

The Cost of Incoherent Context Grows with Capability: A Control for Exemplar Retrieval

Srinivas Harish

New York University
srinivas.harish@nyu.edu

Abstract

Comparisons of retrieved and randomly selected exemplars in mathematical in-context learning confound query relevance with changes in the demonstration pool. We separate these effects by assigning each query the demonstrations retrieved for another query. This derangement preserves the aggregate demonstrations, counts, token budget and retrieval-induced selection bias while changing the query-to-exemplar assignment.

On GSM8K, derangement changes accuracy by -0.24 pp (90% CI $[-0.85, +0.37]$). The pooled effect is equivalent to zero within 1.0 pp across conditions that include embedding retrieval, a gold-computational-graph oracle and a 7B model. On MathQA, the change is -8.36 pp. Near-duplicates occur above cosine 0.95 for 64.6% of MathQA queries and no GSM8K queries. Removing them changes the matching term by -13.56 pp relative to queries unaffected by removal.

Pairing each retrieved problem with another retrieved rationale changes accuracy by -22.40 to -52.24 pp. Across 8 conditions and 3 model families, this change correlates with excess copying of demonstration answers (Spearman -0.976). Across 7 models, the correlation between the accuracy change and unpermuted accuracy is $+0.286$. These results support reporting pool near-duplicate density alongside retrieval gains and checking problem-rationale alignment during prompt construction, since retrieval metrics remain unchanged under this within-demonstration permutation.

1 Introduction

Exemplar retrieval is used in mathematical in-context learning. Embedding retrievers select demonstrations by semantic similarity to the query (Liu et al., 2022; Rubin et al., 2022); complexity- and diversity-based selectors reshape which demonstrations appear (Fu et al., 2023; Zhang et al.,

2023); and structure-aware retrievers match the computational graph of the query to the computational graphs of candidate demonstrations, on the explicit argument that semantic retrieval “prioritise[s] superficial linguistic similarity over deep structural alignment” (Yang et al., 2024b).

Retrieval changes both the assignment of demonstrations to queries and the frequency with which demonstrations are selected from the pool. Comparisons against random selection vary both, leaving their contributions to accuracy unresolved.

Let $R_r(q)$ be the k demonstrations retriever r selects for query q and σ a derangement of the test set. Assigning $R_r(\sigma(q))$ to q preserves the empirical distribution of retrieved sets, including their counts, aggregate token budget and concentration, while randomising the query assignment. Comparing this condition with the original retrieval condition estimates the contribution of matching at a fixed demonstration distribution. Resampling retrieved sets preserves that distribution in expectation; a derangement also prevents queries from retaining their original set, subject to the collision handling in §2.2.

Query assignment and near-duplicates. On GSM8K, deranging the query-to-exemplar assignment changes accuracy by -0.24 pp (90% CI $[-0.85, +0.37]$, TOST $p = 0.0186$). The pooled effect meets the pre-stated equivalence margin of 1.0 pp. Across three models, including a 7B model, no condition changes by more than 2.01 pp and no McNemar p is below 0.122. For the oracle retriever, realised structural similarity falls from 0.941 to 0.425, near the uniform-random level. On MathQA, the same derangement changes accuracy by -8.36 pp ($[-10.34, -6.38]$); the matching term accounts for the retrieval gain ($\Delta_{\text{total}} = +10.20$, $\Delta_{\text{match}} = +12.60$, $\Delta_{\text{pool}} = -1.65$ pp).

No GSM8K test query has a pool neighbour at cosine 0.95 or above, compared with 64.6% on

MathQA, where 26.4% exceed 0.99 and the median nearest-neighbour similarity is 0.973. Such neighbours can share a problem template with different numerals, providing a worked solution method for the query. Within MathQA, correct assignment improves accuracy by +18.11 pp for queries with a near-duplicate and +2.54 pp for those without one. Removing near-duplicates from each query’s candidate list changes the matching term by −13.56 pp relative to unaffected queries.

Fu et al. (2023) previously described MathQA training and development questions with similar wording and small changes to the numerals, reporting retrieval–random gains of +3.5 pp on GSM8K and +36.5 pp on MathQA. Jie and Lu (2023) reproduce this asymmetry and find similar performance for random and least-similar selection on MathQA. The dataset papers also describe differences in construction (Amini et al., 2019; Cobbe et al., 2021). We test the near-duplicate account by isolating assignment from pool composition, measuring near-duplicate availability and manipulating it within MathQA.

Within-demonstration permutations. Permuting problem–rationale pairings reduces accuracy in every math-tuned condition on both corpora, with or without near-duplicates (Table 1). Presentation order, included as a positive control based on prior work (Lu et al., 2022), has no significant effect in any of the four conditions. The general base model has an INTRA change of −2.16 pp, while the least disruptive step permutation, STEPTOPO, changes accuracy by −0.99 pp ($p = 0.203$). In all four math-tuned 1.5B conditions, permuting problem–rationale pairings has a larger effect than permuting query assignments. On MathQA, the respective changes are −25.55 and −11.85 pp; on GSM8K, the assignment effect is near zero, so a ratio would be unstable. For the math-specialised and general 1.5B models, within-set derangement changes accuracy by −22.40 and −2.16 pp, respectively, a factor of 10.4 in magnitude.

Scope. Prior work establishes the importance of internal correspondence (Wang et al., 2023; Madaan and Yazdanbakhsh, 2022) and reports retrieval gains on MathQA associated with near-duplicate examples. Our contribution is a control that separates assignment from pool composition and measures their respective effects.

Contributions.

1. A derangement control for exemplar retrieval and an exact three-term decomposition of any reported gain into pool-reshaping, difficulty-coupling and query-matching terms (§2).
2. A test of the near-duplicate account of mathematical retrieval gains, which the literature states as an observation (Fu et al., 2023; Jie and Lu, 2023). We quantify pool near-duplicate density by cosine, delete near-duplicates from each query’s candidate list, and estimate the effect as a difference-in-differences against an internal placebo (§4).
3. The permutation hierarchy inside the demonstration, placing the internal and external senses of relevance on one commensurable axis, with the manipulations of Wang et al. (2023) and Madaan and Yazdanbakhsh (2022) recovered as token-preserving permutations under one protocol (§5).
4. A comparison across 8 conditions and 3 model families relating accuracy changes to excess answer copying (Spearman −0.976) and unpermuted accuracy (+0.286). A pre-registered control that restores correct problem–answer pairs produces a larger GSM8K loss, indicating that answer-copying alone does not explain it.
5. Recommendations for reporting near-duplicate density and checking problem–rationale alignment during prompt construction (§6).

The released artifacts include per-item generations and CPU scripts for recomputing the reported statistics. GPU computation is required for generation.

2 Permutation as a Measurement Instrument

2.1 The decomposition

Fix a pool P , shot count k , and retriever r mapping a query to an ordered $R_r(q) \subset P$, $|R_r(q)| = k$; write $\text{Acc}[\cdot]$ for accuracy under a prompt-construction rule and U_k for a uniform draw of k from P . The literature reports $\Delta_{\text{total}}(r) = \text{Acc}[R_r(q)] - \text{Acc}[U_k]$. Let σ be a derangement of the test indices ($\sigma(i) \neq i$ everywhere) and σ_s one confined to strata of a nuisance variable s , here the gold step count. Then, writing $A_\pi =$

$\text{Acc}[R_r(\pi(q))]$,

$$\underbrace{A_\sigma - \text{Acc}[U_k]}_{\Delta_{\text{pool}}} + \underbrace{A_{\sigma_s} - A_\sigma}_{\Delta_{\text{diff}}} + \underbrace{A_{\text{id}} - A_{\sigma_s}}_{\Delta_{\text{match}}} = \Delta_{\text{total}}. \quad (1)$$

The terms telescope, so Equation 1 requires no modelling assumption. Δ_{pool} measures the effect of changing the distribution of demonstrations shown; Δ_{diff} measures the effect of coupling demonstration and query difficulty through the specified strata; and Δ_{match} measures the remaining query-specific matching effect. An unrestricted derangement changes both difficulty coupling and query matching, which motivates the stratified condition. U_k samples uniformly from the pool, whereas $R_r(\sigma(q))$ uses the empirical distribution of retrieved sets, which can be highly concentrated (Appendix J). Comparisons of retrieval against random selection estimate the sum of these terms.

2.2 The experimental conditions

Write a demonstration as a triple (x_j, y_j, a_j) : problem, rationale, stated final answer. We call each experimental condition an *arm*: one complete evaluation of the model over the test set, differing from the others only in the permutation applied to the retrieved demonstrations. The unpermuted prompt for query q is the concatenation of $(x_j, y_j, a_j)_{j=1}^k$ for j indexing $R_r(q)$, most-similar demonstration placed adjacent to q , followed by q .

Outside the demonstration. DER replaces $R_r(q)$ by $R_r(\sigma(q))$ for a derangement σ over test indices. Because a collapsed top- k can return the same set to different queries, we derange over sets rather than over indices and repair any position handed its own set back, so no item sees its own treatment. DERSTRAT is the same with σ confined to strata of the gold step count $\min(n_{\text{steps}}, 6)$; singleton strata are left in place and counted without merging strata, and the within-stratum collision repair fails on 18/1319 GSM8K STRUCT items, which do see their own set. ORDER applies a derangement over the k slots, permuting the presentation order of the selected demonstrations.

Inside the demonstration. All three hold $R_r(q)$ fixed. INTRA applies a derangement π over the k slots and prints $(x_j, y_{\pi(j)}, a_{\pi(j)})$. The multiset of problems, the multiset of rationales, the multiset of answers, the token count, the retrieval similarity and the pool composition are identical to the

unpermuted arm for each item, and every rationale shown remains a correct worked solution to a problem that is itself in the prompt. INTRACORRECT prints $(x_j, y_{\pi(j)}, a_j)$ with the same π from the same seed, so the stated answer is correct for the displayed problem. STEPORDER permutes the lines of each y_j ; STEPTOPO and STEPVIOL do so respecting and violating the derivation DAG read off the arithmetic, applied to a rationale in both arms or neither so the two are matched. NUMPERM deranges the numeric literals within y_j , preserving the equation skeleton, the entities and the token count while destroying the arithmetic, the invalid-reasoning manipulation of Wang et al. (2023) realised as a permutation under the same evaluation protocol.

The assignment and within-set permutation controls preserve the aggregate prompt token multiset; within-set permutations preserve it per item. The INTRACORRECT control also changes which final answer accompanies each deranged rationale. Appendix A gives the arm table.

3 Experimental Setup

Corpora and graphs. For GSM8K (Cobbe et al., 2021), the 7473-problem training split forms the pool and all 1319 test problems are queried. MathQA (Amini et al., 2019) supplies a 8000-problem pool and 1000 test problems. Both are in the evaluation suite of Yang et al. (2024b), whose credited mechanism motivates this study. GSM8K solutions carry inline `<<expr=val>>` annotations, from which we recover per step the operator sequence, operand count, and which earlier results the step consumes; MathQA ships the graph in its `linear_formula` field, whose `#k` references give the dependency edges. Graph similarity combines longest-common-subsequence agreement over per-step operator tuples, depth agreement and dependency-pattern agreement.

Retrievers. SEM ranks by cosine similarity of BGE-base-en-v1.5 embeddings (Xiao et al., 2024) of the problem statements, the retriever family used by Yang et al. (2024b). STRUCT ranks by computational-graph similarity to the query’s gold graph, ties broken by fixed random priority rather than pool order. STRUCT is an oracle and is not deployable; it provides a control with verified structural matching before derangement. It places a demonstration carrying the query’s exact gold operator sequence into 69.3% of slots (against 0.8%

for the embedding retriever and 0.7% for a uniform draw) and its gold step count into 98.7% (against 22.2% and 21.2%). On GSM8K, this oracle improves accuracy by about one point.

Models and decoding. Qwen2.5-Math-1.5B, Qwen2.5-Math-7B (Yang et al., 2024a) and, as a same-size non-math control, Qwen2.5-1.5B; all three are base models evaluated few-shot. §5 adds four checkpoints to separate capability from task specialisation: Qwen2.5-0.5B and Qwen2.5-3B, OLMo-2-1B (Team OLMo et al., 2025), all base, and Phi-3.5-mini-instruct (Abdin et al., 2024). Generation used one 32 GB GPU and greedy decoding, so RAND and TRUE are deterministic and the only randomness is the derangement draw, whose seeds are literals in the released `run_arms.py` (12345 for RAND; 1000, 2000, 3000, 4000 plus the replicate index for the others). INTRACORRECT draws INTRA’s permutation from the same seed, matching the two arms item by item. $k = 4$, most-similar demonstration adjacent to the query. All arms answer the same queries, so every contrast is paired. We use exact-binomial McNemar tests per replicate, report the largest replicate p -value, compute paired bootstrap intervals over test items and apply Holm correction across each hierarchy. The random baseline is averaged over independent draws. Appendix A gives full arm definitions and a real prompt fragment under INTRA (Figure 1).

4 The Assignment Term and Near-Duplicate Density

Table 1 reports accuracy under each permutation for both corpora.

GSM8K assignment effects. Retrieval gains over random selection are -0.53 and $+2.12$ pp for the math 1.5B model, -0.30 and $+0.91$ pp for the general 1.5B model, and -0.08 and $+1.14$ pp for the math 7B model. None is significant. Our prediction that the general model would exceed $+3$ pp, based on less specialised pretraining, was not supported.

Derangement changes accuracy by -0.24 pp over 1319 paired items (90% CI $[-0.85, +0.37]$). A TOST equivalence test meets the pre-stated 1.0 pp margin ($p = 0.0186$). The per-condition minimum detectable effect is 2.88 pp at 80% power, so individual conditions cannot exclude a four-point effect; the pooled analysis, clustered by item, excludes a one-point effect.

At 7B, random selection reaches 84.31 accuracy, and derangement changes accuracy by -1.02 pp ($p = 0.364$) and -0.38 pp ($p = 0.878$), with an MDE of 2.54 pp. Related results vary by model size. Table 2 of Yang et al. (2024b) reports $+1.3$ pp at 70B, $+0.1$ at 1–1.5B and performance below random at 8B. Fu et al. (2023) report $+3.5$ pp on GSM8K for complexity-based selection. Our embedding and gold-graph retrievers show no significant gain at either evaluated scale.

MathQA assignment effects. Retrieval improves accuracy by $\Delta_{\text{total}} = +10.20$ pp for SEM and $+4.10$ pp for STRUCT, from a random baseline of 58.20. Derangement changes accuracy by -11.85 pp ($p = 6.9 \cdot 10^{-8}$) and -3.60 pp ($p = 0.064$), respectively. The pooled change is -8.36 pp ($[-10.34, -6.38]$), compared with -0.24 on GSM8K. Equation 1 gives matching terms of $+12.60$ and $+5.40$ pp, with pool terms of -1.65 and $+0.50$ pp. The matching terms account for the retrieval gains, and the encoder comparison in Appendix E gives equal matching terms to two decimal places. On GSM8K, the corresponding terms are $+0.00$ and $+1.40$ pp (Table 10).

Pool near-duplicate density. On GSM8K, the mean nearest-neighbour cosine is 0.807, and 0.0% of 1319 queries have a pool neighbour at 0.95 or above. On MathQA, the mean is 0.947, the median is 0.973, and 64.6% and 26.4% of queries exceed 0.95 and 0.99, respectively. For example, the MathQA query “sari and ken climb up a mountain ... sari wakes up at 07:00” has a nearest pool problem with cosine 0.9997 that instead specifies “... sari wakes up at 05:00”. The gold answers differ, but the neighbour supplies a solution method for the same template.

Despite its low lexical near-duplicate density, GSM8K has a pool problem with the same gold operator signature for 78.9% of queries, with a median of 29 candidates. This structural availability produces no significant retrieval gain in the oracle condition.

Near-duplicate removal. The corpora differ in format, answer type, rationale style and difficulty, so we also manipulate near-duplicate availability within MathQA. For every MathQA query we delete from its candidate list every pool item at cosine $\geq \tau$ of it, for a threshold $\tau = 0.95$, then re-rank. The pool, retriever, model, k , prompt format and test set are unchanged; the candidate lists

corpus	arm	SEM		STRUCT	
		acc (%)	Δ (pp)	acc (%)	Δ (pp)
GSM8K, general 1.5B	uniform random	66.19			
	retrieved (unpermuted)	65.88	-0.30	67.10	+0.91
	DER: query $\leftrightarrow$ set deranged	66.49	+0.61	66.79	-0.30
	DERSTRAT: deranged within difficulty stratum	66.26	+0.38	67.02	-0.08
	INTRA: problem $\leftrightarrow$ rationale deranged in-set	63.72	-2.16	64.97	-2.12
GSM8K, math 1.5B	uniform random	74.45			
	retrieved (unpermuted)	73.92	-0.53	76.57	+2.12
	DER: query $\leftrightarrow$ set deranged	75.32	+1.40	74.56	-2.01
	DERSTRAT: deranged within difficulty stratum	73.92	+0.00	75.17	-1.40
	retrieved (unpermuted), finer-scale stage	73.77		76.80	
	ORDER: presentation order permuted	74.53	+0.76	76.50	-0.30
	STEPTOPO: steps reordered within the derivation DAG	72.78	-0.99	74.75	-2.05
	STEPVIOL: steps reordered against the derivation DAG	71.34	-2.43	71.42	-5.38
	STEPORDER: rationale step order permuted	65.05	-8.72	66.26	-10.54
	NUMPERM: rationale numbers deranged	55.19	-18.57	58.91	-17.89
	INTRA: problem $\leftrightarrow$ rationale deranged in-set	51.52	-22.40	31.31	-45.26
GSM8K, math 7B	uniform random	84.31			
	retrieved (unpermuted)	84.23	-0.08	85.44	+1.14
	DER: query $\leftrightarrow$ set deranged	83.21	-1.02	85.06	-0.38
	DERSTRAT: deranged within difficulty stratum	84.00	-0.23	85.63	+0.19
	INTRA: problem $\leftrightarrow$ rationale deranged in-set	31.99	-52.24	28.28	-57.16
MathQA, math 1.5B	uniform random	58.20			
	retrieved (unpermuted)	68.40	+10.20	62.30	+4.10
	DER: query $\leftrightarrow$ set deranged	56.55	-11.85	58.70	-3.60
	DERSTRAT: deranged within difficulty stratum	55.80	-12.60	56.90	-5.40
	retrieved (unpermuted), finer-scale stage	68.80		63.60	
	ORDER: presentation order permuted	66.30	-2.50	62.70	-0.90
	NUMPERM: rationale numbers deranged	38.40	-30.40	44.00	-19.60
	INTRA: problem $\leftrightarrow$ rationale deranged in-set	42.85	-25.55	46.05	-16.25

Table 1: The permutation hierarchy. Δ is against the unpermuted retrieved arm of the same retriever, except the unpermuted row itself, which is against uniform random selection. Rows above the inner rule permute something outside a demonstration, rows below permute something inside one. All arms answer the same queries; the permutation controls preserve the aggregate token multiset. Accuracies for the finer scales come from a separate generation stage with its own RAND draw, so they are compared against that stage’s unpermuted arm. Exact McNemar and Holm-adjusted p -values, and bootstrap intervals, in Table 4.

exclude near-duplicates. After pruning, no top-4 set contains a near-duplicate, and mean maximum query-demonstration cosine falls from 0.947 to 0.887. We registered the prediction that Δ_{match} would fall from +12.60 pp to roughly +2.5 pp.

Δ_{match} falls from +12.60 to +5.25 pp ([+2.10, +8.45]) for SEM and from +5.40 to -0.15 pp ([−3.15, +2.95]) for STRUCT; the registered prediction was consistent with STRUCT, whose remaining effect is not significant, and below the observed SEM effect, which remains significant at +5.25 pp.

Generation drift and the pruning estimate.

Greedy decoding with identical prompts produces different outputs across generation stages. The two stages’ RAND arms have byte-identical demonstration sets on 1000/1000 items, yet accuracy differs by up to 1.00 pp, with 6.0% of predictions changing. Batch composition changes the kernels’ reduction order, so we use a difference-in-differences estimate to account for drift between stages.

The 354 queries with no pool neighbour above

τ form the placebo group. Their top-4 sets are verified identical across stages for both retrievers. The matching-term change is +1.41 pp in this group and −12.15 pp for the 646 treated queries, giving a difference-in-differences estimate of −13.56 pp (95% CI [−19.60, −7.33], $p = 1.0 \cdot 10^{-4}$) for SEM and −9.90 pp ([−15.30, −4.54], $p = 6.0 \cdot 10^{-4}$) for STRUCT. This adjustment removes the drift measured in the placebo group.

Pruning leaves residual similarity of 0.887, compared with GSM8K’s 0.807, and also changes candidate quality, making $\Delta_{\text{pool}} = -2.10$ and $\Delta_{\text{diff}} = -3.15$ pp negative. Removing near-duplicates reduces most of the assignment benefit, but the pooled change remains −1.61 pp ([−3.40, +0.15]), which is not equivalent to zero. Within-demonstration permutation still changes accuracy by −12.90 pp ($p = 2.0 \cdot 10^{-11}$) and −8.80 pp ($p = 5.6 \cdot 10^{-6}$). These effects are about half their unpruned magnitudes, from a baseline 9.3 points lower, and establish that within-demonstration sensitivity persists after pruning.

In the observational comparison, correct assignment improves accuracy by +18.11 pp for queries whose nearest pool neighbour exceeds cosine 0.95 ($n = 646$), and by +2.54 pp for those below the threshold ($n = 354$). The difference is approximately sevenfold under the same retriever, model and prompt format. The fitted slope against realised similarity is +50.8 pp per unit cosine on MathQA and -5.7 on GSM8K, although the MathQA effect is not monotone across quartiles (Appendix G).

Similarity and nuisance checks. For STRUCT on GSM8K, realised graph similarity falls from 0.941 unpermuted to 0.599 under stratified derangement and 0.425 under unrestricted derangement, compared with 0.418 for uniform selection (Appendix C). Accuracy changes by -2.01 pp (95% CI $[-4.09, -0.04]$, McNemar $p = 0.122$). The INTRA arm matches its unpermuted counterpart to four decimals on token budget, step count, embedding similarity and graph similarity, excluding changes in these quantities as explanations for its accuracy loss.

5 Within-Demonstration Effects

Table 1 compares permutations between and within demonstrations. Deranging the problem-to-rationale correspondence changes accuracy by -22.40 pp ($p = 5.5 \cdot 10^{-49}$) on GSM8K and -25.55 pp on MathQA, compared with -0.24 and -8.36 for deranging the query-to-set assignment. The prompt in the INTRA arm contains k correct worked solutions to k retrieved problems; each solution remains valid for its source problem but is displayed beneath another problem.

Internal and external relevance. Wang et al. (2023) define relevance within a demonstration, where each rationale component corresponds to part of its problem. Retrieval studies instead measure relevance to the test query. We vary both under the same protocol.

Under DER, each rationale remains paired with its original problem, but the set is reassigned to a different query. Under INTRA, the set remains the one retrieved for the query, but its problems and rationales are paired incorrectly. Both conditions preserve individually valid rationales and the aggregate multisets of problems and rationales; INTRA also preserves those multisets for each query. On GSM8K, the respective accuracy changes are -0.24 and -22.40 pp; on MathQA, they are -8.36

and -25.55 pp.

Wang et al. (2023) report GSM8K accuracy decreasing from 48.5 to 27.5 and 11.0 when internal relevance is removed, compared with 39.5 under invalid reasoning. Our NUMPERM (-18.57 pp) and STEPORDER (-8.72 pp) conditions reproduce this ordering. Their no-relevance conditions also change the stated answer. Madaan and Yazdanbakhsh (2022) keep that answer correct in a RANDOM TEXT condition that substitutes a valid rationale for another problem, obtaining 2.98% accuracy on GSM8K. Our INTRACORRECT condition performs this substitution within the prompt. Comparing it with INTRA measures the additional effect of correcting the stated answer while holding the deranged rationale fixed.

Model size and specialisation. On GSM8K, within-set derangement changes accuracy by -2.16 pp ($p = 0.258$) for general 1.5B, -22.40 pp for math-tuned 1.5B and -52.24 pp ($p = 5.7 \cdot 10^{-172}$) for math-tuned 7B. The 7B model achieves 84.31 with random demonstrations and 31.99 under INTRA. None of these models has a significant query-assignment effect.

Four further models distinguish size from task specialisation. OLMo-2-1B changes by -9.60 pp from 32.80 accuracy ($p = 0.0022$, $n = 250$). Phi-3.5-mini changes by -4.00 pp from 78.00, which is not significant ($p = 0.45$, $n = 100$). The general Qwen 0.5B, 1.5B and 3B models have unpermuted accuracies of 36.54, 65.88 and 75.82, with changes of -1.06 , -2.16 and $+0.23$ pp, none significant. General Qwen 3B and Qwen2.5-Math-1.5B have similar unpermuted accuracies of 75.82 and 73.92, but changes of $+0.23$ and -22.40 pp, respectively. Across 7 models from 3 families, the Spearman correlation between unpermuted accuracy and the signed INTRA change is $+0.286$.

Answer copying. Under INTRA, 100.0% of outputs contain a well-formed answer, and generation lengths remain similar to those in the unpermuted condition (Appendix K). We measure copying as the fraction of predictions equal to a printed demonstration answer. Excess copying relative to the unpermuted condition correlates with the signed accuracy change across 8 conditions and 3 families (Spearman -0.976 , Table 3).

OLMo-2-1B’s copy rate increases from 5.2% to 8.8%, and Phi-3.5-mini’s from 5.0% to 6.0%. Their excess rates are +3.6 and +1.0 pp, compared with 43.1 pp for math-tuned 7B. Each excess is

measured against the model’s own unpermuted rate. For math-tuned 1.5B, that rate is 6.4%, and the gold answer occurs among the demonstration answers on 6.1% of items, so a matching answer does not necessarily indicate erroneous copying.

Copying occurs in 46.0% and 69.6% of the math-tuned 1.5B model’s lost items. The non-copying subsets still lose -14.66 and -24.28 pp, with a loss of -26.47 at 7B. These associations are consistent with greater reliance on demonstrated patterns in the math-tuned models, including when those patterns are inconsistent.

Corrected final answers. INTRACORRECT uses the same rationale derangement and seed as INTRA, but restores the correct final answer for each displayed problem. The two conditions are matched item by item. We pre-registered that an effect near zero under INTRACORRECT would support answer-copying as the explanation and require reinterpreting the result as sensitivity to distractor answers.

On GSM8K, INTRACORRECT changes accuracy by -28.54 pp compared with -22.40 for SEM, and by -52.05 compared with -45.26 for STRUCT. The losses increase by roughly six points. Under INTRA, the rationale derives the displayed answer for another problem; under INTRACORRECT, it also conflicts with the displayed answer. This additional inconsistency accounts for the difference between conditions and indicates sensitivity to the relation between rationale and answer. On MathQA, INTRACORRECT is marginally less costly (-23.05 compared with -25.55). Five-way multiple choice may contribute to this difference, since a copied option letter has a one-in-five chance of being correct under uniform guessing.

Derivation dependencies. On the 38.8% of pool rationales that are not linear chains, permuted both ways so the arms are matched, reordering within the dependency structure changes accuracy by -0.99 pp and violating it by -2.43 pp. The arms also differ in step displacement, preventing attribution of the gap to graph structure (Appendix I).

Presentation order. We included a presentation-order condition as a calibration based on established prompt-order sensitivity (Lu et al., 2022). None of its four effects is significant. The changes are $+0.76$ pp ($p = 0.480$) and -0.30 pp ($p = 0.817$) on GSM8K, and -2.50 pp ($p = 0.077$) and -0.90 pp ($p = 0.567$) on MathQA. The same items show significant step-reordering effects of

-2.43 pp ($p = 0.008$) and -2.05 pp ($p = 0.011$). These comparisons demonstrate sensitivity to some permutations but do not establish presentation-order equivalence. The query-assignment analysis uses its separate equivalence test and per-condition MDE (Appendix H).

6 Practical Implications

Pool measurement. Near-duplicate density can be computed with one embedding pass and no language-model generation. Reporting the fraction of queries above a specified nearest-neighbour similarity threshold helps characterise the pool used in a retrieval evaluation. At $\tau = 0.90$, the fractions are 2.4% on GSM8K and 79.8% on MathQA. The retrieval gains observed on MathQA do not reproduce on GSM8K (Appendix G).

Problem-rationale alignment. Prompt builders should verify that each rationale remains paired with its source problem. Misalignment produces accuracy changes of -22.40 to -45.26 pp in the math-tuned 1.5B conditions while leaving the retrieved demonstrations unchanged. Recall@ k , MRR and mean query-demonstration similarity cannot detect this error because they depend on the retrieved set. Table 5 confirms that INTRA matches the unpermuted condition to four decimals on every similarity and nuisance measure.

7 Related Work

Demonstration selection. Similarity-based selection (Liu et al., 2022), learned retrieval (Rubin et al., 2022), complexity-based selection (Fu et al., 2023), clustering-based construction (Zhang et al., 2023) and computational-graph retrieval (Yang et al., 2024b) all report gains against random selection. Peng et al. (2024) find similarity-based selection does not reliably beat random and Zhang et al. (2023) report a GSM8K null. Near-duplicate availability reconciles these with the large MathQA gains of Fu et al. (2023) and Jie and Lu (2023), both of whom attribute the asymmetry to near-duplication; we measure and manipulate it. Its open-domain-QA counterpart is Lewis et al. (2021).

Demonstration content. Min et al. (2022) showed random labels barely hurt classification in-context learning; later work qualified this (Yoo et al., 2022; Kossen et al., 2024), and Pan et al.

(2023) separated task recognition from task learning. Wang et al. (2023) study relevance within demonstrations. We compare this with relevance to the test query under the same protocol.

Relevance controls. Liu et al. (2022) evaluate a farthest-neighbour condition that reduces accuracy by 15 points on NaturalQuestions, where a retrieved neighbour frequently contains the answer. This condition also changes the demonstration distribution. MathQA near-duplicates similarly provide query-specific information through shared solution templates. Prompt-order permutations (Lu et al., 2022) change positions within a selected set. Neither control isolates query assignment at a fixed retrieved-set distribution.

Mathematical NLP. Related work studies retrieval over mathematical content (Ferreira and Freitas, 2020) and permutations of semantic assignments (Williamson et al., 2025; Dias-Alexiou et al., 2025). Our control permutes the assignment of retrieved demonstration sets to queries.

8 Conclusion

Derangement separates the effect of query-specific matching from changes in the retrieved demonstration distribution. The pooled GSM8K assignment effect meets the pre-stated equivalence margin; on MathQA, matching accounts for the retrieval gain, which decreases after near-duplicate removal. Within-demonstration misalignment reduces accuracy on both corpora, with larger losses in math-tuned models and a strong association with excess copying of demonstration answers.

Limitations

The measurements run from 0.5B to 7B, greedy-decoded, at $k = 4$, on two English math-word-problem corpora. The INTRA result covers 7 models from 3 families, but OLMo-2-1B and Phi-3.5-mini are run on subsamples of 250 and 100 items against the 1319 of the others, and Phi-3.5-mini is instruction-tuned where the rest are base checkpoints, so its non-significant result is limited to that evaluation design. The assignment result itself is measured on the original three models only. On GSM8K the absolute retrieval gain is one to two points, so the decomposition there partitions a small quantity. The equivalence test bounds the assignment effect within its stated margin; MathQA provides a comparison with larger retrieval gains. The near-duplicate account rests on one pruning experiment that is partial and also changes candidate quality (residual similarity 0.887 against GSM8K’s 0.807, and it turns Δ_{pool} and Δ_{diff} negative); the density it manipulates is measured under 3 embeddings and 6 thresholds and across 6 corpora, but the manipulation itself was run on MathQA alone. The permutations are those the released annotations can express, so a demonstration property with no annotation cannot be deranged, the topology-split step permutation is available only for the 38.8% of rationales whose derivation is not a linear chain, and the intra-demonstration result describes systems that construct the problem-rationale pairing, such as self-generated exemplars, clustered pseudo-rationales (Zhang et al., 2023) and distillation corpora, rather than systems that inherit it intact from a corpus.

References

- Marah Abdin et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.
- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. *MathQA: Towards interpretable math word problem solving with operation-based formalisms*. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Carolina Dias-Alexiou, Edison Marrese-Taylor, and Yutaka Matsuo. 2025. *An in-depth human study of the mathematical reasoning abilities in large language models*. In *Proceedings of The 3rd Workshop on Mathematical Natural Language Processing (MathNLP 2025)*, pages 186–194, Suzhou, China. Association for Computational Linguistics.
- Deborah Ferreira and André Freitas. 2020. *Natural language premise selection: Finding supporting statements for mathematical text*. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2175–2182, Marseille, France. European Language Resources Association.
- Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2023. Complexity-based prompting for multi-step reasoning. In *International Conference on Learning Representations*.
- Zhanming Jie and Wei Lu. 2023. Leveraging training data in few-shot prompting for numerical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10518–10526.
- Jannik Kossen, Yarin Gal, and Tom Rainforth. 2024. In-context learning learns label relationships but is not conventional learning. In *International Conference on Learning Representations*. ArXiv:2307.12375.
- Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1000–1008.
- Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. *What makes good in-context examples for GPT-3? In Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.
- Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. *Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity*. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.
- Aman Madaan and Amir Yazdanbakhsh. 2022. Text and patterns: For effective chain of thought, it takes two to tango. *arXiv preprint arXiv:2209.07686*.
- Sewon Min, Xinxin Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. *Rethinking the role of demonstrations*:

What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What in-context learning “learns” in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*. ArXiv:2305.09731.

Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2024. Revisiting demonstration selection strategies in in-context learning. *arXiv preprint arXiv:2401.12087*.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. **Learning to retrieve prompts for in-context learning**. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, et al. 2025. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. 2023. **Towards understanding chain-of-thought prompting: An empirical study of what matters**. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, Toronto, Canada. Association for Computational Linguistics.

Dane A Williamson, Yangfeng Ji, and Matthew B. Dwyer. 2025. **Syntactic blind spots: How misalignment leads to LLMs’ mathematical errors**. In *Proceedings of The 3rd Workshop on Mathematical Natural Language Processing (MathNLP 2025)*, pages 1–14, Suzhou, China. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muenighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024a. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Xiaocong Yang, Jiacheng Lin, Ziqi Wang, and Chengxiang Zhai. 2024b. Learning by analogy: Enhancing few-shot prompting for math word problem solving with computational graph-based retrieval. *arXiv preprint arXiv:2411.16454*.

Kang Min Yoo, Junyeob Kim, Hyuhng Joon Kim, Hyunsoo Cho, Hwiyeol Jo, Sang-Woo Lee, Sang-goo Lee, and Taeuk Kim. 2022. Ground-truth labels matter: A deeper look into input-label demonstrations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. ArXiv:2205.12685.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023. Automatic chain of thought prompting in large language models. In *International Conference on Learning Representations*.

A Definitions of the Experimental Conditions

Table 2 defines every arm, and Figure 1 shows a real prompt fragment under INTRA beside its unpermuted counterpart.

arm	demonstrations shown for query q
rand	k drawn uniformly from the pool
r_true	top- k retrieved for q , most similar adjacent to q
r_der	top- k retrieved for $\sigma(q)$
r_derstrat	top- k retrieved for $\sigma_s(q)$, σ_s confined to q ’s step-count stratum
r_order	top- k for q , presented in permuted order
r_intra	top- k for q ; demonstration j ’s question shown with demonstration $\pi(j)$ ’s rationale and answer
r_steporder	top- k for q ; each rationale’s own step order permuted
r_numperm	top- k for q ; numeric literals inside each rationale deranged

Table 2: Arm definitions. σ is a derangement of the test set, π a derangement of the k demonstration slots.

B The Permutation Hierarchy with Exact p-values

Table 4 gives every arm of the GSM8K hierarchy for the math-tuned model under both retrievers, with the paired bootstrap interval, the exact-binomial McNemar p and the Holm-adjusted p , supplementing the accuracies in Table 1.

C Full Similarity and Nuisance Audit

Table 5 is the compact audit quoted in §4 and Table 6 the same audit over every arm. Both report, per arm, the mean prompt tokens and gold steps per demonstration and the mean query-demonstration embedding and graph similarity, confirming that the INTRA arm is matched to its unpermuted arm on all four.

UNPERMUTED (top- k retrieved for the query)

Question: In a basketball game, Cyrus made exactly eighty percent of the shots he attempted. He attempted twenty shots. How ...

Answer: Cyrus made $20 \times 80/100 = 16$ shots. So, he missed $20 - 16 = 4$ shots. The answer is 4.

Question: It's payday but Jebb has to pay 10% for the tax. If his pay is \$650, how much is his take-home pay?

Answer: Jebb has to pay $\$650 \times 10/100 = \$65$. Therefore, Jebb's take-home pay is $\$650 - \$65 = \$585$. The answer is 585.

INTRA (same demonstrations; rationales exchanged)

Question: In a basketball game, Cyrus made exactly eighty percent of the shots he attempted. He attempted twenty shots. How ...

Answer: Jebb has to pay $\$650 \times 10/100 = \$65$. Therefore, Jebb's take-home pay is $\$650 - \$65 = \$585$. The answer is 585.

Question: It's payday but Jebb has to pay 10% for the tax. If his pay is \$650, how much is his take-home pay?

Answer: Cyrus made $20 \times 80/100 = 16$ shots. So, he missed $20 - 16 = 4$ shots. The answer is 4.

Figure 1: Prompt fragment from the released artifacts, showing two of four demonstrations. Both blocks contain the same problems, rationales, answers and tokens. Under INTRA, each rationale appears beneath a different problem while remaining a valid solution to its source problem.

condition	n	Δ accuracy	copy excess
Qwen2.5-Math-7B, sem	1,319	-52.24	+43.1
Qwen2.5-Math-1.5B, struct	1,319	-45.26	+37.4
Qwen2.5-Math-1.5B, sem	1,319	-22.40	+11.1
OLMo-2-1B, sem	250	-9.60	+3.6
Phi-3.5-mini, sem	100	-4.00	+1.0
Qwen2.5-1.5B general, sem	1,319	-2.16	+0.0
Qwen2.5-0.5B general, sem	1,319	-1.06	+0.0
Qwen2.5-3B general, sem	1,319	+0.23	+0.0

Table 3: Signed accuracy change under INTRA and excess answer copying across 8 conditions and 3 model families on GSM8K. Both columns are in percentage points; copy excess is the copy rate minus its unpermuted value. Their Spearman correlation is -0.976 . Across 7 models, the correlation between unpermuted accuracy and the signed change is $+0.286$. The general Qwen checkpoints span 36.54–75.82 accuracy with zero copy excess.

D Near-Duplicate Density Under Three Embedders

The near-duplicate measurement of §4 uses one embedder at one threshold. Table 7 repeats it with 3 embedders over 6 thresholds on both corpora. GSM8K has no near-duplicate under any of them at any threshold from 0.95 up, and MiniLM-L6, which shares neither training nor dimension with the retriever of §3, finds 1.0% of GSM8K queries above 0.85 against bge-base’s 21.8%. On MathQA the density is 64.6% for bge-small and 44.5% for MiniLM-L6 at $\tau = 0.95$, against 64.6% for bge-base.

In all 18 embedder-threshold combinations,

queries with a pool neighbour above τ gain more from correct assignment than queries without one. For each embedder, the gap increases with τ . At $\tau = 0.99$, the gains are +26.1 pp and +3.9 pp for bge-base, and +26.4 pp and +3.9 pp for MiniLM-L6. Agreement with bge-base on the $\tau = 0.95$ flag has Jaccard similarity 0.903 for bge-small and 0.684 for MiniLM-L6.

Additional corpora. Density is a property of a (pool, queries) pair and needs no model, so it can be taken wherever a training split and a test split exist. Table 8 applies one embedder and one procedure to 6 math word-problem corpora. SVAMP has 54.7% of its queries above cosine 0.99 and 64.3% above 0.95, more than MathQA; ASDiv and AQuA have densities of 5.7% and 3.1% at 0.95; MAWPS and GSM8K have densities of 0.0% and 0.0%. These differences support reporting near-duplicate density when comparing retrieval results across corpora.

E A Stronger Retriever

§3’s SEM retriever is bge-base-en-v1.5. Table 9 repeats the MathQA decomposition with bge-large-en-v1.5 in its place, re-embedding the pool and the queries and re-ranking, with both conditions generated in one stage so the comparison uses conditions from the same generation stage.

Replacing bge-base with bge-large changes 74.3% of top-4 sets, with mean Jaccard similarity 0.571 between selections. The total gain decreases from +8.50 to +6.70 pp through changes in Δ_{pool} and Δ_{diff} , while Δ_{match} is +9.10 pp for both to two decimal places. This comparison is consistent with the near-duplicate account despite the differences in retrieved sets.

F The Decomposition, All Conditions

Table 10 applies Equation 1 to all eight corpus-retriever conditions, splitting each reported gain into its pool-reshaping, difficulty-coupling and query-matching parts.

G Observational Split and Dose-Response

MathQA queries whose nearest pool neighbour exceeds cosine 0.95 ($n = 646$) gain +18.11 pp from correct assignment; those below the threshold ($n = 354$) gain +2.54 pp. This approximately sevenfold difference holds within the same corpus, retriever, model and prompt format. The split is observational, and the subgroups differ in baseline

scale	permutation	Δ (pp)	95% CI (pp)	McNemar p	Holm p
SEM (embedding)					
outside	DER: query $\leftrightarrow$ set deranged	+1.40	[-0.53, +3.34]	0.503	1.000
outside	DERSTRAT: deranged within difficulty stratum	+0.00	[-1.93, +1.93]	0.894	1.000
outside	ORDER: presentation order permuted	+0.76	[-1.14, +2.65]	0.480	0.480
inside: steps	STEPTOPO: steps reordered within the derivation DAG	-0.99	[-2.43, +0.38]	0.203	0.406
inside: steps	STEPVIOL: steps reordered against the derivation DAG	-2.43	[-4.17, -0.68]	0.008	0.023
inside: steps	STEPORDER: rationale step order permuted	-8.72	[-10.99, -6.52]	$8.9 \cdot 10^{-14}$	$3.6 \cdot 10^{-13}$
inside: numbers	NUMPERM: rationale numbers deranged	-18.57	[-21.23, -16.07]	$2.1 \cdot 10^{-43}$	$1.0 \cdot 10^{-42}$
inside: pairing	INTRA: problem $\leftrightarrow$ rationale deranged in-set	-22.40	[-25.06, -19.90]	$5.5 \cdot 10^{-49}$	$1.6 \cdot 10^{-48}$
inside: pairing	INTRACORRECT: as INTRA: stated answer kept correct	-28.54	[-31.20, -25.93]	$1.6 \cdot 10^{-67}$	$6.4 \cdot 10^{-67}$
STRUCT (gold graph)					
outside	DER: query $\leftrightarrow$ set deranged	-2.01	[-4.09, -0.04]	0.122	0.244
outside	DERSTRAT: deranged within difficulty stratum	-1.40	[-3.41, +0.53]	0.507	0.507
outside	ORDER: presentation order permuted	-0.30	[-2.20, +1.59]	0.817	0.817
inside: steps	STEPTOPO: steps reordered within the derivation DAG	-2.05	[-3.56, -0.53]	0.011	0.022
inside: steps	STEPVIOL: steps reordered against the derivation DAG	-5.38	[-7.13, -3.64]	$3.0 \cdot 10^{-9}$	$8.9 \cdot 10^{-9}$
inside: steps	STEPORDER: rationale step order permuted	-10.54	[-12.89, -8.19]	$2.7 \cdot 10^{-18}$	$1.1 \cdot 10^{-17}$
inside: numbers	NUMPERM: rationale numbers deranged	-17.89	[-20.39, -15.39]	$3.2 \cdot 10^{-42}$	$1.6 \cdot 10^{-41}$
inside: pairing	INTRA: problem $\leftrightarrow$ rationale deranged in-set	-45.26	[-47.99, -42.61]	$1.4 \cdot 10^{-137}$	$4.3 \cdot 10^{-137}$
inside: pairing	INTRACORRECT: as INTRA: stated answer kept correct	-52.05	[-54.78, -49.39]	$1.2 \cdot 10^{-165}$	$5.0 \cdot 10^{-165}$

Table 4: The full permutation hierarchy on GSM8K, math-tuned model, both retrievers. Δ is against the unpermuted retrieved arm of the same retriever and stage. Intervals are paired bootstrap 95% over test items; McNemar is exact-binomial per replicate, reported at its least significant replicate; Holm is over the nine conditions of that retriever’s hierarchy.

arm	demo tok	demo steps	$\cos(q, d)$	$\text{graph}(q, d)$
rand0	201.1	3.18	0.5754	0.4179
struct_true	199.2	3.24	0.5848	0.9410
struct_derstrat0	199.2	3.24	0.5744	0.5990
struct_der0	199.2	3.24	0.5742	0.4250
sem_true	206.5	3.45	0.7887	0.4472
sem_intra0	206.5	3.45	0.7887	0.4472

Table 5: Realised query–demonstration similarity, GSM8K. Structural match falls from the unpermuted arm through the stratified derangement to the unrestricted one, which sits at the uniform-random level. The INTRA arm is identical to SEM_TRUE on every column by construction, since it contains the same demonstrations. Full table in Appendix C.

difficulty, with random-demonstration accuracies of 59.9 and 55.1. The pruning experiment in §4 provides the intervention estimate.

Assignment gains across quartiles of realised query-demonstration similarity are +10.2, +8.2, +11.4 and +20.6 pp. The sequence is not monotone, although the top quartile has approximately twice the gain of the others. The fitted slope is +50.8 pp per unit cosine on MathQA and −5.7 on GSM8K. GSM8K’s top quartile has mean similarity 0.851 and a gain of +1.36 pp, while MathQA’s bottom quartile has mean similarity 0.828 and a gain of +10.2 pp. Mean similarity alone therefore does not explain the cross-corpus difference. At $\tau = 0.90$, nearest-neighbour similarities exceed the threshold for 2.4% of GSM8K queries and 79.8% of MathQA queries.

H Presentation-Order Sensitivity

The step-reordering conditions detect effects of about two percentage points. MathQA’s order effect of −2.50 pp is larger but remains non-significant, limiting what can be inferred from that result. Lu et al. (2022) study classification tasks in which demonstration order changes the label sequence in the prompt. Our 4-shot mathematical chain-of-thought prompts have no corresponding label sequence, so the two studies evaluate different settings.

I Displacement Matching for the Derivation-DAG Conditions

A rationale’s dependency graph specifies which results each step uses. Topology-preserving reorderings place each step after its dependencies; violating reorderings place at least one step before a required result. A linear chain permits no non-trivial topology-preserving reordering, limiting this

arm	demo tok	demo steps	$\cos(q, d)$	$\text{graph}(q, d)$
rand0	201.1	3.18	0.5754	0.4179
rand1	200.7	3.18	0.5738	0.4184
sem_der0	206.5	3.45	0.6039	0.4254
sem_der1	206.5	3.45	0.6024	0.4227
sem_derstrat0	206.5	3.45	0.6031	0.4254
sem_derstrat1	206.5	3.45	0.6036	0.4257
sem_intra0	206.5	3.45	0.7887	0.4472
sem_intra1	206.5	3.45	0.7887	0.4472
sem_intracorrect0	206.5	3.45	0.7887	0.4472
sem_intracorrect1	206.5	3.45	0.7887	0.4472
sem_true	206.5	3.45	0.7887	0.4472
struct_der0	199.2	3.24	0.5742	0.4250
struct_der1	199.2	3.24	0.5737	0.4247
struct_derstrat0	199.2	3.24	0.5744	0.5990
struct_derstrat1	199.2	3.24	0.5749	0.6050
struct_intra0	199.2	3.24	0.5848	0.9410
struct_intra1	199.2	3.24	0.5848	0.9410
struct_intracorrect0	199.2	3.24	0.5848	0.9410
struct_intracorrect1	199.2	3.24	0.5848	0.9410
struct_true	199.2	3.24	0.5848	0.9410

Table 6: Nuisance matching and realised similarity, every arm, GSM8K. Columns are means over test items: prompt tokens per demonstration, gold steps per demonstration, query–demonstration embedding cosine, and query–demonstration graph similarity. Higher is more similar; both similarities lie in $[0, 1]$.

corpus	embedder	.85	.90	.92	.95	.97	.99
GSM8K	bge-base	21.8	2.4	0.8	0.0	0.0	0.0
	bge-small	13.7	1.1	0.3	0.0	0.0	0.0
	MiniLM-L6	1.0	0.0	0.0	0.0	0.0	0.0
MathQA	bge-base	91.0	79.8	73.9	64.6	52.6	26.4
	bge-small	92.6	81.6	75.9	64.6	51.8	23.8
	MiniLM-L6	69.0	59.9	55.5	44.5	32.7	12.1

Table 7: Near-duplicate density, the share of test queries whose nearest pool neighbour clears the cosine threshold, under three embedders. bge-base is the retriever of §3; MiniLM-L6 is a different family with different training and 384 dimensions. GSM8K holds no near-duplicate under any embedder at any threshold from 0.95 up.

comparison to 38.8% of pool rationales and 42.9% of the demonstrations shown. Each rationale is permuted in both conditions or neither, leaving the prompts byte-identical wherever the comparison is unavailable. Violating the dependency structure reduces accuracy by an additional 1.4 pp. Mean step displacement is 1.59 positions in that condition and 0.81 in the topology-preserving condition. Sensitivity to displacement could therefore explain the ordering without sensitivity to graph structure. A displacement-matched comparison is required to separate these effects.

corpus	pool	test	.85	.90	.95	.99
SVAMP	700	300	85.3	73.7	64.3	54.7
MathQA	8,000	1,000	69.0	59.9	44.5	12.1
ASDiv	2,075	230	28.7	14.3	5.7	4.3
AQuA	12,000	254	9.1	3.5	3.1	2.4
MAWPS	1,089	520	1.0	0.0	0.0	0.0
GSM8K	7,473	1,319	1.0	0.0	0.0	0.0

Table 8: Near-duplicate density across math word-problem corpora using all-MiniLM-L6-v2. Entries give the percentage of test queries whose nearest training-pool neighbour exceeds the column’s cosine threshold. GSM8K and MathQA repeat the MiniLM-L6 measurements in Table 7.

J Retrieved-Pool Concentration

Table 11 shows concentration in the retrieved demonstration distributions, with no significant Δ_{pool} effect in the four conditions considered here. Graph matching is more concentrated because the pool contains 2100 distinct gold operator signatures. Queries with the same signature can therefore receive the same four examples from a small equivalence class. Fixed random tie-breaking changes the median selected pool index from 717 to 3746 while leaving the Gini coefficient at 0.866. Embedding retrieval is less concentrated but raises mean demonstration step count from 3.18 to 3.45 and lowers the share of short (≤ 2 -step) demonstrations from 0.354 to 0.274.

retriever	Δ_{total}	Δ_{pool}	Δ_{diff}	Δ_{match}
bge-base-en-v1.5	+8.50	+0.40	−1.00	+9.10
bge-large-en-v1.5	+6.70	−0.70	−1.70	+9.10

Table 9: MathQA decomposition for two retrievers using Qwen2.5-Math-1.5B, with both evaluated in one generation stage. The selected top-4 sets differ for 74.3% of queries, with mean Jaccard similarity 0.571. Both matching terms equal +9.10 pp to two decimal places and are significant at $p < 10^{-4}$. All effects are in percentage points; $n = 1000$.

K Output Characteristics Under Permutation

Under INTRA, the model produces a well-formed numeric answer in 100.0% of cases, with mean generation length 86.0 tokens compared with 89.2 in the unpermuted condition. The fraction of predictions equal to a printed demonstration answer rises from 6.4% to 17.5% for embedding retrieval and to 43.3% for structural retrieval. Structural retrieval selects examples with similar derivations, which may make their answers more likely to be copied. Table 12 reports accuracy, well-formed answer rate, copy rate and generation length for each condition.

corpus	retr.	Δ_{total}	Δ_{pool}	Δ_{diff}	Δ_{match}
GSM8K, general 1.5B	sem	-0.30	+0.30	-0.23	-0.38
GSM8K, general 1.5B	struct	+0.91	+0.61	+0.23	+0.08
GSM8K, math 1.5B	sem	-0.53	+0.87	-1.40	+0.00
GSM8K, math 1.5B	struct	+2.12	+0.11	+0.61	+1.40
GSM8K, math 7B	sem	-0.08	-1.10	+0.80	+0.23
GSM8K, math 7B	struct	+1.14	+0.76	+0.57	-0.19
MathQA, math 1.5B	sem	+10.20	-1.65	-0.75	+12.60
MathQA, math 1.5B	struct	+4.10	+0.50	-1.80	+5.40

Table 10: The decomposition of Equation 1. The identity is exact; the three terms are rounded independently, so a printed row can differ from its printed Δ_{total} by 0.01. All four terms are in percentage points.

population	unique	Gini	mean steps	≤ 2 steps (share)	top 1% share
whole pool	7473	—	3.17	0.358	—
uniform draw	3733	0.609	3.18	0.354	—
SEM	2941	0.729	3.45	0.274	9.7%
STRUCT	1968	0.866	3.24	0.334	26.1%

Table 11: Demonstration selection statistics on GSM8K, computed from retrieval indices without model generation.

arm	acc (%)	well-formed	copies a demo	gen. tok
rand0	73.46	100.0%	3.7%	90.3
rand1	75.44	100.0%	3.9%	91.7
sem_der0	75.89	100.0%	3.3%	90.2
sem_der1	74.75	100.0%	3.6%	91.0
sem_derstrat0	74.15	100.0%	4.8%	89.9
sem_derstrat1	73.69	100.0%	3.1%	89.6
sem_intra0	50.95	100.0%	18.2%	86.0
sem_intra1	52.08	100.0%	16.8%	84.9
sem_intracorrect0	46.10	100.0%	21.7%	86.5
sem_intracorrect1	44.66	100.0%	22.6%	85.4
sem_true	73.92	100.0%	6.4%	89.2
struct_der0	74.45	100.0%	3.7%	91.5
struct_der1	74.68	99.9%	3.1%	92.0
struct_derstrat0	74.60	100.0%	3.0%	94.4
struct_derstrat1	75.74	100.0%	3.0%	92.5
struct_intra0	31.99	100.0%	42.9%	90.5
struct_intra1	30.63	100.0%	43.7%	89.0
struct_intracorrect0	25.09	100.0%	50.3%	90.1
struct_intracorrect1	23.96	99.9%	52.0%	90.8
struct_true	76.57	100.0%	5.9%	91.3

Table 12: Failure mode per arm on GSM8K. ‘Copies a demo’ is the fraction of predictions equal to one of the answers printed in the demonstration block.