KNIT: RECOVERING THE PASSAGE THAT TOP-k TRUNCATION CUTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieval-augmented generation (RAG) passes the top k passages of a retrieval pipeline to the LLM that answers. For a multi-hop question, the evidence is complete only when the top k holds every passage of the question's evidence chain. But top-k selection scores each passage on its own, by how well it matches the question. The passage that completes a chain is often linked to the others only through a bridge entity the question never names, so it ranks low and falls outside the top k. We introduce KNIT, a training-free layer that selects the top k as a set. KNIT keeps the pipeline's relevance scores and adds links between passages that share an entity, read from a graph built once per corpus. A graph stage selects $2k$ linked passages that together cover the question, and one LLM call keeps k of them. We apply one configuration, fixed on MuSiQue, to five published retrieval pipelines on five corpora. We measure chain completion: the share of questions whose top 5 holds the whole chain. KNIT raises it on all 25 pipeline–corpus pairs, significantly on 23. On the three Wikipedia corpora, the average gain is 10.6 to 13.7 points. KNIT improves even PropRAG, the strongest pipeline on Wikipedia, which already reads such a graph during retrieval. KNIT's gains carry over to question answering on Wikipedia: a 72B reader answers significantly more questions from KNIT's passages on 13 of the 15 Wikipedia pairs. On MuSiQue, our analysis shows that the graph stage's gain comes from the entity links rather than from diversity.

1 INTRODUCTION

Retrieval-augmented generation answers a question with a language model, the *reader*, given a few passages retrieved from a large corpus. A retrieval pipeline retrieves candidate passages for the question, often reranks them, and passes the top k of its final ranking to the reader. Pipelines differ in how they rank, from a lexical index to a graph over the corpus, but every pipeline we study ends the same way: it keeps the k passages whose own scores are highest. The budget k is small, because every passage costs the reader context and compute, so this *top-k selection*, or truncation, decides which evidence the reader sees. For a multi-hop question, whose answer is spread over several passages, the evidence is complete only when the top k holds the whole chain, and completeness is a property of the set, not of any one passage. It matters for the answer. On the three Wikipedia corpora we study, a 72B reader given a pipeline's own top five answers 26 to 44 points more often when the chain is complete than when it is not, on 14 of the 15 pipeline–corpus pairs (Section 4.3).

This top-k selection has a blind spot. The passage that completes a chain is often the one a per-passage score likes least, because it is linked to the others only through a *bridge entity*, the answer of an earlier hop, which the question never names. Figure 1 shows an example from MuSiQue, paraphrased: *Where did a critic of the country that once controlled Estonia say he would intervene in the Korean conflict?* Its gold passages are titled *Korean War*, *Estonia* and *Imperialism*. The pipeline is a dense retriever followed by the monoT5 reranker. It keeps the first two, but it spends its other three places on further *Estonia* passages, a hop it has already covered, and it cuts *Imperialism*, the passage that closes the chain, at rank 8. We call such a passage a *dependent hop*: the hop it answers refers to a bridge entity. Pipelines tend to rank dependent hops last. On MuSiQue, they make up 57.3% of all gold passages, but a dependent hop is the lowest-ranked gold passage on 64 to 87% of questions, depending on the pipeline (Appendix J). However good a per-passage score is, it cannot see what makes such a passage worth keeping, namely its link to the passages already chosen.

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.

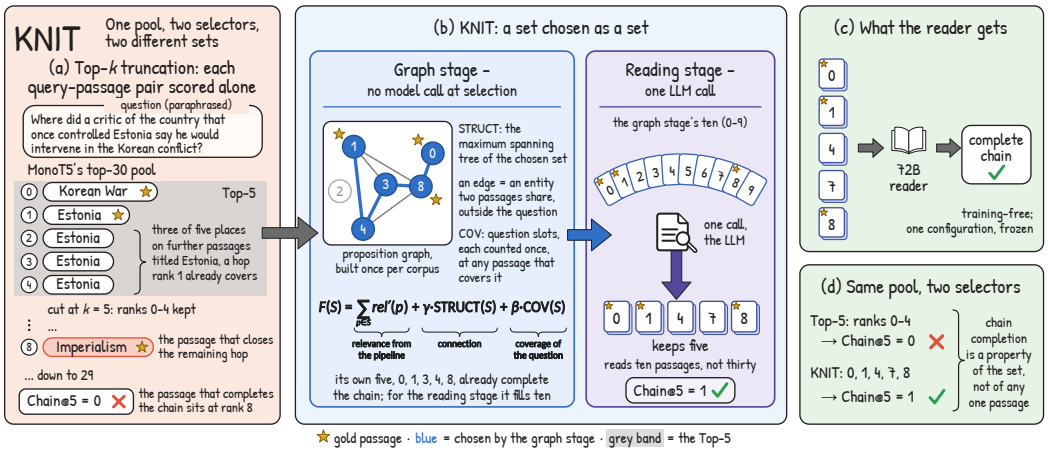

Figure 1: **One pool, two selectors, two different sets.** MonoT5's top-30 pool for one MuSiQue question (Appendix B). Chain@5 is 1 when every gold passage is among the five selected, else 0. The pipeline's five cut the passage that closes the chain, at rank 8 (ranks count from 0). The graph stage alone already keeps it and completes the chain, and so does KNIT. In (b), thick links form the spanning tree that the connection term scores.

Existing remedies do not use that link to select the top k as a set. A pointwise reranker, like this pipeline's monoT5, scores each passage against the question alone, so it cannot see that *Imperialism* completes the chain. Diversity selection, such as maximal marginal relevance (Carbonell & Goldstein, 1998), penalises passages similar to those already chosen, which targets the redundant *Estonia* passages, but it reads similarity, not links, so nothing in it rewards *Imperialism* for completing the chain. Graph-based methods such as HippoRAG 2 (Gutiérrez et al., 2025), PropRAG (Wang & Han, 2025) and GraphER (Miao et al., 2026) do read entity links, but each turns them into one score per passage and keeps the k passages that score highest. Recent work trains set-level objectives (Lee et al., 2025b; Song & Lee, 2026; Jiang et al., 2026) or builds set-aware selectors (Schlatt et al., 2025; Li et al., 2024). To our knowledge, however, no one has measured whether a released selector or reranker, installed at the top k of a strong pipeline it was not trained with, still completes more chains; we measure five off-the-shelf reranking models, Set-Encoder among them, in Section 4.4.

We ask a simple question: *given the candidates a pipeline already retrieves and the scores it already assigns, which k passages should it hand its reader?* We find that what a selector reads, that is, what it takes as input, decides whether it keeps such a passage. The pipeline's scores alone often cut the chain; on MuSiQue, adding similarity gains little, and off-the-shelf reranking models, which ignore the pipeline's scores and read no links, lower the strongest pipeline. We therefore introduce KNIT, a training-free layer that keeps the pipeline's scores, adds the links, and selects the top k as a set rather than each passage on its own. KNIT works in two stages (Figure 1b). First, the *graph stage* reads a *proposition graph*, an index built once per corpus that links two passages when statements extracted from them share an entity. From the pipeline's top 30 candidates, it greedily builds a set of relevant passages that are linked to one another and together cover the question. Second, the *reading stage* makes one LLM call: the graph stage selects $2k$ passages instead of k, and the LLM keeps the k that together answer the question, because only reading their text tells which passages hold the answer.

We fix one configuration on MuSiQue and evaluate it unchanged on five published pipelines, from BM25 to the graph-indexed PropRAG, and on five corpora: three from Wikipedia, one of news and one of science facts, QASC. We measure chain completion, *Chain@k*, the share of questions whose gold passages are all among the k selected, and we compare KNIT with the pipeline's own top k, its *Top-k*. At $k = 5$, KNIT improves Chain@5 on all 25 pipeline–corpus pairs, significantly on 23, and its mean gain is significant on every corpus; on MuSiQue, the graph stage's gain also holds at $k = 3$ and $k = 10$. KNIT even improves PropRAG, the strongest pipeline on Wikipedia, on all five corpora, significantly on the three Wikipedia corpora and on news, and a 72B reader answers significantly more questions from KNIT's passages on 13 of the 15 Wikipedia pairs. Our contributions are threefold:

- **Set selection at the top k.** KNIT, a training-free, two-stage layer, keeps the pipeline's scores, adds entity links from a graph built once per corpus, and selects the top k as a set (Section 2).

- **One configuration for every pipeline, the strongest included.** With one frozen configuration, KNIT improves every pair we test, 23 of 25 significantly, and on 13 of the 15 Wikipedia pairs the gain reaches the reader. Off-the-shelf reranking models, which see the candidates without the pipeline's scores, instead lower the strongest pipeline on MuSiQue (Section 4).
- **Why, and where, it works.** We show that the graph stage chooses as a set, which no fixed per-passage score can do, that on MuSiQue its gain comes from the entity links rather than from diversity, and that the links help where shared entities mark the hops of a chain, as on MuSiQue and QASC, but not on news (Sections 5 and 6).

2 KNIT: SELECTING THE TOP k AS A SET

A chain of evidence is complete only when all its passages are chosen, so KNIT scores whole sets rather than single passages. Because a chain's passages are often joined only through entities the question never names, KNIT links passages through the entities they share and rewards a set that these links tie together. The input is the *candidate pool* P: the pipeline's N highest-ranked passages, with their relevance scores. A top-k selector reads both and returns k passages for the reader. KNIT does so in two stages (Figure 1b).

2.1 THE GRAPH STAGE

We score a set $S \subseteq P$ by three terms:

$$F(S) \;=\; \textstyle\sum_{p \in S} \mathrm{rel}'(p) \;+\; \gamma\,\mathrm{STRUCT}(S) \;+\; \beta\,\mathrm{COV}(S). \tag{1}$$

In words, *relevance* is the pipeline's own score for each passage, *connection* rewards a set whose passages are tied together through entities the question does not name, and *coverage* rewards a set whose passages together cover the question's words, each counted once.

The proposition graph holds each passage's extracted statements and the entities they name, one graph per corpus. On PropRAG, the graph stage reads PropRAG's own graph (Wang & Han, 2025) on Wikipedia; on the four pipelines without a graph, and on every pipeline on news and on QASC, it reads one that we extract with Qwen2.5-14B-Instruct, one call per passage. At selection time, the graph is read with no model call, and entities are matched through their aliases.

Relevance. $\mathrm{rel}'(p)$ is the pipeline's score for p, min–max normalised within the pool, plus a small bonus λ when the words of p's title all appear in the question.

Connection. Two passages are linked when their propositions share an entity. We first remove the question's entities and their aliases, so every link runs through an entity outside the question, a candidate bridge entity. A link's weight is the inverse document frequency (IDF) of the rarest such entity, so links through rare entities weigh more. $\mathrm{STRUCT}(S)$ measures how strongly the chosen passages hang together: it is the weight of the maximum spanning tree of S, the heaviest links that join every passage of S without a loop (the thick links of Figure 1b), scaled to at most 1 for a set of five, and $\gamma\,\mathrm{STRUCT}(S)$ is the *connection term*.

Coverage. Here the question's words are those that are not stopwords. With $H_{\ell p}$ the IDF of word ℓ if passage p's propositions contain it and zero otherwise, $\mathrm{COV}(S) = \sum_{\ell} \max_{p \in S} H_{\ell p}$, so each word counts once, at any passage that covers it. H is normalised so that coverage is on the scale of the relevance sum, and $\beta\,\mathrm{COV}(S)$ is the *coverage term*.

An example. On the question of Figure 1, the graph stage at a budget of five trades the *Estonia* passage at rank 2 for the passage at rank 8. It gives up some relevance but gains more in connection and coverage together, so the set that completes the chain scores higher.

Search. The graph stage builds its set greedily, adding one passage at a time until it reaches its budget. Appendix A gives each term in full and the search as Algorithm 1.

One configuration, frozen: $\gamma = 2.5$, $\beta = 0.5$, $\lambda = 0.2$, $N = 30$ and $k = 5$, with coverage over all of the question's words. The budget k is an input, not a constant; Section 4.1 varies both k and N.

Table 1: **The names this paper uses.** At $k = 5$, every selector but the Top-10 returns five of the same thirty candidates.

Name	What it is
Chain@k	1 if every gold passage of a question is among the k selected, else 0; reported as a percentage of questions, at $k = 5$ unless stated
Dependent hop	a gold passage whose hop refers to a bridge entity, the answer of an earlier hop
Top-5, Top-10	the pipeline's own top five and top ten
Graph stage alone	the graph stage at a budget of five: the top 30 with their scores and the proposition graph, no model call
KNIT	the graph stage selects ten passages, and one LLM call keeps five
The call on the Top-10	the same LLM call, handed the pipeline's own top ten
30-passage reranker	the same LLM, handed the whole top 30 in one call
Plug-ins	Provence and Set-Encoder, used as rerankers in place of the pipeline's ranking; they see the candidates without the pipeline's scores
Rerankers	bge-reranker-v2-m3, Qwen3-Reranker-0.6B and RankZephyr-7B, installed the same way; RankZephyr is handed the pipeline's ranked list

2.2 The reading stage

The reading stage is one call of an LLM, Qwen2.5-32B-Instruct. When it follows, the graph stage selects ten passages instead of five, with nothing else changed, and the LLM keeps the five of them that together answer the question. The LLM thus reads ten passages that the graph stage has already brought together, whereas a reranker handed the whole pool reads thirty. For another target k, the graph stage would select $2k$, but we tested the two-stage form only at $k = 5$.

Cost. KNIT trains nothing. Per question, it spends one LLM call, which on MuSiQue takes about 1,800 prompt tokens and four seconds. The graph stage calls no model and takes tens of milliseconds on one CPU. The proposition graph is built once per corpus, at one 14B call per passage, and on Wikipedia PropRAG has already paid that cost. Appendices A, D and J give the measurements.

3 Experimental set-up

Datasets and metric. We evaluate on five corpora. MuSiQue (Trivedi et al., 2022), 2WikiMultihopQA (Ho et al., 2020) and HotpotQA (Yang et al., 2018) are built on Wikipedia, MultiHop-RAG (Tang & Yang, 2024) on news (question counts in Table 2), and QASC (Khot et al., 2020) on grade-school science, where each question is composed from two one-sentence facts and the passages are QASC's annotated facts. We report Chain@k in points, as a percentage of questions, at $k = 5$, the budget of the chain-level measurement of Lau et al. (2026). MuSiQue's gold sets run to four passages, so at $k = 3$ only 83.4% of its questions can be completed at all. Section 4.1 also reports the graph stage at $k = 3$ and $k = 10$.[1] Gold is matched exactly against each corpus's units: its own passages on Wikipedia, 100-word chunks that we cut on news, and the facts on QASC.

Tuning and held-out data. We split MuSiQue's questions in half by a hash of the question id, within each hop count and independently of any result. The *tuning half* chose the configuration and the reading stage's form; the *held-out half* serves as a test set, except that for PropRAG it also helped choose (γ, β, N) (Appendix E). The four other corpora took no part in tuning. MuSiQue numbers pool both halves unless marked otherwise.

Statistical testing. Every comparison is paired per question. For one pipeline–corpus pair, we use the exact McNemar test (McNemar, 1947) with a Clopper–Pearson interval (Clopper & Pearson, 1934) on the discordant split, and we call a difference *significant* when the test gives $p < 0.05$ and the 95% interval excludes zero, without family-wise correction. For a quantity pooled over pipelines, we use a question-level cluster bootstrap under five seeds, call it significant when its 95% interval excludes zero, and report the widest of the five seeds' intervals unless a table states otherwise.

Pipelines. We install KNIT on five published pipelines, weakest first on MuSiQue: BM25 (Robertson & Zaragoza, 2009); *Hybrid*, BM25 and the dense encoder NV-Embed-v2 (Lee et al., 2025a) fused

[1] Every gain here is paired within our implementation, and absolute values are compared only within it and within one corpus. Our absolute levels are higher than those Lau et al. (2026) report for the same pipelines; for PropRAG the gap lies in retrieval, not in the metric (Appendix N).

Table 2: **One frozen configuration improves every pipeline–corpus pair.** Each entry gives Chain@5 for the pipeline's own top five (Top-5), then for KNIT on the same pool and budget, then the paired gain. Bold marks a gain whose exact 95% interval excludes zero (23 of 25). The last row gives a 72B reader's score given the five passages, averaged over the five pipelines, under the Top-5 and then under KNIT, with the gain: exact match (EM) on the Wikipedia corpora, and on news and QASC the benchmark's own accuracy (Acc.). MuSiQue's column uses all 1,000 questions, including the half that chose the configuration.

Pipeline	MuSiQue $n = 1{,}000$	2WikiMultihopQA $n = 1{,}000$	HotpotQA $n = 1{,}000$	MultiHop-RAG, news $n = 2{,}255$	QASC, science $n = 926$
BM25	10.3→19.9 **+9.6**	32.8→44.1 **+11.3**	49.6→79.0 **+29.4**	11.8→17.1 **+5.2**	13.7→21.0 **+7.2**
Hybrid	20.7→36.1 **+15.4**	41.4→56.4 **+15.0**	73.9→88.9 **+15.0**	13.6→19.6 **+6.1**	23.7→28.6 **+5.0**
MonoT5	24.9→40.0 **+15.1**	44.2→57.0 **+12.8**	74.5→91.6 **+17.1**	11.5→15.9 **+4.3**	27.1→32.8 **+5.7**
Dense	38.2→46.9 **+8.7**	51.1→62.4 **+11.3**	89.4→94.6 **+5.2**	10.1→16.5 **+6.5**	31.7→32.9 +1.2
PropRAG	53.9→58.1 **+4.2**	83.1→92.1 **+9.0**	94.9→96.6 **+1.7**	11.6→17.5 **+5.9**	28.0→29.5 +1.5
KNIT, mean gain	+10.60 [+9.30, +11.90]	+11.88 [+10.74, +13.06]	+13.68 [+12.46, +14.92]	+5.60 [+4.91, +6.29]	+4.13 [+2.76, +5.49]
graph stage alone	+6.92 [+5.86, +8.00]	+5.96 [+5.10, +6.86]	+7.14 [+6.10, +8.20]	+0.76 [+0.28, +1.25]	+2.70 [+1.68, +3.76]
reading stage adds	+3.68 [+2.66, +4.70]	+5.92 [+5.10, +6.78]	+6.54 [+5.60, +7.50]	+4.83 [+4.23, +5.45]	+1.43 [+0.22, +2.61]
72B reader, mean	EM 28.3→33.6 +5.30 [+4.20, +6.42]	EM 45.3→49.7 +4.34 [+3.20, +5.52]	EM 56.4→60.8 +4.42 [+3.40, +5.46]	Acc. 77.3→78.1 +0.84 [+0.30, +1.39]	Acc. 90.1→89.8 −0.30 [−1.04, +0.41]

On MuSiQue's held-out half (500 questions), KNIT is significantly above the Top-5 on all five pipelines: mean +9.60, PropRAG +5.0 [+1.22, +8.42]. For PropRAG that half also helped choose (γ, β, N), so it is only partly held out. The graph stage alone is significantly above the Top-5 on four of the five pipelines, mean +6.72, PropRAG +3.0 [−0.04, +5.69]. Gains are computed before rounding.

by reciprocal rank (Cormack et al., 2009); *MonoT5*, monoT5 (Nogueira et al., 2020) reranking the encoder's top fifty; *Dense*, the encoder alone; and PropRAG (Wang & Han, 2025). Some MuSiQue analyses add five pipelines that we built from a released proposition index and PropRAG, for ten in all (Appendix M).

Baselines. Besides the pipeline's own Top-5, we compare KNIT with five off-the-shelf reranking models, each used in place of the pipeline's ranking. Two are plug-ins: Provence (Chirkova et al., 2025), a context pruner whose reranking head scores each query–passage pair alone, and Set-Encoder (Schlatt et al., 2025), a set-aware cross-encoder that attends across the candidates, which we evaluate over eleven seeded draws. Three are rerankers (Chen et al., 2024; Zhang et al., 2025b; Pradeep et al., 2023): bge-reranker-v2-m3, Qwen3-Reranker-0.6B and the listwise RankZephyr-7B. Table 1 lists every name. We also compare three classical diversity selectors, MMR (Carbonell & Goldstein, 1998), DPPs (Kulesza & Taskar, 2012; Chen et al., 2018) and facility location (Lin & Bilmes, 2011), each with its constant chosen per pipeline on the tuning half (Appendix G).

Readers. Our main reader is Qwen2.5-72B-Instruct under greedy decoding; on MuSiQue, the 7B model of the same family and two readers from other families serve as checks (Appendix K). We score them by SQuAD exact match on Wikipedia and by each benchmark's own accuracy rule elsewhere.

4 Results

We first report KNIT on every pipeline and corpus (Section 4.1), on the strongest pipeline (Section 4.2) and at the reader (Section 4.3). We then compare it with selectors that read something else (Section 4.4).

4.1 Main results

Table 2 shows that KNIT improves Chain@5 on all 25 pipeline–corpus pairs, and the improvement is significant on 23 of them. The two exceptions are Dense and PropRAG on QASC, the two strongest pipelines there. On the three Wikipedia corpora, KNIT improves the five pipelines by 10.6 to 13.7 points on average, and the graph stage alone accounts for about 6 to 7 of these points.

The gains in chain completion hold beyond the tuning data. The four other corpora took no part in tuning, yet on the two other Wikipedia corpora KNIT gains at least as much as on MuSiQue. On MuSiQue's held-out half, which for PropRAG is only partly held out, it still improves all five pipelines significantly (Table 2's note). The graph stage's gain on MuSiQue is also robust to the choices we made. Its mean gain is at least $+5.42$ points at each of the 108 configurations we ranked when choosing the frozen one, although PropRAG alone falls to or below its Top-5 at 20 of them (Appendix E). It stays significant on every pairing of pipeline and graph we tried, and a pool of fifty moves its chain completion by less than two points on every pipeline (Appendices A and C).

The graph stage's gains hold at other budgets. Pooled over ten MuSiQue pipelines, the graph stage's gain over each pipeline's own top k is $+5.72$, $+6.10$ and $+4.48$ points at $k = 3$, 5 and 10, each significant, and it vanishes only at $k = 30$, the whole pool (Appendix F).

Counts of significant pipelines need one qualification: Hybrid, MonoT5 and Dense share a first stage, the encoder's ranking, so their gains are correlated and are not independent evidence. A joint sign test of whether the graph stage gains on every one of the five published pipelines does not reach significance ($p = 0.055$), but a sign-flip test of the mean gain that keeps this correlation (Appendix N) still finds the five-pipeline mean significant on MuSiQue, for KNIT and for the graph stage alone.

4.2 KNIT IMPROVES EVEN THE STRONGEST PIPELINE

PropRAG is the strongest pipeline on all three Wikipedia corpora: it completes the most chains, and it already reads a proposition graph during retrieval. KNIT still improves it on all five corpora, significantly on the three Wikipedia corpora and on news (Table 2). On the three Wikipedia corpora, neither plug-in nor the 30-passage reranker completes more chains on PropRAG than KNIT.

4.3 THE GAINS CARRY OVER TO QUESTION ANSWERING ON WIKIPEDIA

At the 72B reader, KNIT improves exact match significantly on 13 of the 15 Wikipedia pairs, including all five MuSiQue pipelines. The two exceptions are PropRAG on 2WikiMultihopQA and HotpotQA, where its chain gain does not reach the reader. The exact-match gain is smaller than the chain gain because the reader had already answered some of the questions whose chain KNIT completes, and still misses others. A complete chain nonetheless goes with a correct answer on Wikipedia: under the Top-5, the reader answers 26 to 44 points more often when the chain is complete than when it is not, on 14 of the 15 pairs. On MuSiQue, the gain is also significant on all five pipelines with a 7B reader and with readers from two other model families.

On MuSiQue, the gain reaches the reader on PropRAG too. Although PropRAG is where the graph stage gains least (Figure 2c), at the 72B reader KNIT raises its exact match significantly, by $+3.90$ points over its Top-5 with both halves pooled. On the held-out half alone, which for PropRAG also helped choose (γ, β, N), the gain is $+3.40$ [$+0.26$, $+6.18$], still significant. On that half, the gain stays significant on three of the five pipelines.

Five passages from KNIT beat PropRAG's top ten. On MuSiQue, the reader answers significantly more questions from KNIT's five passages than from PropRAG's own top ten, by 3.6 exact-match points, while that top ten does not significantly beat PropRAG's Top-5.

On news and QASC, the reader gains little because it already answers most questions without a complete chain. On news, a complete chain does not predict a correct answer: the reader answers questions with an incomplete chain at least as often as those with a complete one. On QASC, the reader answers 87.80% of the questions with no facts at all, and every retrieved set sits only 1.4 to 2.8 points above that floor (Appendices I, K and L).

4.4 COMPARISON WITH OTHER SELECTORS

Off-the-shelf reranking models lower the strongest pipeline. We use the five off-the-shelf models of Section 3 to rerank each pipeline's top 30 candidates in place of its own ranking, so that they see the candidates without the pipeline's scores, and Table 3 compares them with KNIT on MuSiQue. Four of the five complete about the same share of chains whatever pipeline feeds them, so they improve the weakest pipeline but lower the strongest. The fifth, RankZephyr, is handed the pipeline's ranked list and varies more. All five fall below PropRAG's Top-5, the two plug-ins significantly and by more than 25 points, whereas KNIT improves PropRAG. Without a test, even the graph stage alone, which calls no model, completes more chains on Dense and PropRAG than any of these models. We also ran the two plug-ins with the pipeline's score added to their own: they then lose less than half as much on Dense and PropRAG, but they still lower PropRAG significantly. At the reader, the two plug-ins lower PropRAG's exact match by 7.90 and 10.70 points, while KNIT raises it by 3.90 (Appendix K).

Of the five models, only the two plug-ins ran beyond MuSiQue, on all four other corpora. They also lower PropRAG on the other two Wikipedia corpora and every pipeline on news. On QASC, however, every pipeline's own Top-5 completes fewer chains than either plug-in, and there the plug-ins help:

Table 3: **On MuSiQue, every selector here that replaces the pipeline's ranking lowers the strongest pipeline, PropRAG; KNIT raises it.** Chain@5 ($n = 1{,}000$) of each selector on the same top 30 of each pipeline, pipelines ordered by their own Top-5. The five reranking models replace the pipeline's ranking and see the pool without its scores (RankZephyr is handed it in rank order). Set-Encoder is the median of eleven seeded draws. Bold: the best selector on each pipeline.

Selector	BM25	Hybrid	MonoT5	Dense	PropRAG
Pipeline's own Top-5	10.3	20.7	24.9	38.2	53.9
Off-the-shelf reranking models, in place of the ranking					
Provence	20.0	26.9	26.0	27.3	28.7
Set-Encoder	19.2	23.3	24.3	24.3	24.7
bge-reranker-v2-m3	21.2	33.5	31.3	33.0	34.4
Qwen3-Reranker-0.6B	20.8	30.2	29.8	30.3	31.7
RankZephyr-7B	16.7	26.3	25.4	32.5	38.8
KNIT's LLM over the whole pool					
30-passage reranker	**21.8**	**40.4**	**40.5**	43.2	49.0
Ours: the pipeline's scores and the entity links					
Graph stage alone	16.3	29.1	34.2	45.7	57.3
KNIT	19.9	36.1	40.0	**46.9**	**58.1**

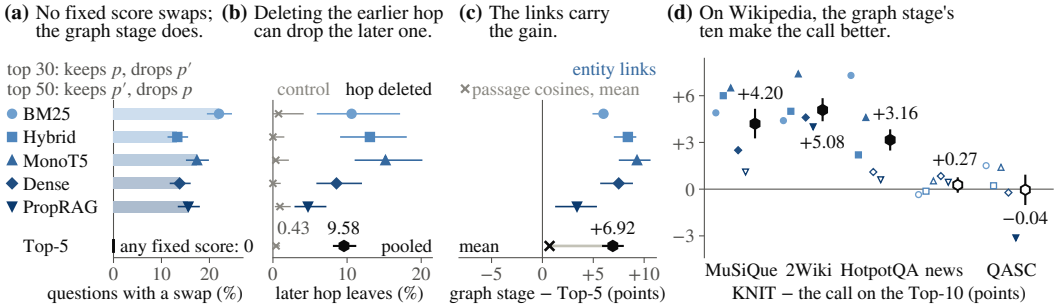

Figure 2: **On MuSiQue, the graph stage chooses as a set, and its gain comes from the links.** (a) $n = 1{,}000$: questions whose top-30 and top-50 selections swap two passages (Appendix C). (b) Questions where the graph stage selects both hops of a chain: how often the later hop leaves when the earlier one, or a rank-matched non-gold passage (control), is deleted (Appendix H). (c) $n = 1{,}000$: the graph stage's gain over the Top-5 on the entity links, per pipeline, and its five-pipeline mean on passage cosines (Appendices E and H). (d) Every question of each corpus: KNIT minus the same call handed the Top-10 (Appendix J). Hexagons pool the pipelines; a filled mark's 95% interval excludes zero.

both improve three or four pipelines significantly and lower none. Without a test, both plug-ins also complete more chains than KNIT on QASC's BM25, Hybrid and PropRAG, and Set-Encoder's median draw does so on all five QASC pipelines (Appendix D).

The 30-passage reranker never significantly beats KNIT on a corpus's strongest pipeline. We also ask KNIT's LLM to pick five of the thirty candidates in one call, a zero-shot listwise reranker (Sun et al., 2023) that we call the *30-passage reranker*. It uses 2.3 to 3.0 times KNIT's prompt tokens. Averaged over the five pipelines, it ties KNIT on QASC and on MuSiQue's held-out half and is significantly ahead by 1.38 to 2.86 points on the other three corpora (Appendix J.2). On MuSiQue it leads on the three weaker pipelines, but it too falls below PropRAG's Top-5, while KNIT leads on Dense and PropRAG (Table 3). KNIT is significantly ahead on two of the strongest pipelines, MuSiQue's PropRAG and QASC's Dense. The reranker leads most on 2WikiMultihopQA, where reading all thirty finds chains the graph stage's ten miss; there KNIT's edge is its lower cost.

5 Analysis: why KNIT gains

What does KNIT bring back, and which part does the work? Figure 2 summarizes our answer: the graph stage chooses as a set, and its gain comes from the links. We run every analysis in this section on MuSiQue unless we name another corpus.

The graph stage brings back the hops a per-passage score buries. Of the gold passages it lifts from outside the top k, 68.4% are dependent hops, against 48.9% of the pool's gold (Appendix J).

No fixed per-passage score reproduces the graph stage's choices. A fixed per-passage score gives each passage one number, so of two passages it always prefers the same one, whatever else is in the

pool. The graph stage does not: handed a pipeline's top thirty and then its top fifty, which still contain two passages p and p', it can keep p and leave out p' in the first pool and do the reverse in the second, which no fixed per-passage score could do. On 13.3 to 22.0% of questions, depending on the pipeline, the graph stage makes such a swap (Figure 2a), and on each of them no fixed per-passage score, however good, can reproduce its choices. This share is a lower bound, since comparing more pools could only find more swaps.[2] The graph stage can swap because its choice depends on the whole pool: its connection and coverage terms score passages together, its terms are normalised within the pool, and one of its two search routes stops after a fixed number of node sets (Appendix A).

Deleting the earlier hop can drop the later one. Under a fixed per-passage score, deleting one passage from the pool removes no other from the selection. We delete the earlier hop of a chain that the graph stage has selected and find that the later hop drops out too on 9.58% of such questions, against 0.43% when a rank-matched non-gold passage is deleted instead; the difference excludes zero on all five pipelines (Figure 2b). The drop-out grows with the weight of the entity link between the two hops, and setting the connection weight γ to zero removes nearly all of it (Appendix H).

The links carry the gain. To test what carries the gain, we replace the entity links by *passage cosines*, the cosine similarity of the passages' embeddings, and keep everything else fixed. The graph stage then loses nearly all of its mean gain over the five pipelines, 6.22 of 6.92 points, significantly on each pipeline (Figure 2c). Rewarding diversity instead gains little. Pooled over the held-out halves of ten MuSiQue pipelines, MMR, DPPs and facility location gain at most 1.32 points, even at their own per-pipeline best; with each of three passage similarities, all three stay significantly below the graph stage (Appendix G; supplementary notes, Section S2).

The reading stage adds what the graph cannot score. The one LLM call judges which passages hold the answer, and handed the graph stage's ten, it adds significantly to the graph stage's gain on every corpus (Table 2). On Wikipedia, however, the call needs those ten. Handed the pipeline's Top-10 instead, the same call completes significantly fewer chains on each Wikipedia corpus, by 3.2 to 5.1 points on average (Figure 2d), and on MuSiQue it then only ties the graph stage alone. On MuSiQue, then, what the call adds over the graph stage alone depends on being handed the graph stage's ten. Outside Wikipedia, the ten lose that advantage: on news and QASC, KNIT ties the call on the Top-10, and on QASC's PropRAG that call does significantly better (Appendices J and L.2).

6 Where the links help, and where they do not

On MuSiQue, QASC and news, we find that rewarding connection helps only where an entity shared by two passages tends to join two hops of a chain. To see this, we take the two pool passages that a question's strongest entity link joins: they are a gold pair about fourteen times as often as a random pair of the pool on MuSiQue and 26 times as often on QASC, but only a third as often on news. The consequence shows on news, where the graph stage gains under one point on average, compared with about six to seven on the Wikipedia corpora. There, coverage used alone improves chain completion on all five pipelines, whereas connection used alone lowers it on all five. Linked pairs on news often sit inside one article, but connection alone stays down even with every same-article link deleted. On QASC, in contrast, each reward gains on its own, and connection carries the graph stage's gain. All of these gains and losses are significant. On these three corpora, then, connection needed shared entities that mark hops and coverage did not. We tested news at one chunking only (Appendix L).

7 Related work

Set selection at the top k. MMR (Carbonell & Goldstein, 1998), DPPs (Kulesza & Taskar, 2012) and submodular coverage (Lin & Bilmes, 2011; Nemhauser et al., 1978) pick a set under a budget, and SMART-RAG (Li et al., 2024) puts a DPP at the top k. All of them read similarity among the candidates, whereas KNIT rewards connection through links extracted from the corpus (Section 5). AdaGReS (Peng et al., 2025) and GeoRAG (Zhang et al., 2026a) select greedily for coverage.

[2] With two nested pools, as here, the test is exact: some fixed per-passage score reproduces a selector's choices on both pools if and only if the selector makes no such swap (Theorem 1 and Corollary 1, Appendix C). Theorem 1 is the characterisation of choice rules by Chambers & Yenmez (2018); its fixed-cardinality statement appears first in Eliaz et al. (2011). Corollaries 1 and 2, on what two nested pools and one pool can reveal, are not in those sources.

Set-aware reranking and pruning. Set-Encoder (Schlatt et al., 2025), Provence (Chirkova et al., 2025), ListT5 (Yoon et al., 2024) and RankZephyr (Pradeep et al., 2023) rerank or prune the candidates. SetR (Lee et al., 2025b), SetCE (Song & Lee, 2026), OptiSet (Jiang et al., 2026), DPS (Meng et al., 2025) and joint passage retrieval (Min et al., 2021) train set-level objectives. DualView (Zhang et al., 2026b) reports rerankers lifting a weak dense first stage. We find that, used without the pipeline's scores, such models also lower the strongest pipeline on MuSiQue (Section 4.4).

Graph and iterative multi-hop retrieval. IRCoT (Trivedi et al., 2023) and multi-hop dense retrieval (Xiong et al., 2021) iterate the query. HippoRAG 2 (Gutiérrez et al., 2025), PropRAG (Wang & Han, 2025) and EviReform (Xu & Li, 2026) read entity or proposition structure during retrieval, and GraphER (Miao et al., 2026) reranks over entity links, but their top-k selection still ranks passages one by one; KNIT instead selects the top k as a set, and it improves even PropRAG (Section 4.2).

Chain-level measurement. Our metric is the full-hit or all-gold at k of CatRAG (Lau et al., 2026), GraphER (Miao et al., 2026) and "Recall Is Not Enough" (Bala, 2026). Reranker benchmarks and studies evaluate at passage level (Thakur et al., 2021; Jacob et al., 2024; Singh et al., 2026), so there a reranker's effect on a whole chain goes unmeasured; DualView is an exception.

8 Limitations

First, KNIT returns pool members only, so a chain with a gold passage outside the pool is out of reach; over ten pipelines on MuSiQue, this holds for over a third of pipeline–question pairs.

Second, KNIT also breaks some chains. Applied to every question, the graph stage sometimes breaks a chain that the pipeline's own five had completed, most often when the lowest-ranked gold passage sits last among them, although over ten pipelines on MuSiQue it completes far more chains than it breaks. The LLM call also breaks some chains that the graph stage alone had completed; it completes more than it breaks, but on Dense and PropRAG the two nearly cancel (Appendices F and J).

Third, KNIT needs a proposition graph per corpus, built at one LLM call per passage, and one LLM call per question (Section 2.2). Fourth, KNIT's LLM, the graph extractor and the main reader all come from one model family, so the reader gain could partly reflect that family, although two readers from other families reproduce it on MuSiQue (Section 4.3). Fifth, every reader result uses one fixed passage order; a shuffled order, tested only under an earlier configuration, left the main reader contrast not significant (Appendix K).

9 Conclusion

We introduced KNIT, a training-free layer that replaces a pipeline's top-k selection with a set chosen from the pipeline's relevance and the corpus's entity links. With one frozen configuration, KNIT improves chain completion on all 25 pipeline–corpus pairs we test, significantly on 23, and it improves even PropRAG, the strongest pipeline on Wikipedia, on all five corpora, significantly on the three Wikipedia corpora and on news. On Wikipedia the gain reaches the reader on 13 of the 15 pairs, and on MuSiQue five passages from KNIT answer more questions than PropRAG's own ten. Keeping the pipeline's relevance matters on Wikipedia and news, where the plug-ins, which discard it, lower PropRAG; on QASC, where they complete more chains than every pipeline's own Top-5, they help. The 30-passage reranker is significantly ahead on average on three corpora, at 2.3 to 3.0 times the prompt tokens, but never significantly beats KNIT on a corpus's strongest pipeline. On MuSiQue, no fixed per-passage score can reproduce the graph stage's choices on 13.3 to 22.0% of questions, and the graph stage's gain comes from the links rather than from diversity. The links help where shared entities mark hops, as on MuSiQue and QASC; on news, where they do not, links used alone hurt. What a selector reads decides the evidence it returns.

REPRODUCIBILITY STATEMENT

Section 2 and Appendices A and J define KNIT in full, and Section 3 describes the pipelines, corpora, readers and tests. The graph stage is a single file with no learned parameters; the caller supplies its configuration, and a released script runs it at the frozen configuration of Table 4. The top-30 pools, the pipeline scores, the proposition graphs and the per-question selections for every pipeline, corpus and design reported here are released with it, as are the reading stage's prompt, the name of its LLM and its per-question scores. A re-implementation written from Section 2.1 with the definitions of Appendix A reproduces every one of the 1,000 selections on each of the five published pipelines, and Appendix A gives the intermediate quantities a re-implementation can check itself against. The tuning and held-out halves follow from $\mathrm{sha256}(\texttt{"split1|"} + \textit{qid})$ and need no stored file. Intervals come from two released functions, the exact McNemar and Clopper–Pearson rule for single pipeline–corpus pairs and the question-level cluster bootstrap for pooled quantities (Appendix N). The main analysis file names each comparison, its metric and its thresholds, and how each outcome is to be interpreted; it is in the anonymous supplementary material with its hash. The measurements added after it was fixed (the science corpus, the released plug-ins and the listwise reranker on the held-out corpora, and the reader's floor and ceiling on QASC) each come with their own rule for reading them, written before the results were read and released beside it; the runs designed after their inputs were read are labelled exploratory where they occur. The departures from these files, cells named but not run included, are recorded where they occur (Appendices D, J, K and L). A released script regenerates Tables 2 and 3, Figure 2 and every appendix figure except Figure 4, a drawing like Figure 1, from the released artifacts, without re-running retrieval or any model; it also regenerates six of the 22 appendix tables in full and four in part. For every other table, those of the supplementary notes included, the release's README names the released script that regenerates it or states that none is released. Four of the files whose hashes this paper or its supplementary notes print are released as copies anonymised for review, with every change listed beside them; the printed hashes are of the originals. Timings and GPU-hours are measurements of our runs, released as records.

AI USE STATEMENT

In this work, we used generative AI tools to help implement the method and the code that cleans and reformats the data and runs the experiments, to help propose and refine hypotheses, to give feedback on the methodology and experiments, to help interpret results, and to help write the text of the proofs in Appendix C. The claims and arguments of those proofs are the authors', except for Theorem 1 and its proof, which are those of Chambers & Yenmez (2018). We have not used generative AI tools to generate synthetic data sets, to develop theoretical models or conceptual frameworks, to formulate mathematical claims or provide the ingredients of their proofs, or to design the research methodology or experiments, and qualitative or thematic data analysis is not applicable to this work. We also used generative AI tools to edit prose, create figures, search the literature and run the experiment pipeline. We have reviewed all AI-assisted work: the study's design and structure, the decisions recorded in its analysis files, the review and revision of all content and the verification of every result remain with the authors, and every reported number was re-derived from the raw result files. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI. The method itself also calls open-weight language models, which extract the proposition graphs and run the reading stage, as Sections 2 and 3 describe.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, et al. Phi-4 technical report, 2024.

Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. Improvements that don't add up: ad-hoc retrieval results since 1998. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 601–610, Hong Kong, China, 2009. Association for Computing Machinery. doi: 10.1145/1645953.1646031.

Ananto Nayan Bala. Recall is not enough: A reader-context diagnostic for budget-constrained retrieval-augmented generation. arXiv:2607.00725v2, 2026. URL https://arxiv.org/abs/2607.00725v2.

Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 335–336, Melbourne, Australia, 1998. Association for Computing Machinery. doi: 10.1145/290941.291025.

Christopher P. Chambers and M. Bumin Yenmez. A simple characterization of responsive choice. *Games and Economic Behavior*, 111:217–221, 2018. doi: 10.1016/j.geb.2018.07.001.

Jianlv 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. arXiv:2402.03216, 2024. URL https://arxiv.org/abs/2402.03216.

Laming Chen, Guoxin Zhang, and Eric Zhou. Fast greedy MAP inference for determinantal point process to improve recommendation diversity. In *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*. Curran Associates, Inc., 2018.

Nadezhda Chirkova, Thibault Formal, Vassilina Nikoulina, and Stéphane Clinchant. Provence: Efficient and robust context pruning for retrieval-augmented generation. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2501.16214.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Büttcher. Reciprocal rank fusion outperforms Condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 758–759, Boston, MA, USA, 2009. Association for Computing Machinery. doi: 10.1145/1571941.1572114.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. Show your work: Improved reporting of experimental results. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2185–2194, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1224. URL https://aclanthology.org/D19-1224/.

Kfir Eliaz, Michael Richter, and Ariel Rubinstein. Choosing the two finalists. *Economic Theory*, 46: 211–219, 2011. doi: 10.1007/s00199-009-0516-3.

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.

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, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Mathew Jacob, Erik Lindgren, Matei Zaharia, Michael Carbin, Omar Khattab, and Andrew Drozdov. Drowning in documents: Consequences of scaling reranker inference. arXiv:2411.11767, 2024. URL https://arxiv.org/abs/2411.11767.

Yi Jiang, Sendong Zhao, Jianbo Li, Bairui Hu, Yanrui Du, Haochun Wang, and Bing Qin. OptiSet: Unified optimizing set selection and ranking for retrieval-augmented generation. arXiv:2601.05027, 2026. URL https://arxiv.org/abs/2601.05027.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. QASC: A dataset for question answering via sentence composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8082–8090, 2020. doi: 10.1609/aaai.v34i05.6319. URL https://ojs.aaai.org/index.php/AAAI/article/view/6319.

Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2–3):123–286, 2012. doi: 10.1561/2200000044.

Kwun Hang Lau, Fangyuan Zhang, Boyu Ruan, Yingli Zhou, Qintian Guo, Ruiyuan Zhang, and Xiaofang Zhou. Breaking the static graph: Context-aware traversal for graph-based RAG. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 5849–5863, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.290. URL https://aclanthology.org/2026.findings-acl.290/.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-Embed: Improved techniques for training LLMs as generalist embedding models. In *International Conference on Learning Representations (ICLR)*, 2025a. URL https://arxiv.org/abs/2405.17428.

Dahyun Lee, Yongrae Jo, Haeju Park, and Moontae Lee. Shifting from ranking to set selection for retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 17606–17619, Vienna, Austria, July 2025b. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.861. URL https://aclanthology.org/2025.acl-long.861/.

Jiatao Li, Xinyu Hu, and Xiaojun Wan. SMART-RAG: Selection using determinantal matrices for augmented retrieval. arXiv:2409.13992, 2024. URL https://arxiv.org/abs/2409.13992.

Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 510–520, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1052/.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947. doi: 10.1007/BF02295996.

Siyuan Meng, Junming Liu, Yirong Chen, Song Mao, Pinlong Cai, Guohang Yan, Botian Shi, and Ding Wang. From ranking to selection: A simple but efficient dynamic passage selector for retrieval augmented generation. arXiv:2508.09497, 2025. URL https://arxiv.org/abs/2508.09497.

Ruizhong Miao, Yuying Wang, Rongguang Wang, Chenyang Li, Tao Sheng, Sujith Ravi, and Dan Roth. GraphER: An efficient graph-based enrichment and reranking method for retrieval-augmented generation. arXiv:2603.24925, 2026. URL https://arxiv.org/abs/2603.24925.

Sewon Min, Kenton Lee, Ming-Wei Chang, Kristina Toutanova, and Hannaneh Hajishirzi. Joint passage ranking for diverse multi-answer retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6997–7008, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.560. URL https://aclanthology.org/2021.emnlp-main.560/.

Mistral AI. Mistral Small 3. https://mistral.ai/news/mistral-small-3, 2025. 24B-parameter model released under the Apache 2.0 licence, 30 January 2025.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978. doi: 10.1007/BF01588971.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. Document ranking with a pre-trained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 708–718, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.63. URL https://aclanthology.org/2020.findings-emnlp.63/.

Chao Peng, Bin Wang, Zhilei Long, and Jinfang Sheng. AdaGReS: Adaptive greedy context selection via redundancy-aware scoring for token-budgeted RAG. arXiv:2512.25052, 2025. URL https://arxiv.org/abs/2512.25052.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! arXiv:2312.02724, 2023. URL https://arxiv.org/abs/2312.02724.

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.

Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. Set-encoder: Permutation-invariant inter-passage attention for listwise passage re-ranking with cross-encoders. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval (ECIR 2025)*, Lecture Notes in Computer Science, pp. 1–19. Springer, 2025. doi: 10.1007/978-3-031-88711-6_1.

Amartya K. Sen. Choice functions and revealed preference. *The Review of Economic Studies*, 38(3), 1971. doi: 10.2307/2296384.

Sadanand Singh, Allam Reddy, and Manan Chopra. Beyond the reranker: Do RAG retrieval enhancements help once a strong reranker is present? arXiv:2606.28367, 2026. URL https://arxiv.org/abs/2606.28367.

Mooho Song and Jay-Yoon Lee. Retrieving a set, not independent passages: Set-level compatibility learning for efficient set exploration. arXiv:2607.05712, 2026. URL https://arxiv.org/abs/2607.05712.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14918–14937, 2023. URL https://aclanthology.org/2023.emnlp-main.923/.

Yixuan Tang and Yi Yang. MultiHop-RAG: Benchmarking retrieval-augmented generation for multi-hop queries. In *Proceedings of the First Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/forum?id=t4eB3zYWBK.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1 (NeurIPS Datasets and Benchmarks 2021)*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/65b9eea6e1cc6bb9f0cd2a47751a186f-Abstract-round2.html.

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, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Jingjin Wang and Jiawei Han. PropRAG: Guiding retrieval with beam search over proposition paths. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6212–6227, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.317. URL https://aclanthology.org/2025.emnlp-main.317/.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oğuz. Answering complex open-domain questions with multi-hop dense retrieval. arXiv:2009.12756, 2021. URL https://arxiv.org/abs/2009.12756.

Xinlong Xu and Yoshua Y. Li. EviReform: Evidence-guided query reformulation for multi-hop graph retrieval. arXiv:2608.13006, 2026. URL https://arxiv.org/abs/2608.13006.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William 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, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Soyoung Yoon, Eunbi Choi, Jiyeon Kim, Hyeongu Yun, Yireun Kim, and Seung-won Hwang. ListT5: Listwise reranking with fusion-in-decoder improves zero-shot retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2287–2308, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.125. URL https://aclanthology.org/2024.acl-long.125/.

Bingxue Zhang, Jianying Jia, and Feida Zhu. Covering the unseen: Information demand coverage optimization for retrieval-augmented generation. arXiv:2606.29328, 2026a. URL https://arxiv.org/abs/2606.29328.

Hengran Zhang, Keping Bi, Jiafeng Guo, Jiaming Zhang, Shuaiqiang Wang, Dawei Yin, and Xueqi Cheng. Distilling a small utility-based passage selector to enhance retrieval-augmented generation. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pp. 22–30. Association for Computing Machinery, 2025a. doi: 10.1145/3767695.3769496.

Litong Zhang, Jiaxin Li, and Kuo Zhao. DualView: Adaptive local-global fusion for multi-hop document reranking. arXiv:2605.18767, 2026b. URL https://arxiv.org/abs/2605.18767.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. arXiv:2506.05176, 2025b. URL https://arxiv.org/abs/2506.05176.

A GUIDE TO THE APPENDICES

Claim in the body	Appendix	Figures and tables
Every definition behind F: the relevance term, the bridge matrix and its two normalisations, the coverage matrix, the greedy search; the frozen constants; the extractor tiers; a re-implementation check	A	Tables 4, 5, 6, 7
How the instance of Figure 1 was picked	B	Table 8
Theorem 1 and the two-pool test; the share of questions on which the graph stage reveals a cycle	C	Table 9
The released plug-ins and rerankers installed in place of the ranking on MuSiQue, the plug-ins also as modifiers there and in place of the ranking on the four other corpora; the numbers behind Figure 5	D	Figure 6; Tables 10, 11, 12; supplementary notes, Section S9
How the configuration was chosen; every candidate gains	E	Table 13
The budget ladder from $k = 3$ to 20; chains completed and broken	F	Tables 14, 15
MMR, DPP, facility location and cohesion at three levels of tuning	G	Figure 7; supplementary notes, Section S9
The terms on their own; the bridge matrix replaced by cosines; the deletion test	H	Tables 16, 17
Every cell with its interval; MuSiQue by split; Recall@5 beside Chain@5	I	Figure 8; supplementary notes, Section S9
The reading stage: its prompt and fallback rule, the dependent-hop share, the earlier form, the same model as a listwise reranker, the top-ten control on every corpus	J	Figure 9; Tables 18, 19, 20
Downstream: KNIT at the 72B reader cell by cell, other reader families, passage order, chain completion against the reader's answer, the plug-ins at the reader, a family-wise reading	K	Figures 8d and 10; Tables 21, 22
News and QASC: the terms on their own, same-article edges, the reader; the science corpus in full	L	Figures 11 and 12; Tables 23, 24; supplementary notes, Section S9
The five in-house pipelines and the ten-pipeline population	M	Table 25
The instrument: the exact test, the bootstrap, the sign tests, and PropRAG's level against the published measurement	N	—

Algorithm 1 KNIT at the truncation (Section 2; every term is defined in this appendix, and the reading stage in Appendix J)

Require: question, pool P (the pipeline's top N, with scores), proposition graph, target k

1: $M_{pq} \leftarrow$ IDF of the rarest entity outside the question shared by the propositions of $p, q \in P$
2: Score a set $S \subseteq P$ by $F(S) = \sum_{p \in S} \mathrm{rel}'(p) + \gamma\,\mathrm{STRUCT}(S) + \beta\,\mathrm{COV}(S)$ ▷ Eq. 1
3: $S_{\mathrm{pair}} \leftarrow$ greedy fill from the pool's best pair, to $2k$ passages if the reading stage follows, else k
4: $S_{\mathrm{path}} \leftarrow$ the same fill from the best connected seed path of two to four passages
5: $S \leftarrow$ the one of S_{pair} and S_{path} with the larger F
6: **return** the k passages of S that one LLM call keeps if the reading stage follows, else S

A IMPLEMENTATION, IN FULL

The greedy search of Section 2.1, which adds one passage at a time, runs from two seeds, the pool's best pair and its best connected seed path, and keeps the set with the larger F. The definitions below reproduce every frozen selection: an independent implementation that follows them gives the same five passages on 1,000 of 1,000 questions on each of the five published pipelines.

Constants. Table 4 gives the frozen configuration constant by constant. In the search of Appendix E, the frozen configuration is the cell written $\mathrm{N30}|\mathrm{all}|2.5|0.5$.

Table 4: **The frozen configuration, constant by constant.** Every row but δ, which is inert here, is read by the search defined below.

constant	value	what it fixes
γ	2.5	the weight on the connection term
β	0.5	the weight on the coverage term
λ	0.2	the title bonus, carried inside rel'
pool depth N	30	candidates read per question
budget k	5	passages returned
coverage slots	`all`	the question's own non-stopword tokens after the 103-word stoplist below, with no cap
structure shape	S3	the maximum spanning tree
seeding construction	C3	the best-pair fill and the seed-path fill, which is what this section implements
near-duplicate block	on, Jaccard 0.7	constrains the search, never the objective
path budget	30,000	node sets recorded before route 2 stops
δ	0.5	a degree penalty read only by another structure shape, so inert at the frozen S3 shape
bridge gauge	$M^{\max}$	the *stack* gauge, defined below

Inputs. Per corpus: passages as `(title, text)`, and a proposition index holding `proposition_units`, `entity_nodes` and `proposition_entity_incidence`. Per question: the pipeline's top-N pool as corpus row ids *in the pipeline's own rank order* (position 0 is its best-ranked passage, and that order is inherited by every tie-break in the search) with the pipeline's raw scores. That is the whole input. For MuSiQue the pool is $N = 30$ over the corpus's N_{pass} passages (Table 6), the document count in every IDF below. The four published pipelines that ship no graph of their own read a proposition index built here, one per corpus, with one open-weight instruction model, Qwen2.5-14B-Instruct, and one prompt; PropRAG reads its own extractor's graph on the Wikipedia corpora, of 59,605 propositions on MuSiQue, and on news and on the science corpus the index built here converted to its cache format. The extraction tier was chosen on MuSiQue's tuning half by the rule in the released selection file, which takes the largest mean gain over those four pipelines with ties inside 0.2 of a point going to the smaller model; Table 5 reads the tiers at the frozen configuration on all 1,000 questions, not the tuning-half values the rule read. The third-party released index (Xu & Li, 2026), whose own counts Table 6 carries, is the *reference graph* of Appendix E. The reference graph and PropRAG's own graph are not ours and are not priced here; they are identified by content:

```
reference graph, sha256
  025db1a9df61be3d98830e6779bda4d58d9d829730d04d497062975179f9b8a8
PropRAG's own graph, sha256
  cc8cb73c30c2a5ab42f6815e5298605853ed778f33e0bdb7ebc0b03c71efd4d2
```

The gain at every extraction size. Table 5 reads the graph stage's gain over each pipeline's own Top-5 on each of these graphs at the frozen configuration.

Table 5: **The graph stage's gain at each extraction size, at the frozen configuration.** Paired chunk-exact Chain@5 over each pipeline's own Top-5, MuSiQue, depth 30, on the four published pipelines that ship no graph of their own. Per pipeline, exact McNemar with a Clopper–Pearson interval on the discordant split. *Every interval excludes zero.* The reference graph is the third-party released index; the second prompt is a second extraction at the deployed size and is not one of the sizes compared.

graph, $n = 1{,}000$	BM25	Hybrid	MonoT5	Dense
7B	$+5.80$ $[+4.72, +6.27]$	$+7.20$ $[+5.39, +8.60]$	$+9.90$ $[+8.22, +11.09]$	$+7.20$ $[+5.33, +8.68]$
14B (deployed)	$+6.00$ $[+4.92, +6.47]$	$+8.40$ $[+7.02, +9.23]$	$+9.30$ $[+7.51, +10.63]$	$+7.50$ $[+5.68, +8.90]$
32B	$+6.20$ $[+5.48, +6.20]$	$+7.90$ $[+6.29, +9.03]$	$+9.90$ $[+8.22, +11.09]$	$+8.30$ $[+6.64, +9.48]$
reference graph	$+7.10$ $[+6.10, +7.45]$	$+8.90$ $[+7.32, +9.98]$	$+11.60$ $[+10.14, +12.50]$	$+9.20$ $[+7.57, +10.33]$
14B, second prompt	$+5.80$ $[+4.72, +6.27]$	$+7.70$ $[+6.28, +8.60]$	$+9.70$ $[+8.15, +10.72]$	$+6.60$ $[+4.80, +8.00]$

Tokenisation. $\mathrm{tok}(s)$ is the sequence of maximal runs matching `[a-z0-9]+` in the NFKD-normalised, lowercased s, applied to question text, proposition text, passage titles, title-plus-text and entity names alike, with stemming, lemmatisation and stopword removal all omitted. NFKD is what separates it from the `\w+` reading: a combining mark is not in `[a-z0-9]` and terminates its run, dropping a trailing mark (`Beyoncé` → `beyonce`) and *splitting* an internal one (`naïve` → `nai`, `ve`). $\mathrm{norm}(s)$ additionally replaces every character outside `[a-z0-9 ]` by a space, collapses whitespace and trims; it decides only whether two candidates carry the same title in the near-duplicate rule and never scores.

The two IDFs. One formula everywhere, $\mathrm{idf}(df) = \log\big(1 + N_{\mathrm{pass}}/(1 + df)\big)$, natural logarithm: differing in value, though not in ordering, from $\log N_{\mathrm{pass}}/df$ and $\log((1 + N_{\mathrm{pass}})/(1 + df))$, and COV's normalisation by its own maximum makes the value matter. *Term IDF, for* COV: a passage's vocabulary $V(p)$ is the union of $\mathrm{tok}(\cdot)$ over the *texts of its propositions*, not the passage text, and $df(t)$ counts passages with $t \in V(p)$. A term absent from every passage's proposition vocabulary scores 0, not the maximum. This is the detail a re-implementation most easily gets wrong, because the usual convention is the opposite. Since $df \geq 1$ forces $\mathrm{idf}_{\mathrm{term}} \geq 0.693$, a zero is strictly below every term that does occur. *Entity IDF, for* STRUCT, written $\mathrm{idf}'(e)$: counted at passage level over the *alias-expanded* incidence A' below rather than the released one, the prime marking that expansion and not the title bonus the prime marks in rel'. It is held in `float32`, because the edge weight is a maximum over these values and the final arbitration compares two objective values at a tolerance of 10^{-12}.

Relevance and the title indicator. For pool member j with pipeline raw score r_j, $\mathrm{rel}'(j) = (r_j - \min_P r)/(\max_P r - \min_P r + 10^{-12}) + \lambda\,\mathrm{title}(j)$, and rel', with the bonus already inside it, is what every later expression called "relevance" reads. The 10^{-12} maps a pool of identical scores to the zero vector rather than dividing by zero, and that case is reached. $\mathrm{title}(j) = 1$ iff the token list $\tau_j = [\, t \in \mathrm{tok}(\text{title of } j) : t \notin \mathcal{D}\,]$ is non-empty *and* every element of τ_j occurs in $\mathrm{tok}(\text{question})$. It is a token-subset test against the question's token set, not a substring test against the question string, and the title is tokenised rather than normalised. $\mathcal{D}$ is the seven-token disambiguator set `film`, `series`, `tv`, `american`, `song`, `band`, `album`, so that a title made only of these is name-free and earns zero through the emptiness clause.

The near-duplicate (twin) relation. a and b are twins iff $\mathrm{norm}(\text{title}_a) = \mathrm{norm}(\text{title}_b)$ *and* $|W_a \cap W_b|/|W_a \cup W_b| \geq 0.7$, where W_a and W_b are the token sets of the two *corpus passages*, title and text joined by one space and tokenised: the only place in the layer where raw passage text is read. It is symmetric with an empty diagonal and constrains the search, not the objective.

Alias expansion of the entity incidence. This is the layer's own work, runs at inference once per (corpus, graph) pair before any question is scored, and a reproduction that omits it does not reproduce

our selections. Let $T_e = \mathrm{set}(\mathrm{tok}(\text{name of } e))$; entities whose name tokenises empty are skipped (57 of them). Let $\mathcal{A}$ be the 52-token closed-class list

```
a an the of in on at to for and or from by with as is are was
were be been being this that these those it its his her their
our your my de la le du des el los las i ii iii iv v vi vii
viii ix x
```

and $C_e = T_e \setminus \mathcal{A}$. Let $\mathrm{tok_df}(t)$ count the entity *names* containing t: names, not propositions and not passages. For each s with $C_s \neq \emptyset$ the *anchor* is $\mathrm{anc}(s) = \arg\min_{t \in C_s}(\mathrm{tok_df}(t), t)$, the rarest content token with the token string breaking ties; if $\mathrm{tok_df}(\mathrm{anc}(s)) > 25$ then s contributes no pairs, which is what keeps the expansion off generic names. Otherwise every $l \neq s$ whose name contains $\mathrm{anc}(s)$ and satisfies $T_s \subsetneq T_l$ forms an ordered alias pair (s, l). With A the released proposition–entity incidence, $A' = A \cup \{(i, s) : (i, l) \in A,\ (s, l) \text{ an alias pair}\}$ adds the short id to every proposition carrying the long one. The pairs are computed once against the original A rather than re-derived from A', so the expansion does not compose transitively and applying it twice yields a different object. Entity IDF is recomputed from A' at passage-level *df*, so a short name absorbed into many long names sees its own IDF fall. Two maps are kept, $\mathrm{up}(s) = \{l : (s, l)\}$ and $\mathrm{down}(l) = \{s : (s, l)\}$.

Question entities, and the widening applied to them. Build a lookup keyed by the token tuple $\mathrm{tok}(\text{name})$ for every entity whose name has 1 to 8 tokens; where two share a tuple the first in ascending `entity_id` order claims it and later ones are unreachable, the collisions counted in Table 6. Matching is longest-first and non-overlapping: with $q = \mathrm{tok}(\text{question})$ and all positions unused, for $L = \min(8, |q|)$ down to 1 and each start $i = 0, \ldots, |q| - L$ ascending, if positions $i, \ldots, i{+}L{-}1$ are unused and $q[i{:}i{+}L]$ is in the length-L table, record that entity and mark those positions used. That yields Q_0; it is not a substring test. The edge builder removes the *widened* set $Q = Q_0 \cup \bigcup_{e \in Q_0}(\mathrm{up}(e) \cup \mathrm{down}(e))$, in both directions, because widening one way only lets the question's own entity re-enter under its other name.

The bridge matrix. The bridge matrix M holds the link weights of Section 2.1. For pool member a let $\mathcal{E}_a$ be the family of sets $\{e \in A'(i) : e \notin Q\}$ over the propositions i of a, dropping empty ones. For $a \neq b$ the released builder computes both

$$M^{\max}_{ab} = \max_{\substack{X \in \mathcal{E}_a,\ Y \in \mathcal{E}_b \\ X \cap Y \neq \emptyset}} \max_{e \in X \cap Y} \mathrm{idf}'(e), \qquad M^{\mathrm{sum}}_{ab} = \max_{\substack{X \in \mathcal{E}_a,\ Y \in \mathcal{E}_b \\ X \cap Y \neq \emptyset}} \sum_{e \in X \cap Y} \mathrm{idf}'(e), \tag{2}$$

where idf' is the alias-expanded entity IDF defined above. Both entries are 0 where no proposition pair shares an entity that is not excluded, and both matrices are symmetric with a zero diagonal. The frozen configuration reads $M^{\max}$, the *stack* gauge.

Write $M = M^{\max}$. Two normalisations follow and the search reads both. The first divides by the largest entry, $M^{\mathrm{n}} = M / \max_{a,b} M_{ab}$. The second divides by $v_1^{\max}$, the largest total bridge weight any five-subset of the pool carries under M^{n}: $v_1^{\max} = \max_{|S|=5} \sum_{\{a,b\} \subseteq S} M^{\mathrm{n}}_{ab}$ over all $\binom{30}{5} = 142{,}506$ five-subsets at the frozen depth, giving $M^{\mathrm{c}} = M^{\mathrm{n}} / v_1^{\max}$. Each division is skipped when its divisor is 0. STRUCT, the fill's marginal and the seed-path weights read M^{c}; the pair seed alone reads M^{n}. Reading M^{c} in the pair seed as well would divide its connection term, and only its connection term, by $v_1^{\max}$, which rescales γ against β inside that one expression.

$\mathrm{STRUCT}(S)$ is the maximum-spanning-tree weight of S under M^{c}. At $|S| = 5$ our release maximises over the 125 spanning trees of K_5.

Coverage. Let $q = \mathrm{tok}(\text{question})$. Remove the 103 stopwords below and deduplicate; every remaining token opens a slot, so n_{slots} is the question's own non-stopword type count and there is no cap. The slots are ordered by $(-\mathrm{idf}_{\mathrm{term}}(t), t)$ ascending (decreasing IDF with the *token string* as tie-break, not question order), which fixes the order of the slots and not which slots exist, and a term absent from every proposition vocabulary has IDF 0 and sorts strictly last. Entry (ℓ, j) of the coverage matrix H is slot ℓ's term IDF when that token is in $V(\text{pool member } j)$ (again the proposition-derived vocabulary, not the passage text) and 0 otherwise; the matrix is divided by its own largest entry (skipped if that is 0) and then by 5, and $\mathrm{COV}(S) = \sum_\ell \max_{j \in S} H[\ell, j]$, so it ranges on $[0, n_{\mathrm{slots}}/5]$ rather than $[0, 1]$. Both five-based normalisers, the subset size of $v_1^{\max}$ and the coverage divisor, stay at five at every budget; the budget enters only the search, as the size of the set it fills (k, or $2k$

before the reading stage) and as the cap on route 2's seed. Neither COV nor STRUCT is a sum over members: a passage's worth depends on the others present.

```
a an the and or but if then than that this these those it its
his her their our your my who whom whose which what when where
why how is are was were be been being am do does did doing
have has had having of in on at to for from by with without as
into onto over under between during before after above below
up down out off again further once here there all any both
each few more most other some such no nor not only own same so
too very s t can will just don should now
```

It is not a standard published list: it carries the question words `what`, `when`, `where`, `who` and `how`, which propositions rarely contain and would otherwise score as rare content terms, and the contraction fragments `s`, `t` and `don`.

The search: the fill. A fill extends S to the stage's budget, $k = 5$ alone and $2k = 10$ when the reading stage follows. Eligible candidates are the pool members outside S that are not a twin of any member of S; if that leaves the eligible set empty the twin block is released for the remainder of this fill, and stays released. That branch is a termination guarantee rather than a scoring rule. Each eligible c scores $\mathrm{rel}'(c) + \gamma \max_{s \in S} M^{\mathrm{c}}_{cs} + \beta\,\mathrm{COV}(S \cup \{c\})$, the middle term 0 when S is empty, largest score winning with *ties going to the lower pool position*. The coverage term is the full $\mathrm{COV}(S \cup \{c\})$ rather than its increment; within a step the two differ by a constant and rank identically.

Table 6: **Anchor counts, on the third-party reference index.** Every row below is counted on the reference index of Appendix E rather than on our graph; the definitions and the MuSiQue corpus of 11,656 passages are the same. Each row is computable before any question is scored and fails independently, so a re-implementation that diverges on a selection can find which definition it read differently. The alias rows depend on the tokeniser, the closed-class list, the anchor's tie-break and the strictness of the subset test at once, and are the ones most likely to differ.

quantity	value
passages / propositions / entities	11,656 / 113,193 / 88,995
incidence pairs, as released	249,860
entities whose canonical name tokenises non-empty	88,938
distinct tokens over entity names	43,842
distinct name token-tuples of length 1–8 (colliding tuples)	87,897 (525)
alias pairs at the *df* cap of 25	39,940
distinct short names / distinct long names among them	11,325 / 30,397
incidence pairs after expansion	337,810
node sets of size 2–4 available at $N = 30$	31,900
five-subsets entering $v_1^{\max}$ at $N = 30$	142,506

Route 1: the pair seed. Over all pairs $a < b$ score $g(a, b) = \mathrm{rel}'(a) + \mathrm{rel}'(b) + \gamma M^{\mathrm{n}}_{ab} + \beta\,\mathrm{COV}(\{a, b\})$, set $g = -\infty$ on twin pairs, take the argmax with ties to the first pair in upper-triangular order, ignoring the $-\infty$ rule if every pair is a twin, and fill from that pair. This is *not* the pool's two top-ranked passages.

Route 2: the seed path. Two nodes are adjacent iff $M^{\mathrm{c}}_{ij} > 0$. Push the one-node walks $([0]), \ldots, ([N-1])$ onto a stack in that order and repeatedly pop from the top, carrying each walk's running sum w of consecutive edge weights. On popping (path, w): if $|\mathrm{path}| \geq 2$ and its *node set* is unrecorded, record that set with weight w, stopping at 30,000 sets; if $|\mathrm{path}| = 4$, do not extend; otherwise push $(\mathrm{path} + [j],\ w + M^{\mathrm{c}}_{\mathrm{last},j})$ for every neighbour j not in the path, ascending. The walk ignores the twin relation. Three things here cannot be derived from the constants: the unit is the node set rather than the path; a node set's weight is that of the *first* walk to reach it in that order, so the order is part of the definition; and the cap counts node sets (Table 6), so at $N = 30$ it can bind only when nearly every node set of size two to four has been recorded. Score each recorded set R as $\sum_{j \in R} \mathrm{rel}'(j) + \gamma\sqrt{\max(w_R, 0)} + \beta\,\mathrm{COV}(R)$ (the square root of the *sum* of consecutive edge weights, not the sum of their roots, and the same expression under every shape) and take the argmax, ties to the first set recorded. Then drop twins: scan the winner's members in ascending pool order, remove the lower-rel' member of the first twin pair (ties removing the earlier), repeat, then truncate

to k; the truncation is inert here, where the size cap of four already bounds the seed below k. Fill from what remains; if no node set was recorded, route 2 is skipped.

Arbitration and output. With $F(S) = \sum_{j \in S} \mathrm{rel}'(j) + \gamma\,\mathrm{STRUCT}(S) + \beta\,\mathrm{COV}(S)$, route 2's set is returned iff $F(S_{\mathrm{path}}) > F(S_{\mathrm{pair}}) + 10^{-12}$. Values of F equal up to rounding occur, and without that tolerance they resolve the other way. The five selected pool positions are sorted ascending, i.e. emitted in pipeline-rank order, before being mapped back to corpus row ids, so a re-implementation comparing selection arrays element by element should sort the same way.

Splits. Within each hop stratum (518/316/166 questions at two, three and four hops; the tuning half draws 259/158/83) questions are ordered by the hex digest of $\mathrm{sha256}(\texttt{"split1|"} + \textit{qid})$ and each stratum's first half is the tuning half. The order is the digest's *string* order.

Three implementation details. Two of them leave every selection measured here unchanged: the floating-point accumulation order inside STRUCT (one unit in the last place), and which of two entities sharing a name token-tuple the question matcher reaches, ours being ascending `entity_id`. The third, the pipeline's own score ties, does not: the emitted pool order is inherited by every tie-break, so a re-implementation that re-sorts the pool will diverge on those questions even with every definition above correct. The pools we release preserve the pipeline's order.

What the proposition graph costs. The graph is built once per corpus, before any question is asked, by one call of the 14B extraction model per passage. Table 7 gives the measured GPU-hours for the five corpora of this paper, each shard run on a single GPU.

Table 7: **Indexing cost of the proposition graph**, one 14B extraction call per passage, GPU-hours summed over the completed single-GPU shards.

Corpus	passages	GPU-hours	passages per GPU-hour
MuSiQue	11,656	12.30	947
HotpotQA	9,811	11.20	876
2WikiMultihopQA	6,119	7.14	857
MultiHop-RAG (news)	12,389	7.91	1,565
QASC	7,671	1.03	7,420

B THE WORKED INSTANCE

Figure 3 draws one MuSiQue question on MonoT5's top-30 pool, Table 8 lists the two selections passage by passage, and Figure 4 follows the same question through both selectors as a compact flow.

Table 8: On this question the Top-5 spends three of its five slots on further passages titled *Estonia*, repeating a hop the gold at rank 1 already covers, while the graph stage gives one of them up and reaches rank 8 for the passage that closes the remaining hop and completes the chain; that single swap is the whole of the difference (one MuSiQue question, MonoT5's own top-30, chunk-exact Chain@5). G marks gold; *rank* is position in the pipeline's own pool; rel' is the objective's relevance term, the min–max normalised pipeline score plus $\lambda = 0.2$ for a title the question names, which is why rank 1 outscores rank 0 here.

	The pipeline's Top-5 (Chain@5 = 0)			The graph stage's five (Chain@5 = 1)		
	chunk	rank	rel'	chunk	rank	rel'
1	*Korean War* G	0	1.0000	*Korean War* G	0	1.0000
2	*Estonia* G	1	1.1488	*Estonia* G	1	1.1488
3	*Estonia*	2	1.1313	*Estonia*	3	0.8920
4	*Estonia*	3	0.8920	*Estonia*	4	0.8860
5	*Estonia*	4	0.8860	*Imperialism* G	8	0.5665

How the instance was chosen. By a hashed rule released with the artifacts: the pipeline is the published pipeline whose bare Top-5 Chain@5 on the 500-question holdout is the median of the five;

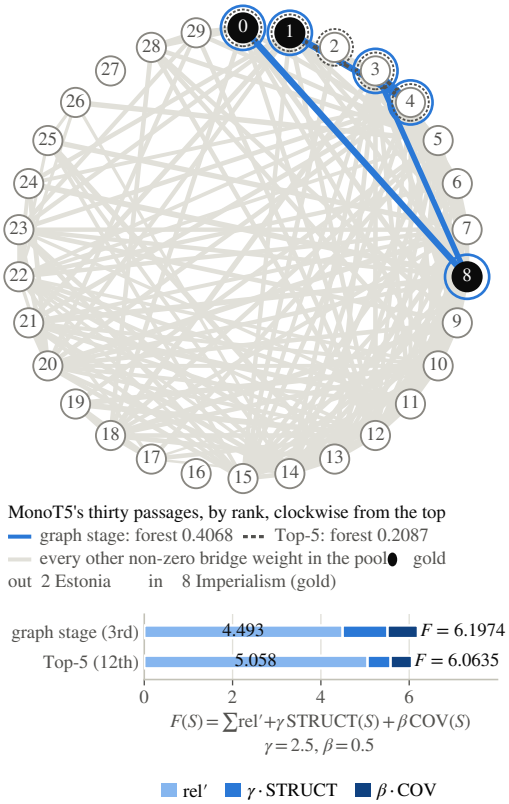

Figure 3: **One question, two sets.** MonoT5's thirty-passage pool for *Where did the arguer that the country in control of Estonia before the Germans had become an imperialist power declare he would intervene in the Korean conflict?*, picked by a released, hashed rule (*How the instance was chosen*, below): chords are the 165 non-zero bridge weights, Top-5's own five (ranks 0–4) and the graph stage's five (0, 1, 3, 4, 8) with their maximum spanning forests, gold filled. The graph stage gives up 0.56 of relevance for +0.50 in the connection term and +0.20 in the coverage term, $F = 6.197$ against 6.064, third of 142,506 subsets where the Top-5 ranks 12th, and completes the chain: the gold at rank 8 is a passage the pipeline buried under three further passages titled *Estonia*. (MonoT5: NV-Embed-v2's top 50 reranked by monoT5; MuSiQue.)

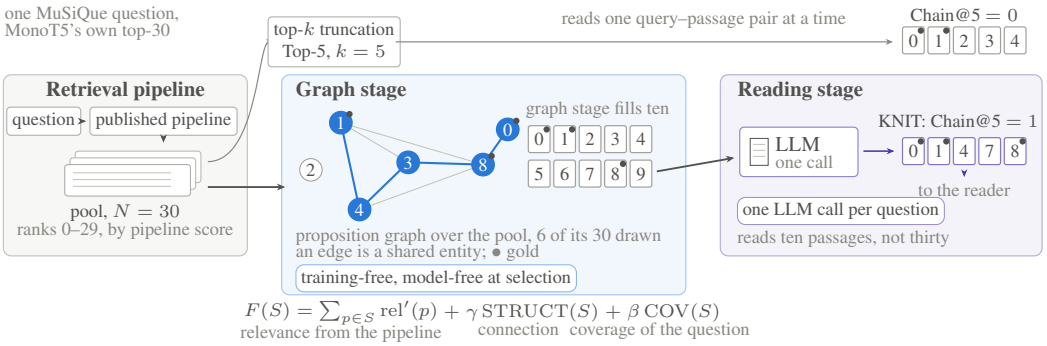

Figure 4: **The same instance as a flow.** The Top-5 scores each query–passage pair alone and keeps the five best; here it misses a gold passage, so the chain is incomplete: Chain@5 is 0. The graph stage reads the edges of a proposition graph over the same pool and fills connected passages that cover the question; its own five already complete the chain. For the reading stage it fills ten, and one LLM call keeps five of them, again a complete chain.

the candidates are the holdout questions the frozen graph stage completes and the pipeline's own Top-5 does not, ordered by the pool position of the deepest gold chunk the pipeline retrieved; the instance is that list's upper median.

C THEOREM 1 AND THE TWO-POOL MEASUREMENT

Fix a query q and a finite universe U of passages. A **k-selection rule** σ assigns to every pool $C \subseteq U$ with $|C| \geq k$ a k-subset $\sigma(C) \subseteq C$. Say σ is **pool-blind-rationalisable** if there is a score $s : U \to \mathbb{R}$ and a fixed strict total order $\prec$ on U, neither depending on C, such that $\sigma(C)$ is the k members of C with the largest s, ties by $\prec$, for every C. Define the **revealed relation** $p \triangleright p'$ iff some pool C has $p \in \sigma(C)$ and $p' \in C \setminus \sigma(C)$.

Theorem 1. *σ is pool-blind-rationalisable if and only if $\triangleright$ is acyclic. Proof.* ($\Rightarrow$) If $p \triangleright p'$ through C then $s(p) > s(p')$, or $s(p) = s(p')$ and $p \prec p'$; so $\triangleright$ is contained in the strict total order "larger s, then earlier in $\prec$", hence acyclic. ($\Leftarrow$) If $\triangleright$ is acyclic its transitive closure is a strict partial order; let $\succ$ be a linear extension. For a pool C, any $p \in \sigma(C)$ and $p' \in C \setminus \sigma(C)$ satisfy $p \triangleright p'$ and so $p \succ p'$, so every chosen member precedes every one not chosen and $\sigma(C)$ is the top-k of C under $\succ$. Take $s(p)$ to be p's position from the bottom of $\succ$ and $\prec$ to be $\succ$ itself; s has no ties, so $\prec$ is never used. □

Corollary 1 (two nested pools). *For pools $A \subset B$, the relation revealed by $\sigma(A)$ and $\sigma(B)$ has a cycle iff there exist $p \in \sigma(A) \setminus \sigma(B)$ and $p' \in \sigma(B) \setminus \sigma(A)$ with $p' \in A$. Proof.* Every edge comes from one of the two pools and two consecutive edges from the same pool would need a passage both chosen and not chosen there, so a cycle alternates pools and has even length. Taking an edge from A and the next from B: $p \in \sigma(A)$ and $p' \in A \setminus \sigma(A)$, $p' \in \sigma(B)$, and the passage after it is not chosen in B; continuing round, the edge that closes the cycle comes from B and points at p, so p is not chosen in B: $p \in \sigma(A) \setminus \sigma(B)$ and $p' \in \sigma(B) \setminus \sigma(A)$ with $p' \in A$, the stated pair. The converse is the 2-cycle $p \triangleright p'$ via A and $p' \triangleright p$ via B. □

Corollary 2 (what one pool cannot show). *With a single pool per query, $\triangleright$ is acyclic for every σ.* With one pool C, the indicator score $s'(p) = \mathbf{1}[p \in \sigma(C)]$ rationalises $\sigma(C)$, so no cycle can appear. Pool dependence is invisible with one pool and observable with two, and every cycle rate below is a lower bound. A deletion test of pool dependence, run on the graph stage's own selections, is in Appendix H.2.

Sources. Theorem 1 is the characterisation, by Chambers & Yenmez (2018), of q-responsive choice by q-acceptance (their q a quota, set here to k) and an acyclic revealed relation, on the pools of size at least k and with the rationalising order written as a score and a fixed tie-break. The proof here is theirs: the same transitive closure and the same linear extension. The fixed-cardinality statement appears first in Eliaz et al. (2011), under Sen's α (Sen, 1971) and a second axiom on the full power set. What is not in those sources is Corollaries 1 and 2, statements about what two nested pools and one pool can witness, and the retrieval reading: the scorers that read (q, p) alone under a pool-independent tie-break (BM25, any dense encoder, any pointwise cross-encoder, any pointwise LLM judge) are pool-blind-rationalisable, so their selections are consistent across pool depths.

In the body's terms, a pool-blind score is a fixed per-passage score, and the swap of Section 5 is the 2-cycle of Corollary 1. The top-50 pool's first thirty rows equal the top-30 pool on 100.0% of questions on every pipeline, so the two are nested and Corollary 1 applies as stated. Table 9 gives the cycle rate each pipeline's two pools reveal, with the two controls.

The cohesion control cycles less at the relevance-heavy end of its constant (9.0% at $c = 0.9$ against 16.2% at $c = 0.1$, non-monotone between), the direction expected as the operator approaches a pool-independent score, so a positive rate is a property of reading the pool rather than an artefact of the instrument. Three properties of the graph stage can make it swap: its connection and coverage terms score passages together, each of its terms is normalised within the pool (Appendix A), and its seed-path search stops at a fixed number of node sets (Table 4). The two-pool measurement does not separate them; the deletion test of Appendix H.2 ties the set-level effect to the connection term.

Table 9: The graph stage's selections are provably not the top-k of any pool-independent per-passage score on 13.3 to 22.0% of questions. The pipeline's own Top-5 cycles on no question, by construction, and our cohesion operator cycles on 9.0 to 17.4%, read at the shallower top-20-against-top-30 pair only. The pipelines are ordered but not separated: every pipeline's interval overlaps at least one other's: BM25's overlaps MonoT5's, and MonoT5's overlaps the other three. "differs" counts questions whose five change at all, and "consistent" those whose new members at $N = 50$ all come from ranks 30–49, which a pool-blind scorer could have produced. Reading fifty rather than thirty moves the frozen graph stage's own Chain@5 by -0.8 to $+1.7$ points, two of five significantly upward (BM25 $+1.30$, MonoT5 $+1.70$). (MuSiQue, $n = 1{,}000$, frozen configuration at $N = 30$ against the same configuration at $N = 50$.)

Pipeline	differs	**cycle rate**	95% CI	consistent	Chain@5 30 → 50
BM25	27.4%	**22.0%** (220)	[19.47, 24.70]	5.4%	16.3 → 17.6
Hybrid	14.5%	**13.3%** (133)	[11.26, 15.56]	1.2%	29.1 → 29.1
MonoT5	21.6%	**17.4%** (174)	[15.10, 19.89]	4.2%	34.2 → 35.9
Dense	17.3%	**13.8%** (138)	[11.72, 16.09]	3.5%	45.7 → 45.8
PropRAG	23.3%	**15.6%** (156)	[13.40, 18.00]	7.7%	57.3 → 56.5
Controls					
the pipeline's own Top-5	0.0%	**0.0%**	—	—	unchanged on 1,000/1,000
cohesion, $c \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, 20 vs. 30	—	cycle rate **16.2% / 17.4% / 15.2% / 12.3% / 9.0%** in that order (Hybrid)			

D RELEASED PLUG-INS AND RERANKERS AT THE TOP-k TRUNCATION

The released checkpoints are `naver/provence-reranker-debertav3-v1` and `webis/set-encoder-large`. The two released plug-ins were installed at the frozen graph stage's own position, replacing the Top-5 of a 30-wide pool and changing nothing else, on all five pipelines (the plug-ins' analysis file pools the four other than MonoT5), at $n = 1{,}000$, against the same chunk-exact gold and under that file's rules. Set-Encoder, the set-aware cross-encoder (Schlatt et al., 2025), is stochastic and ran eleven seeded draws; Provence, the context-pruning head (Chirkova et al., 2025), is deterministic and ran once.

The calibration each model carries. Three released rerankers are installed as selectors. bge-reranker-v2-m3 is *calibrated*: it reproduces its two published BEIR numbers within the tolerance of Appendix G, but its bare first stage also falls within that tolerance on both, so this check does not separate the reranker from its first stage. RankZephyr-7B (Pradeep et al., 2023) is calibrated on the TREC 2019 Deep Learning passage track (DL19) within a tolerance its first stage also meets. Qwen3-Reranker-0.6B publishes no per-dataset number and is not calibrated. Provence and Set-Encoder were reproduced against their own published numbers; what that check can and cannot detect for them is in Appendix G.

Two installation designs. Both designs are read at the frozen configuration's 30-wide pools, the depth of Table 11. Write s_{comp} for the component's own score of a passage. Under design A the component is a *modifier* that keeps the pipeline's own scores: the five passages it selects are the top five of $\text{minmax}(\text{relevance}) + w\,\text{minmax}(s_{\text{comp}})$ at $w = 1$, the weight the objective gives its own relevance term. Under design B it *replaces the truncation*: the five it selects are the top five of s_{comp} alone. Choosing w per pipeline on the tuning half instead would set it to 0 on PropRAG for both components and to 0.5 and 0 on Dense: at $w = 1$ the two strong pipelines give the component more weight than their own tuning would. Neither design measures a component that retrieves for itself: in both, the pipeline decides which thirty of the 11,656 passages are in the pool.

Per pipeline (Table 10) the modifier recovers 52 to 86% of the drop the replacement costs the two strong pipelines.

Table 11 gathers the levels behind Figure 5: every component at the frozen depth, its span across the five pipelines, and whether it resets the level.

The two plug-ins on the other corpora. Figure 6 and Table 12 install both plug-ins in place of the ranking (design B, the design of Figure 5) on the corpora held out of every choice: 2WikiMultihopQA, HotpotQA, news and QASC, on all five pipelines. Nothing was tuned on them: the checkpoints, the pool depth of 30 and the rule for reading the results are MuSiQue's. Level reset recurs on the Wikipedia corpora and news. The differences between a plug-in and KNIT are compared by their levels and were not tested as paired contrasts. Before these cells were seen, the rule for this

Table 10: **The two installation designs, per pipeline.** Each component against the pipeline's own Top-5, in Chain@5 points, with the exact two-sided McNemar test and a Clopper–Pearson interval on the discordant split (MuSiQue, $n = 1{,}000$ questions per pipeline, 30-wide pools to $k = 5$). Set-Encoder is read at the median of its eleven draws in each design.

Pipeline	own Top-5	Component	A, modifier	B, replacement
BM25	10.30	Provence	+8.20 [+7.32, +8.39]	+9.70 [+8.26, +10.60]
		Set-Encoder	+6.70 [+5.71, +7.05]	+8.90 [+7.41, +9.86]
Hybrid	20.70	Provence	+3.40 [+1.71, +4.80]	+6.20 [+3.90, +8.23]
		Set-Encoder	+2.70 [+0.88, +4.31]	+2.60 [+0.25, +4.83]
MonoT5	24.90	Provence	+2.60 [+0.97, +3.98]	+1.10 [−1.19, +3.33]
		Set-Encoder	+0.40 [−1.15, +1.91]	−0.60 [−2.63, +1.47]
Dense	38.20	Provence	−1.50 [−3.34, +0.45]	−10.90 [−13.16, −8.31]
		Set-Encoder	−3.40 [−5.02, −1.53]	−13.90 [−15.63, −11.72]
PropRAG	53.90	Provence	−11.00 [−12.87, −8.74]	−25.20 [−27.03, −22.87]
		Set-Encoder	−14.00 [−15.59, −11.94]	−29.20 [−30.60, −27.24]

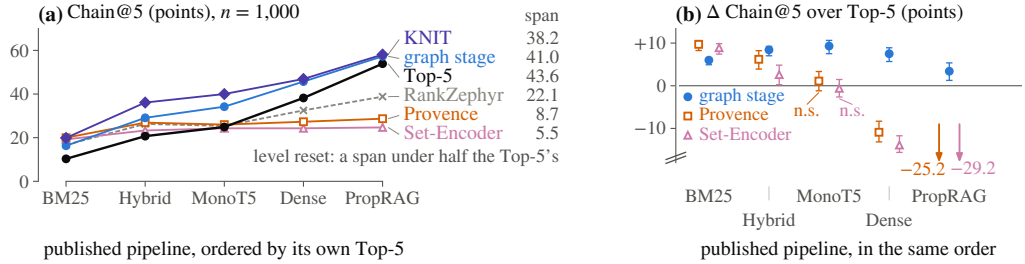

Figure 5: **The plug-ins flatten the pipelines; KNIT tracks them.** MuSiQue. The two plug-ins return about the same Chain@5 whatever pipeline feeds them (*level reset*); the graph stage and KNIT track the pipeline and sit above it. (a) Chain@5 on the five published pipelines, ordered by their own Top-5, with each series' span, its highest minus lowest value, at the right. The plug-ins and RankZephyr run as released and Set-Encoder at the median of eleven draws. The rerankers not drawn, bge and Qwen3, also reset the level. (b) Gain over the Top-5; n.s. marks an interval that covers zero, and PropRAG's two arrowed values lie below the axis. Values and spans are in Table 11, the plug-ins' intervals in this appendix, the graph stage's gains in Appendix E.

Table 11: **Level reset, the numbers behind Figure 5.** Absolute Chain@5 of each component installed in place of the ranking on the five MuSiQue pipelines, ordered by the pipelines' own Top-5 ($n = 1{,}000$; the two plug-ins and the three rerankers as released, at the frozen pool depth of 30, Set-Encoder at the median of its eleven seeded draws), then the reading stage's own 32B model handed the pipeline's top 30 in one call (Section 4.4), then the graph stage and KNIT. *Span* is the range across the five pipelines; a component *resets the level* when its span is under half the Top-5's 43.6. Appendices D and J.2 give every interval.

Component	BM25	Hybrid	MonoT5	Dense	PropRAG	span	resets?
the pipeline's own Top-5	10.3	20.7	24.9	38.2	53.9	43.6	—
Provence	20.0	26.9	26.0	27.3	28.7	8.7	resets
Set-Encoder, median draw	19.2	23.3	24.3	24.3	24.7	5.5	resets
bge-reranker-v2-m3	21.2	33.5	31.3	33.0	34.4	13.2	resets
Qwen3-Reranker-0.6B	20.8	30.2	29.8	30.3	31.7	10.9	resets
RankZephyr-7B	16.7	26.3	25.4	32.5	38.8	22.1	no
the 32B judge, top 30 in one call	21.8	40.4	40.5	43.2	49.0	27.2	no
graph stage alone	16.3	29.1	34.2	45.7	57.3	41.0	no
KNIT	19.9	36.1	40.0	46.9	58.1	38.2	no

comparison named one outcome as adverse to KNIT: a corpus on which the released components raise pipelines and lower none. That outcome appeared, on QASC. Every pipeline's own Top-5 on QASC is below the plug-ins' level, which is the case in which, by the reading of Section 4.4, a component helps.

The rerankers at the frozen depth. RankZephyr reads thirty candidates through its published sliding window, two windows of twenty at a stride of ten taken from the back of the list. Against each pipeline's own Top-5 at depth 30 it is significantly up on BM25 and Hybrid, not significantly

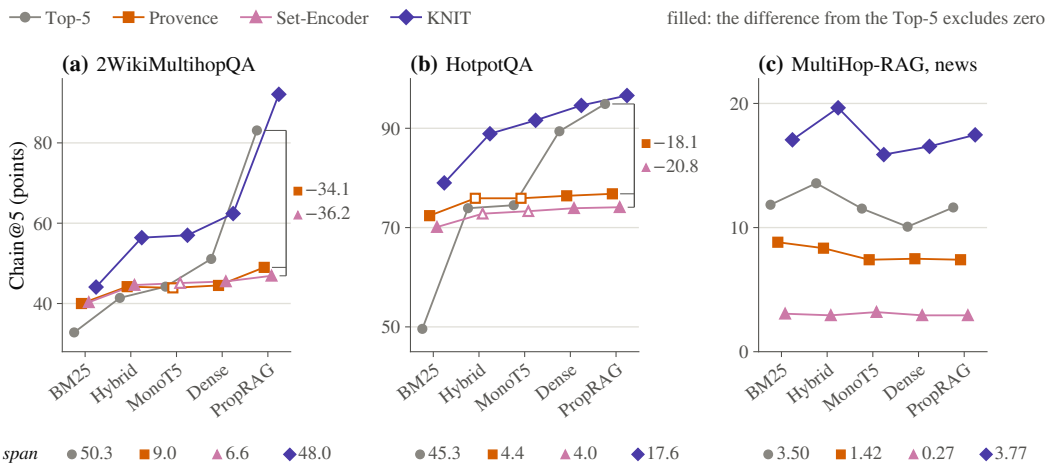

Figure 6: **Level reset on the held-out Wikipedia corpora and on news.** Absolute Chain@5 of each pipeline's own Top-5 (the reference), of the two released plug-ins installed in place of its ranking at depth 30 (Set-Encoder at the median of its eleven draws) and of KNIT on the same pool, pipelines in their MuSiQue order ($n = 1{,}000$ on each Wikipedia corpus, 2,255 on news); a filled mark's paired difference from the Top-5 excludes zero (exact 95% interval). Under each panel is each series' span over the five pipelines, the largest minus the smallest Chain@5: a component that returns the same level whatever pipeline feeds it has a small span, one that tracks the pipeline keeps the pipelines' own. The bracket at PropRAG gives each plug-in's paired difference from the Top-5 there. QASC, where neither plug-in resets the level, is in Table 12.

Table 12: **Level reset does not appear on QASC.** Chain@5 of each pipeline's own Top-5, of the two plug-ins in place of its ranking at depth 30 (Set-Encoder at the median of eleven draws) and of KNIT; Δ is the paired difference from the Top-5, bold where its exact interval at $\alpha = 0.05$ excludes zero. *Span* is the largest minus the smallest level in a column. Levels differ by as little as 0.21 (Set-Encoder 33.15 against KNIT 32.94 on Dense).

Pipeline	Top-5	Provence	Δ [95% CI]	Set-Encoder	Δ [95% CI]	KNIT	Δ
BM25	13.71	23.11	**+9.40** [+7.74, +10.50]	22.35	**+8.64** [+6.90, +9.86]	20.95	**+7.24**
Hybrid	23.65	32.07	**+8.42** [+5.77, +10.77]	32.72	**+9.07** [+6.57, +11.21]	28.62	**+4.97**
MonoT5	27.11	31.32	**+4.21** [+2.04, +6.13]	33.37	**+6.26** [+3.99, +8.22]	32.83	**+5.72**
Dense	31.75	32.18	+0.43 [−1.90, +2.74]	33.15	+1.40 [−0.85, +3.57]	32.94	+1.19
PropRAG	27.97	30.67	+2.70 [−0.27, +5.58]	31.53	**+3.56** [+0.71, +6.29]	29.48	+1.51
span	18.04	9.07		11.02		11.99	

different on MonoT5 and significantly down on Dense and on PropRAG, there by 15.1 points: up on the weak pipelines and down on the strong, as the two pointwise rerankers are.

Whether a released component demotes the *dependent* hop (the gold passage whose sub-question refers to a previous answer) more than the hop-1 passage is left open: of the four cells the analysis file counts (the two plug-ins on the two strong pipelines), two sit above their own depth-matched null and two sit inside it. The analysis file also names a Qwen3-Reranker-8B cell, which was not run.

Inference cost at the same position. On one CPU, each arm in one process with threads unpinned and no GPU on either side, the graph stage costs 11.9–31.0 ms per question at the frozen depth, against 4.636 s and 5.272 s for Set-Encoder on BM25's and PropRAG's pools, over 1,500 and 900 timed queries. The competitor is timed at its own best case, its model constructed before the clock starts. The graph stage's timing excludes building the proposition graph, which is done offline and is not priced here, so the comparison covers inference time only. Set-Encoder was timed on 20-passage pools; the comparisons above give it 30, and its released configuration scores 100.

Related methods not run. The closest prior set selector (Lee et al., 2025b) emits an adaptive-size subset rather than a fixed five, so it has no comparable operating point at this budget. A released utility-based selector (Zhang et al., 2025a) states no licence in either its repository or its model card, so we did not run it. SetCE (Song & Lee, 2026), OptiSet (Jiang et al., 2026) and DPS (Meng et al., 2025) emit a query-dependent variable-size set, and IRCoT (Trivedi et al., 2023) accumulates passages across an iterative retrieval loop rather than reading a fixed pool, so none of the four has a fixed-five operating point to compare against; AdaGReS (Peng et al., 2025) ships an open implementation whose only fixed-count path scores relevance alone, its redundancy term switched

off, so its contribution has no fixed-count operating point. GeoRAG (Zhang et al., 2026a) releases no code for its sub-query generation and coverage pipeline. A sixth published pipeline is absent: HippoRAG 2 (Gutiérrez et al., 2025) on MuSiQue, whose runs in our environment did not reach retrieval.

E How the configuration was chosen

Two proposition graphs carry the search. *Our graph*, which we extract at the 14B tier (Appendix A), is the one every number at the frozen configuration reads unless it names another; PropRAG's numbers on the Wikipedia corpora read PropRAG's own graph. The *reference graph* is a third-party released index (Xu & Li, 2026), the reference against which the extractor tier was chosen.

Under an earlier configuration ($N = 20$, eight coverage slots), the gain on PropRAG was not significant. We then chose the configuration again, this time on the five published pipelines rather than on the four, two of them in-house, that had chosen the earlier one (Appendix H); we call this second choice the *re-freeze*. One part of it is not blind: we widened the grid in light of an earlier value on PropRAG at depth 30, a Chain@5 of 56.8 at $n = 1{,}000$, so PropRAG's held-out half is part of the choice of (γ, β, N), though no part of the choice of the slot rule. The candidates are 130 cells on the reference graph, a union rather than a full factorial grid, built in three stages: stage 1 crossed the pool depth N with the slot rule, stage 2 crossed both with γ and β, and stage 3 varied the other constants of Table 4 (shape, seeding, λ, the near-duplicate block, the path budget and δ) around the cell the first two stages chose. The ranks below are those of the 108 stage-1 and stage-2 cells, evaluated on our graph.

The selection rule is in the released selection file, and it is lexicographic rather than the best mean. It ranks a cell by how many of the five pipelines it lifts significantly on the tuning half, then by its five-pipeline mean on that half; it admits a cell only if that mean is at or above the earlier configuration's own tuning-half mean; and it breaks remaining ties by the smaller N, then by fewer slots. We write a cell as $\mathrm{N}|\mathrm{slots}|\gamma|\beta$, at the frozen shape and seeding S3|C3, with `all` the parameter-free slot rule; the frozen cell, whose constants Appendix A lists, is $\mathrm{N30}|\mathrm{all}|2.5|0.5$. A cell's rank counts only the cells strictly above it (*competition* rank), because cells tie.

Where the rule lands. PropRAG's own tune-half argmax *is* the frozen cell. Each of the other four pipelines has a tune-half argmax that beats the frozen cell, so PropRAG is the pipeline the selection rule was built to protect and the other four pay for it. Under that rule the frozen cell is first of 108 on the half that chose it, at $+7.12$ points and significant on 5 of 5 pipelines. On the *holdout* half the rule returns $\mathrm{N30}|\mathrm{all}|3.0|0.5$ at $+7.16$ instead, and the frozen cell falls to competition rank 22 at $+6.72$, significant on 4 of 5.

Table 13: **The candidate cells, on our graph.** Chunk-exact Chain@5 gain over each pipeline's own Top-5 on the five published pipelines, MuSiQue, for the tune half ($n = 500$), the holdout half ($n = 500$) and all 1,000 questions; bold marks a contrast whose exact interval excludes zero. *Rank* is the competition rank among the 108 stage-1 and stage-2 cells of the union under the selection rule applied to that split (cells strictly above; the rule ranks by the number of pipelines lifted significantly, then by the five-pipeline mean).

Candidate	split	BM25	Hybrid	MonoT5	Dense	PropRAG	mean	rank
frozen: $\mathrm{N30}\|\mathrm{all}\|2.5\|0.5$								
	tune	$\mathbf{+5.8}$	$\mathbf{+8.4}$	$\mathbf{+8.4}$	$\mathbf{+9.2}$	$\mathbf{+3.8}$	$+7.12$	1
	holdout	$\mathbf{+6.2}$	$\mathbf{+8.4}$	$\mathbf{+10.2}$	$\mathbf{+5.8}$	$+3.0$	$+6.72$	22
	all 1,000	$\mathbf{+6.0}$	$\mathbf{+8.4}$	$\mathbf{+9.3}$	$\mathbf{+7.5}$	$\mathbf{+3.4}$	$+6.92$	8
earlier constants at the frozen depth and slot rule: $\mathrm{N30}\|\mathrm{all}\|2.0\|1.0$ (the § marks of Appendix L)								
	tune	$\mathbf{+7.0}$	$\mathbf{+9.0}$	$\mathbf{+10.0}$	$\mathbf{+8.8}$	$+3.0$	$+7.56$	18
	holdout	$\mathbf{+5.8}$	$\mathbf{+8.8}$	$\mathbf{+10.6}$	$\mathbf{+6.2}$	$+0.2$	$+6.32$	45
	all 1,000	$\mathbf{+6.4}$	$\mathbf{+8.9}$	$\mathbf{+10.3}$	$\mathbf{+7.5}$	$+1.6$	$+6.94$	38

The five-pipeline mean across the union. Over the five published pipelines at $n = 1{,}000$ on our graph, the five-pipeline mean gain over each pipeline's own Top-5 is positive at all 108 stage-1 and stage-2 cells of the union, from $+5.42$ to $+7.54$ points. Every mean on either half is positive. The frozen cell gives $+6.92$ and ranks 8th under the selection rule applied to all 1,000. Behind that mean

the pipelines differ: of the 540 pipeline $\times$ cell values the only ones at or below the pipeline's own Top-5 are PropRAG's, 20 of its 108 cells.

F BUDGET, DEPTH AND THE CEILING

The default budget is five and the pool is thirty; every arm below is the frozen configuration, its normalisers included, carried unchanged and read at $n = 1{,}000$, unless its own text says otherwise.

The graph stage and the pipeline's own top-k spend k passages on the same pool, so the budget cancels from the paired difference, but Chain@k moves its own ceiling with k, so the columns of Table 14 at different k are not one quantity.

The graph stage's value lives at small budgets, because the incumbent top-k already converts more of the reachable ceiling as the budget approaches the pool depth. Pooled over the ten MuSiQue pipelines under a question-level cluster bootstrap, the gain over each pipeline's own top-k on the same top-30 pool is significant at every budget measured below 30 and reaches exactly 0.00 only at $k = 30$, where the budget equals the pool (Table 14).

Table 14: **The budget ladder.** Graph stage minus each pipeline's own top-k on the same top-30 pool, chunk-exact Chain@k, pooled over the ten MuSiQue pipelines by a question-level cluster bootstrap; *arms* counts how many of twelve (the ten pipelines plus Graph-B on 2Wiki and on HotpotQA, $n = 1{,}000$ each) are significant on their own; the two *converts* columns are the mean over the ten pipelines of each pipeline's share of its own reachable ceiling (cov@30 restricted to what fits in k) that each arm turns into a complete chain. A dash is a quantity this study did not read.

k	pooled gain [95% CI]	arms sig. / 12	reachable ceiling	top-k converts	graph stage converts
3	$+5.72$ [$+4.77$, $+6.68$]	12	60.50	37.47%	48.07%
5	$+6.10$ [$+5.13$, $+7.08$]	12	64.48	51.98%	62.57%
10	$+4.48$ [$+3.63$, $+5.33$]	11	64.48	71.50%	79.70%
20	$+1.23$ [$+0.53$, $+1.92$]	4	64.48	90.36%	–
30	0.00	–	64.48	–	–

Across the ten MuSiQue pipelines the per-pipeline gain at $k = 3/5/10$ runs from Fusion's $+2.10/+2.30/+0.90$, the last not significant, up to Graph-A's $+9.90/+11.20/+9.10$; Fusion and Graph-A to Graph-D are the in-house pipelines of Appendix M.

Section 4.1 reports the pooled gains at $k = 3$, 5 and 10. The reachable ceiling is lower at $k = 3$ because MuSiQue gold sets run to four passages: at $k = 3$ only 83.4% of its questions are completable at all.

The graph stage can only recover a chain the pipeline retrieved, so its ceiling is the pipeline's own cov@30, and the mean reachable share is 64.48%, the other side of the 35.52% of pipeline–question cells the body reports as unreachable.

F.1 WHERE THE GRAPH STAGE LOSES

The graph stage is applied to every question, so on some questions it replaces a complete chain with an incomplete one. Those breaks are not spread evenly. The break profile below is measured at the frozen configuration.

Breaks concentrate at the Top-5 boundary. Ranks here count from zero, so rank 4 is the last Top-5 slot. Pooled over the ten MuSiQue pipelines at the frozen configuration, the graph stage breaks 19.8% [16.3, 23.7] of the chains whose deepest gold sits at that rank against 4.3% [3.6, 5.1] where it sits earlier, and the rate climbs across the four ranks (Table 15). Over the same questions the stage completes 833 chains the Top-5 misses against 223 broken, for the net of $+6.10$ points Table 14 carries at $k = 5$.

Rank 4 is the worst rank on nine of the ten pipelines. The exception is BM25, whose rank-4 cell is $0/15$ and too thin to order.

Table 15: **Where the graph stage loses, at the frozen configuration.** Chains the Top-5 completes that the graph stage breaks, by the zero-based rank of the deepest gold passage in the Top-5 (MuSiQue, $N = 30$, all-token slots, $\gamma = 2.5$, $\beta = 0.5$, $k = 5$; ten pipelines, all 1,000 questions each), exact Clopper–Pearson intervals. No question's deepest gold sits at rank 0. In the two combined rows the interval treats pipeline–questions as independent, which they are not, so it is narrower than the truth; the per-pipeline range beside it assumes nothing.

Deepest gold's rank	breaks / completions	break rate (%)	per-pipeline range (%)
1	27 / 1,496	1.8 [1.2, 2.6]	0.0–3.6
2	44 / 854	5.2 [3.8, 6.9]	2.1–9.1
3	59 / 645	9.1 [7.0, 11.6]	4.5–12.9
4	93 / 470	**19.8** [16.3, 23.7]	0.0–29.6
1–3 combined	130 / 2,995	4.3 [3.6, 5.1]	
all ranks	223 / 3,465	6.4 [5.6, 7.3]	

G COMPARATORS, AND THE TUNING BUDGET

MMR (Carbonell & Goldstein, 1998), facility location (Lin & Bilmes, 2011) and the determinantal operator (DPP in Figure 7a) are the classical members read here, beside the cohesion operator of ours (MMR with the sign of its second term reversed), each read at its own trade-off constant c. Each classical member is read at its own per-pipeline best, a freedom the graph stage does not have: 7,090 operating points per pipeline over the same top-20 pools at the same budget, 65.6 times the 108 candidates that Appendix E ranks to choose the graph stage's configuration (54.5 times the full union of 130).

Under the earlier configuration. With $N = 20$ and eight coverage slots, the family was swept over ten pipelines; there the strongest member is cohesion, and the earlier configuration's graph stage leads it by 0.60 points on the ten-pipeline holdout mean, [−0.40, +1.62], not significant: it loses significantly on Hybrid (−4.0) and wins significantly on Graph-D (+4.0). The constant is chosen on the tuning half over the family's full grid; the same split over the five-point grid, Table 16's procedure, gives 0.62 [−0.40, +1.66], also not significant. On the five published pipelines alone both rules put cohesion ahead instead. Under all four rules for choosing the comparator's cell (the hundred points the released grid file lists, the 7,090-point sweep, and an oracle over that sweep read on all 1,000 questions or on the holdout alone), the graph stage is significantly above the family on none of the five pipelines built from published components and significantly below on one, Hybrid, at pooled margins −0.60, −0.56, −0.40 and −1.28 points. The last, the oracle read on the holdout, is significant: [−2.40, −0.12].

The diversity operators on a content similarity. No diversity operator on any substrate is significantly above the Top-5 on either of the two strongest pipelines. At depth 30 the largest gain any of the nine content cells reaches is +0.40 on Dense and +0.00 on PropRAG, where three of the arms reaching it have no discordant pair at all. The one cell significant on the pooled reading is up on four of the ten and on none of the other six: BM25 +2.20 [+0.24, +3.51], Hybrid +2.60 [+0.92, +3.30], MonoT5 +3.60 [+1.47, +4.92] and Fusion +1.80 [+0.24, +2.50]. Three of those four are the three weakest pipelines by held-out Top-5 and the fourth is the second strongest by that measure, so the gain is not confined to weak pipelines.

The tuning budget at the frozen depth. How much each side is tuned can decide such a comparison (Armstrong et al., 2009; Dodge et al., 2019), so we read it at three levels of tuning. In absolute chain completion, cohesion's constant alone moves its Chain@5 by 3.8 to 16.2 points per pipeline, against a graph-stage-minus-cohesion margin of at most 3.4 in absolute value at the tuned constant (Hybrid, where cohesion at its tuned constant leads).

A separate comparator, the operator's own per-pipeline choice rather than one frozen constant, gives the same 3.4-point margin on Hybrid at depth 30: cohesion stays ahead there against it too (−3.4 points [−6.24, −0.20] on the holdout, significant). Held to the budget the graph stage is held to, one constant for all five published pipelines ($c = 0.5$, which their pooled tuning halves select), frozen cohesion's five-pipeline held-out mean is +4.88 against the graph stage's +6.72, a paired +1.84 [+0.52, +3.20] for the graph stage.

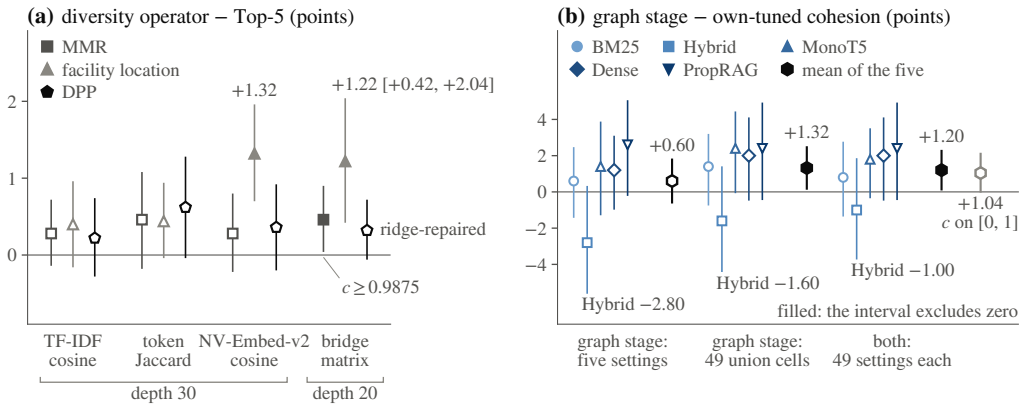

Figure 7: **The comparator family at its own best.** (a) Chain@5 of each diversity operator (MMR, facility location, DPP) minus the Top-5 at the constant c its own tuning half chose, on ten pipelines' held-out halves, pooled by the question-level cluster bootstrap: three content similarities at the frozen depth of 30, and the bridge matrix on 20-passage pools. MMR's bridge-matrix optimum sits at $c \geq 0.9875$ on all ten pipelines, and DPP's bridge-matrix kernel is ridge-repaired, a misspecified control. (b) The graph stage minus own-tuned cohesion, held-out Chain@5 on the five published pipelines at depth 30 ($n = 500$ each): the graph stage held to five points on its connection-weight axis (the analysis file's primary comparison), the graph stage searching the 49 depth-30 cells of the candidate union, and both searching 49 points, where the grey hexagon lays cohesion's 49 constants on the closed range $[0, 1]$ instead. Exact 95% intervals per pipeline, a question-level cluster bootstrap for the hexagon, the mean of the five. Filled marks exclude zero. Every number drawn is in the supplementary notes, Section S9.

A matched-budget comparison. Held to five points on its connection-weight axis ($\gamma \in \{1.5, 2.0, 2.5, 3.0, 3.5\}$ at the frozen β and depth), the graph stage stays ahead of own-tuned cohesion, by $+0.60$ [-0.64, $+1.84$] pooled and up on four of the five pipelines. That interval covers zero, so the pooled five-point margin is not significant. Searching the 49 depth-30 cells of the candidate union instead, the graph stage leads by $+1.32$ [$+0.12$, $+2.52$], and that margin is significant. Given the same 49 points on both sides, cohesion tuned over 49 evenly spaced trade-off constants on the closed range of the five its analysis file names, the graph stage stays ahead by $+1.20$ [$+0.08$, $+2.32$] pooled and on four of the five pipelines; Hybrid again runs the other way, at -1.00 [-3.73, $+1.86$]. The pooled interval clears zero by 0.08 of a point. The analysis file names the five-point comparison the primary one. On it the graph stage is ahead of cohesion on four of the five pipelines in point estimate only; the margin is significant when the graph stage searches the candidate union, and not when cohesion's 49 constants are spread over $[0, 1]$ instead ($+1.04$ [-0.04, $+2.16$], the grey hexagon of Figure 7b). Per pipeline the three values are drawn in Figure 7b (Hybrid's printed) and listed in the supplementary notes, Section S9, and no pipeline is significant either way. Hybrid is at -2.80 in that figure and -3.4 against the separate per-pipeline comparator above, because cohesion is own-tuned in both while the graph stage is held to five points in that figure's left group and frozen for that comparator.

Each operator's point was the argmax of tuning-half gain over the pipeline's own Top-5; three of the fifteen arms tied, broken toward the candidate first in sorted key order. Each point was then read on the held-out half ($n = 500$ per pipeline).

Calibration. Each released component with a published number (all but Qwen3-Reranker-0.6B) was first reproduced against a number its own paper published, at a tolerance of twice the bootstrap standard error of our own per-topic values, 0.034 to 0.064 at nDCG@10 on each component's own published data. A setting here is one dataset read over one first stage. On none of the two plug-ins' six settings is this check sharp enough to catch every installation fault, and the same holds for the three settings on which the rerankers are checked (Appendix D). On Set-Encoder's two checked settings the tolerance is smaller than that component's published lift over its own first stage (0.057, 0.066) but larger than the published gap between its large and base checkpoints (0.001, 0.013). On Provence's two settings the bare first stage falls within the tolerance, so a reproduction that ran no model at all would pass the check. Set-Encoder's other two settings are left out of the check: there our BM25 first stage is at 0.512 and 0.477 against the published 0.480 and 0.494, so on those two of the six settings nothing installed would pass.

H THE GRAPH SIGNAL: TERMS, SUBSTRATE, AND THE DELETION TEST

Everything below is recomputed from the persisted per-question selection arrays through a separate implementation of the gold sets, the split and Chain@5.

Table 16: **The ablation.** Paired differences in chunk-exact Chain@5 with intervals; bold marks a significant contrast. Each block is read on its own population. *Terms*: the tuning halves of the four pipelines that chose the earlier selection (Dense, PropRAG, Graph-B, Graph-D; the full objective's Chain@5, 48.90), except the second title row, which is read over ten pipelines at $n = 1{,}000$. *Substrate*: nine pipelines (the ten of Appendix M minus Fusion, whose pool in this ablation is twenty deep) at $n = 1{,}000$ and the frozen configuration. *Cohesion*: the graph stage minus cohesion on the five published pipelines' holdout, c being the cohesion operator's trade-off constant; the re-tuned row leaves the graph stage $+0.12$ ahead, while over that family's full grid at the earlier depth of 20 the same contrast is -0.56, the other way. Pooled intervals: question-level cluster bootstrap, 20,000 draws.

Block	Arm	Δ	95% CI
Terms	$-$STRUCT (γ=0)	$\mathbf{-5.35}$	$[-7.00, -3.70]$
	$-$COV (β=0)	$\mathbf{-3.50}$	$[-4.70, -2.35]$
	$-$title (λ=0), tuning halves	-0.40	$[-1.35, +0.50]$
	$-$title, ten pipelines	$\mathbf{-0.45}$	$[-0.78, -0.14]$
	$-$relevance (STRUCT + COV only)	$\mathbf{-15.70}$	$[-18.35, -13.10]$
Substrate	bridge matrix $\rightarrow$ NV-Embed-v2 cosine	$\mathbf{-5.10}$	$[-6.00, -4.20]$
Cohesion	graph stage minus cohesion, frozen c=0.9	$\mathbf{+4.36}$	$[+3.12, +5.60]$
	graph stage minus cohesion, frozen c=0.7	$\mathbf{+2.56}$	$[+1.60, +3.52]$
	graph stage minus cohesion, frozen c=0.5	$\mathbf{+1.84}$	$[+0.52, +3.20]$
	graph stage minus cohesion re-tuned per pipeline, five-point grid	$+0.12$	$[-1.12, +1.40]$

The ablation is read at the frozen configuration on the tuning halves of the four pipelines that chose the earlier configuration ($N = 20$, eight coverage slots): Dense, PropRAG and the in-house Graph-B and Graph-D of Appendix M, at $n = 500$ per half.

H.1 THE SUBSTRATE: REPLACING THE BRIDGE MATRIX

Holding every pipeline, pool, term, constant and construction fixed, $M^{\max}$ is replaced by the passage-cosine substrate E, built with NV-Embed-v2 over the same top-30 pools (Lee et al., 2025a); every reading below is the one the analysis file names. The substitute is not neutral: the dense retriever *is* NV-Embed-v2, Hybrid fuses it with BM25 and MonoT5 reranks an NV-Embed-v2 first stage, so on three of the nine rows, and three of the five published pipelines, E is the pipeline's own geometry.

The frozen configuration loses 5.10 points under the substitution, significantly down on nine pipelines of nine (Table 16); pooled over the five published pipelines alone the difference is -6.22 $[-7.18, -5.28]$.

A support-matched matrix D, fixed by the same analysis file, reproduces M's density on every pipeline; the deletion test of Appendix H.2 uses it as a second substrate.

The two substrates are not interchangeable, and the numbers this paper reports are measured on the bridge relation; the loss under E does not show that the bridge relation is the right substrate.

H.2 THE DEPENDENCY, TESTED BY DELETION

The graph stage's five passages are not five independent choices. On MuSiQue questions where the graph stage selects both hops of a chain, and where the winning fill does not change under either deletion, the antecedent is removed from the pool, with the pipeline's next passage, at rank 30, taking its place and every normaliser recomputed. The dependent hop departs as well on 9.58% of those questions, against 0.43% when a rank-matched non-gold passage is removed instead: an excess of $+9.15$ points $[+7.26, +11.12]$ over $n = 1{,}399$ questions, with an interval excluding zero on all five published pipelines (Table 17). The excess stands against a bound: any selector explained by a fixed per-passage score with a pool-independent tie-break loses the deleted passage and nothing else, at

most one departure (q-responsiveness, Chambers & Yenmez, 2018; Appendix C). Over the $6{,}093$ distinct deletions run here the pipeline's own Top-5 never lost two. The response rises with the bridge weight the two passages share and the rise is carried by the top quartile: 3.4%, 2.6%, 7.8% and 24.6% across quartiles of the pooled primary population, the two lowest indistinguishable; a value at a cut point is placed in the lower quartile. Switching the connection term off removes 96% of the excess, against 4% for the coverage term.

The excess is present in one stratum of the seed split and covers zero in the other: $+9.88$ [$+7.85$, $+12.02$] where the deleted antecedent belongs to the graph stage's seed ($n = 1{,}296$) and $+0.00$ [-2.73, $+2.97$] where it does not ($n = 103$). That split cannot separate the seed from the bridge relation, since seed membership is defined by the bridge weights.

The control is matched on pool rank and not on selection, and whether a passage is selected is what decides whether it can depart. The pooled $+9.15$ is therefore a mixture: $+7.52$ [$+5.52$, $+9.70$] on the $1{,}103$ questions where the control is itself selected and about twice that where it is not; the unrestricted excess is the primary one.

Each arm below is read on its own recomputed population (Table 17). With either term switched off the excess still excludes zero under the primary test, so neither term carries it alone, and the 96% and 4% above are effect sizes reported beside that rule rather than in place of it: the three populations differ, so the two shares are not a decomposition of one quantity. Run instead with the bridge matrix replaced by the passage-cosine substrate E and by the support-matched D, everything else fixed, the excess is $+3.53$ and $+4.05$ against the bridge matrix's $+9.15$; it excludes zero on both substrates and on four of the five published pipelines under E, so part of the effect belongs to set selection as such rather than to the bridge relation, while each substrate's excess is below half the bridge matrix's with each interval disjoint from it, so its size is substrate-dependent.

These arms measure the graph stage alone; the reading stage, which can restore a passage the graph stage dropped, was not measured. The primary arm, the two term arms and the two substrate arms, their per-question arrays and the rules that read them ship with the artifacts.

Table 17: **Deleting the antecedent of a selected chain.** Co-departure rate of the dependent hop (%) when the antecedent is deleted (gold) and when a rank-matched non-gold passage is deleted (control), with Clopper–Pearson 95% intervals, and the paired excess with a question-level bootstrap interval; MuSiQue, frozen configuration, questions where the graph stage selects both hops and the winning fill does not change under either deletion. The lower block reports the pooled excess under each arm and stratum, and under one restriction beside the primary (control passage itself selected). The two term arms and the two substrate arms change the selection on the unperturbed pool, so each is read on its own recomputed population.

Pipeline	n	gold	control	excess
BM25	132	10.61 [5.92, 17.15]	0.76 [0.02, 4.15]	$+9.85$ [$+4.55$, $+15.15$]
Hybrid	237	13.08 [9.06, 18.05]	0.00 [0.00, 1.54]	$+13.08$ [$+8.86$, $+17.72$]
MonoT5	257	15.18 [11.02, 20.16]	0.39 [0.01, 2.15]	$+14.79$ [$+10.51$, $+19.46$]
Dense	350	8.57 [5.86, 12.01]	0.00 [0.00, 1.05]	$+8.57$ [$+5.71$, $+11.43$]
PropRAG	423	4.73 [2.91, 7.21]	0.95 [0.26, 2.40]	$+3.78$ [$+1.65$, $+6.15$]
Pooled	1,399	9.58 [8.09, 11.24]	0.43 [0.16, 0.93]	$+9.15$ [$+7.26$, $+11.12$]

Arm or stratum, pooled	n	excess
$\beta = 0$, coverage term off	1,366	$+8.78$ [$+6.93$, $+10.73$]
$\gamma = 0$, connection term off	1,606	$+0.37$ [$+0.12$, $+0.70$]
antecedent in the seed	1,296	$+9.88$ [$+7.85$, $+12.02$]
antecedent not in the seed	103	$+0.00$ [-2.73, $+2.97$]
control passage itself selected	1,103	$+7.52$ [$+5.52$, $+9.70$]
cosine substrate E in place of the bridge matrix	1,389	$+3.53$ [$+2.31$, $+4.89$]
support-matched substrate D	1,135	$+4.05$ [$+2.48$, $+5.80$]

I EVERY CORPUS, CELL BY CELL

A cell is one pipeline on one corpus, every arm reading the same 30-wide pool: *Top-5* is that pool's own first five in the pipeline's own rank order, *graph stage* its five under the frozen configuration, and *cov@30*, the share of questions whose gold chain lies in the 30-wide pool, the cell's ceiling.

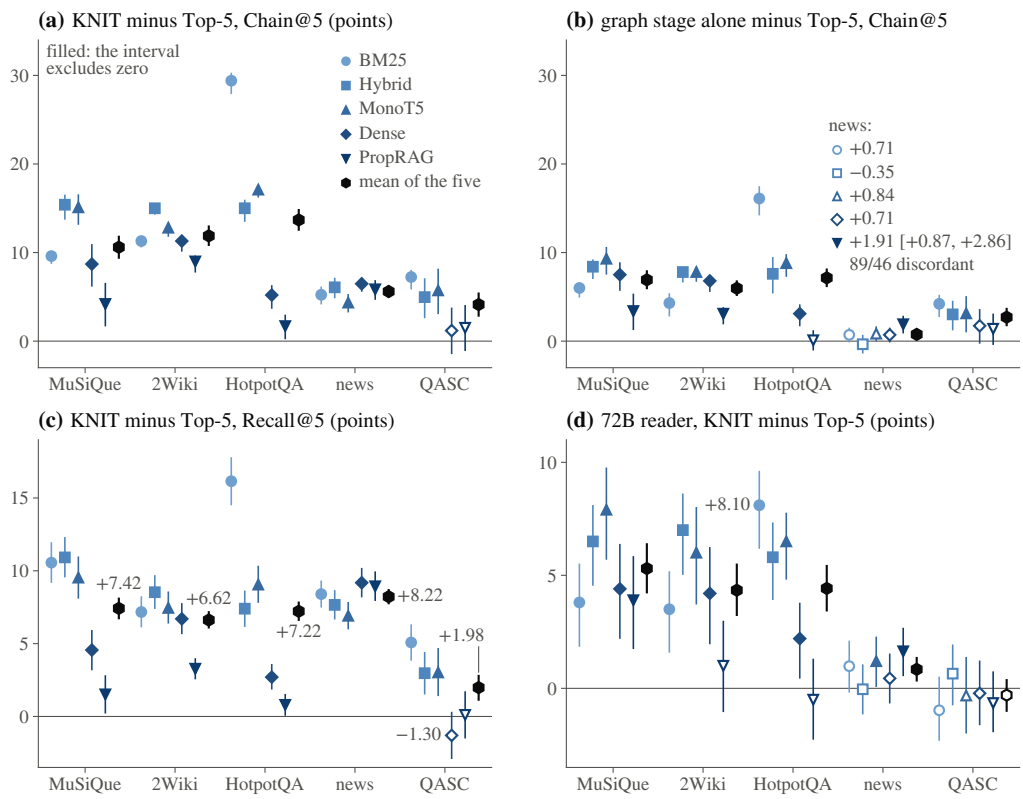

Figure 8: **Every cell at a glance.** Each pipeline's gain over its own Top-5 on the 25 cells, with its 95% interval; pipelines are in the order of their Top-5 on MuSiQue, the hexagon is the five-pipeline mean, and a filled mark is an interval that excludes zero. (a) KNIT, Chain@5. (b) The graph stage alone, Chain@5, on the scale of (a); the news cells are printed, PropRAG's with its interval and discordant split. On HotpotQA PropRAG's Top-5 already completes 94.9 of 100 chains at a cov@30 of 99.7, so little headroom remains there. (c) KNIT, Recall@5: the share of each question's gold passages among the five, averaged over questions. (d) The 72B reader: exact match on the Wikipedia corpora, each benchmark's own accuracy rule on news and QASC. Single Chain@5 and reader cells use the exact interval, Recall@5 cells and every mean the question-level bootstrap; MuSiQue pools both halves. (a) and the means of (d) are Table 2's, (a)'s intervals with the released artifacts; (b), (c) and the cells of (d) are printed in the supplementary notes, Section S9.

Comparability across the corpora. Across the corpora of Figure 8, a gain is a within-corpus paired contrast and is comparable; the Top-5 and graph-stage absolute Chain@5 are not, because gold is taken from `supporting_facts` titles on 2WikiMultihopQA and HotpotQA and from paragraph support indices on MuSiQue.

I.1 MuSiQue by split, the corpus that chose the configuration

The halves are read in Table 13, and are printed with every interval in the supplementary notes, Section S9. The holdout half is the out-of-sample value, which the note under Table 2 prints; the value on all 1,000 questions is half in sample.

PropRAG is the cell the re-freeze was searched for and the only one whose graph-stage result differs between the two splits. At the holdout's own discordant ratio (34 against 19) the effect would need about sixty discordant pairs to be significant and has fifty-three; one discordant question changing sides would decide it. The holdout does not refute that cell; it fails to establish it. How that cell entered the choice of configuration is in Appendix E.

PropRAG on the transfer corpora. PropRAG's index ships pre-built for MuSiQue only, so its 2WikiMultihopQA and HotpotQA graphs were built by running its own released pipeline on those corpora, and its cells in Figure 8 read its own graph and ranking at a pool depth of 200 sliced to 30.

I.2 Recall@5 beside Chain@5

KNIT's Recall@5 gain over the Top-5 excludes zero on 23 of the 25 cells, the same 23 on which its Chain@5 gain does, and pooled over the five pipelines it is $+7.42$ on MuSiQue, $+6.62$ on 2WikiMultihopQA, $+7.22$ on HotpotQA, $+8.22$ on news and $+1.98$ on QASC, every interval excluding zero.

J The reading stage

The reading stage is one call of Qwen2.5-32B-Instruct, the judge. The graph stage is run at a budget of ten, the frozen configuration otherwise unchanged. Its ten passages are shown in the pipeline's rank order, numbered 1 to 10, each as its title and full text, and the judge is asked for the identifiers of the five that jointly answer the question. The instruction is the published zero-shot listwise instruction (Sun et al., 2023) with its ask changed from ranking every passage to choosing five, rendered through the checkpoint's own chat template, under greedy decoding with at most 64 new tokens. The reply is read left to right for integers between 1 and 10, duplicates dropped. If it yields five, those five passages are returned; otherwise the first five of the ten as shown are, which happens on 0.0 to 0.9% of questions across the twenty pairs of the four corpora other than QASC and on 0.0 to 2.3% on QASC. The prompts average 1,718 to 1,896 tokens on MuSiQue, at most 2,630 on the other Wikipedia corpora and on news, and about 470 to 480 on QASC's one-sentence facts, under a budget of 8,192 that no prompt reaches. On MuSiQue the call takes 3.82 to 3.94 seconds a question across the five pipelines, generation only, at batch size one on two A40 cards. The prompt is released verbatim with the artifacts.

How the form was chosen. An earlier form of the reading stage (Section J.1) had the judge rate up to five whole candidate sets. On MuSiQue's tuning half the form above completes 2.84 more chains per hundred questions than the earlier one, five pipelines pooled [$+1.40$, $+4.32$], and it was adopted on that contrast, as the configuration of the graph stage was chosen on the same half. The held-out half and the four other corpora took no part. The earlier form was not run on QASC, which was added after the choice.

Where the deepest gold sits. The gold each pipeline buries deepest is a *dependent* hop, whose sub-question refers to a previous answer, on most questions of every pipeline: a rate of 0.64 to 0.87 against their share of all gold passages, 0.573. The burial rests on depth and relevance: on the four pointwise pipelines a passage with the same features is rarely in the pool beside the buried hop (supplementary notes, Section S5).

What the graph stage brings in from below the cut. At the frozen configuration and depth 30 the gold the graph stage pulls in from below a pipeline's own Top-5 is enriched in those dependent hops: 68.4% of it against a base rate of 48.9% among all gold inside the pool (Table 18); it is computed from gold labels and is a diagnostic, not a rule. The brought-in interval and the base interval are disjoint on four of the five pipelines; BM25's brought-in interval overlaps the base interval but not the interval of the gold it did not bring in. The pooled row is descriptive: its units are five passes over the same $1{,}000$ questions and three of the five pipelines share a first stage, so the per-pipeline rows, not the pooled row, carry the evidence. The pooled set the analysis file names (the questions on which all five pipelines bring in at least one gold passage) is empty, each pipeline bringing one in on 119 to 168 of $1{,}000$ questions and those sets barely intersecting; the per-question excess in its place, $+2.73$ [$+2.09$, $+3.37$] dependent passages per 100 questions, is a substitute chosen once that emptiness was found, a departure from the analysis file.

J.1 The earlier form: a judge over candidate sets

The earlier form read whole candidate sets rather than single passages. The graph stage's own search produced up to five of them, S_1 being the set the stage returns, and one judge call rated each on a scale of 0–100; a fixed rule combined that rating with the graph stage's own objective, returned one set, and the reader answered from it.

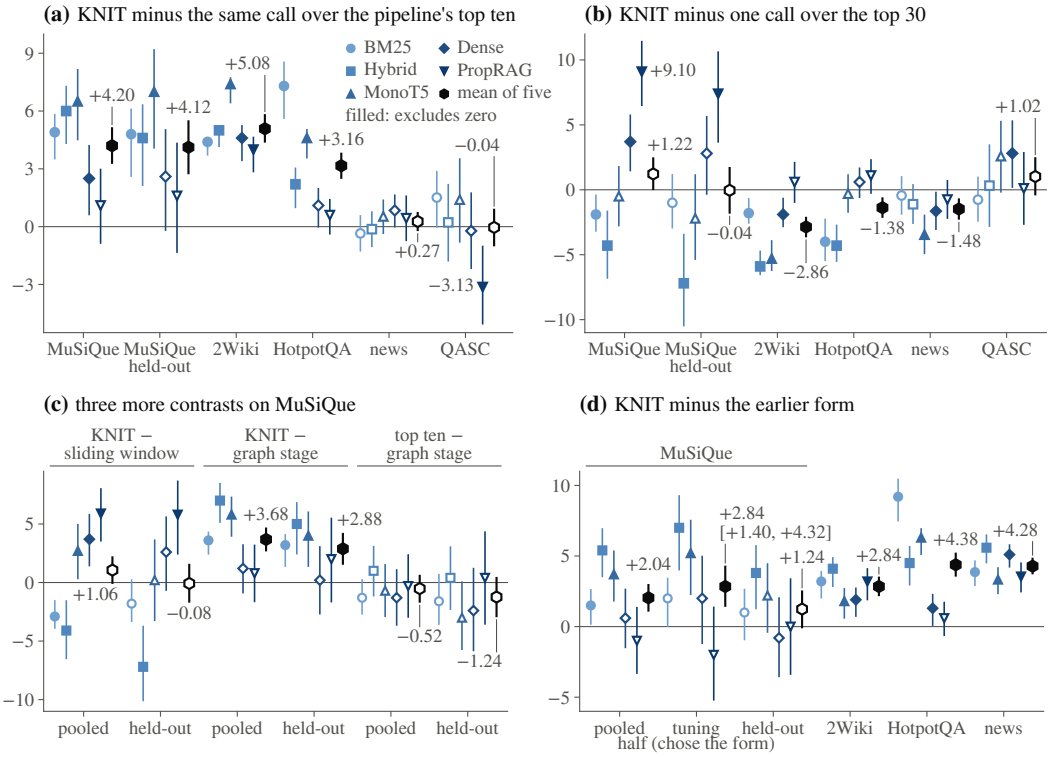

Figure 9: **The reading stage against its controls.** Chain@5 differences, paired per question, with their 95% intervals, exact for a pipeline and a question-level bootstrap for the five-pipeline mean (the hexagon); a filled mark is an interval that excludes zero. (a) KNIT minus the same call handed the pipeline's own top ten in place of the graph stage's ten. (b) KNIT minus the same 32B model handed the top 30 in one call, on MuSiQue (both halves pooled, and its held-out half) and on the four corpora of Table 20: 2WikiMultihopQA, HotpotQA, news and QASC. (c) On MuSiQue: KNIT minus the same model as the published sliding-window reranker over the top 30, KNIT minus the graph stage's own five, and the top-ten call minus those five. (d) KNIT minus the earlier form of the reading stage; the form was chosen on MuSiQue's tuning half, whose mean is the contrast that chose it. MuSiQue is read with both halves pooled and on its held-out half. Printed values are the means and two cells the body cites; every number drawn is in the supplementary notes, Section S9, or in Table 20.

Table 18: **What the graph stage brings in from below the cut.** Among the gold passages the graph stage brings into its five from below the pipeline's own Top-5, inside the depth-30 pool, the share that are dependent hops (a hop whose own sub-question names a previous answer through a back-reference) against the share among all gold of the same pool and among the gold the stage did not bring in. MuSiQue, frozen configuration, 1,000 questions per pipeline; the unit is one gold passage of one question of one pipeline and the intervals are exact Clopper–Pearson. The base is restricted to gold inside the pool, because a passage outside it can be brought in by nothing.

Pipeline	brought in	dependent, brought in	dependent, all gold in pool	dependent, the rest	lift
BM25	173	47.4% [39.8, 55.1]	37.8% [35.4, 40.2]	36.7% [34.2, 39.2]	1.25
Hybrid	180	71.7% [64.5, 78.1]	48.8% [46.6, 50.9]	46.7% [44.4, 48.9]	1.47
MonoT5	166	71.1% [63.5, 77.8]	48.4% [46.2, 50.5]	46.5% [44.2, 48.7]	1.47
Dense	153	83.7% [76.8, 89.1]	50.7% [48.6, 52.8]	48.3% [46.2, 50.5]	1.65
PropRAG	128	70.3% [61.6, 78.1]	54.9% [52.9, 56.9]	54.0% [52.0, 56.1]	1.28
Pooled, descriptive	800	68.4% [65.0, 71.6]	48.9% [47.9, 49.8]	47.3% [46.3, 48.3]	1.40

J.2 The same judge as a zero-shot listwise reranker

KNIT's judge reads the graph stage's ten. The natural alternative hands the same model the pool itself. We run the judge, Qwen2.5-32B-Instruct, as a zero-shot listwise reranker (Sun et al., 2023), its instruction rendered through the checkpoint's own chat template, under greedy decoding, on the same top-30 pools of the five published pipelines on MuSiQue. Two forms are read. *Sliding window* is the published policy: a window of twenty at a stride of ten, taken from the back of the list, two calls per question, its first five passages kept. *One call* shows all thirty passages at once and asks for the five

that jointly answer the question, the ask KNIT's judge is given over ten. No prompt is truncated: the longest one-call prompt is 8,990 tokens against a budget of 12,288, and the longest sliding-window prompt 6,767 against 8,192.

Table 19 gives the levels and Figure 9b and c the paired contrasts. Averaged over the five pipelines neither form differs significantly from KNIT. The rule for reading this comparison names the held-out half as its primary comparison. On the held-out half, KNIT minus the one-call reranker is -1.00 $[-2.98, +1.21]$ on BM25, -7.20 $[-10.52, -3.39]$ on Hybrid, -2.20 $[-5.40, +1.20]$ on MonoT5, $+2.80$ $[-0.38, +5.68]$ on Dense and $+7.40$ $[+3.63, +10.66]$ on PropRAG, -0.04 $[-1.84, +1.76]$ over the five, and the reranker minus PropRAG's own Top-5 is -2.40 $[-6.57, +1.90]$, against -4.90 $[-7.83, -1.84]$ with both halves pooled. The sliding-window form leaves 4.6 to 7.6% of its replies unparsed, above 5% on three pipelines; an unparsed reply leaves its window in the pipeline's order.

Table 19: **KNIT beside a same-model listwise reranker.** Chain@5 on MuSiQue, $n = 1{,}000$ (both halves pooled). KNIT's judge keeps five of the graph stage's ten; the other three uses of the same Qwen2.5-32B-Instruct call read the pipeline's top 30 in one call, the published sliding-window form over the same 30 (two calls), and the pipeline's top 10. The last rows give prompt tokens and calls per question and the share of replies that could not be parsed, as ranges over the five pipelines; a dash marks a quantity that is undefined or was not read here.

				the same 32B model, reading		
Pipeline	Top-5	graph stage	KNIT	30, one call	30, sliding	top 10
BM25	10.3	16.3	19.9	21.8	22.8	15.0
Hybrid	20.7	29.1	36.1	40.4	40.2	30.1
MonoT5	24.9	34.2	40.0	40.5	37.3	33.5
Dense	38.2	45.7	46.9	43.2	43.2	44.4
PropRAG	53.9	57.3	58.1	49.0	52.2	57.0
mean	29.6	36.5	40.2	39.0	39.1	36.0
prompt tokens / question	0	0	1,718–1,896	4,625–5,213	6,325–7,076	1,643–1,841
calls / question	0	0	1.0	1.0	2.0	1.0
unparsed replies	–	–	0.1%–0.9%	0.0%–0.3%	4.6%–7.6%	0.1%–0.7%

The shortlist. What the judge is handed matters as much as the call. Shown the pipeline's own top ten in place of the graph stage's ten, the same call, with the same instruction, completes 4.20 fewer chains per hundred questions (4.12 on the held-out half), significantly on both populations, and it does not pass the graph stage's own five, which call no model. The graph stage's ten differ from the pipeline's top ten on 88.5 to 96.0% of questions, 91.6% over the five pipelines, with 1.75 to 2.85 of the ten drawn from below rank ten on average. They hold the complete chain more often than the top ten on every pipeline, and the call converts that. Against the graph stage's own five, the call completes 351 chains they miss and breaks 167, over the five pipelines, and on Dense and PropRAG the two nearly cancel (Figure 9c draws the five differences). The prompts of the two are the same size, 1,643 to 1,896 tokens. The same control was run on the four other corpora (Figure 9a). The graph stage's ten are a better shortlist than the pipeline's own on the Wikipedia corpora and not on the two outside it.

At the reader. Of the reranker's two forms, the sliding window completes more chains on the tuning half, and its selections were read by the 72B reader on the two strongest pipelines under the reader settings of Appendix K. KNIT minus the reranker is $+1.90$ $[-0.11, +3.78]$ on PropRAG and -0.90 $[-2.80, +1.06]$ on Dense, so neither difference excludes zero: the gap in chain completion on these pipelines does not carry to exact match.

The reranker on the corpora held out of the choice. The one-call form was run, unchanged, on QASC, on news, on 2WikiMultihopQA and on HotpotQA, with the rule for reading each outcome written before any cell was read (Table 20). On news, reading thirty passages buys the reranker 5.7 to 8.1 points over every pipeline's Top-5. Against KNIT it is 1.48 points ahead pooled, $[-2.30, -0.66]$ for KNIT minus the reranker, significantly on MonoT5 and Dense and not significantly different from it on the other three. On 2WikiMultihopQA the 30-passage reranker is above KNIT on four of the five pipelines, not significantly different from it on PropRAG. On this corpus the gap comes from depth: the chains held entirely within the thirty exceed those held within the graph stage's ten by 2.1 to 6.1 points, and each call completes nearly all of what it is shown. On HotpotQA the

30-passage reranker is above KNIT on the two weakest pipelines, and not significantly different from it on MonoT5, Dense and PropRAG. As on 2WikiMultihopQA the gap is depth where there is one: the chains held within the thirty exceed those within the graph stage's ten by 5.7 and 6.2 points on BM25 and Hybrid, and by 1.2 to 3.3 on the other three, where the two calls do not differ significantly.

Table 20: **The same judge as a one-call listwise reranker, on the corpora held out of the choice.** The reading stage's 32B model is handed each pipeline's top 30 in one call and asked for the best five, with the prompt, decoding and fallback rule of the MuSiQue cells of Appendix J.2. Chain@5 of the pipeline's own Top-5, of the reranker and of KNIT; then, paired per question with the exact interval at $\alpha = 0.05$ (bold: excludes zero), KNIT minus the reranker; last, prompt tokens a question, reranker and KNIT.

Pipeline	Top-5	reranker	KNIT	KNIT − reranker	tokens
QASC, science, $n = 926$					
BM25	13.71	21.71	20.95	-0.76 [-2.46, $+1.02$]	1,112 / 471
Hybrid	23.65	28.29	28.62	$+0.32$ [-2.86, $+3.50$]	1,112 / 472
MonoT5	27.11	30.24	32.83	$+2.59$ [-0.22, $+5.31$]	1,120 / 479
Dense	31.75	30.13	32.94	$\mathbf{+2.81}$ [$+0.15$, $+5.35$]	1,118 / 475
PropRAG	27.97	29.37	29.48	$+0.11$ [-2.70, $+2.91$]	1,137 / 481
mean, five				$+1.02$ [-0.43, $+2.51$]	
2WikiMultihopQA, $n = 1{,}000$					
BM25	32.80	45.90	44.10	$\mathbf{-1.80}$ [-2.61, -0.64]	4,574 / 1,832
Hybrid	41.40	62.30	56.40	$\mathbf{-5.90}$ [-6.57, -4.68]	5,026 / 1,936
MonoT5	44.20	62.30	57.00	$\mathbf{-5.30}$ [-6.25, -3.88]	6,093 / 2,421
Dense	51.10	64.30	62.40	$\mathbf{-1.90}$ [-2.88, -0.61]	5,541 / 2,302
PropRAG	83.10	91.50	92.10	$+0.60$ [-1.01, $+2.15$]	7,571 / 2,630
mean, five				$\mathbf{-2.86}$ [-3.66, -2.08]	
MultiHop-RAG, news, $n = 2{,}255$					
BM25	11.84	17.52	17.07	-0.44 [-1.92, $+1.05$]	5,274 / 1,933
Hybrid	13.57	20.75	19.65	-1.11 [-2.63, $+0.44$]	5,236 / 1,925
MonoT5	11.53	19.33	15.88	$\mathbf{-3.46}$ [-4.95, -1.91]	5,191 / 1,910
Dense	10.07	18.18	16.54	$\mathbf{-1.64}$ [-3.10, -0.15]	5,195 / 1,906
PropRAG	11.62	18.23	17.47	-0.75 [-2.24, $+0.75$]	5,302 / 1,937
mean, five				$\mathbf{-1.48}$ [-2.30, -0.66]	
HotpotQA, $n = 1{,}000$					
BM25	49.60	83.00	79.00	$\mathbf{-4.00}$ [-5.48, -2.22]	4,989 / 1,732
Hybrid	73.90	93.20	88.90	$\mathbf{-4.30}$ [-5.55, -2.68]	5,008 / 1,728
MonoT5	74.50	91.90	91.60	-0.30 [-1.77, $+1.20$]	5,194 / 1,807
Dense	89.40	94.00	94.60	$+0.60$ [-0.63, $+1.72$]	5,033 / 1,776
PropRAG	94.90	95.50	96.60	$+1.10$ [-0.31, $+2.36$]	5,381 / 1,811
mean, five				$\mathbf{-1.38}$ [-2.16, -0.58]	

K Downstream, at the frozen configuration

The reader results behind Table 21 are the frozen configuration measured at two reader sizes, 72B as the primary and 7B for the direction of the reader-size trend, on every MuSiQue pipeline (Figure 10); the 72B reader accuracy row of Table 2 uses the same 72B reader on every corpus.

K.1 KNIT at the reader

Each pipeline's Top-5, graph-stage and KNIT selections were read by the same reader under the same prompt, answer budget and greedy decoding, on one card class. Figure 10 shows MuSiQue at the 72B and 7B readers, two readers from other model families, and KNIT against the earlier form of its reading stage on 2WikiMultihopQA, HotpotQA and news at 72B; Figure 8d shows KNIT against the Top-5 at the 72B reader on every corpus.

Over the five MuSiQue pipelines the two-stage gain rises by 0.87 exact-match points per natural-log unit of reader parameters between 7B and 72B [0.33, 1.41], the graph stage's own gain by 0.70 [0.29, 1.12], a slope that does not depend on the reading stage's form, and the reading stage's increment by 0.16 [-0.34, $+0.67$], an interval that covers zero. Two reader sizes give a direction, not a trend.

Table 21: **Downstream contrasts.** PropRAG on MuSiQue unless a row says otherwise; $n = 1{,}000$ with MuSiQue's two halves pooled, paired per question, reader 72B and judge 32B unless stated. n_{10} counts the questions only the first arm answers correctly and n_{01} those only the second does; the interval is exact on that split, and bold marks one excluding zero; a p below 10^{-3} is printed as a bound. RS = reading stage, GS = graph stage, KNIT = GS + RS, EM = exact match; token F1 agrees with exact match on every row it was computed for.

Contrast, metric	Δ [95% CI]	n_{10}/n_{01}, p
Primary		
KNIT (GS + RS, 32B judge) vs. Top-5, EM	$\mathbf{+3.90}$ $[+1.74, +5.85]$	77/38, $< 10^{-3}$
KNIT vs. GS, EM	$\mathbf{+2.30}$ $[+0.10, +4.38]$	69/46, 0.040
GS vs. Top-5, EM	$+1.60$ $[-0.18, +3.25]$	45/29, 0.081
Secondary		
KNIT vs. Top-5, EM, Dense	$\mathbf{+4.40}$ $[+2.19, +6.39]$	83/39, $< 10^{-4}$
GS vs. Top-5, EM, Dense	$\mathbf{+3.80}$ $[+1.96, +5.37]$	62/24, $< 10^{-4}$
KNIT vs. GS, EM, Dense	$+0.60$ $[-1.37, +2.53]$	49/43, 0.602
Provence (rerank head) vs. Top-5, EM	$\mathbf{-7.90}$ $[-10.12, -5.40]$	46/125, $< 10^{-4}$
Set-Encoder vs. Top-5, EM, median draw	$\mathbf{-10.70}$ $[-12.63, -8.40]$	33/140, $< 10^{-4}$

At the 72B reader, KNIT's five passages are 3.60 exact-match points above PropRAG's own top ten $[+1.38, +5.64]$, while the pipeline's ten are 0.30 above its five $[-1.77, +2.35]$.

Every reader cell of the frozen configuration ran on one card class (NVIDIA A40, four cards per allocation, fp16 weights sharded across them), because a change of card class was measured here to move 8 of 1,000 raw generations. Every reader value here uses the selector's own passage order. Passage order is a nuisance factor a set selection cannot fix, and it was bounded only under the earlier configuration: shuffling the order in which the passages reach the reader takes the primary contrast of the earlier configuration's reader study (the two-stage form over the Top-5 at the 72B reader; released in the supplementary material) to $+1.70$ $[-0.25, +3.53]$, not significant.

The main reader is Qwen2.5-72B-Instruct and the judge is Qwen2.5-32B-Instruct, one model family, so the downstream reading could be measuring that family's tokenizer and alignment.

Two open-weight readers from other families, Mistral-Small-24B-Instruct (Mistral AI, 2025) and phi-4 (Abdin et al., 2024), answer from the same stored selections under the same system message, greedy decoding and SQuAD exact-match scorer. Each message is rendered through the checkpoint's own chat template, so the rendered prompt differs between readers by construction; every contrast is KNIT's selection minus the pipeline's own Top-5 within one reader, and no contrast between readers is computed. These are checks beside the main reader: they replace no number and enter no pooled statistic of the body. The judge was not varied: it is the 32B model of the main reader's family throughout.

Chain completion and the reader's answer. Table 22 joins, question by question, the chain completion of the Top-5 and of KNIT with the reader's correctness under each. On the Wikipedia corpora the reader answers 54 to 66% of the questions whose chain the Top-5 completes and 14 to 33% of the rest (2WikiMultihopQA's PropRAG, whose Top-5 completes 83% of chains, is the exception, at 59 against 49), so on 14 of the 15 Wikipedia pairs a completed chain goes with 26 to 44 more exact-match points. Across the Wikipedia corpora the questions whose chain KNIT completes or breaks carry 63 to 93% of the exact-match gain on thirteen of the fifteen cells; on PropRAG's two cells outside MuSiQue the unchanged-chain questions take back most or all of what the completed ones add. On news the order reverses. The reader answers 77 to 79% of the questions whose chain is incomplete and 70 to 73% of those the Top-5 completes. The questions whose chain KNIT completes are answered at about the same rate before and after, and almost the whole of the small news gain comes from questions whose chain status did not change. On news, whether the annotated chunks are read together is not what decides the reader's answer.

K.2 The analysis file

The downstream grid is named in one analysis file. Its sha256 is `1d5b5ec4...44e02`, and the reader scripts take their thresholds from it; the manifest of arms, readers and grid is released with the artifacts.

The file fixes two pipelines, and on them it fixes two further things. The first is the primary contrast: KNIT (both stages) against the Top-5 at the 72B reader in exact match, paired McNemar at $n = 1{,}000$,

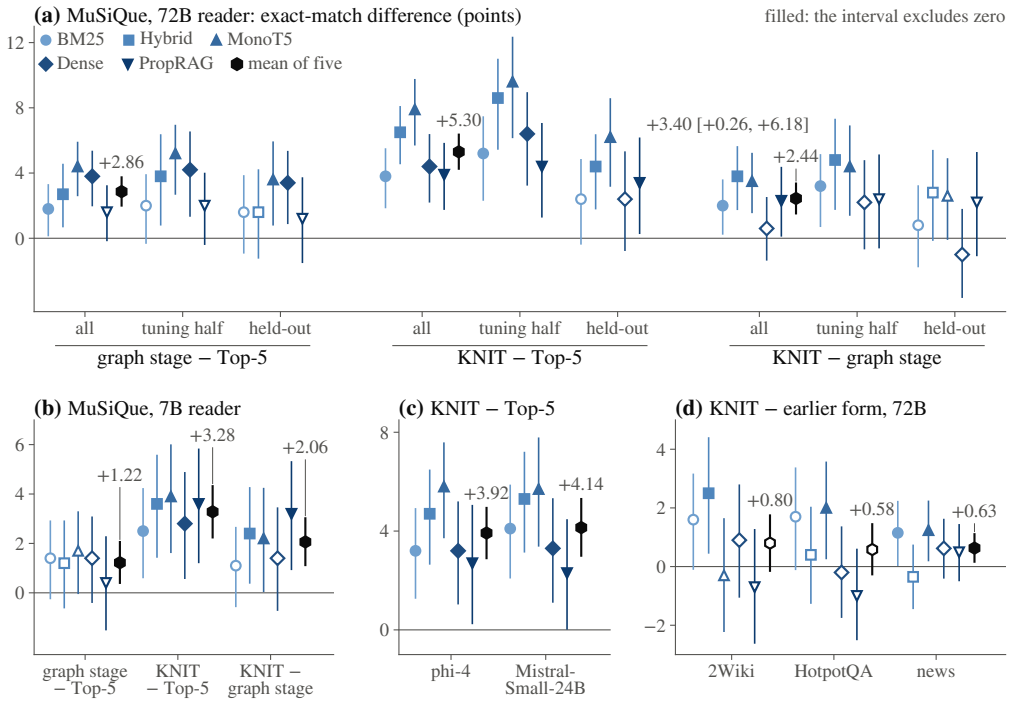

Figure 10: **KNIT at the reader.** Exact-match differences in points, paired per question (the benchmark's token-overlap accuracy on news), with the exact 95% interval; filled marks exclude zero, and the hexagon is the five-pipeline mean with its question-level bootstrap interval. (a) MuSiQue at the 72B reader: the graph stage over the Top-5, KNIT over the Top-5 and KNIT over the graph stage, on all 1,000 questions, on the tuning half, on which the configuration was chosen, and on the held-out half. (b) The same three contrasts at the 7B reader, on all 1,000. (c) KNIT over the Top-5 on MuSiQue under phi-4 and Mistral-Small-24B. (d) KNIT over the earlier form of its reading stage at the 72B reader; KNIT over the Top-5 on these corpora is Figure 8d. Every number drawn and the levels are in the supplementary notes, Section S9.

which the frozen configuration puts at $+3.90$ $[+1.74, +5.85]$ on PropRAG (Table 21). The second is how each outcome is to be interpreted.

The same file fixes a reader-size test in advance: the per-tier gain of the reading stage over the graph stage, its slope across $\log$ reader size, and an interaction term. The frozen configuration's slope above covers zero, so the data do not show that the reading stage's gain depends on the reader's size. The interaction term, which would test whether the reading stage needs an LLM at least as capable as the reader, is not evaluated at the frozen configuration.

The non-Qwen reader the analysis file names has licence-gated weights and was not run, so the file's check of the reader-size test in a second model family has no cell, and every series of reader sizes it governs uses one model family; two ungated readers from other families are shown in Figure 10c.

K.3 The released plug-ins at the reader

The two released plug-ins were read at the 72B reader on PropRAG in the top-k truncation's position, their depth-30 selections (Appendix D; Set-Encoder at its median draw) against the pipeline's own Top-5 on the same 1,000 questions under the reader conditions of Section K.1; only Provence's reranking head is read, since its sentence-level pruner cannot move a passage-id metric.

K.4 A family-wise reading of the printed exact-match contrasts

The family is Table 21's eight MuSiQue contrasts. Under Holm's family-wise bound at 0.05 five of the eight survive; the other three are PropRAG's reading-stage step ($p = 0.040$, significant on its own test but not at the family-wise bound of $0.05/3$) and two contrasts not significant on their own tests, PropRAG's graph stage alone and Dense's reading-stage step.

Table 22: **Chain completion and the 72B reader's answer.** For each cell of Table 2 read at the 72B reader: chain completion and reader accuracy (exact match on Wikipedia, the benchmark's token-overlap accuracy on news, accuracy on QASC's eight-way questions) under the pipeline's own Top-5 and under KNIT; the reader's accuracy under the Top-5 on the questions whose chain the Top-5 completes and on those it does not; the number of questions whose chain KNIT completes and the Top-5 does not, and the reverse; and, of the chains KNIT completes, the share the reader had already answered under the Top-5 and the share it answers with the chain. Percentages are of questions; the completes and breaks columns are counts.

Pipeline	Chain@5	reader	reader, Top-5, chain		chains KNIT		of those completed, answered	
	Top-5→KNIT	Top-5→KNIT	complete	incomplete	completes	breaks	before	after
MuSiQue, $n = 1{,}000$								
BM25	10.3→19.9	18.8→22.6	56.3	14.5	97	1	23.7	52.6
Hybrid	20.7→36.1	24.2→30.7	54.1	16.4	166	12	24.1	55.4
MonoT5	24.9→40.0	27.3→35.2	55.4	18.0	170	19	25.9	57.1
Dense	38.2→46.9	31.9→36.3	55.5	17.3	134	47	14.9	44.8
PropRAG	53.9→58.1	39.2→43.1	56.0	19.5	100	58	27.0	57.0
2WikiMultihopQA, $n = 1{,}000$								
BM25	32.8→44.1	40.5→44.0	65.2	28.4	113	0	23.9	49.6
Hybrid	41.4→56.4	42.0→49.0	61.6	28.2	150	0	12.0	48.0
MonoT5	44.2→57.0	41.7→47.7	59.5	27.6	130	2	10.0	49.2
Dense	51.1→62.4	45.7→49.9	58.5	32.3	117	4	9.4	44.4
PropRAG	83.1→92.1	56.8→57.8	58.5	48.5	95	5	50.5	70.5
HotpotQA, $n = 1{,}000$								
BM25	49.6→79.0	49.1→57.2	65.3	33.1	302	8	41.4	66.6
Hybrid	73.9→88.9	54.6→60.4	66.0	22.2	159	9	26.4	57.9
MonoT5	74.5→91.6	55.5→62.0	64.6	29.0	172	1	32.6	65.7
Dense	89.4→94.6	60.4→62.6	65.0	21.7	64	12	25.0	59.4
PropRAG	94.9→96.6	62.2→61.7	64.1	27.5	34	17	32.4	52.9
news, $n = 2{,}255$								
BM25	11.8→17.1	76.7→77.6	72.7	77.2	157	39	72.0	74.5
Hybrid	13.6→19.6	77.6→77.6	72.2	78.5	190	53	75.3	76.3
MonoT5	11.5→15.9	77.1→78.3	70.4	77.9	141	43	78.0	78.0
Dense	10.1→16.5	78.0→78.4	71.8	78.6	167	21	74.9	75.4
PropRAG	11.6→17.5	77.1→78.8	72.5	77.7	180	48	73.9	76.1
QASC, $n = 926$								
BM25	13.7→21.0	90.3→89.3	99.2	88.9	73	6	90.4	95.9
Hybrid	23.7→28.6	89.7→90.4	98.2	87.1	84	38	88.1	98.8
MonoT5	27.1→32.8	89.5→89.2	97.6	86.5	104	51	88.5	98.1
Dense	31.7→32.9	90.5→90.3	98.3	86.9	78	67	92.3	96.2
PropRAG	28.0→29.5	90.6→90.0	98.5	87.6	78	64	91.0	94.9

L THE TWO CORPORA OUTSIDE WIKIPEDIA

Three of the paper's five corpora are built on Wikipedia. The other two are: a corpus of science facts, read in Section L.2, and a news collection (Tang & Yang, 2024), re-chunked at 100 words with 12% overlap: 12,389 passages at a mean of 97.2 words over 609 articles, and 2,255 questions with gold passages. The benchmark ships no proposition graph and a capitalised-span entity index is not what the Wikipedia rows read, so a graph was extracted here with the same builder, prompt and extraction tier, giving 87,236 propositions, 61,631 entities and 153,184 incidences over those passages; the rows below read it. Every question carries 2 to 4 gold rows, so the $|\text{gold}| \leq 5$ primary stratum the analysis file names is the whole question set and Chain@5 is attainable everywhere; the uncapped control that file also asks for, Chain@$|\text{gold}|$, is not defined for an unordered five-passage set, and the $|\text{gold}|$ strata of Figure 11c stand in for it, a departure from that file as written.

The rule for this corpus is in the released analysis file. The graph stage alone is significantly up on PropRAG, run over the graph extracted here, at $+1.91$ [$+0.87$, $+2.86$] on an 89/46 discordant split from a Top-5 of 11.62. The four graph-less pipelines are at $+0.71$, -0.35, $+0.84$ and $+0.71$, every interval covering zero (Figure 8b).

The marks labelled § in Figure 11a re-read the same cells at $\gamma = 2.0$, $\beta = 1.0$. Under the frozen configuration PropRAG is significant on news and no graph-less news pipeline is; under the § tuple all four graph-less news pipelines are significant; the same constants leave PropRAG on MuSiQue not significant (Table 13).

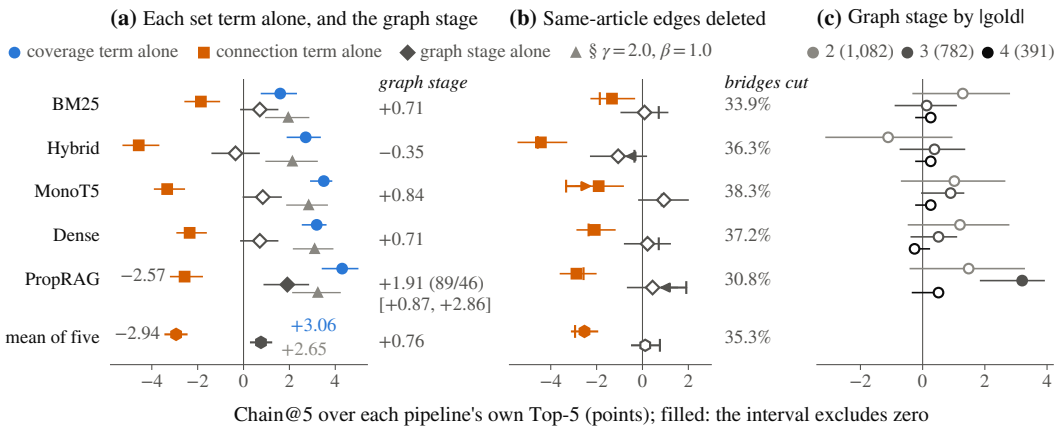

Figure 11: **On news the two set terms move in opposite directions, and deleting same-article edges does not change that.** Chain@5 over each pipeline's own Top-5 at the frozen configuration (MultiHop-RAG, $n = 2{,}255$), with the exact 95% interval; filled marks exclude zero. (a) The coverage term alone and the connection term alone; the frozen graph stage alone, its values at the right (PropRAG's with its interval and discordant split); and the same stage at $\gamma = 2.0$, $\beta = 1.0$ (§), shown for comparison only. (b) Every bridge edge whose two pool rows share a normalised title deleted: each series runs from a tick, with every edge kept, to a mark with its interval, without those edges; the share of bridges cut is at the right. (c) The frozen graph stage by gold cardinality, with each stratum's question count; a mark at the end of its interval is a stratum whose discordant questions all fall one way. Hexagons are five-pipeline means with question-level bootstrap intervals; the coverage and § means are printed without one. Table 23 gives what the mask moves, pooled; every number drawn is in the supplementary notes, Section S9.

L.1 The mechanism, measured by the arms the analysis file names

The analysis file fixes two arms and one diagnostic. The gain available here is the coverage term's, net of a connection term that is significantly down on all five pipelines (Figure 11). Raising the connection weight and halving the coverage weight, which is what the frozen configuration does relative to the earlier-constants candidate ($\gamma = 2.0$, $\beta = 1.0$), moves the four graph-less cells from significant to not significant.

The comparison the graph stage actually makes fails. The diagnostic reads four of the five pipelines: BM25, Hybrid, MonoT5 and Dense. A selection only ever compares passages within one question, and there the mean within-question AUC of the edge weight separating gold pairs from the rest is below 0.5 on five of their eight pipeline-by-depth cells.

A question's strongest bridge is a gold pair below chance. One fixed pool is read first: the top 20 passages by TF-IDF cosine for each of the 2,255 questions, 428,450 pairs in all. In the pool of a question with two or more gold passages, gold-gold pairs are 0.7% of the pairs, and on the proposition graph such a question's strongest bridge is a gold pair 0.2% of the time. That is a lift of 0.33 over the prior (computed before rounding), and the lift is below 1 on each of the four pipelines at every depth (0.29 to 0.90). The same lift is 1.43 on the capitalised-span substrate and 13.91 on MuSiQue's proposition graph.

The strongest bridge is a same-article pair most of the time. On the fixed pool, replacing the capitalised-span entity index with the proposition graph moves the share of questions whose strongest bridge is a same-article pair from 71.7% to 68.3%, and the pooled AUC from 0.546 to 0.573. The same edge builder on MuSiQue's proposition graph over PropRAG's pools gives 25.4% and 0.702, on a different corpus and pool. The reason is in the corpus: 96.2% of the gold pairs of the fixed pool are cross-article (95.0 to 96.5% across the eight pipeline cells), while overlapping 100-word windows of one article repeat its names. Adjacent chunks are therefore bridged 80.0% of the time, and 34.5% of all bridged pairs sit inside one article against 3.8% of gold pairs.

Every passage has propositions. All 12,389 passages carry at least one proposition (mean 7.04, median 7) and no gold row of any question lacks one, so the failure is not an extraction hole.

Scope of the diagnostic. This is a property of the corpus and graph pair measured here, not of the extraction step alone: the entity substrate was replaced, the question-entity exclusion was widened by aliases, and the same-article concentration survived both. A different chunking is not measured, and the 80.0% adjacency figure follows directly from the 12% overlap of these chunks. On the one pipeline the graph stage lifts, the coverage term carries the gain against a connection term at -2.57 on its own. The deletion test of Appendix H.2 has no cell on this corpus: MultiHop-RAG carries no hop labels, so neither the antecedent nor the dependent hop is defined here, and we cannot test for a positive excess on news, which would have argued against coverage transferring while connection does not.

The same-article concentration is not what costs the connection term its points. The measurements above hold at the frozen depth; the next one tests whether they explain the cost. Deleting every bridge edge whose two pool rows carry the same normalised title cuts a mean of 35.3% of a pipeline's bridges, confirming the 34.5% above at this depth, and moves the connection-only arm's selection on more than half the questions. The term is still significantly down on all five pipelines (Figure 11b) and the recovery covers zero (Table 23). What the mask does cost is the full graph stage's one significant news gain: PropRAG's $+1.91$ falls to an interval covering zero and the pooled stage changes by -0.64 $[-1.13, -0.16]$, so the same-article edges carry something the full stage uses. The same mask on MuSiQue cuts a far smaller share of that corpus's bridges and moves its full graph stage by -0.34 $[-0.80, +0.12]$, about as far as the mask moves news's connection-only arm ($+0.41$ $[-0.02, +0.83]$) and in the other direction; neither interval excludes zero, so the mask is not repairing news selectively. This run is exploratory: it was designed after the rows above were read, and it replaces no number of the frozen configuration.

Table 23: **What the same-article mask moves, pooled.** The lower block of the mask table (supplementary notes, Section S9): Chain@5 pooled over the five pipelines with the question-level bootstrap; the coverage-only arm, which leaves out the connection term, is a check on the masked run, and bold marks an interval excluding zero.

Corpus and arm	masked − unmasked, pooled	selections moved	bridges cut
news, connection only	$+0.41$ $[-0.02, +0.83]$	1,330–1,622 of 2,255	30.8–38.3%
news, full graph stage	$\mathbf{-0.64}$ $[-1.13, -0.16]$	1,277–1,535 of 2,255	30.8–38.3%
news, coverage only (check)	$+0.04$ $[-0.00, +0.08]$	2–23 of 2,255	30.8–38.3%
MuSiQue, full graph stage	-0.34 $[-0.80, +0.12]$	187–208 of 1,000	7.0–8.5%

L.2 A SCIENCE CORPUS

QASC (Khot et al., 2020) is the second corpus outside Wikipedia, and unlike the news collection it is made of single sentences. Each of its grade-school science questions is annotated with the two facts whose composition answers it. We read all 926 validation questions. The benchmark's retrieval corpus of 17 million sentences is out of reach for a graph that costs one LLM call per passage, so the corpus here is closed: the 7,671 distinct facts the training and validation questions are annotated with, 8.95 words on average, none with a title. The query is the question text alone; the eight answer choices go to the reader only. Gold is the two annotated facts, so Chain@5 asks for both among the five selected, as on the other corpora. The graph was extracted with the builder, prompt and extraction tier of the news graph and holds 10,206 propositions and 7,298 entities; with no titles, the title bonus of the relevance term fires on no passage. How the results would be read was written into the analysis file before the corpus was assembled: every question, the frozen configuration, KNIT against each pipeline's own Top-5 as the primary contrast, the graph stage alone and the control below as secondary contrasts, and how each outcome would be read. Nothing was tuned on this corpus. Table 24 gives every cell. KNIT is significantly above the Top-5 on BM25, Hybrid and MonoT5 and lowers no pipeline. On Dense and PropRAG, whose own Top-5 are the two highest on this corpus, its gain is not significant. The graph stage alone gains $+2.70$ $[+1.68, +3.76]$, significantly on the same three pipelines, and here the connection term carries it: run alone it gains $+1.75$ $[+0.73, +2.79]$ and the coverage term $+0.93$ $[+0.52, +1.36]$, both significant.

What the connection term reads here. The bridge diagnostic of Section L.1, run with the same instrument, puts this corpus on MuSiQue's side. At the frozen depth of 30 a question has one gold pair among the 435 pairs of its pool, a prior of 0.23%. Its strongest bridge is that pair 6.0% of the

Table 24: **The science corpus under the frozen configuration.** QASC, all 926 validation questions over 7,671 single-sentence facts; Chain@5 asks for both annotated facts among the five selected. Paired differences with the exact interval at $\alpha = 0.05$; n_{10}/n_{01} is questions won by the first-named arm and the reverse; bold marks an interval that excludes zero. Upper table: the graph stage alone and KNIT, each against the pipeline's own Top-5. Lower table: what the reading stage adds to the graph stage's five; the control, which is the same call with the same instruction over the pipeline's own top ten; KNIT against that control. Single-term arms, five-pipeline means over Top-5: connection only $+\mathbf{1.75}$ [+0.73, +2.79], coverage only $+\mathbf{0.93}$ [+0.52, +1.36].

Pipeline	cov@30	Top-5	graph stage	Δ over Top-5 [95% CI]	n_{10}/n_{01}	KNIT	Δ over Top-5 [95% CI]	n_{10}/n_{01}
BM25	28.51	13.71	17.93	$+\mathbf{4.21}$ [+2.72, +5.23]	48/9	20.95	$+\mathbf{7.24}$ [+5.84, +8.05]	73/6
Hybrid	58.96	23.65	26.67	$+\mathbf{3.02}$ [+1.22, +4.56]	48/20	28.62	$+\mathbf{4.97}$ [+2.59, +7.09]	84/38
MonoT5	63.61	27.11	30.24	$+\mathbf{3.13}$ [+0.99, +5.08]	62/33	32.83	$+\mathbf{5.72}$ [+3.05, +8.18]	104/51
Dense	68.03	31.75	33.48	+1.73 [−0.29, +3.62]	49/33	32.94	+1.19 [−1.46, +3.79]	78/67
PropRAG	64.90	27.97	29.37	+1.40 [−0.43, +3.11]	40/27	29.48	+1.51 [−1.11, +4.07]	78/64
mean, five				$+\mathbf{2.70}$ [+1.68, +3.76]			$+\mathbf{4.13}$ [+2.76, +5.49]	

Pipeline	KNIT over the graph stage alone	control	KNIT over the control
BM25	$+\mathbf{3.02}$ [+1.60, +4.04]	19.44	+1.51 [−0.06, +2.89]
Hybrid	+1.94 [−0.21, +3.97]	28.40	+0.22 [−1.81, +2.22]
MonoT5	$+\mathbf{2.59}$ [+0.19, +4.86]	31.43	+1.40 [−0.83, +3.55]
Dense	−0.54 [−2.92, +1.86]	33.15	−0.22 [−2.20, +1.78]
PropRAG	+0.11 [−2.32, +2.54]	32.61	$-\mathbf{3.13}$ [−5.08, −0.99]
mean, five	$+\mathbf{1.43}$ [+0.22, +2.61]		−0.04 [−1.02, +0.93]

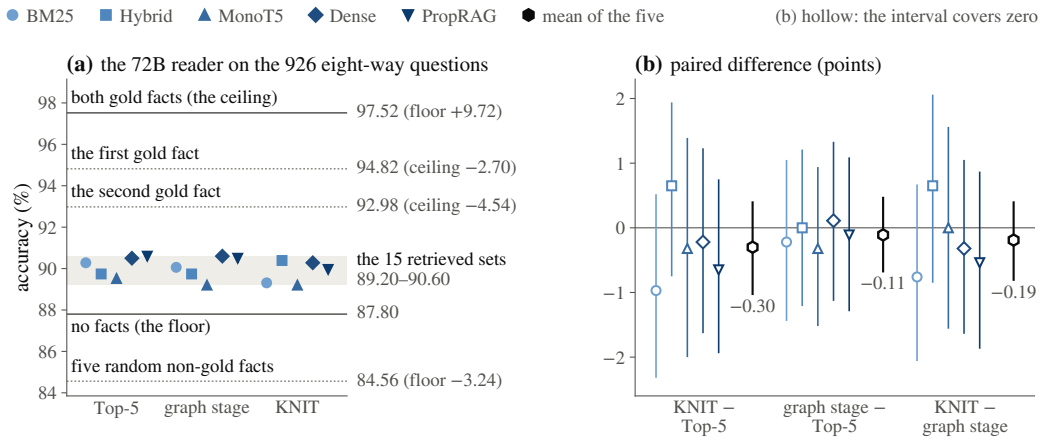

Figure 12: **On QASC the reader's knowledge, not the selection, sets the accuracy.** (a) Accuracy of the 72B reader on all 926 eight-way questions with no facts (the floor), five random non-gold facts, one gold fact (the first or the second) and both (the ceiling), and under each of the fifteen retrieved five-fact sets (the Top-5, the graph stage and KNIT on five pipelines), drawn as levels without intervals inside the grey band they span. Each reference line prints its accuracy and, except the floor, its paired difference from the floor or the ceiling; all four differences exclude zero (exact 95% interval). (b) Paired differences between the three retrieved sets on each pipeline, exact 95% intervals, the hexagon the mean of the five: none excludes zero. The reader answers each question with one letter under greedy decoding, and a reply that names no letter counts as wrong (at most 0.11% of a cell); every arm uses the same prompt, which admits an empty list of facts, and the five random facts are drawn from the corpus excluding the two gold facts with one fixed seed. Every number drawn is in the supplementary notes, Section S9.

time, a lift of 26.1 over the prior, from 13.0 on PropRAG to 67.6 on BM25; the same lift is 13.91 on MuSiQue and 0.33 on news. The mean within-question AUC of the edge weight separating the gold pair from the rest is 0.692 (0.679 to 0.726 by pipeline), where on news it is below 0.5 on five of eight cells. Only questions whose two gold facts are both in the pool enter, as on the other corpora: 2,630 of the 4,630 pipeline-question pairs, from 264 questions on BM25 to 630 on Dense. On the 220 questions for which every pipeline retains both facts the lift still runs from 19.8 (PropRAG) to 75.1 (BM25), so the spread across pipelines is not a difference in which questions are read. With every title empty, its same-article rows are not defined on this corpus and are not reported.

The shortlist does not repeat its Wikipedia advantage here. Against the same call over the pipeline's own top ten, KNIT does not differ significantly on four pipelines and is significantly behind on PropRAG, by 3.13 points [−5.08, −0.99], at a five-pipeline mean of −0.04 [−1.02, +0.93] (Table 24).

At the reader. Figure 12 shows the 72B reader on all five pipelines, three five-fact sets each. Its accuracy on the eight-way questions lies between 89.20 and 90.60% on every one of the fifteen sets, whose chain completion runs from 13.71 to 33.48: on this corpus the reader's accuracy does not follow which of these fifteen sets it is given. KNIT minus the Top-5 is −0.30 [−1.04, +0.41] pooled and is not significant on any pipeline; neither stage's own step is. The reader was given the five facts and the eight choices under a prompt frozen before any reader ran.

The reader's floor and ceiling. Why the fifteen sets sit so close together is answered by four further runs of the same reader under the same prompt, with the reading of every outcome written before they ran (Figure 12a). With no facts at all the reader answers 87.80% of the questions, the floor, and with both gold facts 97.52%, the ceiling; both gold facts add 9.72 points to that floor and one gold fact alone 2.7 to 4.5 points less than both, while the five-fact sets hold both on 13.7 to 33.5% of questions and their three other facts cost accuracy, as the random sets show. The fifteen retrieved sets stand 1.4 to 2.8 points above the floor and 6.9 to 8.3 below the ceiling. Every accuracy above is measured against the 87.80 floor.

Chain completion and the reader's answer on this corpus. Where both gold facts are among the Top-5 the reader answers 97.6 to 99.2% of the questions; where they are not, 86.5 to 88.9%, the no-facts floor (Table 22). Of the 73 to 104 chains KNIT completes on each pipeline, the reader had already answered 88.1 to 92.3% under the Top-5, and it answers 94.9 to 98.8% with both facts: +0.3 to +1.1 points over the corpus. KNIT also drops a gold fact the Top-5 held on 6 to 67 questions, which costs up to 0.6 points, and on the questions whose chain status does not change its other passages cost 0.1 to 1.4 points, as the random-fact sets of Figure 12a would predict. That is where the chain-level gain is spent.

M THE IN-HOUSE PIPELINES

Every table in the body is read on the five pipelines every stage of which is published. Five further pipelines were built here: four in-house retrieval configurations retrieving over the released proposition index (Xu & Li, 2026), and one a reciprocal rank fusion (Cormack et al., 2009) of PropRAG with the strongest of those four. Table 25 gives the frozen configuration's gain on each of the ten pipelines.

The five in-house pipelines. **Graph-A, Graph-B, Graph-C** and **Graph-D** are the four in-house configurations, ordered by their own bare Chain@5 on MuSiQue (24.3, 40.5, 42.2, 42.3); **Fusion** is the reciprocal rank fusion of PropRAG with Graph-D (bare 49.2). The graph stage's bridge matrix reads the index built here on all five, whatever index the pipeline retrieved over, on MuSiQue as on the transfer corpora.

Where results depend on the ten-pipeline population. Table 2, the reader-level contrasts and the tuning-budget comparison use the five published pipelines; results pooled over ten pipelines say so where they appear. The unreachable share of Section 8 is 35.52% of 10,000 pipeline–question pairs over ten pipelines.

N THE INSTRUMENT

The resolution rule. For a single cell the instrument is the exact two-sided McNemar test with the 95% Clopper–Pearson interval on the discordant split $n_{10}/(n_{10}+n_{01})$, carried onto the difference scale by $\Delta = (n_d/n)(2\pi - 1) \cdot 100$, with $n_d = n_{10} + n_{01}$ and π the split n_{10}/n_d at each interval endpoint; the map is monotone at fixed n_d and n, so the endpoints are exact and no delta-method approximation enters. Pooled quantities are read instead under the question-level cluster bootstrap. A contrast is significant, or resolved, when its interval excludes zero, and for a paired binary outcome

Table 25: The frozen configuration is significantly up on ten of the ten pipelines. "Top-5" names the pipeline's own top-5 of its own 30-wide pool; "Graph stage" replaces that Top-5 and changes nothing else; Δ is paired per question. Per-pipeline intervals are Clopper–Pearson on the discordant split, the Mean rows' are the question-level cluster bootstrap, SIG marks an interval excluding zero (MuSiQue, $n = 1{,}000$).

	Pipeline	Top-5	Graph stage	Δ	95% CI
published	BM25	10.3	16.3	$+6.00$ SIG	$[+4.92, +6.47]$
	Hybrid	20.7	29.1	$+8.40$ SIG	$[+7.02, +9.23]$
	MonoT5	24.9	34.2	$+9.30$ SIG	$[+7.51, +10.63]$
	Dense	38.2	45.7	$+7.50$ SIG	$[+5.68, +8.90]$
	PropRAG	53.9	57.3	$+3.40$ SIG	$[+1.25, +5.36]$
in-house	Graph-A	24.3	35.5	$+11.20$ SIG	$[+9.38, +12.53]$
	Graph-B	40.5	45.0	$+4.50$ SIG	$[+2.29, +6.49]$
	Graph-C	42.2	44.7	$+2.50$ SIG	$[+0.88, +3.88]$
	Graph-D	42.3	48.2	$+5.90$ SIG	$[+3.81, +7.69]$
	Fusion	49.2	51.5	$+2.30$ SIG	$[+0.30, +4.15]$
	Mean, published five	29.60	36.52	$+6.92$ SIG	$[+5.86, +8.00]$
	Mean, in-house five	39.70	44.98	$+5.28$ SIG	$[+4.00, +6.58]$
	Mean, all ten	34.65	40.75	$+6.10$ SIG	$[+5.13, +7.08]$

that is the same event as $p < 0.05$: over an exhaustive grid of 36,900 (n, n_{10}, n_{01}) cells the identity holds with 0 violations.

Dependence between pipelines, and the absolute level. *The sign-flip test of the mean gain*, which Section 4.1 cites, keeps the dependence between pipelines. Its statistic is the mean over pipelines of each pipeline's mean per-question paired difference, the graph stage's Chain@5 minus the Top-5's on MuSiQue's $1{,}000$ questions. Its null draws one sign per question and applies it to every pipeline at once, so the dependence between pipelines that share a first stage is kept, and the p-value is the share of $2{,}000{,}000$ such draws whose statistic is at least the observed one; no draw reached it on any of the pipeline sets below, which bounds p by 5×10^{-7} throughout. Read the same way on the five published pipelines, KNIT's two-stage gain of $+10.60$ is also beyond every draw. *The joint sign test*, under the same per-question null, asks instead whether every pipeline's mean gain is positive, and it fails on one pipeline set, the one Table 2 is read on. At $2{,}000{,}000$ draws its values are 0.01153 over ten pipelines and 0.03804 over the six pipelines that took no part in choosing the earlier configuration (all but Dense, PropRAG, Graph-B and Graph-D); over the five published pipelines it is 0.055, above 0.05, and 0.130 when those five are collapsed to their three independent sources. It discards magnitude, and at five pipelines its nominal floor is 0.031. A count of significant pipelines is descriptive and not a test, since it treats correlated pipelines as separate evidence. The interval this paper reports for the ten-pipeline mean is the question-level cluster bootstrap, which comes out at $[+5.13, +7.08]$ around $+6.10$ at the frozen configuration (Appendix F).

The absolute Chain@5 differs from a published value of the same corpus. This paper's PropRAG bare chunk-exact Chain@5 is 53.90 against 33.8 (Lau et al., 2026) at the same metric, k and question count, and our implementation's BM25 is at 10.30 against that table's 6.4. Every contrast here is a within-pipeline paired delta, and no delta's validity depends on the absolute Chain@5, though its size depends on the headroom that level leaves and the level-reset finding is indexed on absolute Chain@5 itself (Section 4.4 and Appendix D). The difference is real and it tracks the pool: the same comparison reports PropRAG Recall@5 at 61.8 where our implementation has 78.33, and BM25 alone spreads 11.9 points across four published papers on this corpus (Gutiérrez et al., 2025; Wang & Han, 2025; Xu & Li, 2026; Lau et al., 2026).