.03 **for BM25.** A row reading *Interest expense, net 1,802 1,673* carries almost no sentence-level signal, and against a paragraph that merely discusses interest expense, a general-domain cross-encoder prefers the paragraph. Prefixing each chunk with its schema nodes helps every method (sign test $p = 0.031$) but recovers only 16% of the loss, $+0.009$ [-0.010, 0.029] for SCOPE: the reranker demotes table rows not because it cannot tell what they are, but because it is a prose-relevance model. Under both regimes, SCOPE stays decisive over vanilla SIRA and BM25 and indistinguishable from all three agent budgets, so the ordering of Table 1 is a property of the candidate pools rather than of any one scorer.

The recall loss does not reach the answer either. Containment labels favor exactly the rows the reranker demotes, so we re-ran the answer call on the reranked top-12 window for SCOPE and the five-turn agent, $1{,}078$ calls, changing nothing else. For SCOPE, the reranker costs 0.054 Recall@10 and 0.033 of its delivery, and answer accuracy does not fall with them: $+0.013$ [-0.017, 0.043], the opposite sign to the -0.013 the delivery drop predicts. On the five-turn agent it costs 0.064 recall and 0.046 of delivery and gives -0.020 [-0.048, 0.008] against a predicted -0.018. Neither interval clears zero, so we claim no improvement and no loss; what we can say is that a *decisive* retrieval loss produced no detectable answer loss on one method and a predicted one on the other, and that Recall@10 did not tell us which. Section 5.3 showed gains failing to convert, and this is the same failure run backward. We tested one reranker, and a larger or table-aware cross-encoder may behave differently.

Table 7: **Reranking the top** 100 **makes structured retrieval worse.** Recall@10 and table-heavy Recall@10 with no reranker, with `bge-reranker-base`, and with the same reranker shown each chunk's schema labels. 460 locatable test questions, and intervals are document-clustered and paired against the un-reranked ranking.

	Recall@10			table-heavy Recall@10		
Method	none	rerank	+schema	none	rerank	+schema
SCOPE	**0.797**	**0.743**	**0.752**	0.825	**0.722**	**0.738**
ReAct, 1 turn	0.791	0.736	0.748	**0.829**	0.701	0.724
ReAct, 2 turns	0.804	0.741	0.749	0.818	0.711	0.727
ReAct, 5 turns	0.800	0.736	0.745	0.816	0.711	0.727
Vanilla SIRA	0.721	0.714	0.729	0.689	0.664	0.695
BM25	0.700	0.715	0.725	0.694	0.664	0.683

G CHUNK GRANULARITY DECIDES WHETHER STRUCTURE CAN HELP

UDA's own RAG baseline chunks at 3000 characters, and we chunk at 800. At 3000, the task is saturated: BM25 alone reaches Recall@10 0.881 on dev, expansion changes the retrieved set for 5 of 107 questions, and isolating the structural axis moves NDCG@10 by $+0.001$ to $+0.008$. At 800, BM25 falls to 0.695, and structure produces the gains reported above.

The mechanism clarifies it: at 3000 characters, an entire financial table is one chunk, so there are no rows for a structural key to localize, since the key resolves to a real node and then selects the chunk

BM25 ranked first anyway. **Structure can only pay when the unit of retrieval is finer than the unit of layout.**

Three points of procedure. We changed the granularity on development evidence alone on 2026-09-17, before scoring any test questions, and recorded this in advance, committing to report a null at the finer setting if one occurred. 800 characters ($\approx$ 200 tokens) is ordinary RAG granularity, and evaluating where the baseline already sits at 0.881 measures the ceiling rather than the method (Fuhr, 2017). The coarse setting also handicaps the *baselines*: our dense encoder (Wang et al., 2022) accepts 512 tokens, so a 3000-character chunk ($\approx$ 750 tokens) is silently truncated.

H Robustness Throughout Expander Models

Figure 4 and Table 8 report the sweep.

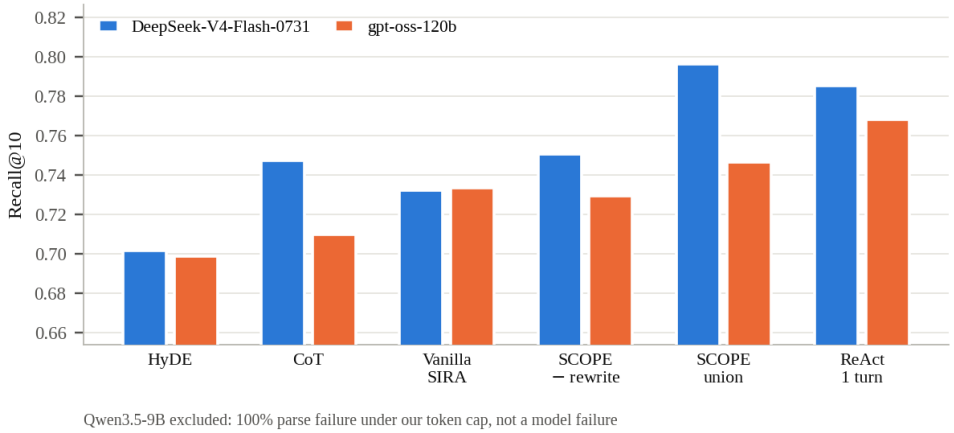

Figure 4: **Expander robustness.** Recall@10 with expansion produced by two additional models, with all other settings fixed. The SCOPE-family row is the union variant, not the configuration reported in Table 1: the sweep predates it. The expansion family outranks the non-agentic baselines under both expanders. Against the one-turn agent, the difference changes sign across them (-0.010 Llama, $+0.011$ DeepSeek, -0.022 gpt-oss), and we report no intervals for this sweep, so none of the three is established as a difference.

Each number above comes from one model, which also follows SIRA's BEIR protocol. We re-ran the retrieval stage across model families (Table 8).

The expansion family stays ahead of the non-agentic baselines under both expanders. Still, its standing against the one-turn agent changes sign ($+0.011$ on DeepSeek, -0.022 on gpt-oss, -0.010 on Llama), and the failure column explains the spread. As the predictor is asked for more fields, the share of unusable responses rises monotonically (0.080 for terms alone, 0.229 with structural keys, 0.597 once a rewrite is added). Hence, the method asking for the most structure is most often reduced to plain BM25. **A retrieval method depending on structured generation inherits that generation's variance**, and single-model evaluation cannot detect it.

We deliberately exclude two things. A fourth expander, Qwen3.5-9B, failed to parse on every question, and the cause was ours, not the model's: a reasoning model spends its budget on chain-of-thought and never reaches the JSON under our 400-token cap. And we do not extend this comparison to the free-text baselines, although we ran them: HyDE and CoT use a 220-token budget, and at that budget they failed more often than any JSON method, which measures the interaction between budgets and verbose models rather than output formats. These experiments therefore cannot support cross-format claims, and we do not make them.

Table 8: **Expander sweep** (retrieval only). Recall@10, with the share of expansions yielding no usable structured output in parentheses.

Expander	Family	Vanilla SIRA	− rewrite	SCOPE	ReAct-1
DeepSeek-V4-Flash	DeepSeek	0.732 (.004)	0.751 (.000)	**0.796** (.000)	0.785
Llama-3.3-70B	Meta	0.721 (.000)	0.740 (.000)	0.781 (.000)	**0.791**
gpt-oss-120b	OpenAI	0.733 (.080)	0.729 (.229)	0.746 (.597)	**0.768**

I Pre-Registered Gates

The five gates below were fixed before any test question was scored, together with the thresholds and the comparison each would be judged on. All five failed. We report them unchanged rather than re-pointing them at configurations chosen after seeing the data, because a gate that moves to meet its result is not evidence of anything. Three of the five measure effects that are real and statistically solid but smaller than the bar we set, and the remaining two failed on their own terms. Separately, our finding that iteration buys no ranking makes the cheap one-turn agent the more relevant latency comparison. The post-hoc block below the rule was added after the fact and is marked as such.

Table 9: **Verification gates**, fixed before test scoring. All five failed. The full SCOPE configuration post-dates the pre-registration, so we report its comparisons separately and never substitute it. *n.d.*: the interval spans zero, so the two are not distinguishable at this sample size.

Gate	Result	Outcome
G1	table-heavy R@10: (SCOPE − rewrite) − SIRA $\geq +0.08$	**fail** ($+0.057$ [0.022, 0.100])
G2	table-heavy NDCG@10, same threshold	**fail** ($+0.045$ [0.021, 0.069])
G3	table-heavy QA, same threshold	**fail** ($+0.033$ [0.009, 0.059])
G4	latency ReAct / (SCOPE − rewrite) $\geq 3\times$ (retr.)	**fail** ($1.96\times$)
G5	latency ReAct / (SCOPE − rewrite) $\geq 3\times$ (e2e)	**fail** ($1.25\times$)
post hoc, not pre-registered		
	SCOPE − 5-turn ReAct, R@10	n.d. (-0.003 [-0.036, 0.027])
	SCOPE − Vanilla SIRA, R@10	decisive ($+0.076$ [0.039, 0.114])
	SCOPE − Vanilla SIRA, table-heavy	decisive ($+0.136$ [0.079, 0.193])

J Method Inventory

Paper names are descriptive, while code names are what appear in the released result directories and in the workbook of Section L. They are not renamed, so older result files stay valid.

Gates G1, G4 and G5 are written against `doc_sira`, not `doc_sira_v2`, because that is the configuration that was pre-registered. Section 5.6 reports them as written.

K Token and Cost Accounting

Table 10 compares against the five-turn agent in full.

Retrieval-stage tokens are the expansion, rewrite, and agent-loop calls: every token spent before the generator is invoked. They are what Table 1 reports and what the $5.5\times$ figure refers to. The answer call is identical across methods (same reader, same 12-chunk budget) and dominates:

We name the stage each multiplier refers to wherever it appears. Against the five-turn agent, SCOPE is $5.5\times$ cheaper *at the retrieval stage* and $1.65\times$ cheaper end-to-end, while against the one-turn agent, which matches it on retrieval, it is more expensive on both. Latency, not token count, is the agent's real cost: the five-turn agent serializes 2.9 search round-trips on average that SCOPE issues as one.

Paper name	Code name	Configuration
SCOPE	`doc_sira_v2`	Semantic + structural + locator-preserving rewrite, replace mode, no router, $\beta = 0.5$
SCOPE − rewrite	`doc_sira`	The pre-registered configuration: router + semantic + structural, no rewrite
SCOPE, union variant	`doc_sira_rewrite_union`	As above but the rewrite is unioned alongside the existing search rather than replacing it, $\beta = 1.0$
Rewrite only	`doc_sira_rewrite`	Rewrite with the structural axis disabled ($\beta = 0$)
SCOPE-R	`doc_sira_refine`	One grounded refinement pass over the first retrieval
+ structural PRF	`doc_sira_sprf`	Pseudo-relevance feedback on structural keys from the top 3
+ propagation	`doc_sira_prop` `doc_sira_rewrite_prop`	Schema-graph score propagation (Section 5.5), negative result
Vanilla SIRA	`vanilla_sira`	Query-side validated expansion only, $\alpha = 0.25$
ReAct, n turns	`react_1turn` `react_2turn` `react`	Search–observe–refine with RRF over all issued queries

Table 10: **Cost at comparable retrieval quality.** Recall@10 differences against all three agent budgets span zero (Section I), and these are the resources each spends. Every ratio divides the two column means. The pre-registered latency gates G4 and G5 were written against the no-rewrite configuration, whose router leaves it 0.37 fewer calls on average (1.63 against 2.00): $1.96\times$ [1.78, 2.20] retrieval-stage and $1.25\times$ [1.17, 1.37] end-to-end.

	SCOPE	ReAct, 5 turns	ratio
Recall@10	0.797	0.800	−0.003 [-0.036, 0.027]
LLM calls per question	2.00	3.90	$1.95\times$
Retrieval-stage tokens	501	2753	$\mathbf{5.5}\times$
End-to-end tokens	3307	5469	$1.65\times$
Retrieval-stage latency	2.30 s	3.78 s	$1.65\times$
End-to-end latency	4.72 s	5.61 s	$1.19\times$

L Released Artifacts

The release includes a single workbook that consolidates every table in this paper plus the per-question records behind them, so no number here requires re-running anything to check. Its sheets are: `summary` (all methods × subsets), `paired_ci` (all headline comparison with its document-clustered bootstrap interval and a `decisive` flag), `gates`, `conversion`, `model_sweep`, `action_variants`, the four development sheets (`dev_variants`, `dev_grids`, `dev_routing`, `dev_headline`), `prep_docs`, and `per_question`: one row per question per method, carrying the retrieved ranking, the routing decision, the expansion that was issued and whether it parsed. The README sheet records the code hash, model, chunk granularity, and calibration constants used to build the export.

Method	retrieval-stage	end-to-end	ratio vs. SCOPE
BM25 / Dense E5	0	≈2806	0.85×
SCOPE	501	3307	1.00×
ReAct, 1 turn	201	≈3007	0.91×
ReAct, 5 turns	2753	5469	1.65×