DO STRUCTURAL MODIFICATIONS ISOLATE COMPOSITIONAL REASONING IN KNOWLEDGE GRAPH COMPLETION?

Anonymous authors
Paper under double-blind review

ABSTRACT

Knowledge graph completion models are often credited with compositional reasoning: answering a query by chaining relations along a path. A natural test is a structural modification: the training graph is rewired so that its two-hop paths are destroyed, and a drop in accuracy would be read as evidence that the model used those paths. In this paper, we ask whether this modification isolates composition and find that it does not. The rewiring does remove two-hop rules, as a symbolic rule miner confirms, but it also rewrites up to a quarter of all training triples. To separate the two changes, we build a control graph that is rewired by the same amount but keeps the amount of two-hop structure unchanged. For two path-based models on two benchmarks, including the ULTRA foundation model, this control alone reproduces about three quarters of the drop in MRR. In a grid on CoDEx-S at a fixed edit budget and below the modification's closure loss, the drop follows symmetric and reciprocal triples more closely than two-hop structure. Models with no mechanism for following paths lose more than half as much accuracy as ULTRA under the same rewiring. What remains depends on choices the test leaves open: its size changes with the evaluation metric, and its sign can flip with how the modified graphs are built. A drop under this modification therefore cannot, on its own, show that a model composes. We propose that such studies report a control rewired by the same amount, results on queries with no two-hop path, how the modified graphs were built, and more than one metric. We release our constructions and controls.

1 INTRODUCTION

Knowledge graph completion models, and knowledge graph foundation models in particular (Galkin et al., 2024; Huang et al., 2025; 2026), are often credited with compositional reasoning: answering a query $(h, r, ?)$ by chaining relations along a path. The claim shapes how these models are described and how new architectures are compared.

A natural way to test the claim is a structural modification of the training graph. The graph is rewired so that its two-hop paths are destroyed, the model is measured again, and a drop in accuracy is read as evidence that the model was using those paths. The test is attractive: it is cheap, it needs no access to model internals, and it applies to any model. We read twelve studies that draw conclusions about path structure, and none runs it yet (Table 7). We therefore examine it before it is adopted rather than after.

However, the test has a gap that is visible before any model is trained. A drop points to a mechanism only if the modification changed that mechanism and nothing else that matters. The rewiring that removes a third of the closed two-hop structure of CoDEx-S also rewrites a quarter of all its training triples (24.81%; Figure 1). The two changes arrive together, and a comparison of two graphs cannot say how the drop divides between them. The same problem is known for interventions on model internals (Makelov et al., 2024; Zhang & Nanda, 2024); to our knowledge it has not been examined for interventions on the training graph.

In this paper, we ask whether a drop under this modification shows compositional reasoning, and find that it does not. Across $1{,}376$ runs on seven graphs, we add the controls the two-graph test leaves

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.

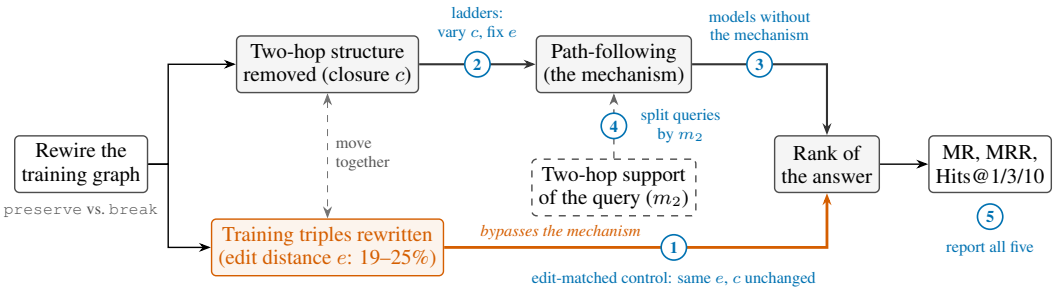


Figure 1: **Rewiring removes two-hop structure and rewrites training triples at the same time, so a drop in accuracy alone cannot be attributed to path-following.** The rewritten triples (orange) can lower the rank of the answer without passing through the mechanism the claim is about. Numbered circles (blue) mark where each of our controls enters; Figure 3 follows one real query per benchmark through the arms.

out. The central one is a control graph rewired by the same amount, with its two-hop structure kept at the clean level, so that only the rewiring differs. We also apply the modification to models with no query-conditioned traversal mechanism at inference. We split the test queries by their two-hop support before the modification, and we rebuild the modified graphs under rules we can state.

Our contributions are as follows.

- **A confound, and a control that exposes it** (Sections 3 and 4.1). The control alone reproduces about three quarters of the MRR drop (76.0% on CoDEx-S and 78.7% on FB15k-237 for frozen ULTRA-3g), and models with no traversal mechanism lose more than half as much as ULTRA (Section 4). The share is what the control reproduces, not a decomposition of the drop: it is larger at Hits@1 than at Hits@10, and it moves with which edges are rewired.
- **The answer depends on how the modified graphs are built** (Sections 4.2 and 5). What remains after matching is positive in the released graphs and negative under a documented acceptance rule at a comparable edit budget; in the released graphs the drop is also absent on the almost half of CoDEx-S test queries with no two-hop path, and reverses on FB15k-237. Crossing closure loss with reciprocity at a fixed edit budget, over 45 constructions and four families, the closure coefficient is positive and small up to 23% closure loss, and symmetric-pair loss carries the fit.
- **A checklist and a release** (Section 6). Six things a result of this kind must report, and the code and per-query ranks behind every table.

Our claim is that this evidence cannot tell whether these models compose, not that they do not. Matched contrasts control the edit count; they do not by themselves identify composition, and slopes and strata are descriptive. The checks that would constrain it are cheap, and we hope they become part of how such claims are made.

2 Related Work

Auditing how knowledge graph completion is evaluated. The field has a strong habit of checking its own measurements. Ruffinelli et al. (2020) show that training choices can reverse the ranking of architectures, and Akrami et al. (2020) remove leaked inverse and duplicate relations. Shomer et al. (2025) rebuild an inductive benchmark to remove a shortest-path shortcut, and Gul & Bhat (2026) show that rank statistics disagree across published results. These studies change the benchmark, the protocol or the test queries and keep the metric fixed. We keep the benchmark and the model fixed, and ask whether two training graphs differ only in the quantity a claim is about.

Interventions and controls on graphs. Li et al. (2023a) replace the message-passing graph with a random graph of the same edge count. It is the one prior intervention on the graph with a matched control, and it matches the number of edges rather than a measured structural dose. Yang et al. (2017) and Sadeghian et al. (2019) put an embedding model through the same data split they use to argue for rule-based reasoning, a non-compositional control under a different manipulation. Gregucci et al.

(2026) locate an effect by splitting test queries by structural exposure rather than by intervening. No study we found runs an edit-matched control and a non-compositional control together, or asks whether such a control depends on how the modified graphs were built (Table 7).

The same problem for interventions on model internals. In interpretability, an intervention can reach the outcome through a pathway other than the one it is credited to (Makelov et al., 2024), and the metric used to read it can change the conclusion (Zhang & Nanda, 2024). Causal claims there are argued to need their identification assumptions stated (Lin & Liu, 2026; Joshi et al., 2026). We bring these concerns to interventions on the training graph, where the second pathway can be measured directly. The twelve studies, and the criterion behind each entry, are in Section A.2.

3 PRELIMINARIES

Task and rank statistics. A knowledge graph is a set of triples (h, r, t). For a test query $(h, r, ?)$ a model ranks every candidate tail under the filtered protocol, and likewise for $(?, r, t)$. We report mean rank (MR), mean reciprocal rank (MRR) and Hits@1, 3 and 10; they read different parts of the rank distribution, so the choice among them is part of the result. Every effect is a within-model contrast between two training graphs, never a comparison of absolute accuracy across models. Two evaluation protocols exist, full test and a fixed 1,200-query subset held identical across graphs; they are never mixed in a row (Section B.2).

The structural modification. From each training graph we build four arms by within-relation double-edge swaps: degree-preserving randomisation (Maslov & Sneppen, 2002) applied one relation at a time, so every entity's per-relation degree and every relation's head and tail multisets are preserved exactly (Section B.1). `clean` is the original graph; `preserve` is rewired to keep two-hop support, `break` to destroy it, and `random` without regard to it, which still lowers closure and so is not a neutral control. The conventional contrast is `preserve` minus `break`. The claim it tests is narrow: that accuracy depends on two-hop, path-mediated inference over the training graph, which is what we mean by composition. The graphs are CoDEx-S and CoDEx-M (Safavi & Koutra, 2020), FB15k-237 (Toutanova & Chen, 2015), NELL-995 (Xiong et al., 2017), WN18RR (Dettmers et al., 2018), and `codex_m_E`, an inductive CoDEx-M split built by Shomer et al. (2025) to remove a shortest-path shortcut; a seventh, `hetionet_E`, is used only in a cross-architecture replication (Section C.5). A swap can create a triple that coincides with a test triple; removing every such query from both arms changes no sign and moves no contrast by more than 0.012 on MRR or any Hits level (Table 29).

Closure and edit distance. Closure, the number of training triples in at least one closed two-path, is the quantity the test means to change. Edit distance, the fraction of training triples that differ from `clean`, also changes, and the test does not control it. On CoDEx-S `preserve` displaces 13.14% and `break` 24.81%, so `break` applies 1.89× as many edits as `preserve` (Table 9). We call these the two doses. A third quantity the rewiring moves, the shortest-path shortcut of Shomer et al. (2025), is measured and does not drive the result (Section C.5). A fourth is not matched by any arm: the swap destroys 24% of CoDEx-S's symmetric training triples (Section D.3); Section 5 varies it independently of closure and measures its effect.

Limitation of the two-graph comparison.

Remark 1 (Why a second arm does not settle it)**.** *An intervention moving a target quantity c and a second quantity e together yields, from a two-arm contrast, only the directional derivative along the operator's path; recovering ∂c at fixed e needs a third arm.*

This is the chain rule, not a theorem: we claim that the design as run does not *estimate* the compositional effect, not that it cannot be estimated in principle. The obstruction is measurable. `preserve` and `break` alone leave the two doses perfectly collinear (variance inflation 1.44 across seven CoDEx-S arms, 1.01 across the edit-matched ladder, where separate conditional slopes can be fitted; Table 20). Table 1 fits them and the edit slope exceeds the closure slope in every one of ten fits, at a ratio that is not stable; clustering on the arm widens ULTRA's closure interval to include zero (Section D.2), so we read it as descriptive and rest no claim on it.

Table 1: **The edit slope exceeds the closure slope in every fit.** MRR change per ten points of edit distance and per ten points of two-path closure, fitted where each dose varies independently. Each fit uses five arm means, three seeds each on the full test set: `clean`, `preserve`, `break` and `random` from the fine-tuned campaign and `matched` from a second run of the same protocol. The ratio runs from 1.83× to 5.77× here and 1.66× to 5.12× on individual runs (Table 20). Descriptive: runs are treated as independent; clustered intervals are in Section D.2.

Graph	Model	per 10 pts edits	per 10 pts closure	edit / closure	R^2
CoDEx-S	ULTRA	−0.0679	**−0.0192**	3.54×	0.943
CoDEx-S	ComplEx	−0.0363	**−0.0073**	4.97×	0.955
FB15k-237	ULTRA	−0.0256	**−0.0051**	5.02×	1.000
FB15k-237	ComplEx	−0.0126	**−0.0069**	1.83×	0.994
CoDEx-S	AnyBURL	−0.0670	**−0.0136**	4.93×	0.956
FB15k-237	AnyBURL	−0.0254	**−0.0044**	5.77×	0.987

Five arms per fit; both doses measured on the produced graph, not targeted.

Added controls. The Edit-Matched Arm (`matched`) is rewired to `break`'s edit count with closure held at the clean value (24.84% edits against 24.81%, closure drift −0.52%). It supplies the comparison unit the test lacks: a rule that ignores closure never reaches `break`'s closure loss, so `break` has no untargeted counterpart (positivity fails; Section D.3). The Edit-Matched Closure Ladder adds two rungs between `matched` and `break` at the same edit budget, so closure varies alone. The Closure-Removal Ladder on `codex_m_E` drives closure from 13.7% of the graph to 1.1% at a fixed 25% edit budget. The acceptance rule, not the closure target, sets the edit budget: a strictly closure-decreasing rule reaches `break`'s closure loss at 11.20% edits against 24.81%. Only `matched` was rebuilt at five construction seeds. `preserve`, `break` and `random` are single realisations, because their builder is not recoverable from our run tree, so intervals over constructions carry no `break`-side variance.

Models. ULTRA (Galkin et al., 2024), a path-based knowledge graph foundation model (the ULTRA-3g checkpoint), is used frozen, fine-tuned per arm, or trained from scratch. ComplEx (Trouillon et al., 2016), DistMult (Yang et al., 2015) and CompGCN (Vashishth et al., 2020) perform no query-conditioned path traversal at inference (the two bilinear scorers cannot represent the composition pattern (Sun et al., 2019)), and serve as controls for that mechanism. AnyBURL (Meilicke et al., 2019) is a symbolic rule miner and Personalized PageRank a relation-blind structural baseline; MOTIF and HYPER (Huang et al., 2025; 2026) appear in replications (Section C.5). A dash in any table means not recorded, never zero, and cells excluded from analysis are listed, with the reason, in a register (Section F.2).

Hypotheses. Write ρ for the share of the conventional contrast that the Edit-Matched Arm reproduces and κ for the MRR difference between that arm and `break` under a given construction procedure (the matched-control contrast). If the drop measured composition, two things would be expected; if a matched control were independent of how the graphs are built, a third would.

- H1 The control reproduces little of the drop ($\rho \approx 0$), and about equally at Hits@1 and Hits@10.
- H2 Test queries with no two-hop path show no drop, and the drop grows with two-hop support.
- H3 κ is the same under any acceptance rule that reaches the same closure target.

Sections 4 and 5 test these. Each could have come out in favour of the conventional reading, and we report intervals wherever the design supplies them.

4 A PERFORMANCE DROP DOES NOT ISOLATE COMPOSITIONAL REASONING

This section tests H1 and H2 with three controls: a graph rewired by the same amount with closure intact (Section 4.1), models with no traversal mechanism (below), and a split of the test queries by two-hop support (Section 4.2). A fourth, a ladder that removes closure at a fixed edit budget, is in Section D.1: at a 2–11% edit budget MRR tracks the edit share, not closure (Figure 7).

Models with no query-conditioned traversal mechanism give a first check needing no new graph. We find that ComplEx, DistMult and CompGCN lose between half and three fifths of ULTRA's MRR

Table 2: **How much of the drop the Edit-Matched Arm reproduces depends on which rank statistic reports it.** The Edit-Matched Arm preserves closure while matching `break` on edit distance. *Panel A*: arm levels for ULTRA on CoDEx-S. *Panel B*: the share (`preserve` − `matched`)/(`preserve` − `break`) for every family and graph with a matched arm. Statistics are rows because the statistic is what varies in the finding. On CoDEx-S every column falls as k rises; on FB15k-237, where the conventional contrast is far smaller (Table 27), the ordering holds for AnyBURL, is flat for ULTRA and reverses for ComplEx.

Panel A: arm levels, ULTRA on CoDEx-S, three seeds

Metric	`clean`	`preserve`	`matched`	`break`	conventional	reproduced	**share**
H@1 ↑	0.3816	0.3431	0.1920	0.1624	+0.1807	+0.1511	**83.6**%
H@3 ↑	0.5288	0.4892	0.3667	0.2939	+0.1953	+0.1225	**62.7**%
H@10 ↑	0.6780	0.6281	0.5600	0.4747	+0.1534	+0.0681	**44.4**%
MRR ↑	0.4823	0.4415	0.3128	0.2638	+0.1777	+0.1287	**72.4**%
MR ↓	39.45	58.10	73.99	101.16	+43.05	+15.89	**36.9**%

Panel B: share reproduced, every family with a matched arm

Metric	CoDEx-S			FB15k-237		
	ULTRA	ComplEx	AnyBURL	ULTRA	ComplEx	AnyBURL
H@1 ↑	83.6%	92.1%	88.1%	63.3%	34.4%	84.7%
H@3 ↑	62.7%	76.2%	73.3%	60.1%	50.1%	70.4%
H@10 ↑	44.4%	59.3%	64.4%	61.9%	57.0%	65.3%
MRR ↑	72.4%	77.9%	80.3%	61.8%	46.1%	75.1%
MR ↓	36.9%	86.1%[†]	—	74.9%	54.5%	—

Full-test protocol, three seeds; MR sign inverted so that positive always means worse. AnyBURL mean rank is undefined by construction (top-10 truncation). Arm levels come from the fine-tuned campaign: `clean`, `preserve` and `break` from its main run and `matched` from a second run of the same protocol, both on seeds 1–3 and scored on the full test set. AnyBURL's arms come from its own rule-mining runs (three mining seeds). AnyBURL's share is 75.5–79.1% (SD 1.2 pts) across five constructions of the matched arm; for frozen ULTRA-3g the CoDEx-S share is 72.7% on the 1,200-query protocol and 76.0% on full test (the five-construction mean), and 72.2–78.2% across those constructions, so the spread down a column is not construction noise. [†]Two seeds (Section F.1).

drop under the same modification, summed over four graphs and three seeds (Section C.1). The per-graph share varies widely, so we read this as an ordering rather than an estimate. Models that cannot use the mechanism still lose more than half of what the mechanism is credited with.

4.1 An edit-matched control reproduces most of the drop

An arm that keeps closure intact but pays `break`'s edit budget would behave like `preserve` if the drop tracked closure alone (H1). It does not (Table 2). For frozen ULTRA-3g on full test, the Edit-Matched Arm reproduces 76.0% of the conventional contrast on CoDEx-S and 78.7% on FB15k-237. Each is a mean over five constructions of the matched arm, with 95% two-level intervals $[68.5, 82.9]$ and $[73.7, 84.2]$. MOTIF-3g gives 76.1% and 72.8% $[66.7, 79.3]$ on the same arms, so no cell falls below 72.8%. A rule scorer composing only through two-hop paths gives 35.6% and 35.7%, against 76.0% and 78.7% for frozen ULTRA on the same arms (Section C.7). Across every family and graph with a matched arm (Panel B, Figure 5), the share runs from 46.1% (ComplEx, FB15k-237) to 80.3% (AnyBURL, CoDEx-S).

These shares are conditional on the released construction. We also rebuilt both arms under documented rules at a comparable budget. There the conventional contrast is 2.2 times smaller (+0.048 against +0.107 MRR, against the same documented `preserve`), and the Edit-Matched Arm reproduces 121% $[98, 156]$ of it. Once both arms are built by rules we can state, the control reproduces all of the drop on CoDEx-S.

The share is not one number, and that is part of the result (H1). For frozen ULTRA-3g it falls from 87.0% at Hits@1 to 46.6% at Hits@10 on CoDEx-S, and from 81.9% to 69.9% on FB15k-237. MOTIF-3g falls the same way, and so do the fine-tuned families of Table 2 on CoDEx-S. On FB15k-237 the fine-tuned ordering is flat for ULTRA and reverses for ComplEx. What the control reproduces is concentrated at the head of the ranking, and what it does not is concentrated in recall. The share is therefore large at Hits@1 and small on mean rank, and in neither case a fraction of the

Table 3: **The reported aggregate averages a negative contrast on 45% of the test set with a positive one on the rest. Two-Hop Exposure Stratification** of the edit-matched contrast `matched − break` by $m_2(q)$, the number of entities adjacent to both head and tail on the *clean* arm, a pre-treatment covariate. **On the 826 queries with no support to remove the matched arm is worse on every statistic**; from $m_2 \geq 2$ every statistic favours it. Bold: 95% bootstrap interval excludes zero.

m_2	n	% of test	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑	MR ↓	MRR 95% CI
aggregate	1,828	100.0%	**+0.0365**	**+0.0200**	**+0.0583**	**+0.0703**	+3.55	[+0.026,+0.047]
0	826	45.2%	**−0.0220**	**−0.0266**	−0.0139	−0.0121	**+16.47**	[-0.034,-0.010]
1	254	13.9%	+0.0207	+0.0098	**+0.0394**	+0.0394	−8.83	[-0.004,+0.046]
2–5	248	13.6%	**+0.0499**	**+0.0403**	**+0.0504**	**+0.0907**	−1.54	[+0.019,+0.082]
6–20	214	11.7%	**+0.1061**	**+0.0631**	**+0.1589**	**+0.1986**	**−10.86**	[+0.072,+0.140]
21+	286	15.6%	**+0.1557**	**+0.1136**	**+0.2150**	**+0.2220**	**−7.58**	[+0.125,+0.188]
gradient (21+ − 0)			+0.1777	+0.1402	+0.2289	+0.2341	−24.05	

CoDEx-S, frozen ULTRA-3g, 1,828 test queries, single seed; 2,000 bootstrap draws over queries; per-query score is the mean of the head-side and tail-side statistic. MRR point estimates reproduce Table 26; all five statistics with intervals, and the conventional `preserve − break` version, are in Table 21.

Table 4: Paired contrasts in MRR by pre-treatment two-hop support m_2, on both graphs and for both contrasts. Frozen ULTRA-3g, full test, 95% percentile bootstrap over queries (2,000 draws); the matched column is one construction of the matched arm (five-construction means are in Table 23; intervals are in Tables 21 and 22, except for the FB15k-237 conventional column). n counts test triples. The conventional contrast is absent at $m_2 = 0$ on CoDEx-S and negative on FB15k-237, and the edit-matched arm is worse than `break` there on both graphs.

CoDEx-S				FB15k-237			
m_2	n	conventional	matched	m_2	n	conventional	matched
0	826	−0.010	−0.022	0	5,352	−0.018	−0.027
1	254	+0.088	+0.021	1	4,136	+0.029	+0.006
2–5	248	+0.178	+0.050	2–5	6,840	+0.073	+0.027
6–20	214	+0.338	+0.106	6–20	3,642	+0.086	+0.030
21+	286	+0.517	+0.156	21+	496	+0.075	+0.028
gradient		+0.527	+0.178	gradient		+0.094	+0.054

drop caused by edits (Section C.2.1). Matching the edit count also does not fix which edges move. Two arms at one budget, with closure matched to 0.16% on both counters, differ by 12.8 points in this share, against 1.15 points of construction noise (Section C.3.4). The share is therefore what an edit-matched control reproduces, not a decomposition of the drop. It also moves with a model's reach, lowest where the operator acts: a MINERVA-style walker gives 77.9/51.7/83.7/71.9% at one to four hops on CoDEx-S, the two-hop value separating from all others over five constructions, and 23.3% against 62.9% at two and three hops on FB15k-237 (Section C.7).

4.2 The drop is absent on queries with no two-hop path

For a test query $q = (h, r, t)$, let $m_2(q)$ be the number of entities adjacent to both h and t in the *clean* training graph, ignoring direction and relation. The closure counter, by contrast, counts *directed* two-paths. Because m_2 is measured before any modification, splitting by it is effect modification by a pre-treatment covariate (VanderWeele, 2009), not conditioning on a consequence of the treatment. A query with $m_2 = 0$ has no two-hop path for the modification to remove; on CoDEx-S that is 826 of 1,828 test queries, 45.2%. The measure is specific: $m_2 = 0$ says there is no two-hop path, not that no compositional information reaches the answer.

If the drop measured composition, it should be near zero on these queries and larger where there is more support (H2). In the released graphs we observe three things (Table 3, Table 4, Figure 2). (i) The conventional contrast is absent at $m_2 = 0$ on CoDEx-S, as H2 predicts, and negative on FB15k-237. (ii) The edit-matched contrast is *negative* there. On CoDEx-S the arm that preserves closure ranks these queries worse than `break`, by 0.0220 MRR $[-0.034, -0.010]$ and 0.0266 Hits@1, and it is worse on all five statistics, three of which exclude zero. The residual persists across four sensitivity checks: degree (Section C.3.3), closure created by the construction (61 of 826 queries;

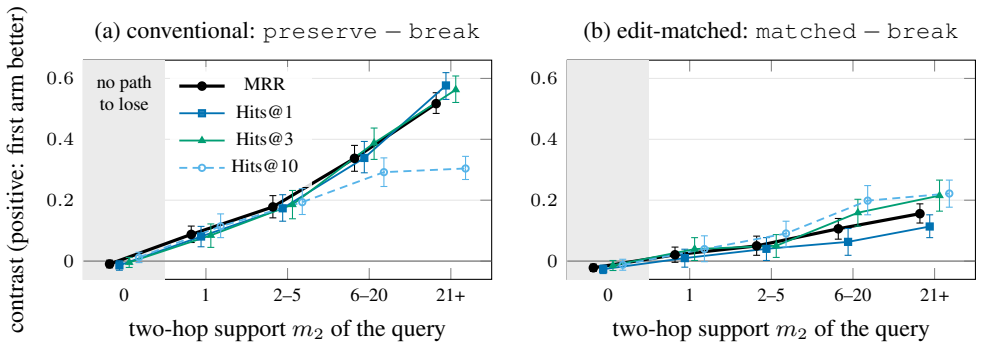


Figure 2: **Once edits are matched, the contrast is smaller in every stratum with two-hop support, and it is negative on queries with no two-hop path.** CoDEx-S, frozen ULTRA-3g; per-stratum contrasts from Table 21 with 95% bootstrap intervals, both panels on the same axis. The shaded stratum ($m_2 = 0$) holds 45% of the test set: there the edit-matched contrast is below zero on all four statistics and its interval excludes zero on MRR and Hits@1, while every conventional interval shown covers zero (MR, not plotted, excludes it). We claim only this sign change; the ordering among the supported strata does not replicate across constructions or graphs.

Table 5: The matched-control contrast $\kappa = f(\text{matched}) - f(\text{break})$ in MRR, by construction procedure, at one target closure per graph (CoDEx-S 33.2% of closed triples destroyed; FB15k-237 26.8%). Frozen ULTRA-3g, full test, five constructions per row, 95% t-intervals. The rows marked released use the released arms, whose acceptance rule is not identified; every other row is a rule we state; λ is the probability with which a closure-neutral swap is accepted. The last two columns read each row against a *documented* `preserve` (closure pinned, `preserve`'s own budget, paired by seed), so both arms follow stated rules; ρ is omitted where that contrast is indistinguishable from zero.

				vs. documented `preserve`	
Graph	Procedure	Edits	κ (MRR)	Δ_{conv}	ρ
CoDEx-S	released (rule unidentified)	24.81%	+0.0366 [+0.0321, +0.0412]	+0.107	66%
	strict decrease	11.45%	+0.0031 [−0.0001, +0.0063]	−0.007	—
	$\lambda = 0.05$	15.89%	−0.0039 [−0.0084, +0.0007]	+0.016	—
	$\lambda = 0.1$	18.77%	−0.0055 [−0.0131, +0.0022]	+0.031	120%
	$\lambda = 0.2$	22.67%	−0.0098 [−0.0160, −0.0036]	+0.048	121%
	$\lambda = 0.4$	27.71%	−0.0104 [−0.0201, −0.0006]	+0.076	114%
	$\lambda = 0.6$	31.38%	−0.0080 [−0.0159, −0.0001]	+0.093	109%
FB15k-237	released (rule unidentified)	19.43%	+0.0091 [+0.0085, +0.0097]	+0.036	75%
	strict decrease	12.94%	+0.0040 [+0.0023, +0.0056]	+0.013	69%

dropping the 69 affected strengthens the residual to −0.0298), per-query edit exposure (correlation −0.0029) and leakage (worth 0.0010; Section C.6). (iii) With more support the contrast turns positive, reaching +0.1557 [+0.125, +0.188] at $m_2 \geq 21$ on CoDEx-S. The sign change replicates across five constructions, a second graph and retraining (Table 22). The ordering between strata does not replicate, so we claim the sign change and not a monotone gradient.

Overall, the reported aggregate (+0.0365 matched, +0.1524 conventional) is not an estimate of what any query experiences. It averages a negative effect over 45% of the test set against a positive one over the rest. Two limits apply. The split does not identify a mechanism, because the contrast inside each stratum is the same two-arm contrast as the aggregate. And the pattern belongs to the released construction. Rebuilt under a documented rule at the same doses ($\lambda = 0.2$, five constructions), the CoDEx-S gradient falls from +0.343 to +0.033, and the conventional contrast at $m_2 = 0$ turns positive, +0.017 [+0.008, +0.027]. How the graphs are built is the subject of Section 5.

5 The result depends on how the modified graphs are built

A construction procedure has two parts: an acceptance rule, which decides whether a candidate swap is kept, and a stopping condition, which fixes the dose. The released arms state the dose but not the rule. If a matched control were independent of the procedure, κ would not change with the

rule (H3). To test this, we rebuilt the modified graphs under rules we can write down, at the same target closure, and measured $\kappa = f(\text{matched}) - f(\text{break})$ on full test, with five constructions per rule (Table 5; Figure 8 plots it).

We find that κ depends on the rule. On CoDEx-S the released construction gives $\kappa = +0.0366$ MRR (95% t-interval $+0.0321$ to $+0.0412$). A strict rule, which accepts a swap only if it lowers closure, reaches the same target closure with an edit budget of 11.45% instead of 24.81%, and gives $\kappa = +0.0031$ (-0.0001 to $+0.0063$). Relaxing that rule, by accepting closure-neutral swaps with probability λ, raises the edit budget past the released one. κ does not follow it upward: it is -0.0039 at 15.89% edits, -0.0098 (-0.0160 to -0.0036) at 22.67%, -0.0104 at 27.71% and -0.0080 at 31.38%. The positive value is therefore not an effect of the budget; among the procedures we can document, it is specific to the released one. On FB15k-237 the comparison separates in the same direction and by less: $+0.0091$ ($+0.0085$ to $+0.0097$) for the released rule against $+0.0040$ ($+0.0023$ to $+0.0056$) for the strict rule. At Hits@1 the strict rule's value is indistinguishable from zero.

We also read each rule against a *documented* `preserve`, so that both arms follow stated rules. The Edit-Matched Arm then reproduces 121% $[98, 156]$ of the drop at a budget like the released one, and at least 100% under every relaxed rule, against 66% for the released pair. The released `break` is also unusually damaging for its dose ($+0.107$ MRR against that `preserve`, where $\lambda = 0.2$ costs $+0.048$ at a comparable budget), and the same rebuild changes the split of Section 4.2. Both released arms are atypical of the rules that reach their doses, and the differences push the reported contrast the same way.

A control at one edit budget cannot separate closure from what moves with it. We therefore built a grid that crosses closure loss with reciprocity at a fixed edit budget, and scored four families on it (Section D.1.1). In every family the coefficient on closure loss is positive, the opposite of the sign the compositional reading predicts, and small: over the grid's range it moves MRR by $+0.003$ to $+0.009$, against -0.012 to -0.030 for the loss of symmetric pairs, which carries the fit. The grid stops at 23% closure loss, below `break`'s 33%, and on CoDEx-S symmetric-pair loss and reciprocity nearly coincide.

Overall, what an edit-matched control measures is a property of the procedure as much as of the graph; which edges a rule prefers also matters (Section 4.1). A result of this kind is not interpretable unless the acceptance rule and the stopping condition are reported with it.

6 A REPORTING CHECKLIST

Everything below is regenerable from the release: the builders of every arm we constructed, the per-query rank dumps and m_2 files, the evaluation scripts with their checksums, and the scripts that emit every generated table. A result of this kind is interpretable only with six things stated beside it.

1. **Edit distance beside the structural dose.** How many training triples moved, not only how much structure was removed.
2. **The acceptance rule and the stopping condition.** A dose does not determine a construction, and the matched-control contrast changes sign with the rule (Table 5).
3. **An arm matched on the edit budget.** Without it, most of the drop cannot be attributed.
4. **A second axis, or a design that crosses them.** At one edit budget closure cannot be separated from the symmetric pairs the rewiring also removes (Section D.1.1).
5. **The contrast on queries with no two-hop path to lose.** An internal negative control from the same runs.
6. **Intervals clustered on the construction, and both a head and a recall statistic.** Construction varies more than training seeds, and the answer changes along the ranking.

7 DISCUSSION

The remedy is cheap. Where the checks of Section 6 fire, the fix is a third arm; the two ladder rungs took 97 and 154 seconds of CPU to build (Section B.1).

Applied to a published finding. Li et al. (2023a) read a null against a same-edge-count random graph as evidence against message passing. That arm moves 99.985% of edges and destroys 72.49% of closure, so the null does not point to any one cause (Section E).

The modification does remove composition. AnyBURL mines 81% fewer compositional rules from the modified CoDEx-S graph (694 to 133), against a tenth of all rules mined (Table 31). A scorer built from counted rules also loses accuracy under `break` (+0.0310 MRR, 10.1% of its clean score; Section C.7), although a random rewiring costs it almost as much. The target dose is therefore real; what the two-graph test cannot do is separate it from the edits. For the three trained backbones on record, the drop is not a failure to fit: the arms reach similar training loss and differ in test accuracy (Section C.8).

When the conventional reading would be right, and what generalises. Had the checks of Section 4 come out the other way they would have strengthened the compositional reading without establishing it. We observed the opposite of each, and a model could still compose and show all of them, which is why we claim that this evidence cannot tell, not that composition is absent. Nor is the problem specific to graphs. Interventions on model internals also move more than their target (Makelov et al., 2024; Lin & Liu, 2026; Joshi et al., 2026), and a scalar summary hides the split (Zhang & Nanda, 2024). Here, the acceptance rule, not the target quantity, sets the size of the side effect.

8 Limitations and Conclusion

What we claim. Table 3 is one construction on one graph under a frozen checkpoint. We claim the sign change and not the ordering between strata: the sign replicates across five constructions, two graphs and three training seeds (Table 23, Table 22), while the ordering of adjacent upper strata does not. Table 34, Table 18 and Table 41 rest on single runs or unequal samples and are read only as sign disagreements at the noise floor. Intervals are clustered on the arm: a variance decomposition separates a construction component from the training-seed component ($p = 0.009$ on MRR) without pinning down its size (Section D.2).

A two-arm contrast at one edit budget cannot hold fixed everything the operator moves. Two quantities travel with closure: reciprocity, which the Edit-Matched Arm also lowers by 54–68% of `break`'s fall, and the loss of symmetric triples. The grid of Section 5 varies both at a fixed budget and finds that the drop tracks symmetric-pair loss rather than closure (Section D.1.1); a design that leaves the two confounded cannot say which one a drop measures.

The control has a regime and an operating point. Our search found no closure-preserving graph at the near-total edit budgets of structure-free arms (Section E), and on FB15k-237 a one-hop walker reaches 0.012–0.014 MRR on every arm, so no contrast is defined there. The value also depends on the acceptance rule (Table 5). The released `preserve` and `break` arms are single builds from a builder we could not recover; five rebuilds of each bound how far they sit from the rules that reach those doses (Section D.1.2), and both differences enlarge the conventional contrast, so our shares are conservative for these builds.

The design is two-hop only (Section C.7), and frozen ULTRA never trains on a modified graph, so optimisation effects cannot explain its drop; retraining reproduces the pattern on both graphs (Table 22).

Conclusion. A drop under this structural modification does not show compositional reasoning. The modification changes at least four things at once, one of them unmatched by any single arm, and the answer the same runs give depends on the rank statistic and on how the graphs were built. This evidence cannot yet tell whether these models compose; the checks that would constrain it are cheap, and we release them.

Reproducibility Statement

The supplementary material contains the code that builds every arm we constructed, the evaluation and table-generation scripts, the released arms with checksums, and the per-query ranks. Arm con-

struction and its invariants are described in Section B.1, the evaluation protocol and hyperparameters in Section B.2, and the checks applied to every reported number in Section F.1.

AI USE STATEMENT

We used generative AI tools to help design and script experiments, implement analysis code, interpret results, find related work, and draft, edit and format the paper. All AI-assisted code ran under the checks of Section F.1, every reported number was traced to the run records, and the authors reviewed all AI-assisted text and code and take full responsibility for this submission.

REFERENCES

Farahnaz Akrami, Mohammed Samiul Saeef, Qingheng Zhang, Wei Hu, and Chengkai Li. Realistic re-evaluation of knowledge graph completion methods: An experimental study. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020.

Yael Artzy-Randrup, Sarel J. Fleishman, Nir Ben-Tal, and Lewi Stone. Comment on "network motifs: Simple building blocks of complex networks" and "superfamilies of evolved and designed networks". *Science*, 305(5687):1107, 2004.

Pablo Barceló, Mikhail Galkin, Christopher Morris, and Miguel Romero Orth. Weisfeiler and leman go relational. In *LoG*, 2022.

Maya Bechler-Speicher, Ben Finkelshtein, Fabrizio Frasca, Luis Muller, Jan Tonshoff, Antoine Siraudin, Viktor Zaverkin, Michael M. Bronstein, Mathias Niepert, Bryan Perozzi, Mikhail Galkin, and Christopher Morris. Position: Graph learning will lose relevance due to poor benchmarks. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Peru Bhardwaj, John Kelleher, Luca Costabello, and Declan O'Sullivan. Poisoning knowledge graph embeddings via relation inference patterns. In *ACL-IJCNLP*, 2021.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. In *International Conference on Learning Representations*, 2018.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2D knowledge graph embeddings. In *AAAI*, 2018.

Bailey K. Fosdick, Daniel B. Larremore, Joel Nishimura, and Johan Ugander. Configuring random graph models with fixed degree sequences. *SIAM Review*, 60(2):315–355, 2018.

Mikhail Galkin, Xinyu Yuan, Hesham Mostafa, Jian Tang, and Zhaocheng Zhu. Towards foundation models for knowledge graph reasoning. In *ICLR*, 2024.

Cosimo Gregucci, Obaidah Theeb, Daniel Hernandez, Antonio Vergari, and Steffen Staab. Half a link can be enough to predict a whole link: Understanding generalization in knowledge graph foundation models. *arXiv preprint arXiv:2606.18001*, 2026.

Haji Gul and Ajaz Ahmad Bhat. When metrics disagree: A meta-analysis of knowledge-graph-completion model benchmarking. *arXiv:2606.10287*, 2026.

Xingyue Huang, Miguel Romero Orth, İsmail İlkan Ceylan, and Pablo Barceló. A theory of link prediction via relational Weisfeiler-Leman on knowledge graphs. In *NeurIPS*, 2023.

Xingyue Huang, Pablo Barcelo, Michael M. Bronstein, Ismail Ilkan Ceylan, Mikhail Galkin, Juan L. Reutter, and Miguel Romero Orth. How expressive are knowledge graph foundation models? In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *PMLR*, pp. 25021–25058, 2025. arXiv:2502.13339.

Xingyue Huang, Mikhail Galkin, Michael M. Bronstein, and İsmail İlkan Ceylan. HYPER: A foundation model for inductive link prediction with knowledge hypergraphs. In *ICLR*, 2026.

Siddharth Joshi, Aaron Mueller, David Klindt, Wieland Brendel, Patrik Reizinger, and Dhanya Sridhar. Causality is key for interpretability claims to generalise. *arXiv preprint arXiv:2602.16698*, 2026.

Jaejun Lee, Chanyoung Chung, and Joyce Jiyoung Whang. InGram: Inductive knowledge graph embedding via relation graphs. In *ICML*, 2023.

Juanhui Li, Harry Shomer, Jiayuan Ding, Yiqi Wang, Yao Ma, Neil Shah, Jiliang Tang, and Dawei Yin. Are message passing neural networks really helpful for knowledge graph completion? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023a.

Juanhui Li, Harry Shomer, Haitao Mao, Shenglai Zeng, Yao Ma, Neil Shah, Jiliang Tang, and Dawei Yin. Evaluating graph neural networks for link prediction: Current pitfalls and new benchmarking. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks)*, 2023b.

Xi Victoria Lin, Richard Socher, and Caiming Xiong. Multi-hop knowledge graph reasoning with reward shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

Yu Lin and Zhen Liu. Position: Mechanistic interpretability must disclose identification assumptions for causal claims. *arXiv preprint arXiv:2605.08012*, 2026.

Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. Is this the subspace you are looking for? an interpretability illusion for subspace activation patching. In *ICLR*, 2024.

Sergei Maslov and Kim Sneppen. Specificity and stability in topology of protein networks. *Science*, 296:910–913, 2002.

Christian Meilicke, Melisachew Wudage Chekol, Daniel Ruffinelli, and Heiner Stuckenschmidt. Anytime bottom-up rule learning for knowledge graph completion. *IJCAI*, 2019.

Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

Chiara Orsini, Marija M. Dankulov, Pol Colomer-de Simón, Almerima Jamakovic, Priya Mahadevan, Amin Vahdat, Kevin E. Bassler, Zoltán Toroczkai, Marián Boguñá, Guido Caldarelli, Santo Fortunato, and Dmitri Krioukov. Quantifying randomness in real networks. *Nature Communications*, 6:8627, 2015.

Pouya Pezeshkpour, Yifan Tian, and Sameer Singh. Investigating robustness and interpretability of link prediction via adversarial modifications. In *NAACL*, 2019.

Andrea Rossi, Donatella Firmani, Paolo Merialdo, and Tommaso Teofili. Explaining link prediction systems based on knowledge graph embeddings. In *SIGMOD*, 2022.

Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. You CAN teach an old dog new tricks! on training knowledge graph embeddings. In *ICLR*, 2020.

Ali Sadeghian, Mohammadreza Armandpour, Patrick Ding, and Daisy Zhe Wang. DRUM: End-to-end differentiable rule mining on knowledge graphs. In *NeurIPS*, 2019.

Tara Safavi and Danai Koutra. CoDEx: A comprehensive knowledge graph completion benchmark. In *EMNLP*, 2020.

Harry Shomer, Jay Revolinsky, and Jiliang Tang. Towards better benchmark datasets for inductive knowledge graph completion. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 5777–5788, 2025. doi: 10.1145/3711896.3737438. arXiv:2406.11898.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotaTE: Knowledge graph embedding by relational rotation in complex space. In *ICLR*, 2019.

Komal K. Teru, Etienne Denis, and William L. Hamilton. Inductive relation prediction by subgraph reasoning. In *ICML*, 2020.

Kristina Toutanova and Danqi Chen. Observed versus latent features for knowledge base and text inference. In *Workshop on CVSC*, 2015.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *ICML*, 2016.

Tyler J. VanderWeele. On the distinction between interaction and effect modification. *Epidemiology*, 20(6):863–871, 2009.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. Composition-based multi-relational graph convolutional networks. In *ICLR*, 2020.

Wenhan Xiong, Thien Hoang, and William Yang Wang. DeepPath: A reinforcement learning method for knowledge graph reasoning. In *EMNLP*, 2017.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *ICLR*, 2015.

Fan Yang, Zhilin Yang, and William W. Cohen. Differentiable learning of logical rules for knowledge base reasoning. In *NeurIPS*, 2017.

Xiao Yang et al. Are heterogeneous graph neural networks truly effective? a causal perspective. *arXiv preprint arXiv:2510.05750*, 2025.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *ICLR*, 2024.

Hengtong Zhang, Tianhang Zheng, Jing Gao, Chenglin Miao, Lu Su, Yaliang Li, and Kui Ren. Data poisoning attack against knowledge graph embedding. In *IJCAI*, 2019.

Yongqi Zhang and Quanming Yao. Knowledge graph reasoning with relational digraph. In *WWW*, 2022.

Yongqi Zhang, Zhanke Zhou, Quanming Yao, Xiaowen Chu, and Bo Han. AdaProp: Learning adaptive propagation for graph neural network based knowledge graph reasoning. In *KDD*, 2023.

Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal Xhonneux, and Jian Tang. Neural Bellman-Ford networks: A general graph neural network framework for link prediction. In *NeurIPS*, 2021.

A Claim map and related work

A.1 Claim map

This appendix opens with a map from each claim in the main text to the table that supports it and the protocol that table was measured under (Table 6).

The rest follows the paper. Section B gives the construction and the measurement protocol; Section C holds the analyses behind Section 4.1 and Section 4.2; Section D holds those behind Section 5 and Section 8; Section E re-analyses a published finding; and Section F states the verification protocol and the run records.

A.2 Related work and prior-work verification

Knowledge graph completion audits its measurements more than most fields: training and tuning (Ruffinelli et al., 2020), dataset artefacts (Akrami et al., 2020), inductive splits and shortcuts (Shomer et al., 2025), metric disagreement (Gul & Bhat, 2026). Those vary benchmark, protocol or query population with the metric fixed; we vary the metric on the same runs and ask whether two arms differ only in the quantity the claim is about. Structure-free and wrong-structure controls (Bechler-Speicher et al., 2025; Li et al., 2023b;a; Yang et al., 2025) apply the same move to architectures; we re-analyse the one of them that edits the graph rather than the model (Section E); expressiveness results (Barceló et al., 2022; Huang et al., 2023; 2025) say what a model can represent, while intervening on the graph is how the field argues that it does, the step we audit.

Table 6: What supports each claim in the main text, and the protocol each number is measured under. Full test means the complete test split scored with one scorer (per-query mean of the head and tail statistic); the 1,200-query protocol appears only where it is named.

Claim	Evidence	Protocol
The modification rewrites up to a quarter of training triples, not only closure	Table 9, Section 3	Released arms; training split
An edit-matched control reproduces about three quarters of the MRR drop	Table 2, Section 4.1	Full test; frozen ULTRA-3g and MOTIF-3g; five constructions of the matched arm
Models with no traversal mechanism lose more than half as much as ULTRA	Section 4, Section C.1	Trained four-graph campaign; three seeds
The reproduced share is larger at Hits@1 than at Hits@10	Section 4.1, Section C.2	Full test; query bootstrap
The matched-control contrast changes sign with the acceptance rule	Table 5, Figure 8, Table 39	Full test, five constructions per rule; 1,200-query values named where used
In the released graphs, the drop is absent or reversed on queries with no two-hop path	Table 4, Table 21, Table 22	Full test; 2,000-draw query bootstrap; released arms
Edge selection at a fixed dose moves the reproduced share	Table 25	Full test; three constructions per side
The modification removes compositional rules	Table 31	AnyBURL rule counts on CoDEx-S and FB15k-237
At a fixed edit budget the drop follows symmetric-pair loss, not closure	Table 35, Section D.1.1	Full test; 45 constructions on CoDEx-S, four families
The released arms are atypical of the rules that reach their doses	Table 36, Section D.1.2	Full test; five rebuilds per arm; frozen ULTRA-3g

Training-graph interventions are not new in themselves. Poisoning and attribution work on knowledge graph embeddings adds, removes or rewires training triples and measures the effect on predictions (Pezeshkpour et al., 2019; Zhang et al., 2019; Bhardwaj et al., 2021; Rossi et al., 2022); Bhardwaj et al. (2021) find that attacks built on the symmetry pattern transfer across every model and dataset they test, which is consistent with the sensitivity to symmetric pairs we report. Those studies ask which edits move a prediction, not whether a drop isolates one mechanism. Network science has long shown that structural conclusions depend on the randomisation ensemble they are measured against (Milo et al., 2002; Artzy-Randrup et al., 2004; Orsini et al., 2015; Fosdick et al., 2018); the within-relation double-edge swap is one such ensemble, and the choice of acceptance rule (Section D.3) is a choice among them.

Table 7 was built by reading the primary text of each study rather than its abstract or a secondary summary. A cell reads *edit-matched* only where the paper states that the amount of perturbation (edge count or edit budget) is held fixed between the compared conditions, and *non-compositional model control* only where the same intervention is applied to a family with no query-conditioned multi-hop path mechanism at inference and its response compared; reporting such a model in a leaderboard does not count. The rank-statistic column records what the intervention table itself reports, which is frequently narrower than the paper's main results table.

Two rows show that each control has been run before, though separately. Li et al. (2023a) re-run message passing over a random graph with the edge count matched to the original (their Table 7, on MRR and three Hits levels), which is a genuine edit-budget-matched control; it differs from ours in that it destroys structure at a fixed budget rather than preserving it, and it is used to argue a null rather than to attribute an effect. Yang et al. (2017) and Sadeghian et al. (2019) report an embedding model under the same entity-disjoint resplit used to argue for rule-based reasoning, which is a non-compositional family subjected to the same manipulation, though the manipulation is a resplit rather than a budgeted edit. The defensible claim, and the one Table 7 supports, is that the two controls

Table 7: **The two controls this paper pairs have never been run together.** Studies that draw a conclusion about multi-hop or path structure from an intervention or ablation, verified against the primary text; rank statistics are those the *intervention* table reports, often fewer than the paper's main results. **Neither ingredient is unprecedented**: Li et al. (2023a) match a random graph's edge count, and Yang et al. (2017); Sadeghian et al. (2019) put an embedding model through the same resplit; **but no study runs both, and none applies the perturbation to a non-compositional family at a fixed edit budget**. Parenthesised entries are near-misses (Section A.2).

Study	edit-matched control?	non-compositional model control?	rank statistics
GraIL (Teru et al., 2020)	no	no	AUC-PR
Neural-LP (Yang et al., 2017)	no	**yes** (TransE)	H@10
DRUM (Sadeghian et al., 2019)	no	**yes** (TransE, Tab. 5)	MRR, H@1, H@3, H@10
NBFNet (Zhu et al., 2021)	no	no	MRR
RED-GNN (Zhang & Yao, 2022)	no	no	MRR
AdaProp (Zhang et al., 2023)	no	no	MRR, H@10
InGram (Lee et al., 2023)	no	no	MRR, H@10
ULTRA (Galkin et al., 2024)	no	no	MRR, H@10
MOTIF (Huang et al., 2025)	no	no	MRR, H@10
Shomer et al. (2025)	no	(PPR: structural, not negative)	ΔSPD, MRR, H@10
Gregucci et al. (2026)	n/a	no	MRR
Li et al. (2023a)	**yes** (edge count)	(MLP variant, not intervened)	MRR, H@1, H@3, H@10
This paper	**yes**	**yes**	MR, MRR, H@1, H@3, H@10

have never been run together. Section E re-analyses Li et al. (2023a)'s arm under our protocol and reports what the matched control can and cannot add to it.

B Construction and evaluation protocol

This section supports Section 3: it states how every arm is built and how every number is measured.

B.1 Arm construction and structural invariants

This section supports Section 3: it states the operator, the acceptance rules, and the invariants each arm is built to satisfy.

Table 8: The clean message-passing graph of each benchmark, with the two structural quantities the modification moves. Closure counts training triples on at least one directed two-path; symmetric counts training triples whose reverse is also present; reciprocity is the share of connected entity pairs joined in both directions.

Graph	entities	relations	train	valid	test	closed	symmetric	recip.
CoDEx-S	2,034	42	32,888	1,827	1,828	14,288 (43.4%)	5,050 (15.4%)	0.155
CoDEx-M	17,050	51	185,584	10,310	10,311	42,151 (22.7%)	6,538 (3.5%)	0.036
FB15k-237	14,505	237	272,115	17,535	20,466	169,248 (62.2%)	33,972 (12.5%)	0.282
NELL-995	74,432	200	149,678	543	2,818	31,123 (20.8%)	17,184 (11.5%)	0.171
WN18RR	40,559	11	86,835	3,034	3,134	8,014 (9.2%)	29,658 (34.2%)	0.343
CoDEx-M$_E$ (ladder base)	8,003	40	69,073	3,837	3,837	9,453 (13.7%)	396 (0.6%)	0.006

Table 8 lists them. The two graphs the main text reports on are the two with the most closure to remove: 43.4% of CoDEx-S's training triples and 62.2% of FB15k-237's lie on a directed two-path, against 9.2% on WN18RR. WN18RR is the mirror case: it carries the most symmetric triples (34.2%) and the least closure, which is why its contrast sits at the noise floor throughout.

The operator. All base arms, and the starting arms of the reciprocity factorials, are built by within-relation double-edge swaps: pick two triples (h_1, r, t_1) and (h_2, r, t_2) sharing relation r and replace them with (h_1, r, t_2) and (h_2, r, t_1). Arms differ only in which candidate swaps are accepted: `preserve` accepts swaps that retain closure instances, `break` accepts swaps that destroy them, `random` accepts without regard to closure, and `matched` accepts swaps that keep the closure count near the `clean` value and runs until a target edit count is reached. Rules enforced at build time: one canonical copy per arm read by every model family; all arms of a graph share one entity and one relation dictionary; validation and test are never perturbed; an md5 is recorded per arm. Candidate swaps are filtered against validation and test at build time, but the filter is not perfect: a swap can create a triple that coincides with a test triple, and the perturbed arms retain some. We measure the residue per arm rather than assert it away, and Section C.4.2 re-scores every contrast

with the affected queries removed. Only `matched` was re-seeded at construction, five builds per graph; `preserve`, `break` and `random` are single realisations, because their builder is not recoverable from our run tree and a reimplementation calibrated to reproduce the released arm would measure the reimplementation rather than the builder.

Proposition 1 (Swap-level invariants)**.** *A within-relation double-edge swap preserves exactly, per relation: (i) the head and tail multisets, and hence the triple count; (ii) every entity's per-relation degree; and (iii) half-link coverage in the sense of Gregucci et al. (2026). These invariants, and any statistic that is a function of them, are all the operator fixes; they do not imply invariance of the quantities varied here. The number of two-paths joining a head to its answer, the transitivity and the fraction of reciprocated edges may each move, and their joint behaviour is an empirical matter rather than a consequence of (i)–(iii).*

Proof. A swap maps $\{(h_1, r, t_1), (h_2, r, t_2)\}$ to $\{(h_1, r, t_2), (h_2, r, t_1)\}$. The head multiset of r contains h_1, h_2 before and after; the tail multiset contains t_1, t_2 before and after, giving (i). Per-relation degrees are the multiplicities in those multisets, giving (ii). Half-link coverage classifies a test query (h, r, t) by membership of h in r's training head set and t in its training tail set, which are determined by the multisets, giving (iii). Compositions of swaps preserve what every swap preserves. □

Table 9: Edit distance per arm: the percentage of training triples differing from `clean`. The final column is the ratio the conventional `preserve`–`break` contrast silently applies.

	edit distance				closure destroyed
Graph	`preserve`	`break`	`random`	break/preserve	by `break`
CoDEx-S	13.14%	24.81%	15.58%	1.89×	33.23%
FB15k-237	9.99%	19.43%	15.64%	1.95×	26.83%
WN18RR	2.15%	2.40%	2.22%	1.12×	28.06%
NELL-995	7.03%	7.50%	7.46%	1.07×	22.56%

The CoDEx-M factorial, measured rather than asserted. Table 9 and Table 10 cover the four graphs that carry base arms. CoDEx-M has none (it carries the reciprocity × closure factorial instead), and Section D.4 relies on its sign reversal; its cells start from swap-built closure arms and then set reciprocity by the relocation step described under *The reciprocity factorials* below. Its structural statistics, recomputed from the released arms on the same definitions as Table 10 and averaged over the five construction seeds in each cell, are in the supplementary material.

Three things follow, and all three are checks the design needed rather than results in its favour by construction. *Edit distance is held*: it spans 24.92–25.25% across all nine cells, a range of 0.33 points, so the contrasts within the grid are taken at effectively fixed edit dose. *The two factors are crossed, not confounded*: closure destruction moves 2.73%, 14.14%, 25.59% along its own axis while reciprocity along that axis stays within 0.000887; reciprocity moves 0.02895, 0.03475, 0.04368 along its axis while closure destruction stays within 0.57 points. *Assortativity is not the third mover here*: it spans 0.00112 across the whole grid against a `clean` value of −0.1896. That last point matters, because on CoDEx-S reciprocity and assortativity are collinear at $r = -0.9932$ and conditioning on assortativity reverses the reciprocity coefficient (Table 43). On CoDEx-M assortativity does not move, so whatever drives the reversal reported in Section D.4, it is not that.

`random` is not a zero-dose control. Its achieved closure dose ranges from 2.60% to 11.93% across the four graphs in closure instances (Table 10), and from 2.4% to 13.3% in closed edges (Table 13). We report it rather than assume it, and we do not use it as a null.

What else the operator moves. Triangles fall 8.3–23.7% under `break` and transitivity 14.4–24.3%; assortativity changes by less than 0.01 on three of four graphs and the effective diameter is unchanged on three of four; the relation-degree fingerprint is bit-identical across all sixteen graph-arm combinations (Table 10). Reciprocity is the largest remaining mover, falling 5.3–34.7% under `break`. Conditioning the operator on reciprocity reduces that drop from −24.4% to −0.4% on CoDEx-S and from −5.3% to −0.04% on WN18RR at the cost of a lower acceptance rate, which is why reciprocity is the second factor of the grids in Section D.4. That the operator can

hold reciprocity fixed does not make reciprocity the cause of any residual: Table 43 shows it cannot be separated from assortativity. Invariant (iii) holds in the produced files, a check that implementation matches algebra (supplementary material); NELL-995 needs one note, because it ships its message-passing edges in `facts.txt` beside `train.txt` and its released arms edit both. Read on `train.txt` alone, those arms keep only 95.0–95.4% of per-relation degrees; read on the 149,678-triple graph the models are actually given, all 132,643 per-relation degrees are preserved exactly (Table 13). Every NELL-995 row of Table 13 and of Figure 3 is computed on that graph.

Table 10: Higher-order statistics per arm. Closure and triangles fall under `break` as intended; reciprocity moves as well.

Graph	Arm	Closures	Triangles	Transitivity	Reciprocity
CoDEx-S	`clean`	179,619	67,571	0.06124	0.15547
CoDEx-S	`preserve`	179,662	67,895	0.06130	0.15244
CoDEx-S	`break`	125,977	52,924	0.04645	0.11756
CoDEx-S	`random`	171,953	64,225	0.05766	0.14457
FB15k-237	`clean`	1,063,593	365,165	0.01640	0.28191
FB15k-237	`preserve`	1,067,229	374,566	0.01662	0.25994
FB15k-237	`break`	771,905	334,959	0.01404	0.18405
FB15k-237	`random`	936,757	379,811	0.01618	0.21719
WN18RR	`clean`	9,664	4,403	0.00950	0.34339
WN18RR	`preserve`	9,664	4,400	0.00947	0.33552
WN18RR	`break`	6,766	3,361	0.00719	0.32532
WN18RR	`random`	8,724	4,097	0.00880	0.33207
NELL-995	`clean`	139,003	53,256	0.00756	0.17133
NELL-995	`preserve`	139,036	53,345	0.00751	0.15947
NELL-995	`break`	103,189	44,503	0.00619	0.14706
NELL-995	`random`	135,384	53,399	0.00749	0.15393

The Edit-Matched Arm. The target is `break`'s edit count; closure drift is the residual (Table 11). Our first construction matched the *swap count* rather than the edit count. Because `preserve` accepts swaps at a much higher rate than `break` on dense graphs, 4,079 swaps produced only 12.20% edits, statistically indistinguishable from plain `preserve`, and useless as a matched arm. Targeting the edit count directly gives 24.84% against `break`'s 24.81%. The lesson generalises: match the dose you care about, not a proxy for it. Across five independent constructions of the CoDEx-S matched arm the share reproduced by the matched arm spans 75.5–79.1% (SD 1.2 points). Section C.3.4 shows that this spread measures construction noise within one selection rule and not the selection rule itself. DistMult was also run on the matched arm (levels in the released run ledger); no DistMult share is printed because its `preserve` and `break` levels come from a different seed set.

Table 11: Construction of the Edit-Matched Arm. The target is `break`'s edit count; closure drift is the residual.

Graph	target edits	achieved	edit %	closure drift	swaps
CoDEx-S	8,158	8,168	24.84%	−0.52%	14,400
FB15k-237	52,870	52,910	19.44%	−0.03%	59,800

The Edit-Matched Closure Ladder. The ladder builder changes one line relative to the matched-arm builder: it accepts a swap if it moves the closure count toward a specified target, and holds it inside a 1% band once there, while edits accumulate to the same target as before (Table 12). Both rungs land within 1.1 points of their closure target and within 1.0% of `break`'s edit count. Every closure figure in the paper is the number of training edges that participate in at least one closed two-path, and all arms were recounted with one script. Two closure counters exist in our code and agree on CoDEx-S (14,288 either way) but not on FB15k-237 (181,836 against 169,248); an intermediate rebuild that mixed them reported an irreproducible coefficient, and the lesson is the paper's own: a dose is only meaningful relative to the instrument that set it. The two CoDEx-S rungs took 97 and 154 seconds of CPU to build and 24 additional training runs to evaluate.

The builders' incremental closure counter. The matched and ladder builders, and the ceiling search of Section E, track closure incrementally through a successor map keyed on the head entity

alone, so deleting (a, r, b) removes $a \to b$ even when (a, s, b) remains. On directed pairs carrying more than one relation — 0.84% of pairs on CoDEx-S and 9.48% on FB15k-237 — the incremental count therefore falls below the true one. The count error has one sign; the acceptance error does not. A swap that lifts the true count above the target band can be carried back inside it by the undercount, so the search can accept a graph a correct counter would reject as readily as reject one it would accept, and both directions are realisable on the released builders by a single legal degree-preserving swap. Re-running the CoDEx-S matched build from the released code reproduces the shipped arm byte for byte ($14{,}400$ swaps, $8{,}168$ edits, identical md5) and ends at an incremental $14{,}136$ against a true $14{,}214$: the builder steered as though it had lost 1.06% of closure while it had lost 0.52%. What the recount establishes is the structural description of the arms we shipped and evaluated: the -0.52% and -0.03% of Table 11, and every closure figure in this paper, are exact counts on those graphs. It does not establish which graphs a corrected builder would have selected, or how the models would score on them. We therefore report these arms as the output of this search, with their properties measured exactly, and claim nothing about the arms a correct search would have produced — including for the ceiling of Section E, whose reach and stalling rate are properties of that search.

Table 12: Ladder construction on CoDEx-S.

Arm	target closure	achieved	closure loss	edits	edit %
`ladder-11`	12,716	12,654	11.44%	8,162	24.82%
`ladder-21`	11,145	11,293	20.96%	8,233	25.03%
`matched` (ref.)	—	14,214	0.52%	8,168	24.84%
`break` (ref.)	—	9,540	33.23%	8,158	24.81%

The Closure-Removal Ladder on `codex_m_E`. Train and inference graphs share no entities; all forty relations are shared; no test triple appears in the inference graph. Closure is 13.7% against CoDEx-S's 43.4% and reciprocity 0.0063 against 0.1555, so both doses our intervention manipulates largely disappear when a benchmark is rebuilt to remove an unrelated shortcut. The arms perturb the inference graph and are evaluated with a frozen checkpoint. Edit distance across the three perturbed rungs is 25.30%, 25.06% and 25.26% (a spread of 0.24 points), so closure falls from 13.73% to 1.11% at a common edit budget. One caveat: on this graph the edit operator also perturbs reciprocity (0.0063 to 0.0018), so `sc_preserve` is not a pure edit-only arm; this does not affect the conclusion, since closure adds nothing on top, but the doses are less cleanly separated here than on CoDEx-S.

The reciprocity factorials. Closure and reciprocity are set independently at three levels each, with edit distance held constant across every cell and leakage zero throughout; five arm-construction seeds per design cell on CoDEx-M, three on the retrained CoDEx-S grid. Edit distance spans only 0.4 points across the CoDEx-M grid (24.91–25.28%), but its residual variation correlates with reciprocity, which is why every reported contrast is taken within a closure block, where edit distance is constant, rather than across the grid.

How the cells are built, and what that costs. Each cell starts from a swap-built arm at its closure level, which keeps every per-relation degree, and then moves reciprocity by relocating single triples within their relation until the target is met. Relocation does not preserve degrees: across the 45 cells of each grid, 94.5–99.4% (CoDEx-S) and 96.6–99.0% (CoDEx-M) of `clean` per-relation degrees are unchanged, and the departure grows with the reciprocity move (CoDEx-M, seed 1: 96.7–96.9% at high reciprocity against 98.7–99.0% at low; Table 13). Proposition 1 therefore covers the base arms and the starting arms of the factorials, not the factorial cells themselves. The builders' leakage filter also has an error: a training triple that coincided with a validation or test triple was moved to a random tail drawn as an index into the entity dictionary rather than as an entity name, so 0.33–0.49% (CoDEx-S) and 0.14–0.19% (CoDEx-M) of training triples in every cell point to an entity that occurs nowhere else. Mapping the indices back to names changes closed edges by at most 1.3% and reciprocity by at most 0.0002 in any cell, far less than the gaps between closure levels (on CoDEx-M about 3, 14 and 25% of closed edges lost at seed 1), and would have created nine leaked triples across the 45 CoDEx-S cells; the cells are reported as trained. That check is on graph statistics rather than on model output: what repairing the indices would do to the fitted coefficients is unmeasured, against a closure contribution that is itself small (Section D.1.1).

What each arm does to the graph. Table 13 measures every arm on its own training graph, with one script. Closed edges are triples (h, r, t) with some m, $h \to m \to t$; m_2 is the paper's support count (entities adjacent to both h and t, any direction and relation), here recomputed on each arm; *supported lost* is the share of test queries with $m_2 > 0$ on `clean` left at $m_2 = 0$; rule groundings are the clean groundings of the fifty most frequent two-hop compositions $r_1(x, y) \wedge r_2(y, z) \Rightarrow r(x, z)$ whose three triples all remain. Figure 3 follows one test query per benchmark through three arms and Figure 4 compares the arms across graphs. Three readings follow. First, `break` removes compositional structure on every graph: 22.6–33.2% of closed edges, 11.6–44.3% of the supported test queries' every two-hop path, and 29–54% of the top-rule groundings. Second, `matched` makes the same number of edits as `break` on CoDEx-S and FB15k-237 and keeps closure within 0.5%, yet it still leaves 12.3% and 6.0% of supported queries without a path and removes about a fifth of top-rule groundings: holding closure fixed does not hold every path fixed. Third, `random` is not neutral: it removes 8.5–13.3% of closed edges on three graphs. Degrees are kept exactly on CoDEx-S, FB15k-237, WN18RR and NELL-995; only the CoDEx-M rows fall short of 100%, for the reason given in the factorial paragraph above.

Table 13: **Graph anatomy per arm.** Measured on each arm's training graph. Edits: share of `clean` training triples absent from the arm. Degrees kept: share of `clean` (entity, relation, direction) degrees unchanged. m_2=0: test queries with no entity adjacent to both head and answer. Supported lost: of the queries with $m_2 > 0$ on `clean`, the share left at $m_2 = 0$. CoDEx-M rows are seed 1 of each closure × reciprocity cell (closure high/mid/low, reciprocity high/mid/low). Every NELL-995 row is computed on the graph the models read: its message-passing edges, which ship in a separate file, together with its training triples.

Graph	Arm	edits (%)	degrees kept (%)	closed edges lost (%)	m_2=0 (%)	supported lost (%)	top-50 rule groundings kept (%)	reciprocity
CoDEx-S	`clean`	0.0	100.0	0.0	45.2	0.0	100.0	0.155
	`preserve`	13.1	100.0	0.3	46.5	4.1	98.9	0.152
	`random`	15.6	100.0	13.3	53.8	19.7	89.3	0.145
	`break`	24.8	100.0	33.2	59.4	29.4	45.6	0.118
	`matched`	24.8	100.0	0.5	49.7	12.3	79.3	0.135
FB15k-237	`clean`	0.0	100.0	0.0	26.2	0.0	100.0	0.282
	`preserve`	10.0	100.0	0.2	27.1	2.4	99.4	0.260
	`random`	15.6	100.0	12.5	29.2	7.6	85.4	0.217
	`break`	19.4	100.0	26.8	32.2	11.6	71.2	0.184
	`matched`	19.4	100.0	0.0	26.9	6.0	79.8	0.215
WN18RR	`clean`	0.0	100.0	0.0	81.9	0.0	100.0	0.343
	`preserve`	2.2	100.0	0.0	82.4	2.6	100.0	0.336
	`random`	2.2	100.0	8.5	82.6	4.0	90.3	0.332
	`break`	2.4	100.0	28.1	84.0	11.6	70.2	0.325
NELL-995	`clean`	0.0	100.0	0.0	77.0	0.0	100.0	0.171
	`preserve`	7.0	100.0	−0.1	78.0	5.4	99.8	0.159
	`random`	7.4	100.0	2.5	80.6	18.5	91.3	0.154
	`break`	7.4	100.0	22.6	86.4	44.3	69.3	0.147
CoDEx-M	`clean`	0.0	100.0	0.0	62.8	0.0	100.0	0.036
	`chi_rhi`	25.2	96.9	3.2	66.8	15.7	61.1	0.044
	`chi_rmid`	25.0	98.2	2.5	66.5	15.0	61.4	0.034
	`chi_rlo`	24.9	98.9	2.1	66.3	14.6	61.6	0.029
	`cmid_rhi`	25.2	96.8	14.3	68.8	21.7	60.1	0.043
	`cmid_rmid`	25.1	97.9	13.4	68.6	21.0	60.3	0.035
	`cmid_rlo`	25.0	99.0	13.4	68.4	20.5	60.5	0.028
	`clo_rhi`	25.2	96.7	25.3	71.8	29.6	58.9	0.044
	`clo_rmid`	25.0	98.0	25.0	71.5	29.0	59.2	0.034
	`clo_rlo`	24.9	98.7	25.3	71.4	28.7	59.3	0.029

B.2 Evaluation protocol and compute

All effects are within-model, within-block contrasts. Training budget, seed count and evaluation subset are identical on both sides of every difference reported, so they cancel; absolute accuracy is not compared across models. Ranking is filtered, in both directions. For ULTRA, MOTIF and HYPER the filter is built as in ULTRA's reference evaluation, from the known triples of the graph being scored: that arm's own training triples together with the shared validation and test triples (on CoDEx-S clean, 32,888 + 1,827 + 1,828). The filtered candidate set therefore differs slightly between arms, and we have not re-scored them against one fixed clean filter. The per-query score is

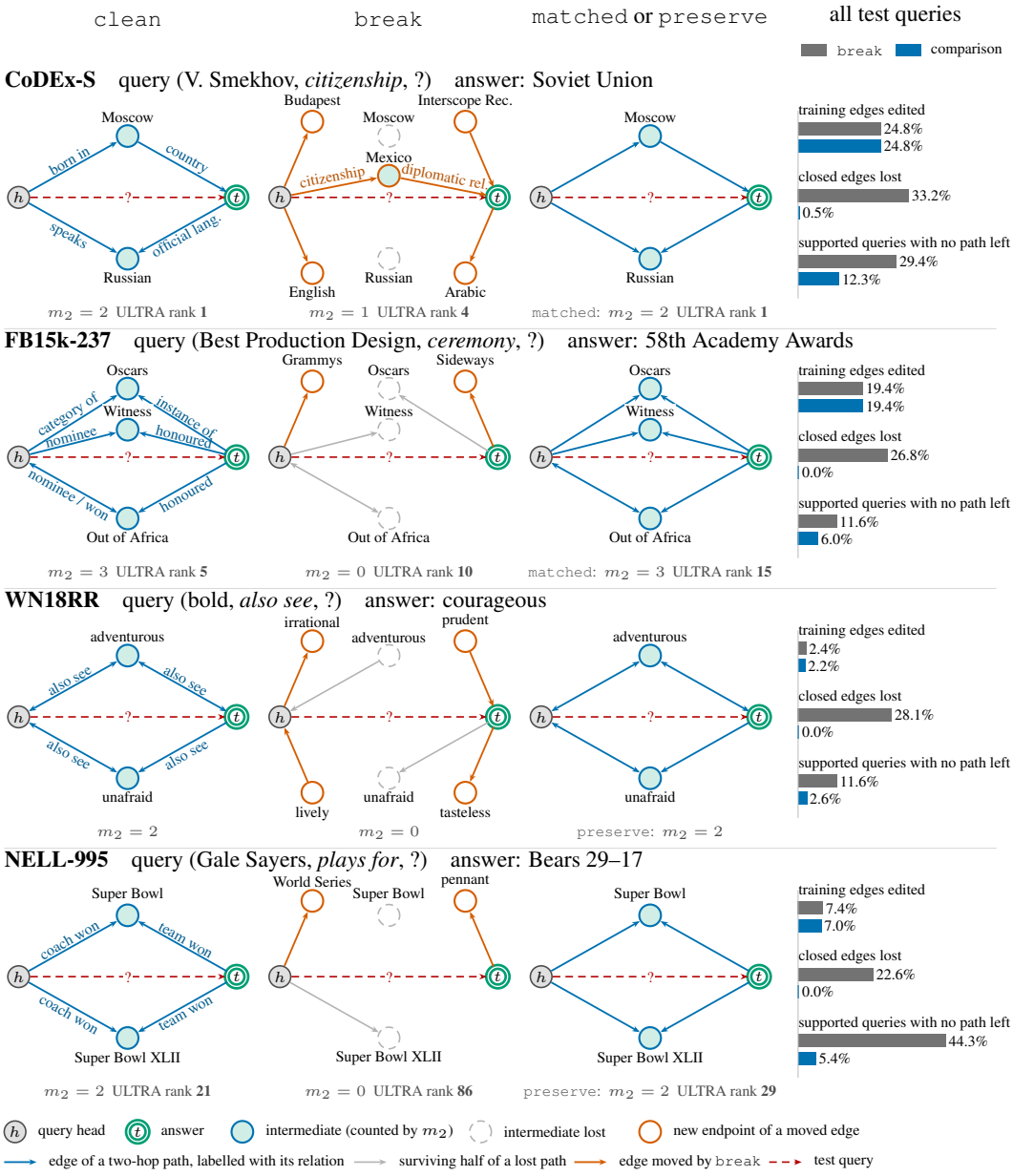


Figure 3: **One real test query per benchmark: `break` cuts every two-hop path of the query and the comparison arm keeps each one; the bars show the same contrast over all test queries.** Columns 1–3 draw the query's two-hop neighbourhood (h left, answer t right, each edge named by its relation) in `clean`, `break`, and `matched` (CoDEx-S, FB15k-237) or `preserve` (WN18RR, NELL-995), which keeps these paths edge for edge. Under `break`, grey is the surviving half of a path and orange an edge moved with its relation and query-side entity kept (Smekhov is now born in Budapest); a moved edge can open a new path (Mexico), which m_2 counts, so support loss is a lower bound. Footers give m_2 and, where per-query ranks were recorded, the answer's rank under frozen ULTRA-3g. On FB15k-237 `matched` keeps all three paths yet ranks the answer 15th against 10th under `break`, which one query cannot attribute (Figure 2 reads contrasts over all queries). The right column gives, over the whole benchmark, the share of training edges edited, of `clean`'s closed edges lost and of supported test queries left with no two-hop path. Each query is the first in file order with two or three intermediates, all lost under `break` and all kept under the comparison arm, a rule fixed before any rank was read.

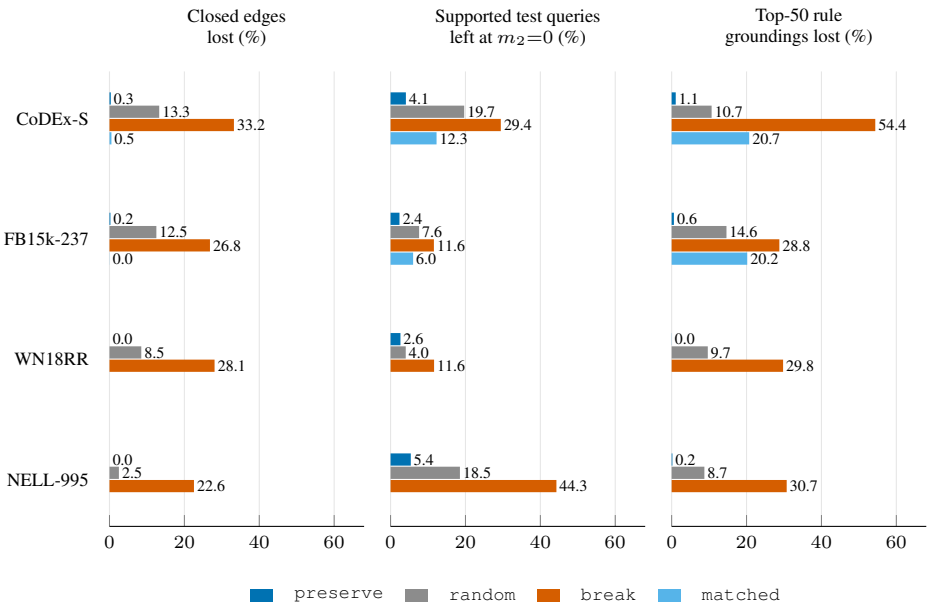


Figure 4: **`break` removes compositional structure on all four graphs; `matched`, at the same edit count, removes almost no closed edges.** Bars are percentages of the `clean` quantity: closed edges lost, supported test queries left with no two-hop path ($m_2 = 0$), and clean groundings of the fifty most frequent two-hop rules lost. `matched` is shown for CoDEx-S and FB15k-237 only. Values are from Table 13; negative closure changes (NELL-995 `preserve`, -0.1%) are drawn at zero.

the mean of the head-side and tail-side statistic for every metric. Only runs with a zero exit status are recorded, and manifests are rebuilt from on-disk metadata before analysis.

Compute. Every run in this paper was produced on one node with two NVIDIA H100 PCIe GPUs (80 GB each), 128 CPU cores and 377 GB of RAM. Retraining a model on one arm costs about 1.8 GPU-hours (mean 6,348 seconds over the nine retraining runs recorded in the release); scoring a frozen checkpoint over a full test set costs about 9 seconds on CoDEx-S and about 100 seconds on FB15k-237; building one arm costs 97–154 seconds of CPU. The cost of adding the controls this paper asks for is therefore a small multiple of one training run. The frozen results use the `ultra_3g` checkpoint throughout. Its pretraining mixture contains FB15k-237, WN18RR and CoDEx-M, so on those graphs "frozen" means that the checkpoint is never trained on the arm, not that the graph is unseen; CoDEx-S, NELL-995 and hetionet are not in it.

Evaluation sets. Two protocols exist, and which one each campaign uses is stated here. The zero-shot four-graph campaign ranks a fixed subset of 1,200 queries drawn once per graph and held identical across arms, so contrasts are paired; that subset is 65.6% of the CoDEx-S test set and 5.9% of FB15k-237's, so the cross-graph comparison holds the number of ranked queries fixed rather than the fraction of each test set. Everything else uses each graph's full test set: the trained four-graph campaign, the matched arm, the ladders, the stratified analysis, *the CoDEx-S reciprocity factorial* (1828 test triples) and *the CoDEx-M reciprocity factorial* (20,622 queries in each of 45 cells). Both halves of the sign reversal in Section D.4 are therefore on the same protocol. The CoDEx-M factorial was first run on the 1,200-query subset, which gave -0.02314 on MRR against -0.02210 on full test; the full-test campaign is the one reported and the subset values are superseded. Pairing scores both arms of a contrast on one query set, so the contrast is well defined and free of any difference in which queries each arm saw; it does not make the contrast invariant to which queries the set contains. The subset is a sample of the test set, and a contrast estimated on it can differ from the full-test contrast in both sampling error and composition. That the two agree to 0.001 is a measured result, not a guaranteed one. One counting convention differs between the two graphs and is worth naming: CoDEx-M's 20,622 counts both directions of its 10311 test triples, while the CoDEx-S coverage figure above counts triples.

Seeds and budgets. Three random variables are varied in this work and are not interchangeable: *arm construction*, which builds the perturbed graph; *model training*, nested inside one arm; and the *rule-miner seed*, which varies the instrument rather than the arm. Every interval we report names which of the three it is over. 3 training seeds per cell in the trained four-graph campaign; 5 arm-construction seeds per design cell in the CoDEx-M factorial; one run per cell in the zero-shot cross-graph campaign and the fixed-budget ladder. Where five seeds are available the CoDEx-M contrast is -0.0233, -0.0241, -0.0228 and -0.0221 at two, three, four and five seeds. The seed noise floor (pooled within-cell SD over the five construction seeds) is 0.0034 for the frozen foundation model and 0.0010 for the embedding models on identical arms; ULTRA's reciprocity contrast is 6.6 times its floor, ComplEx's and DistMult's 1.2 and 1.4 times theirs. Because this floor is a construction-seed quantity, we use it as a conservative proxy where the comparison is over training seeds and say so at each use. Table 14 gives the hyperparameters, read from the released configurations rather than restated from memory.

Table 14: Hyperparameters, read from the released configurations and run scripts rather than restated from memory. They are fixed per family across arms and never tuned per arm, since tuning per arm would itself be a dose.

Family	Configuration
ULTRA (zero-shot frozen)	6 message-passing layers of width 64; `distmult` message, `sum` aggregate; short-cut and layer-norm on; 0 training epochs (frozen checkpoint)
ULTRA (fine-tuned / from scratch)	same architecture; AdamW, learning rate 5×10^{-4}, batch size 64, 256 strict negatives, adversarial temperature 1; ten epochs unless a table says otherwise
ComplEx, DistMult	dimension 500, learning rate 0.1, regularisation 5×10^{-3}, 50 epochs, batch size 1,000
AnyBURL	CPU, 16 worker threads, mining snapshot at 10 seconds, identical budget on every arm
Personalized PageRank	non-learnable; no hyperparameters fitted

Which protocol each number is on. Table 15 gives the quantity, protocol, campaign, regime and n for every reciprocity contrast in this paper, because the same claim appears at several values. The CoDEx-M factorial is reported on the full test set; the subset row is listed only so that the two values can be told apart.

Table 15: **Every reciprocity contrast, with the protocol it was measured on.** Estimator is *paired* where the contrast is taken within a closure-by-seed block and *pooled* where it is a difference of level means. The CoDEx-M subset row is superseded and is listed so that the two published values can be told apart.

quantity	protocol	campaign	estimator	value
CoDEx-S, ULTRA frozen	full test (1828)	factorial	paired	$+0.02578$
CoDEx-S, ULTRA fine-tuned	full test	factorial	pooled	$+0.0175$
CoDEx-S, ULTRA from scratch	full test	factorial	pooled	$+0.0164$
CoDEx-M, ULTRA frozen	full test (20,622)	factorial	paired	-0.02210
superseded	subset (1,200)	factorial	paired	-0.02314
CoDEx-M, ULTRA from scratch	full test	factorial	paired	$+0.000016$
same runs	full test	factorial	pooled	$+0.000204$

C Additional analyses for the edit-matched control and the two-hop split

This section supports Section 4.1 and Section 4.2. It collects the models without a traversal mechanism, the rank-statistic analysis, the robustness and reach of the stratified contrast, replication beyond ULTRA, the negative contrast on queries with no two-hop path, the positive controls, and the training curves.

C.1 Models without a traversal mechanism

This section supports Section 4: it asks how much of the drop appears in models that cannot follow a path at inference.

A bilinear scorer has no query-conditioned traversal mechanism to remove, so if the drop measured that mechanism it should sit near zero. CompGCN does perform message passing, so it controls for query-conditioned traversal specifically, not for every representation of relational structure. Summed over four graphs and three seeds (Table 16), ComplEx reproduces 59.7% of ULTRA's `preserve−break` MRR loss, DistMult 53.2% and CompGCN 50.5%. By mean rank they degrade further than the foundation model (Table 17), and on the three graphs other than CoDEx-S ComplEx reproduces

Table 16: **Models with no query-conditioned traversal mechanism reproduce most of the effect, and they reproduce the recall half far better than the precision half.** `preserve − break` summed over four graphs; each family as a percentage of ULTRA's. **The bilinear controls recover 83–86% of ULTRA's Hits@10 loss and only 44–49% of its Hits@1 loss.** The **Relation-Blind Structural Baseline**, below the rule, shows the same asymmetry in extreme form: 1.4% of the Hits@1 loss against 21.8% at Hits@10.

Family	traverses paths?	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
ULTRA	yes (path-based)	0.2339	0.2316	0.2559	0.2212
ComplEx	**no** (bilinear)	59.7%	48.8%	66.2%	86.2%
DistMult	**no** (bilinear)	53.2%	43.5%	57.7%	82.9%
CompGCN	**no** (query-agnostic MP)	50.5%	44.0%	57.8%	60.4%
PPR (relation-blind)	**no** (topology only)	7.2%	1.4%	−4.6%	21.8%

Four graphs × three seeds; ULTRA's row is absolute, the rest are shares of it. Mean rank is omitted: one excluded run in the CoDEx-S `preserve` arm makes the summed MR share uninterpretable (Section F.1), and ULTRA's own MR effect is negative on WN18RR, so the denominator nearly cancels. The companion `clean − break` contrast is Table 17; the two answer different questions.

103% of ULTRA's summed loss, seed bootstrap [82%, 127%]: parity. The pooled share hides its own spread (per graph ComplEx runs from 46% on CoDEx-S to 217% on WN18RR), so we read it as an ordering, not an estimate. All three fit the perturbed graphs as closely as the clean one (Section C.8). The Relation-Blind Structural Baseline row of the summed `preserve−break` table is not on the same footing: it was run at one construction on the fixed 1,200-query protocol, while its denominator is trained full-test ULTRA, so its shares are indicative only.

Table 17: Summed `clean−break` loss over four graphs (trained four-graph campaign, three seeds), each family as a percentage of ULTRA's loss on the same statistic. MR loss is `break−clean`.

Family	MR	MRR	H@1	H@3	H@10
ULTRA	100%	100%	100%	100%	100%
ComplEx	121%	55%	43%	61%	76%
DistMult	250%	56%	46%	61%	77%
CompGCN	160%	55%	51%	59%	60%

The split across the rank statistic is the one Section 4.1 found. The bilinear controls recover 83–86% of ULTRA's Hits@10 loss and only 44–49% of its Hits@1 loss; the Relation-Blind Structural Baseline, which ignores relation type altogether, shows the asymmetry in extreme form, on NELL-995, which carries essentially all of its summed effect (Section C.5), 1.4% of the Hits@1 loss and 21.8% of the Hits@10 loss. Whatever the intervention does to recall is reproduced by families with no traversal mechanism to lose; an added arm and a substituted model locate the non-compositional component in the same place.

The `clean−break` contrast on five statistics. Table 17 is the companion of Table 16 on the wider contrast that never touches the `preserve` arm. Its fifteen cells are reproduced exactly by the generator of the main tables through an independent pipeline (Section F.1). By mean rank the non-compositional families degrade 121–250% as far as the foundation model.

C.2 Rank-statistic sensitivity

This section supports Section 4 and Section 6: it asks how far the reading depends on which rank statistic is reported.

In four places in this paper the reading changed with the statistic, a second obstacle to the inference, at a different level of the pipeline and cheaper to check. MR, MRR and the Hits levels are different estimands, so disagreement is not itself a defect; it is evidence that the intervention acts unevenly along the rank distribution. The share reproduced by the matched arm is 83.6% at Hits@1 and 36.9% at mean rank (Table 2); the closure cost at a fixed edit budget is +0.0033 on Hits@1 and −0.0050 on Hits@10 (Table 34); on the matched arm of CoDEx-S the contrast is +0.0365 on MRR and +3.55 on mean rank in the direction of *worse* (Table 3), same runs, opposite signs; and the

Table 18: **The reported effect is concentrated in one benchmark, and on the graph where it is smallest the sign depends on the metric.** Zero-shot `preserve − break` for frozen ULTRA on four graphs, single seed. **CoDEx-S alone carries 71% of the summed MRR effect**, and **on WN18RR the contrast is +0.0001 on MRR and −0.0108 on Hits@10**. Both sit within seed noise, so Section C.2.1 does not count this graph as evidence of rank-statistic instability; it is reported here as the power limit it is.

Graph	MRR ↑	H@1 ↑	H@10 ↑	share of total (MRR)
CoDEx-S	**+0.1641**	**+0.1725**	**+0.1425**	71%
FB15k-237	+0.0449	+0.0442	+0.0558	19%
NELL-995	+0.0219	+0.0233	+0.0275	9%
WN18RR	**+0.0001**	**+0.0067**	**−0.0108**	0%
total	**+0.2310**	**+0.2467**	**+0.2150**	100%
total without CoDEx-S	+0.0669	+0.0742	+0.0725	29%

Frozen zero-shot ULTRA, 1,200-query protocol, one construction seed per graph. Mean rank and Hits@3 were not recorded for this campaign. NELL-995 figures are post-correction on the full 149,678-edge graph.

ordering of the four graphs by effect size is shared by MRR and the Hits levels and broken by mean rank (Section C.2.1).

The pattern is not random. MRR and Hits@1 read the head of the rank distribution, MR its tail and Hits@10 the boundary between retrieval and failure, so a perturbation lengthening long ranks without displacing short ones moves one and not the other. Across all 13,122 comparable arm pairs in the record, 24% order two arms differently under MR than under MRR, against 6.1%–11.7% among the Hits levels, and the reversals are most frequent for fine-tuned ULTRA (40% of its pairs; Section C.2.1). Excluding the graphs on which MR is least comparable it rises to 25.47%; against that, the pooled MR-MRR rank correlation of −0.743 is largely a scale artefact, being −0.919 within CoDEx-S (Section C.2.1). A head-versus-tail sign disagreement says the intervention moved the answer's rank in one part of the distribution and not the other, a weaker finding than either statistic alone reports. Where the five agree we say so: the reciprocity contrast of Section D.4 has the same sign on MRR and every Hits level recorded on both graphs (Table 41; mean rank on the frozen CoDEx-S grid is the exception), which is why we trust it and not the four-graph ordering. Agreement is necessary and not sufficient; where two statistics disagree, the effect belongs in the record as acting on part of the rank distribution rather than on ranking.

The reversal rate without mean rank. The reversal rate above includes mean rank. Mean rank is the noisiest statistic in our record and no study in Table 7 reports it for an intervention, so the rate restricted to the statistics the field does report is the one a reader should carry: 6.02% at Hits@1, 8.24% at Hits@3 and 11.61% at Hits@10, against 23.90% for MR. Excluding NELL-995 and WN18RR, which Section C.2.1 says not to count, the figures are 5.48%, 8.24% and 10.81%. The rank-statistic claim therefore has two forms: a strong one that depends on mean rank, and a weaker one (roughly one pair in ten) that does not. We rest the contribution on the weaker one.

C.2.1 Agreement across the five statistics

All five statistics for ULTRA on every arm of the trained four-graph campaign are in the supplementary material. On CoDEx-S and FB15k-237, where the arms separate, all five agree on the arm ordering; on NELL-995 and WN18RR they do not, and this is a power limit rather than a rank-statistic failure: the `preserve − break` contrast is 0.017 and 0.002 there against 0.178 on CoDEx-S, so those arms sit within seed noise of one another and their ordering is whichever statistic one reads. We therefore do not count those two graphs as evidence of rank-statistic instability; Section C.2 rests on the cases where the arms do separate. The four statistics that read the head of the distribution order the four graphs identically on the conventional contrast; mean rank does not, placing NELL-995 above FB15k-237 and turning WN18RR's contrast negative.

Measured over every arm-level cell in the released results (Table 19), rank correlation with MRR is −0.76 for MR and at least 0.93 for each Hits level; 24% of the 13,122 comparable arm pairs order two arms differently under MR than under MRR, against 6.1%–11.7% among the Hits levels. Reversal is not spread evenly: 40% of fine-tuned ULTRA pairs reverse, against 24% of zero-shot ULTRA pairs and 22.6%–23.2% for the embedding families. Discrimination is not free of noise:

the median seed-to-seed range is 2.7% of the cell mean on MR against 1.1% on MRR and 1.8% on Hits@1, so MR is the most responsive of the five and the least stable, which is why we report it beside the others rather than in place of them.

Table 19: Metric disagreement across the released results: Spearman correlation with MRR, and the share of arm pairs whose ordering reverses. Computed over cells carrying both statistics. Restricted to the cells carrying all five statistics, so every column is computed on the same cells; the denominators differ only through ties on the metric compared.

Metric	ρ_s with MRR	Pairs reversed	%
MR	-0.756	3127 / 13122	23.8
H@1	$+0.962$	791 / 12970	6.1
H@3	$+0.991$	1083 / 13042	8.3
H@10	$+0.928$	1527 / 13020	11.7

Pairing denominators. Table 19 reports more comparable pairs for mean rank than for Hits@1, the opposite of the ledger's cell counts, where MR is recorded on fewer cells than Hits@1: a recount over each metric's own cells gives 13197 MR-comparable pairs against 15377 at Hits@1. The table is not counted that way. It is restricted to the cells carrying all five statistics, so every column is computed on the same cells, and the denominators can then differ only through ties: a pair is dropped when its two cells tie on MRR or on the metric being compared. No pair ties on MRR here, and none ties on mean rank, so that column keeps all 13,122 pairs; Hits@1 is the coarsest of the four and ties on 152, the most of any. That is the ordering the table shows, so the printed denominators are correct for the quantity it reports.

C.3 Robustness of the stratified contrast

This section supports Section 4.2: it asks how far the stratified reading depends on the specification, the statistic, degree, and which edges a rule happens to move.

C.3.1 The dose decomposition on individual runs

Table 20 refits the specification of Table 1 on CoDEx-S from individual runs over a wider arm set that adds the ladder rungs. Every closure interval excludes zero; the edit slope exceeds the closure slope for every family; the ratio spans $1.66\times$ to $5.12\times$. It is the fit in which the two slopes can be separated: over these seven arms the variance inflation factor between edit distance and closure is 1.44, and over the four ladder rungs alone 1.01, against the perfect collinearity of a two-arm contrast.

Table 20: Two-dose decomposition on CoDEx-S with 95% intervals, per-seed, $n = 21$, df$= 18$.

Model	edit per 10 pts	closure per 10 pts	R^2	n	df
ULTRA	-0.0627 $[-0.0759, -0.0494]$	$\mathbf{-0.0180}$ $[-0.0281, -0.0079]$	0.925	21	18
ComplEx	-0.0353 $[-0.0405, -0.0301]$	$\mathbf{-0.0069}$ $[-0.0109, -0.0030]$	0.957	21	18
DistMult	-0.0257 $[-0.0312, -0.0201]$	$\mathbf{-0.0155}$ $[-0.0199, -0.0112]$	0.977	18	15
AnyBURL	-0.0631 $[-0.0738, -0.0525]$	$\mathbf{-0.0124}$ $[-0.0205, -0.0044]$	0.944	21	18

C.3.2 The stratified contrast: all five statistics, both contrasts, and its replication

Table 26 is the release's own MRR-only table, which Table 3 reproduces. Table 21 gives both contrasts on all five statistics with bootstrap intervals; it is the source of Figure 2. Table 22 repeats the matched contrast on a second graph and a second training regime: the sign change survives in all four cells (negative at $m_2 = 0$, positive in the top stratum, intervals excluding zero at both poles) but the shape does not. The crossing sits at $m_2 = 1$ everywhere except FB15k-237 retrained, which is still negative at $m_2 = 1$, and FB15k-237 frozen is not monotone at the top. We claim the sign change, not the ordering. One channel does not corroborate: a frozen per-query counterfactual attribution (for each query, the largest drop in the answer's score when one of K sampled edges incident to the query is removed, against the largest drop over K random edges) returned a null once neighbourhood size was controlled ($t = +1.40$ for the exposure cell against $t = -24.19$ for log pool size). On the arm channel the same covariate is inert: conditioning the per-query contrast on

Table 21: **The stratified contrast on all five rank statistics, for the edit-matched contrast and for the conventional contrast.** Same queries, runs and strata as Table 3; every cell is a mean over queries in the stratum with a 95% percentile bootstrap interval (2,000 draws). For MR a positive value means the first-named arm ranks worse. The conventional contrast is steeper at Hits@1 than at Hits@10 in the top stratum (+0.5769 against +0.3042); the matched contrast is not (+0.1136 against +0.2220). The difference between the two panels is the cost of the extra edits; it is larger at Hits@1 than at Hits@10 in three of the four strata with support (not at $m_2 = 1$), the same asymmetry Table 2 reports at the aggregate level.

m_2	n	MR ↓	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
Panel A: `matched − break` *(edit-matched)*						
aggregate	1,828	+3.55 [-0.7,+8.2]	+0.0365 [+0.026,+0.047]	+0.0200 [+0.008,+0.032]	+0.0583 [+0.043,+0.072]	+0.0703 [+0.055,+0.086]
0	826	+16.47 [+9.3,+23.6]	−0.0220 [-0.034,-0.010]	−0.0266 [-0.041,-0.013]	−0.0139 [-0.031,+0.002]	−0.0121 [-0.030,+0.006]
1	254	−8.83 [-22.3,+4.0]	+0.0207 [-0.004,+0.046]	+0.0098 [-0.020,+0.039]	+0.0394 [+0.002,+0.077]	+0.0394 [-0.002,+0.083]
2–5	248	−1.54 [-17.4,+15.0]	+0.0499 [+0.019,+0.082]	+0.0403 [+0.002,+0.077]	+0.0504 [+0.012,+0.087]	+0.0907 [+0.046,+0.131]
6–20	214	−10.86 [-19.0,-1.9]	+0.1061 [+0.072,+0.140]	+0.0631 [+0.019,+0.107]	+0.1589 [+0.115,+0.203]	+0.1986 [+0.152,+0.248]
21+	286	−7.58 [-9.1,-6.3]	+0.1557 [+0.125,+0.188]	+0.1136 [+0.077,+0.152]	+0.2150 [+0.164,+0.266]	+0.2220 [+0.177,+0.266]
Panel B: `preserve − break` *(the conventional contrast)*						
aggregate	1,828	−11.77 [-15.8,-7.5]	+0.1524 [+0.138,+0.166]	+0.1584 [+0.141,+0.176]	+0.1685 [+0.151,+0.185]	+0.1294 [+0.114,+0.144]
0	826	+6.76 [+0.1,+13.9]	−0.0096 [-0.022,+0.003]	−0.0139 [-0.030,+0.001]	−0.0042 [-0.021,+0.012]	+0.0121 [-0.005,+0.029]
1	254	−38.51 [-51.2,-25.4]	+0.0876 [+0.061,+0.115]	+0.0807 [+0.047,+0.114]	+0.0846 [+0.045,+0.122]	+0.1142 [+0.077,+0.155]
2–5	248	−37.94 [-50.7,-25.0]	+0.1779 [+0.142,+0.215]	+0.1734 [+0.131,+0.218]	+0.1855 [+0.139,+0.232]	+0.1935 [+0.153,+0.238]
6–20	214	−22.64 [-32.6,-13.1]	+0.3375 [+0.295,+0.380]	+0.3388 [+0.290,+0.393]	+0.3879 [+0.334,+0.437]	+0.2921 [+0.245,+0.339]
21+	286	−10.68 [-12.2,-9.2]	+0.5172 [+0.485,+0.553]	+0.5769 [+0.531,+0.619]	+0.5629 [+0.521,+0.608]	+0.3042 [+0.268,+0.344]

Table 22: The sign change replicates across graphs and training regimes. Each entry is the matched−break contrast in reciprocal rank within an m_2 stratum, with a 95% bootstrap interval over 2000 resamples. Every cell is negative at $m_2 = 0$ and positive in the top stratum.

graph	regime	stratum	n	matched−break
CoDEx-S	frozen	0	826	−0.0220 [−0.0335, −0.0103]
		1	254	+0.0207 [−0.0040, +0.0442]
		2–5	248	+0.0499 [+0.0188, +0.0837]
		6–20	214	+0.1061 [+0.0705, +0.1391]
		21+	286	+0.1557 [+0.1244, +0.1867]
CoDEx-S	retrained	0	826	−0.0283 [−0.0404, −0.0155]
		1	254	+0.0289 [+0.0060, +0.0530]
		2–5	248	+0.0940 [+0.0607, +0.1295]
		6–20	214	+0.2005 [+0.1610, +0.2384]
		21+	286	+0.2022 [+0.1665, +0.2371]
FB15k-237	frozen	0	5352	−0.0266 [−0.0309, −0.0224]
		1	4136	+0.0061 [+0.0016, +0.0104]
		2–5	6840	+0.0265 [+0.0222, +0.0307]
		6–20	3642	+0.0298 [+0.0241, +0.0354]
		21+	496	+0.0279 [+0.0155, +0.0403]
FB15k-237	retrained	0	5352	−0.0411 [−0.0455, −0.0366]
		1	4136	−0.0034 [−0.0082, +0.0014]
		2–5	6840	+0.0286 [+0.0240, +0.0331]
		6–20	3642	+0.0401 [+0.0334, +0.0461]
		21+	496	+0.0498 [+0.0337, +0.0658]

log pool size leaves m_2 at $t = 7.02$ against $t = 0.51$ for pool size itself (Section C.3.3). The two channels disagree, and both belong in the record.

Five independent constructions of the matched arm. The replication above varies the graph and the training regime at one arm construction each. Table 23 varies the construction: five independent builds of the matched arm per graph, each with its own per-query rank dump, scored against the same `break` arm and stratified by the same pre-treatment m_2. The first CoDEx-S construction is Table 3. In every one of the ten graph-by-construction cells the matched arm is worse at $m_2 = 0$ and better in the top stratum, on every statistic, with one exception (Hits@10, CoDEx-S construction 5, +0.0018). The monotone ordering is a CoDEx-S phenomenon and does not travel.

Table 23: **The stratified contrast replicates across 5 independent constructions of the matched arm, on two graphs and all five rank statistics.** Each cell is the mean over the constructions with the range across them in brackets; Table 3 is construction 1 of the CoDEx-S block. **The matched arm is worse at $m_2 = 0$ and better in the top stratum in every one of the 50 graph-by-construction-by-statistic cells**, with a single exception (CoDEx-S, Hits@10, construction 5, $+0.0018$). The monotone ordering is a CoDEx-S phenomenon: it holds in 5 of 5 constructions there on MRR, Hits@1, Hits@10, and in at most 2 of 5 on FB15k-237. We claim the sign change, not the ordering.

m_2	n	MR ↓	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
CoDEx-S, 5 constructions						
aggregate	1,828	$+3.80$ [+2.3,+5.5]	$+0.0366$ [+0.033,+0.042]	$+0.0205$ [+0.017,+0.024]	$+0.0601$ [+0.051,+0.073]	$+0.0690$ [+0.063,+0.078]
0	826	$+16.05$ [+13.5,+18.0]	-0.0262 [-0.031,-0.022]	-0.0321 [-0.037,-0.027]	-0.0180 [-0.030,-0.010]	-0.0096 [-0.015,+0.002]
1	254	-3.36 [-8.8,+0.1]	$+0.0198$ [+0.005,+0.029]	$+0.0098$ [-0.006,+0.018]	$+0.0307$ [+0.010,+0.043]	$+0.0382$ [+0.033,+0.041]
2–5	248	-1.68 [-7.0,+3.0]	$+0.0410$ [+0.022,+0.059]	$+0.0319$ [+0.012,+0.050]	$+0.0427$ [+0.022,+0.065]	$+0.0782$ [+0.054,+0.095]
6–20	214	-13.24 [-16.5,-10.7]	$+0.1083$ [+0.098,+0.115]	$+0.0706$ [+0.063,+0.079]	$+0.1528$ [+0.133,+0.159]	$+0.1883$ [+0.173,+0.203]
21+	286	-7.68 [-8.0,-7.5]	$+0.1756$ [+0.156,+0.184]	$+0.1346$ [+0.114,+0.145]	$+0.2573$ [+0.215,+0.287]	$+0.2262$ [+0.220,+0.240]
FB15k-237, 5 constructions						
aggregate	20,466	-3.67 [-4.8,-2.8]	$+0.0091$ [+0.008,+0.010]	$+0.0073$ [+0.006,+0.008]	$+0.0096$ [+0.009,+0.011]	$+0.0152$ [+0.014,+0.016]
0	5,352	$+12.64$ [+7.9,+16.6]	-0.0247 [-0.027,-0.022]	-0.0252 [-0.027,-0.023]	-0.0271 [-0.030,-0.023]	-0.0200 [-0.021,-0.017]
1	4,136	-2.54 [-4.9,-0.5]	$+0.0059$ [+0.004,+0.008]	$+0.0056$ [+0.004,+0.008]	$+0.0045$ [+0.003,+0.005]	$+0.0085$ [+0.003,+0.011]
2–5	6,840	-14.05 [-14.9,-12.3]	$+0.0259$ [+0.024,+0.027]	$+0.0222$ [+0.020,+0.024]	$+0.0281$ [+0.026,+0.031]	$+0.0369$ [+0.036,+0.038]
6–20	3,642	-9.02 [-10.8,-7.4]	$+0.0285$ [+0.025,+0.032]	$+0.0266$ [+0.022,+0.031]	$+0.0322$ [+0.027,+0.036]	$+0.0297$ [+0.026,+0.034]
21+	496	-6.85 [-8.2,-5.7]	$+0.0260$ [+0.019,+0.037]	$+0.0240$ [+0.019,+0.030]	$+0.0256$ [+0.015,+0.038]	$+0.0456$ [+0.035,+0.067]

`matched` − `break`, ULTRA zero-shot frozen; strata are pre-treatment m_2 on the clean arm, identical across constructions. Monotone in k of 5 constructions: CoDEx-S: MRR 5, Hits@1 5, Hits@3 4, Hits@10 5, MR 0; FB15k-237: MRR 1, Hits@1 2, Hits@3 1, Hits@10 0, MR 0.

C.3.3 Degree control

A reviewer may object that m_2 is a proxy for entity degree: high-degree queries have more two-hop paths and larger candidate pools, so the gradient of Table 3 could be a degree gradient. We partition the same 1,828 queries by $\deg_h \times \deg_t$, both measured on the *clean* training graph and therefore pre-treatment, into terciles cut at 1,547 and 6,504 (sizes 609/607/612), and recompute the edit-matched contrast within each cell (Table 24). At $m_2 = 0$ the contrast is negative in all three terciles (-0.0253, -0.0181 and -0.0221) and every bootstrap interval over the evaluation population excludes zero. A per-query regression of the contrast on m_2 with log pool size as a covariate gives m_2 a coefficient of $+0.00264$ at $t = 7.02$, against $t = 0.51$ for pool size itself; m_2 alone gives $+0.00275$, so conditioning on degree attenuates it by four percent. The two covariates are correlated at $+0.587$, a variance inflation factor of 1.53, so the two coefficients are separately fittable here. The monotone *ordering* does not survive the partition ($m_2 = 1$ in the top tercile is -0.0393 against $+0.0207$ overall), which is consistent with Section 8: we claim the sign change, not the ordering.

Table 24: **The $m_2 = 0$ residual survives conditioning on degree.** Edit-matched contrast `matched` − `break` by m_2 stratum and by tercile of $\deg_h \times \deg_t$ on the `clean` graph. Bold: 95% percentile bootstrap interval over 2,000 resamples of the evaluation population excludes zero.

m_2	tercile	n	MRR ↑	95% CI	H@1 ↑
0	all	826	-0.0220	$[-0.034, -0.010]$	-0.0266
	T1	377	$\mathbf{-0.0253}$	$[-0.046, -0.006]$	-0.0305
	T2	320	$\mathbf{-0.0181}$	$[-0.036, -0.001]$	-0.0203
	T3	129	$\mathbf{-0.0221}$	$[-0.045, -0.001]$	-0.0310
1	all	254	$+0.0207$	$[-0.003, +0.046]$	$+0.0098$
	T1	110	$+0.0365$	$[-0.000, +0.076]$	$+0.0136$
	T2	88	$+0.0393$	$[-0.005, +0.086]$	$+0.0398$
	T3	56	-0.0393	$[-0.081, +0.002]$	-0.0446
2–5	all	248	$+0.0499$	$[+0.021, +0.080]$	$+0.0403$
6–20	all	214	$+0.1061$	$[+0.071, +0.139]$	$+0.0631$
21+	all	286	$+0.1557$	$[+0.124, +0.187]$	$+0.1136$

C.3.4 Edge selection at a fixed budget

The Edit-Matched Arm holds two quantities: edit count and closure. It does not hold the selection of edges to displace. We test whether that matters by building two further arms to the same budget with the acceptance rule weighted by incident degree (one toward high-degree triples, one toward

low) at three construction seeds each, and scoring all six on the frozen checkpoint under the protocol of Section B.2.

The arms are matched on what the design controls. Edit distance is 24.89% against 24.92%; closure is $14{,}148$ against $14{,}147$ triples and agrees to 0.16% on the instance-level counter as well; the triple count and the fraction of test queries with any two-hop support are unchanged (Table 25). They are not matched on the outcome: the share reproduced by the matched arm is 79.5% under high-degree selection and 66.7% under low, a gap of 12.8 points with a standard deviation of 1.15 across the three paired seeds, an order of magnitude above the construction spread of 1.2 points reported above for the released arm.

What differs is compositional structure that closure does not count. Triangles are $64{,}445$ under high-degree selection against $69{,}346$ under low, a difference of 7.1%; transitivity is 0.0574 against 0.0616; mean two-hop support per test query is 4.92 against 4.97. On all three the more depleted arm is the more damaged arm, and the low-degree arm ends with slightly *more* triangles than `clean` while paying a 25% edit budget.

Two candidate explanations fail a sign test and we record the failure rather than the correlation. Reciprocity is 0.1399 under high-degree selection and 0.1320 under low: the high-degree arm loses *less* reciprocity from `clean` and is nonetheless the more damaged, so reciprocity cannot mediate this gap. Assortativity behaves the same way and moves by 0.6%. Neither closure counter separates the arms at all.

We therefore read this as a limit on what closure measures rather than as a further confound. Closure counts triples participating in a closed two-path; it is matched here to a sixth of a percent while the structure the model appears to use moves by seven. This is the dose-side analogue of Section C.2: there a scalar rank statistic projects a per-query distribution, here a scalar dose projects a structured treatment, and the projection is not the informative one. It also supplies the obvious reading of Table 34, where the first twenty-one points of closure loss cost nothing and the last third costs everything, a threshold in closure is what one expects if closure is not the operative quantity. The limit: six arms on one graph, and triangles, transitivity and m_2 are mutually correlated, so we identify a family rather than a member.

Table 25: **Matched on the dose, unmatched on the structure.** Two arms at `break`'s edit budget with closure held at the `clean` value, differing only in whether the acceptance rule favours high- or low-degree triples; means over three construction seeds each. The final column records whether the quantity moves in the direction that would explain the damage gap. The $m_2 > 0$ row uses this study's own per-arm counter, whose `clean` value is not reported because the released code does not reproduce it. The row is read as a within-table comparison between the two selection arms only, and their difference is unaffected.

quantity	`clean`	high-degree	low-degree	explains the gap?
held by construction				
edit distance (%)	0.00	24.89	24.92	—
closure (triples)	14,288	14,148	14,147	no signal
closure (instances)	179,619	171,113	170,831	no signal
training triples	32,888	32,888	32,888	—
fraction $m_2 > 0$	—	0.3881	0.3889	no signal
free, and moving				
triangles	67,571	64,445	69,346	**yes**
transitivity	0.0612	0.0574	0.0616	**yes**
mean m_2 per query	5.225	4.915	4.972	**yes**
reciprocity	0.1555	0.1399	0.1320	no (sign reversed)
assortativity	−0.2595	−0.2566	−0.2580	no (sign reversed)
share reproduced (%)	—	**79.5**	**66.7**	gap +12.8

C.4 Reach of the dose, per graph and family

This section supports Section 4.1: it reports the contrast for every graph and family, how far each dose reaches, and what the test-leakage residue is worth (Section C.4.2).

Table 26: The edit-matched contrast stratified by pre-treatment two-hop support, MRR only, as generated in the release; $n = 1{,}828$, CoDEx-S, frozen zero-shot ULTRA.

m_2	n	matched	break	matched − break
0	826	0.2409	0.2629	−0.0220 [−0.0335, −0.0103]
1	254	0.2864	0.2657	+0.0207 [−0.0040, +0.0442]
2–5	248	0.3601	0.3102	+0.0499 [+0.0188, +0.0837]
6–20	214	0.4217	0.3156	+0.1061 [+0.0705, +0.1391]
21+	286	0.4514	0.2957	+0.1557 [+0.1244, +0.1867]

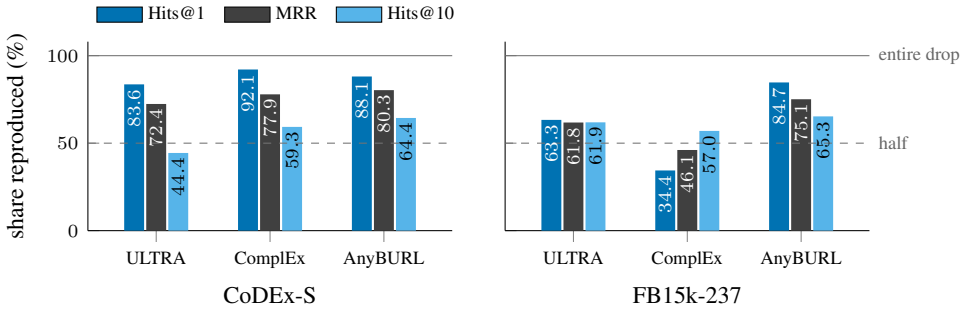


Figure 5: **An edit-matched control reproduces at least half of the drop in 15 of 18 cells.** Share of the conventional contrast reproduced by the Edit-Matched Arm, from Table 2 Panel B (fine-tuned campaign, three seeds, full test). On CoDEx-S the share falls from Hits@1 to MRR to Hits@10 for every model. The three cells below half (dashed line) are ULTRA at Hits@10 on CoDEx-S and ComplEx at Hits@1 and MRR on FB15k-237.

C.4.1 The conventional contrast per graph and family

Table 27 gives `preserve − break` for all sixteen graph-family cells in the trained campaign. The two graphs whose contrast applies the larger edit ratio (Table 9) carry the two largest effects in every family; on WN18RR the bilinear models show a larger contrast than the foundation model, and the per-graph ComplEx-to-ULTRA ratio runs from 46% on CoDEx-S to 217% there, which is why Table 16 is read as an ordering rather than an estimate. Part of the spread across graphs is dose delivery. Table 28 counts, on each clean message-passing graph, the test queries that carry any two-path from head to answer: the intervention can reach 73.8% of FB15k-237's test set and 54.8% of CoDEx-S's, but only 23.0% of NELL-995's and 18.1% of WN18RR's, a factor of 4.1. The four averages in Table 18 are therefore not four measurements of one quantity at one strength; each is attenuated by how much of its own test set the operator can act on. NELL-995 ships its message-passing edges in a separate file; a screen that reads the training file alone sees 37,420 of 149,678 edges and reports 96.1% of the test set unsupported, a property of the file layout and not of the graph (Section F.1).

Table 27: `preserve − break` (MRR) by graph and family, trained four-graph campaign, three training seeds.

Graph	ULTRA	CompGCN	ComplEx	DistMult
CoDEx-S	+0.1777	+0.0714	+0.0816	+0.0885
FB15k-237	+0.0370	+0.0352	+0.0323	+0.0324
WN18RR	+0.0018	+0.0027	+0.0039	+0.0074
NELL-995	+0.0174	+0.0089	+0.0216	−0.0038

Table 28: Reach of the intervention: test queries carrying compositional support in the clean arm. The two support columns differ only where the graph carries self-loops.

Graph	Test queries	Compositionally supported		Self-loops
		any two-path	non-degenerate	
CoDEx-S	1,828	1,002 (54.8%)	1,002 (54.8%)	0
FB15k-237	20,466	15,114 (73.8%)	15,114 (73.8%)	1,625
NELL-995	2,818	677 (24.0%)	648 (23.0%)	42
WN18RR	3,134	568 (18.1%)	568 (18.1%)	7

C.4.2 TEST-LEAKAGE RE-SCORING CONTROL

The swap operator can create a triple that coincides with a test triple, so the perturbed arms are not leak-free even though the `clean` arms are (Table 29, lower block): 35 of CoDEx-S's 1,828 test queries under `preserve`, 90 under `break` and 71 under `matched`; 453, 378 and 729 of FB15k-237's 20,466. These counts reproduce the retained test files of the re-scoring control exactly. NELL-995's leakage lists are empty for every arm and WN18RR's hold a single triple, so neither graph needs re-scoring; they also have no per-query rank dump.

A contrast is only meaningful if both arms are scored on the same queries, so we exclude a query whose triple is leaked in any of `preserve`, `break` and `matched` and re-score every contrast on the one retained set that results, 175 queries dropped on CoDEx-S (9.6%) and 1,294 on FB15k-237 (6.3%). Table 29 gives the result on all five statistics. Nothing of consequence moves. The largest change to any contrast on any statistic is 0.0115 (CoDEx-S `preserve`−`break` on Hits@3, $0.1685 \rightarrow 0.1570$); on MRR the largest is 0.0066 ($0.1524 \rightarrow 0.1458$, same contrast). The edit-matched contrast (the one the paper's claim rests on) moves by 0.0002 on MRR on both graphs. Mean rank, on a different scale, moves by up to 1.24 places against contrasts of -11.8 to -22.7. No sign changes anywhere, and the share reproduced by the matched arm on these rank dumps moves from 76.1% to 74.9% on CoDEx-S and from 78.7% to 79.4% on FB15k-237. The contamination objection is answered by measurement rather than by construction.

Table 29: **Excluding every leaked test query changes no sign and moves no contrast by more than 0.012 on any rate statistic.** The swap operator can create a triple that coincides with a test triple: the perturbed arms carry the leaked-query counts below, against zero on both `clean` arms. A query is excluded if its triple is leaked in *any* of `preserve`, `break` and `matched`, so both contrasts are scored on one common retained set and stay paired. Mean rank is on a different scale and moves by up to 1.24 places, which is small beside the contrasts themselves (-11.8 to -22.7). ULTRA zero-shot frozen, per-query score as in Table 3.

Graph	Contrast	queries	MR ↓	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
CoDEx-S	`preserve` − `break`, full test	1,828	−11.77	+0.1524	+0.1584	+0.1685	+0.1294
	leak-free	1,653	−12.49	+0.1458	+0.1525	+0.1570	+0.1285
	`matched` − `break`, full test	1,828	+3.55	+0.0365	+0.0200	+0.0583	+0.0703
	leak-free	1,653	+4.31	+0.0367	+0.0236	+0.0532	+0.0659
FB15k-237	`preserve` − `break`, full test	20,466	−21.49	+0.0427	+0.0402	+0.0442	+0.0505
	leak-free	19,172	−22.73	+0.0450	+0.0421	+0.0465	+0.0534
	`matched` − `break`, full test	20,466	−2.82	+0.0091	+0.0077	+0.0091	+0.0143
	leak-free	19,172	−2.53	+0.0093	+0.0078	+0.0090	+0.0147
leaked test queries, by arm							
CoDEx-S	`clean` 0, `preserve` 35, `break` 90, `matched` 71	union 175	9.57% of 1,828 test queries excluded				
FB15k-237	`clean` 0, `preserve` 453, `break` 378, `matched` 729	union 1,294	6.32% of 20,466 test queries excluded				

Leaked-query counts reproduce the retained test files of the Test-Leakage Re-scoring Control exactly (CoDEx-S: $1{,}828 - 90 = 1{,}738$ for `break`). On these rank dumps the MRR nuisance share moves from 76.1% to 74.9% on CoDEx-S and from 78.7% to 79.4% on FB15k-237 when the leaked queries are dropped.

C.5 REPLICATION BEYOND ULTRA

This section supports Section 4 and Section 3: it asks whether the pattern survives a second foundation model and a relation-blind baseline.

C.5.1 CROSS-ARCHITECTURE KGFM REPLICATION

The arms in the body are built by us, on graphs we also criticise, by editing the training graph and retraining. Table 30 reports the same edit-matched contrast under none of those conditions: `hetionet_E`, a biomedical graph on the entity-inductive split of Shomer et al. (2025), with HYPER and MOTIF evaluated zero-shot on frozen checkpoints, ported and run independently. Holding edit distance at `break`'s level while preserving compositional support leaves a residual of +0.0418 for HYPER and +0.0400 for MOTIF against conventional contrasts of +0.1173 and +0.1049; the two models agree on the residual to within 0.0018. The share that survives matching is larger here (36% and 38%) than for frozen ULTRA-3g on CoDEx-S (24%, the complement of its 76.0% share), so the confound is not an artefact of the graph the body uses. The estimand is not the same (these arms perturb the inference graph, whose entities are unseen in training), so the two sets of numbers should not be pooled. One build seed per arm; no interval is reported.

Table 30: Zero-shot MRR per arm on `hetionet_E`. $\Delta_{\mathrm{p-b}}$ is the conventional contrast; $\Delta_{\mathrm{m-b}}$ is what survives holding edit distance at `break`'s level.

Model	`clean`	`preserve`	`matched`	`break`	$\Delta_{\mathrm{p-b}}$	$\Delta_{\mathrm{m-b}}$
HYPER	0.5310	0.3331	0.2577	0.2159	+0.1173	+0.0418
MOTIF	0.4980	0.3172	0.2523	0.2123	+0.1049	+0.0400

C.5.2 The relation-blind structural baseline and the shortest-path dose

Rewiring edges changes shortest-path structure, and Shomer et al. (2025) show that ΔSPD predicts the performance of Personalized PageRank across datasets. Our operators move it: on CoDEx-S ΔSPD goes from $+0.114$ under `preserve` to -0.090 under `break`, changing sign. It does not drive the result. Under `preserve` − `break` on CoDEx-S the baseline loses 0.0024 MRR against ULTRA's 0.1641 on the same protocol; within the factorial, ΔSPD is flat across reciprocity levels. On all five statistics the baseline responds 33% on mean rank against 4.5% on reciprocal rank, and on NELL-995 its mean rank *improves* by 39% under `break`: a statistic on which a compositionally inert scorer moves this much is measuring the edit budget, not the mechanism. The summed asymmetry quoted in Section C.1 is carried almost entirely by this graph: NELL-995 contributes the whole of the Hits@1 difference and 97% of the Hits@10 difference, and on CoDEx-S the baseline's Hits@1 is zero in both arms. Two of the four graphs are heavily shortcut-contaminated in the sense of Shomer et al. (2025) (the baseline alone reaches 0.2801 MRR on NELL-995 and 0.1735 on WN18RR) and these are the two graphs on which the conventional effect is smallest. The baseline was run at a single construction seed on the fixed-subset protocol, so every Hits figure for it is a multiple of $1/1200$.

C.6 The negative contrast on queries with no two-hop path

The -0.0220 MRR residual at $m_2 = 0$ (Section 4.2) is the paper's least intuitive number, and three mechanisms have been proposed to explain it away. None of the three does. A fourth, that the acceptance rules place edits in different neighbourhoods in the two arms, is not tested here.

Spurious closure created by the construction. A swap can create a two-hop path where the clean graph had none, so a query counted at $m_2 = 0$ might not be at $m_2 = 0$ in the arm it is scored on. This happens, and it is small. Of the 826 CoDEx-S queries with clean $m_2 = 0$, `preserve` creates support for 17, `break` for 36, `random` for 40 and `matched` for 41. The contrast is `matched` against `break`, and the two sets overlap in 8 queries, so support differs for 61 of the 826: 33 gain it under `matched` alone and 28 under `break` alone. Dropping all 69 queries that either arm gives support moves the in-stratum contrast from -0.0220 to -0.0298 on the remaining 757, so the residual does not rest on the queries whose support the operator created.

Differential edit exposure. The matched arm might displace more edges incident to the $m_2 = 0$ queries than `break` does. Within the $m_2 = 0$ stratum we cut queries into terciles of per-query edit exposure and recompute: -0.0311, -0.0102 and -0.0246. The contrast is negative in all three and non-monotone in exposure; exposure and contrast correlate at -0.0029. Whatever produces the residual is not tracking how many edits a query saw.

Test leakage. 29 of the 826 queries are leaked into at least one of the two arms, 10 under `break` and 22 under `matched`; over the full test set the leaked counts are 90 under `break`, 71 under `matched` and 35 under `preserve`, and the common evaluation set excludes the union of all three, 175 of 1,828. Removing the 29 from both arms moves the in-stratum contrast from -0.0220 to -0.0230, and removing them together with the created-support queries leaves -0.0296 on 731.

What this leaves. Three candidate confounds are measured and none accounts for the residual, which strengthens the negative control rather than the explanation: we still cannot say what the residual *is*. It is a real, replicated difference between two arms on queries with no two-hop support to lose, and the honest reading is that the intervention changes something about these queries that neither closure nor edit count nor leakage names. One candidate fits the pattern and is untested: a rule that targets closure also decides where edits land, so `break`'s edits sit in closed neighbourhoods while `preserve` and `matched` must place theirs elsewhere. On these queries `break` indeed

scores above `preserve` (0.263 against 0.253 on CoDEx-S, 0.268 against 0.250 on FB15k-237). Measuring each arm's local edit exposure by stratum would test it. That is the paper's thesis applied to the paper's own result.

C.7 Positive control and rule counts

AnyBURL is run on CPU, three mining seeds per graph-arm cell, with identical mining budgets across arms; the variance this supports is over the instrument, not the arm. CoDEx-S `clean` yields MRR 0.4263 against ULTRA's 0.4823 and ComplEx's 0.3877: a credible baseline, not a crippled control. Its shares are in Table 2. Two further uses. First, the intended dose is real and countable: AnyBURL recovers compositional rules whose support the intervention removes, and on CoDEx-S the compositional subset collapses from 694 rules under `clean` to 133 under `break` while the total mined moves far less; on FB15k-237, where the effect on accuracy is an order of magnitude smaller, it falls only from 4,500 to 3,444 (Table 31). Second, a positive control: a scorer built from counted rules (definitionally compositional) evaluated on the 15,114 of 20,466 FB15k-237 test queries that are compositionally supported gives `preserve` − `break` = +0.0310, 10.1% of the clean score. Random rewiring at the same edit budget costs this scorer almost as much, so the check shows that the dose is real, not that the two-graph test separates it from the edit dose; the edit-matched control on scorers that compose by construction follows.

Table 31: AnyBURL scores per arm beside the number of rules mined and the compositional subset of them.

Graph	Arm	MRR	Rules	Compositional
CoDEx-S	`clean`	0.4263	618,112	694
CoDEx-S	`preserve`	0.3795	509,540	760
CoDEx-S	`preserve_matched`	0.2579	452,433	597
CoDEx-S	`break`	0.2281	557,997	133
CoDEx-S	`random`	0.3190	430,125	508
FB15k-237	`clean`	0.2865	162,323	4,500
FB15k-237	`preserve`	0.2660	190,196	4,369
FB15k-237	`preserve_matched`	0.2375	187,442	4,710
FB15k-237	`break`	0.2281	165,436	3,444
FB15k-237	`random`	0.2392	169,425	4,065

The edit-matched control on scorers that compose by construction. To calibrate the control, we rerun AnyBURL on every arm of CoDEx-S and FB15k-237 (three mining seeds for `clean`, `preserve` and `break`, one per matched construction, the same 10-second budget) and apply three subsets of the same learned rules: all rules; cyclic rules with a body of two or three atoms, which predict (h, r, t) only through a path from h to t; and cyclic rules of two atoms alone, which predict only through a two-hop path, the structure `break` removes. A scorer of the last kind composes by construction. For it, the Edit-Matched Arm reproduces 35.6% of the drop on CoDEx-S (29.8–42.4% across the five constructions) and 35.7% on FB15k-237, against 76.0% and 78.7% for frozen ULTRA-3g. Admitting three-atom paths raises the share to 58.9% and 65.6%, because `break` spares most three-hop paths (Section C.7); in this rerun, under our classification of the rules, cyclic two-atom rules on CoDEx-S fall from 682 to 117 but three-atom rules only from 1,474 to 1,273. With all rules, most of which are not paths, the share is 78.4% and 73.7%, close to ULTRA's. The control therefore separates a two-hop compositional scorer from ULTRA, and ULTRA's share is the one shown by a scorer that mostly does not predict through two-hop paths. The matched arm is not free for the compositional scorer either: it lowers cyclic two-atom rules on CoDEx-S from 682 to 553, since closure is counted on untyped edges while rules are typed, so the share is an upper bound for a purely two-hop scorer. We also train a MINERVA-style walker, the policy-gradient model of Das et al. (2018) as implemented by Lin et al. (2018), on the same arms of CoDEx-S: an agent starts at h, takes up to three steps along edges and is rewarded for stopping at the answer, so its prediction is a path. The walker is scored on a protocol of its own, which we state because it differs from the one the other campaigns use: one tail query $(h, r, ?)$ per test triple (1,828 on CoDEx-S, 20,466 on FB15k-237), with no head queries and no reciprocal expansion; all other known answers for (h, r) masked, including those reachable through the inverse edges the runner adds; and the rank taken within the top 128 scored candidates, so a target the beam never reaches contributes zero to MRR and to every Hits while still counting in the denominator. The supplementary material carries the

query list, the entity and relation mappings and these settings. Across the thirty-two CoDEx-S runs the test and validation files are byte-identical, the arms share one entity set and one relation set, and no test query occurs in any training graph. It reaches MRR 0.477 on `clean`, and the Edit-Matched Arm reproduces 83.7% of its drop (60.0–107.1% across the five constructions), close to ULTRA's share rather than to the two-hop scorer's. Two things limit what this adds. The agent may walk three hops, and `break` leaves three-hop paths in place (Section C.7), so it can route around the edges the operator removes; and $\kappa = 0.011$ MRR here is of the same order as the training-seed difference we measure on `clean`, `preserve` and `break` (≤ 0.007 MRR over two seeds), so its share is imprecise. Read together, the three scorers say that the control's answer tracks how much of a model's mechanism runs through the two-hop paths the operator removes, not how compositional the model is in general.

The hop budget of the walker. The walker's reach is a parameter, and the share the Edit-Matched Arm reproduces moves with it (Table 32). On CoDEx-S, where five matched constructions exist at every budget, ρ is 77.9% at one step, 51.7% at two, 83.7% at three and 71.9% at four, and the two-step range, 46.6–55.0%, is disjoint from the range at every other budget. A two-step walker is the case in which the agent's mechanism and the structure the operator removes coincide, and it is there that an edit-matched control reproduces least of the drop. FB15k-237 puts its minimum at two steps as well, 23.3% against 62.9% at three and 64.1% at four, on one construction and one training seed per budget; at one step every arm is at the floor (0.012–0.014 MRR) so no contrast is defined there. Its four budgets were run under one early-stopping rule, which the CoDEx-S ladder was not: the three-hop CoDEx-S runs used a patience of 10 against 6 at the other budgets, and at one of the three training seeds the matched arm was stopped at 6 while `preserve` and `break` used 10. The three-step seed range below is set by the two seeds whose three arms share a stopping rule. Two cautions. As the budget grows `preserve` and `break` converge – the gap is 0.238 MRR at two steps on CoDEx-S and 0.070 at three – so ρ's denominator shrinks and the estimate loses precision: across three training seeds ρ spans 46.5–55.5% at two steps but 68.9–94.3% at three. And the ranges separate between one and two steps and between two and three, but not between three and four, so we read the two-step minimum and not a trend.

Table 32: Hop budget of the MINERVA-style walker. ρ is the share of the `preserve−break` contrast in test MRR that the Edit-Matched Arm reproduces; the three MRR columns give the values it is formed from. CoDEx-S rows are the mean over the five matched constructions at training seed 1, with the range over those five; FB15k-237 has one construction and one training seed per budget.

Graph	hops	`preserve`	`matched`	`break`	ρ (%)	range over constructions	n
CoDEx-S	1	0.143	0.084	0.067	77.9	76.3–80.3	5
	2	0.344	0.221	0.106	51.7	46.6–55.0	5
	3	0.430	0.371	0.360	83.7	60.0–107.1	5
	4	0.414	0.361	0.340	71.9	64.9–81.1	5
FB15k-237	1	0.013	0.014	0.012	n/a	all arms at the floor	1
	2	0.262	0.245	0.189	23.3	one construction	1
	3	0.340	0.318	0.305	62.9	one construction	1
	4	0.322	0.297	0.283	64.1	one construction	1

Three-hop paths. m_2 and closure are two-hop quantities. On each arm we also record whether a test query keeps a three-hop path h–a–b–t (Table 33). `break` leaves one to 295 of the 295 CoDEx-S queries it strips of every two-hop path, and to 1744 of 1748 on FB15k-237. The modification is therefore a two-hop intervention, and a query with no path in Section 4.2 has no two-hop path; it is not disconnected. On the queries that keep only a three-hop path, ULTRA-3g still loses MRR under `break`, and the Edit-Matched Arm accounts for most of that loss. Composition over three or more hops is not tested by this design; it would need an operator that removes longer paths as well.

C.8 Training curves

A perturbed graph is a different training problem, so a drop in test accuracy could in principle be a drop in *fit*: the model simply fails to train on the perturbed arm. That is a different claim from the one this paper makes, and it is separable from it, because the two trained backbones in our record log a training curve. All 48 CompGCN runs (four graphs × four arms × 3 training seeds, 100 epochs)

Table 33: Frozen ULTRA-3g MRR (mean of head and tail reciprocal rank) by what `break` leaves a test query. $\Delta_{\text{conv}} = \texttt{preserve} - \texttt{break}$ and $\kappa = \text{matched} - \texttt{break}$; matched is the mean over five constructions.

Graph	Test query under `break`	n	`preserve`	matched	`break`	Δ_{conv}	κ
CoDEx-S	keeps a two-hop path	707	0.655	0.414	0.299	+0.357	+0.115
CoDEx-S	loses every two-hop path, keeps a three-hop path	295	0.406	0.314	0.289	+0.116	+0.024
CoDEx-S	no two-hop path in `clean`	826	0.253	0.237	0.263	-0.010	-0.026
FB15k-237	keeps a two-hop path	13366	0.374	0.333	0.314	+0.061	+0.019
FB15k-237	loses every two-hop path, keeps a three-hop path	1744	0.330	0.269	0.237	+0.092	+0.032
FB15k-237	loses every two- and three-hop path	4	0.328	0.223	0.025	+0.302	+0.198
FB15k-237	no two-hop path in `clean`	5352	0.250	0.243	0.268	-0.018	-0.025

record training loss and validation MRR at every epoch, and the 60 ComplEx and DistMult runs record training loss every ten epochs. These are not new runs: every one of the 108 joins to a row of the released ledger by source filename, so the curves are the training histories of the results the paper reports.

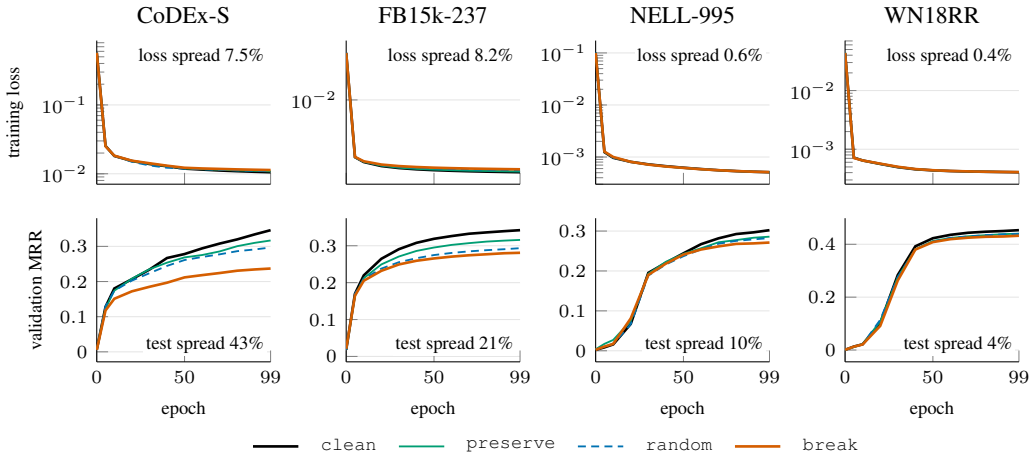


Figure 6: **The arms fit the training data alike and generalise differently.** CompGCN, mean over 3 training seeds. Top: training loss, log scale, the four arms are indistinguishable. Bottom: validation MRR on the same runs, where they separate. The annotations give the across-arm spread in final training loss and in test MRR, relative to the smallest value.

The arms converge, and they converge alike. No run diverged, none stopped early, and every arm used the identical epoch budget by construction. Figure 6 shows the result: on the log scale the four training-loss curves lie on top of one another while the validation curves fan out. The across-arm spread in final training loss is 7.5% on CoDEx-S, 8.2% on FB15k-237, 0.6% on NELL-995 and 0.4% on WN18RR; the spread in test MRR across the same arms is 42.8%, 21.4%, 9.7% and 4.3%. On the bilinear scorers the gap is wider still: 1.0% of training loss against 38.7% of test MRR for ComplEx, 1.8% against 41.7% for DistMult.

What the loss ordering does and does not say. We report the ordering rather than assert independence. On CoDEx-S and FB15k-237 the final training loss orders exactly as the accuracy does (`clean` lowest, then `preserve`, `random`, `break`), so the perturbed graphs are marginally harder to fit; on NELL-995 and WN18RR the two orderings disagree. Even where they agree the magnitudes are not comparable, on CoDEx-S the CompGCN arms span 0.010534 to 0.011327 in training loss while their test MRR spans 42.8%. The matched arm makes the same point against the arm it is matched to: ComplEx reaches a *lower* final loss under `matched` than under `break` (3.4457 against 3.4513, lower by 0.16%) and scores 6.7% higher. Fit does not order the result.

The budget is a real limit, and it cuts the other way. Validation MRR has not plateaued at 100 epochs in any arm. On CoDEx-S the gain from epoch 70 to epoch 99 is 0.0381 for `clean` and 0.0128 for `break`: the clean arm is still improving roughly three times as fast when the budget ends. If those rates held, a longer budget would widen the reported gap rather than close it, which would make these CompGCN effects a lower bound at this budget. We did not run the longer budget,

Table 34: **Removing compositional structure is nearly free until most of it is gone.** *Panel A*, the **Edit-Matched Closure Ladder**, holds edit distance at 24.81–25.03% on all four rungs, so only closure varies: **the whole effect appears in the last third**. *Panel B*, the **Closure-Removal Ladder**, holds edit distance at 25.06–25.30% across its three perturbed arms: destroying 92% of closure moves no statistic by more than 0.005.

	Arm	edits	closure loss	MR ↓	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
Panel A: Edit-Matched Closure Ladder (CoDEx-S, three seeds; ComplEx rung levels in Table 37)								
ULTRA	`matched`	24.84%	0.52%	73.99	0.3128	0.1920	0.3667	0.5600
	`ladder-11`	24.82%	11.44%	71.94	0.3277	0.2063	0.3867	0.5718
	`ladder-21`	25.03%	20.96%	74.77	0.3243	0.2040	0.3817	0.5689
	`break`	24.81%	33.23%	101.16	0.2638	0.1624	0.2939	0.4747
	slope, low segment (per 10 pts)			+0.38	+0.0056	+0.0058	+0.0074	+0.0044
	slope, high segment (per 10 pts)			**+21.50**	**−0.0493**	**−0.0339**	**−0.0716**	**−0.0768**
ComplEx	*slope, low segment* (per 10 pts)			+1.61	−0.0001	−0.0008	+0.0027	+0.0004
	slope, high segment (per 10 pts)			**+0.15**	**−0.0146**	**−0.0025**	**−0.0245**	**−0.0441**
Panel B: Closure-Removal Ladder (`codex_m_E`, ULTRA zero-shot, one run per arm)								
ULTRA	`clean`	—	13.69%	—	0.3278	0.2450	—	0.4825
ULTRA	`sc_preserve`	—	13.73%	—	0.2489	0.1750	—	0.3975
ULTRA	`sc_mid`	—	7.07%	—	0.2469	0.1750	—	0.3908
ULTRA	`sc_break`	—	**1.11%**	—	0.2493	0.1783	—	0.3925
	edit cost (`clean`→`sc_preserve`)			—	−0.0788	−0.0700	—	−0.0850
	closure cost (`sc_preserve`→`sc_break`)			—	+0.0004	+0.0033	—	−0.0050

Closure loss in Panel A is relative to the clean graph; in Panel B the absolute closure of each arm is shown. Dashes are metrics the campaign did not record. A positive Δ on MR means worse.

so this is an extrapolation from the rates at the end of the one we did. The budget is identical across arms, which is what the matched protocol requires; it is not tuned per arm, and no arm is given more of it.

What this does not cover. The trainers log validation MRR, not validation loss, so a train/validation *loss* gap is not recoverable from the record and we do not report one. The frozen and zero-shot ULTRA arms have no training curve at all, by construction. This section therefore answers the question for the two trained backbones on record, and leaves it open for the fine-tuned foundation model, where it would need a re-run with validation loss logged.

D Additional analyses of construction dependence

This section supports Section 5 and Section 8. It collects the fixed-budget closure ladders and the closure-by-reciprocity grid, the variance components, the acceptance rules, and the reciprocity factorials.

D.1 Fixed-budget closure ladders

This section supports Section 4: it asks what removing closure costs when the edit budget is held fixed.

Table 34 reports two closure ladders under one column skeleton. On `codex_m_E`, holding edit distance at 25.06–25.30% across the three perturbed arms (a spread of 0.24 points) and driving closure from 13.73% of the graph to 1.11%, a 92% removal of closed two-paths, changes MRR by +0.0004, Hits@1 by +0.0033 and Hits@10 by −0.0050, while the edit operator reaching the first perturbed arm at unchanged closure costs −0.0788 MRR: the intended dose contributes nothing detectable and the unmeasured dose everything. The limit is power: one run per arm, so this is a sign disagreement at ±0.005, not an exact zero.

The Edit-Matched Closure Ladder on CoDEx-S gives the shape. With edit distance pinned between 24.81% and 25.03% on all four rungs, the first twenty-one points of closure loss cost nothing beyond seed noise (ULTRA's MRR improves, from 0.3128 to 0.3277) and the whole effect appears between 21% and 33%, where ULTRA's slope is −0.0493 MRR per ten points and ComplEx's −0.0146 (Table 37). The dose-response is a threshold, not a slope: a mechanism consuming two-hop support in proportion to its supply would gradient from the first rung, whereas a threshold is what one expects if the last third is where the operator starts to touch the edges the answers depend on, compositional or not. A two-arm contrast samples only the ends of this curve.

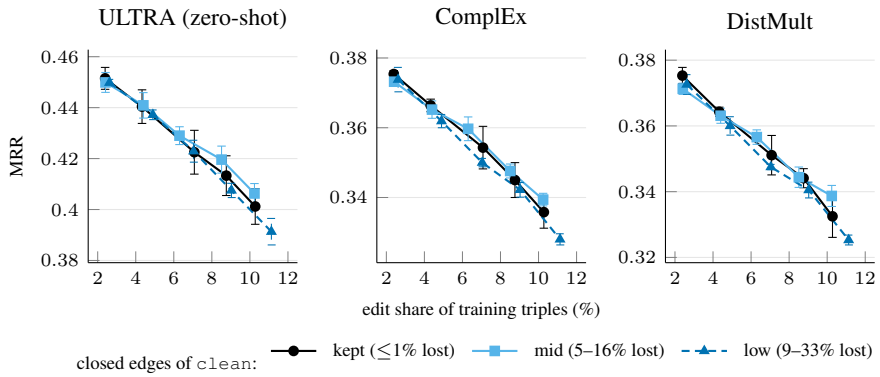


Figure 7: **At a 2–11% edit budget, MRR falls with every edit step at every closure level; removing up to a third of closed edges moves it by at most 0.010.** Low-Edit Closure Ladder, CoDEx-S. Points are means over four constructions, bars ±1 SD across constructions; the x-axis is the realised edit share and line style the closure level, labelled by the share of `clean`'s closed edges lost.

A third ladder, at a low edit budget. The Low-Edit Closure Ladder crosses five edit levels (2.4–11.1% of training triples, realised) with three closure levels on CoDEx-S, four constructions per cell and one run per construction, for ULTRA zero-shot, ComplEx and DistMult (180 runs). Figure 7 plots it. At the level that keeps closure, MRR falls by 0.050, 0.040 and 0.043 from the lowest to the highest edit level. At a fixed edit level, moving to the lowest closure level (up to 33% of closed edges removed, as much as `break` removes) changes MRR by between +0.0004 and −0.0099, although those arms also carry up to 0.9 points more edits; the middle closure level lies above the kept level in 10 of 15 cells. At this budget the closure effect is small and not monotone, while the edit effect is large and falls at every step on every line. The budget matters: at a 25% edit budget the last third of closure loss does cost MRR (Table 34, Panel A), so we report the closure effect per budget rather than as one coefficient.

D.1.1 Separating closure from symmetric-pair loss

This subsection supports Section 5: it asks which of the quantities the operator moves the drop follows. The two-arm contrast moves closure and the symmetric pairs together, so neither arm can say. To separate them we built a grid of constructions on CoDEx-S at a fixed edit budget, crossing three levels of closure loss with three levels of reciprocity, and scored four families on it (45 constructions; 27 for ULTRA). Across the grid closure loss spans 1.25–22.75% and symmetric-triple loss spans 10.10–22.61%, and the two are near-orthogonal ($r = -0.14$), so a linear model estimates separate conditional coefficients.

Table 35: Closure and symmetric-pair coefficients at a fixed edit budget (CoDEx-S, MRR per point of loss, 95% intervals). The fit also carries a reciprocity term, reported in the text. Read as association under the implemented construction.

Model	n	closure	symmetric pairs	R^2
ULTRA	27	+0.00011 [0.00001, 0.00021]	−0.00210 [−0.00241, −0.00179]	0.96
AnyBURL	45	+0.00038 [0.00029, 0.00046]	−0.00269 [−0.00294, −0.00243]	0.97
ComplEx	45	+0.00013 [0.00003, 0.00022]	−0.00128 [−0.00156, −0.00099]	0.85
DistMult	45	+0.00008 [−0.00003, 0.00019]	−0.00122 [−0.00156, −0.00088]	0.76

In every family the closure coefficient is positive, which is the opposite of the sign the compositional reading predicts, and small. Over the range the grid spans, the fitted change in MRR from closure loss is +0.003 to +0.009; from symmetric-pair loss it is −0.012 to −0.030, three to eight times larger. The reciprocity term is negative in all four families (−0.099 to −0.147 per unit) and, on CoDEx-S, is not separable from same-relation symmetry: 5,050 of the 5,070 two-way pairs share their relation. An independent re-implementation of the geometry and of the fit reproduces every sign and magnitude in Table 35.

D.1.2 How typical are the released arms?

This subsection supports Section 8: it asks how far the released arms sit from the rules that reach their doses. We rebuilt each arm five times with our own released builder, not the unrecovered builder of the released arms: `preserve` at the released edit budget with closure pinned, `break` under the strict rule (a swap is kept only if it lowers closure) at the released closure, and `matched` at the rebuilt `break`'s edit count. Every rebuild is scored on the full test set with the frozen scorer. A re-score of the released `break` arm reproduces its shipped per-query ranks exactly, so the differences below are in the graphs and not in the scorer.

Table 36: Released arms against five rebuilds each at the same doses (CoDEx-S, frozen ULTRA-3g; `clean` MRR 0.4716).

	edits		symmetric-pair loss (%)		MRR	
Arm	released	rebuilt	released	rebuilt	released	rebuilt
`preserve`	4,323	4,323–4,324	1.43	5.66–6.85	0.433	0.381–0.388
`break`	8,158	3,667–3,815	23.96	3.49–4.63	0.281	0.390–0.404
matched	8,168	3,667–3,816	12.63	4.71–4.95	0.318	0.396–0.405

Both released arms sit outside the rebuilt range, and both differences enlarge the conventional contrast: the released `preserve` spares the symmetric pairs that any rule reaching its dose removes, and the released `break` reaches its closure dose with twice the edits a strictly decreasing rule needs. Rebuilt at the same doses, the ordering of the two arms reverses: every rebuilt `break` (0.390–0.404 MRR) outscores every rebuilt `preserve` (0.381–0.388), so the conventional contrast is negative for every rebuilt pair. On the rebuilt sets the Edit-Matched Arm reproduces all of the MRR gap in each of the five, and 55–130% of the Hits@1 gap.

Table 37: The Edit-Matched Closure Ladder on CoDEx-S for all four families (MRR), the source of the slopes in Table 34.

Arm	edits	closure loss	ULTRA	ComplEx	DistMult	AnyBURL
`matched`	24.84%	0.52%	0.3128	0.2975	–	0.2579
`ladder-11`	24.82%	11.44%	0.3277	0.3006	0.2977	0.2691
`ladder-21`	25.03%	20.96%	0.3243	0.2973	0.2920	0.2711
`break`	24.81%	33.23%	0.2638	0.2794	0.2721	0.2281
slope, low segment, per 10 pts†			+0.0056	−0.0001	−0.0060	+0.0065
slope, high segment 21 → 33%, per 10 pts			**−0.0493**	**−0.0146**	**−0.0162**	**−0.0351**

† 0.5 → 21% closure loss, except DistMult, which has no `matched` level and spans 11 → 21%.

D.2 Variance components and clustered intervals

This section supports Section 8: it asks how the construction and training-seed components of variance compare, and what that implies for the intervals in the main text.

The dose is assigned to arms, not to runs. The intervals of Table 20 treat runs as independent observations. They are not: every run scoring a given arm inherits that arm's single realisation of the intervention, so the arm is the unit of assignment. Refitting the same model with standard errors clustered on the arm moves ULTRA's closure coefficient from -0.0180 $[-0.0281, -0.0079]$ with 18 degrees of freedom to $[-0.0431, 0.0071]$; collapsing to arm means first (4 degrees of freedom) gives $[-0.0460, 0.0100]$. ComplEx behaves the same way. DistMult's interval still excludes zero. We report both and rest no claim on the unclustered fit.

How large the missing component is. Clustering is a correction whose size the data can report directly, because `matched` was re-seeded at construction. Training all five constructions at three training seeds each (fifteen runs on CoDEx-S, ULTRA trained from scratch at depth six for ten epochs, all fifteen completing) lets the two variance components be estimated separately (Table 38). The standard deviation between constructions is 0.00538 MRR and the mean standard deviation within a construction is 0.00302, but their ratio is not the ratio of the components: the numerator carries seed noise, since the variance of five construction means has expectation $\sigma_c^2 + \sigma_e^2/3$, and the denominator is a mean of three-point standard deviations, which is biased low. A one-way random-effects decomposition of the same fifteen runs separates them. On MRR the construction component

is distinguishable from zero ($F(4, 10) = 6.23$, $p = 0.009$), with $\hat{\sigma}_c = 0.00493$ against $\hat{\sigma}_e = 0.00373$, a ratio of 1.32 rather than the 1.78 of the raw standard deviations. Five constructions do not pin that ratio down: its 95% interval is $[0.36, 4.25]$ and includes 1. Treating runs as independent understates the standard error by 1.66 at the point estimate, and that factor inherits the same interval. On Hits@10 the decomposition separates nothing ($F(4, 10) = 1.14$, $p = 0.39$). We therefore read Table 38 as evidence that a construction component exists and is not negligible, not as a measurement of its size. This is measured on one protocol and one graph and we do not transport the magnitude; the ordering is what matters, and it is the same ordering the clustered intervals above assume.

Table 38: **Arm construction contributes more variance than training seed.** Five independent matched constructions on CoDEx-S, three training seeds each, depth six, ten epochs. Hits figures are read from the run logs rather than the driver's summary (see the note below).

	MRR	Hits@1	Hits@10
grand mean over 15 runs	0.31836	0.19615	0.56338
SD between the five constructions	0.00538	0.00552	0.00356
mean SD within a construction (training seed)	0.00302	0.00347	0.00525
ratio, between over within	**1.78**	1.59	0.68

Parsing of the evaluation summaries. The driver used for Table 38 writes Hits@1 and Hits@10 into its summary file by a pattern that also matches the trainer's `hits@10_50` line, producing 0.0 and 50.0 in those two columns. Every Hits figure in Table 38 is therefore read from the run logs, not from that file. We checked the released run ledger for the same signature: the ten rows whose source columns showed it are flagged in the run ledger and retain only their MRR, and every remaining Hits and MRR value lies in $[0, 1]$, so the error did not reach any reported result. One cell of the fifteen failed a dataset check on a cold cache and was re-run; the guard refused rather than scoring a partially built dataset, which is the behaviour Section F.1 argues for.

Blocking on closure does not remove closure. The reciprocity factorial of Section D.4 pairs within closure blocks. Within those blocks reciprocity and closure are still correlated at -0.991, -0.947 and -0.962 respectively, and the blocks span only 2.4–2.8% of closure. Attributing the residual closure movement its fitted share accounts for roughly a sixth of the $+0.0258$ contrast. The block design reduces the closure dose; it does not zero it.

Multiplicity. Tables 3 and following carry well over one hundred and fifty nominal 95% intervals. At that count roughly eight exclusions of zero are expected by chance alone. Conjunctive statements ("every interval in this family excludes zero") are unaffected. Disjunctive ones are: in particular "three of five statistics exclude zero" at $m_2 = 0$ (Section 4.2) should be read as a description of that table and not as five independent tests. The claim we rest on there is the *sign*, which agrees on all five statistics and replicates across five constructions and two graphs.

D.3 Acceptance rules, positivity, and an untracked dose

This section supports Section 3 and Section 5: it asks how much of each dose is forced by the operator and how much is a choice of its acceptance rule.

Section 3 reports the two doses the arms deliver. This appendix asks a prior question: how much of each dose is *forced* by the operator, and how much is a choice made by its acceptance rule.

The rules. The arm builder proposes within-relation degree-preserving double-edge swaps and accepts or rejects each one. Three rules bracket the design. STRICT accepts a swap only if it strictly lowers the closure count; NON-INCREASING accepts any swap that does not raise it; ACCEPT-ALL accepts every legal swap and is the `random` arm's rule. We ran all three on `clean` CoDEx-S at three construction seeds each, recomputing closure exactly every 200 accepted swaps, and recorded the whole trajectory in the (edit distance, closure) plane rather than only its endpoint (Table 40).

The edit dose is not forced by the target dose. STRICT reaches `break`'s closure loss of 33.23% at an edit distance of 11.20% (seeds 10.91, 11.33, 11.36), against the released `break` arm's 24.81%,

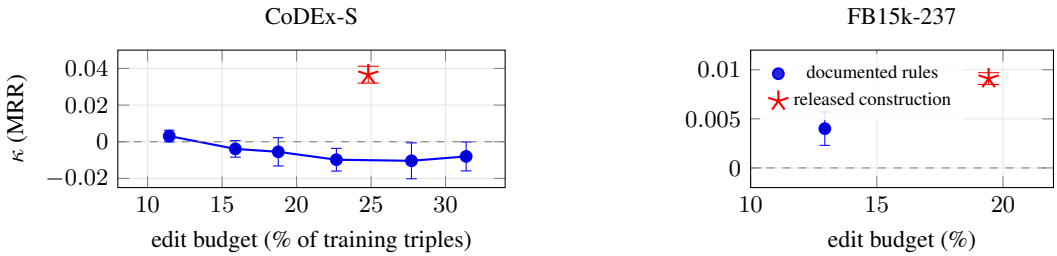


Figure 8: **On CoDEx-S no documented acceptance rule reproduces the released construction's contrast.** The matched-control contrast κ against the edit budget a procedure spends to reach one target closure (frozen ULTRA-3g, full test, five constructions per point, 95% t-intervals). Documented rules (circles) span 11.45–31.38% of edits and all lie below the released construction's +0.0366 (star); on FB15k-237 both rules stay positive.

the same target dose at 45% of the edit dose. The released arm lies between STRICT and NON-INCREASING and matches neither exactly; we cannot say which rule produced it, because its builder is not in our run tree (Section 3), and that gap is the reason the question cannot be closed here. The consequence for the claim is a scoping one and we state it plainly: it is not true that this operator *must* move the two doses together. It is true that the operator as implemented (here, and in the work we audit) does, and that is the object of study.

The decomposition is a property of the operator, not of closure. The paragraph above is a statement about construction, and it can be turned into a measurement. We materialised the STRICT arms (stopping each at the released `break` arm's closure of 9,540 triples exactly rather than at a tolerance) built an edit-matched partner for each at its own edit budget, and scored every cell under the frozen checkpoint on the same 1,200-query evaluation population the released numbers use. These are new runs on independent random streams, stopped at the first swap that reaches the target closure rather than at the nearest point on the 200-swap recount grid, so their edit distances (11.15–11.60%, mean 11.45%) sit just above the trajectory values quoted above and are read the same way. The released arms were re-scored in the same pass and reproduce the released ledger to every digit we hold, so the two operating points are comparable rather than merely adjacent (Table 39). The closure-targeting arm of each block delivers the identical target dose: closure falls from 14,288 to 9,540, 33.23% destroyed.

At the released operating point (24.81% edit distance), the edit-matched arm leaves +0.0480 [+0.0414, +0.0547] as its matched-control MRR contrast. At the STRICT operating point, 11.45% edit distance for the same target dose, it leaves +0.0056 [+0.0012, +0.0100]: both positive, both measured on 5 construction seeds, intervals disjoint, and smaller by a factor of 8.60. This is not the whole effect shrinking with the smaller edit budget, as a share of its own arm's drop from `clean`, the matched-control contrast still falls from 24.76% to 7.73%. Nor is it a property of MRR: the same comparison on the two Hits levels this scorer records gives ratios of 12.76 at Hits@1 and 4.44 at Hits@10, so the direction survives the rank statistic and the magnitude does not, exactly as Section C.2 would predict.

What this costs the control we propose. We read this as a limit on our own instrument, not as a result about closure. The edit-matched arm does at each operating point exactly what it is built to do: it holds the edit dose fixed while closure changes, and it does so cleanly in both blocks. Closure does not thereby vary alone: reciprocity and symmetric-pair loss travel with it (Section D.1.1). What the arm removes is variation in the edit dose, not variation in everything except closure. What it does not deliver is a number that belongs to closure. Hold the target dose exactly fixed, change only the acceptance rule that sets the edit dose, and the matched-control contrast moves by a factor of 8.60. Two readings survive the data. Either the closure effect interacts with the edit dose (destroying closure costs more in a graph already a quarter rewritten than in one an eighth rewritten) or the released arm's larger component is carried by something its extra 13.35 points of edit distance move and its matched partner does not reproduce. Two operating points cannot separate these, and the design does not contain a third. Under either reading the same conclusion follows: a matched contrast measures a closure effect *at an operating point the acceptance rule chose*, and two studies'

Table 39: **One target dose, two acceptance rules, and a matched-control contrast that moves by a factor of** 8.60 **on the** 1,200**-query protocol.** CoDEx-S under the frozen ULTRA checkpoint, 1,200 queries, one evaluation population throughout. On full test the same two rules give $+0.0366$ and $+0.0031$ (Table 5), a factor of 11.8. The closure-targeting arm of each block destroys 33.23% of closure and its matched partner almost none; the blocks differ only in the acceptance rule, and so in the edit budget at which the matched comparison is made. Errors are standard deviations over construction seeds. The released `break` arm is a single construction, so the released matched-control contrast carries no break-side construction variance. The released-arm rows reproduce the run ledger exactly, which is a precondition the generator checks before it will emit this table.

arm	edit distance (%)	closure destroyed (%)	MRR	n
`clean`	0.00	0.00	0.4641	1
shipped operator				
`break`	24.81	33.23	0.2701	1
`matched`	24.88	0.59	0.3182 $\pm$0.0053	5
matched-control contrast			$+\mathbf{0.0480}$ $\pm$0.0053	5
STRICT *rule, same target dose*				
STRICT	11.45	33.23	0.3918 $\pm$0.0073	5
STRICT-matched	11.45	0.54	0.3974 $\pm$0.0053	5
matched-control contrast			$+\mathbf{0.0056}$ $\pm$0.0035	5

closure effects are not comparable until their acceptance rules are. That is the operator dependence Section 3 identifies in the interventions we audit, now found in the repair we propose for them.

What this result will not carry. What the intervals license is the separation of the two components, not the ratio, which is a point estimate at 5 seeds. The released component's spread is over 5 matched constructions against one `break` construction, so it carries no break-side construction variance; both components come from one graph, one checkpoint and one evaluation subsample. These 20 cells were run after the run ledger was sealed and carry MRR and two Hits levels but not mean rank, so they are released as a record of their own rather than merged into it. And the STRICT arms are not the released arms: they are a different operator reaching the same target dose, which is what makes the comparison informative and also why it cannot be folded into Table 2.

Positivity fails, which is why a matched arm is needed at all. ACCEPT-ALL runs to its swap cap at 21.22% edit distance having destroyed 12.98% of closure, and NON-INCREASING to 24.33% edits at 27.77% destroyed. Neither reaches 33.23% at any attainable budget. So in the joint dose plane the closure-targeting and non-targeting processes occupy disjoint regions: at `break`'s operating point there is no non-targeted unit to compare it to. The counterfactual "the same edits, placed without regard to closure" has no support in the data, and an estimator that appears to form it is extrapolating. The Edit-Matched Arm is precisely a construction that supplies a comparison unit at `break`'s edit budget, which is the positive case for it.

Table 40: **The acceptance rule, not the target dose, sets the edit budget.** Three acceptance rules on `clean` CoDEx-S, three construction seeds each; terminal values are means. Closure is the number of training triples on at least one directed two-path. The released arms are recomputed from the released files and reproduce the manifest exactly.

acceptance rule	terminal edit distance	terminal closure destroyed	edits to reach 33.23%
STRICT ($<$)	14.45%	38.34%	**11.20%**
NON-INCREASING ($\leq$)	24.33%	27.77%	not reachable
ACCEPT-ALL	21.22%	12.98%	not reachable
released `break`	24.81%	33.23%	—
released `matched`	24.84%	0.52%	—

A fourth dose the design does not measure. Recounting the released arms directly, the within-relation swap destroys symmetric training triples at a rate the design never records: on CoDEx-S the symmetric-triple rate falls from 0.1536 to 0.1168 under `break`, a 24% loss, and on WN18RR from 0.3416 to 0.3238. Symmetry is preserved by neither the degree constraint nor the closure target, and it is not among the quantities any arm is matched on. It is a fourth structural coordinate moving with the intervention. We report it rather than control it: controlling it needs a fifth arm we have not built.

Table 41: **Orthogonalising the measured doses gives a contrast that does not transfer, and where it lives is itself graph-specific.** High-minus-low reciprocity, paired by closure × seed, at fixed edit distance. **The sign reverses between CoDEx-S and CoDEx-M on all four rate statistics.** Mean rank rises on both, the head-versus-tail split of Section C.2. The From-Scratch Initialisation Control removes the effect on CoDEx-M but retains nearly two-thirds of it on CoDEx-S (+0.0164).

Study	Initialisation	MR ↓	MRR ↑	H@1 ↑	H@3 ↑	H@10 ↑
CoDEx-S						
Retrained Reciprocity Factorial	trained	−0.05	+0.01747	+0.01665	+0.02535	+0.01228
Zero-Shot Reciprocity Factorial	frozen pretrained	+23.04	+0.02578	+0.02175	+0.03782	+0.03470
repeated runs	frozen pretrained	—	+0.02562	+0.02100	—	+0.03122
From-Scratch Initialisation Control	random	—	+0.01640	—	—	—
CoDEx-M						
Zero-Shot Reciprocity Factorial	frozen pretrained	+374.02	−0.02210	−0.01832	−0.02502	−0.02986
Bilinear Reciprocity Control (ComplEx)	trained	+23.03	−0.00119	−0.00108	−0.00139	−0.00154
Bilinear Reciprocity Control (DistMult)	trained	+16.66	−0.00137	−0.00130	−0.00143	−0.00175
From-Scratch Initialisation Control	random	−0.11	+0.00002	+0.00103	+0.00130	+0.00078

Fifteen closure-by-seed pairs per row; nine for the retrained factorial, six for the CoDEx-S from-scratch control, four to five for the CoDEx-M one (non-MRR statistics recorded on 8 of 40 runs). Dashes: not recorded. CoDEx-S from-scratch value from Table 44. Protocols, stated here because they were not: CoDEx-S ranks its full test set (1828 triples), CoDEx-M the full-test campaign that superseded the 1,200-query subset (−0.02314 on MRR).

Provenance of the released arms. Section 3 states that the builder for `preserve`, `break` and `random` is not recoverable. We record the sharper version here, because it affects what a reader can check. The negative-control builder distributed with our supplementary code implements exactly the acceptance rules above, but is hard-coded to three NELL inductive splits that appear in no result in this paper, and writes to a separate output tree. It therefore regenerates a different family of arms than the ones every headline number is computed on. The arms themselves are released, and their checksums are in the manifest, so the results are *verifiable*; they are not, from the released code alone, *regenerable*. Anyone extending this work should rebuild the arms from a builder that reproduces the released files bit-for-bit against those checksums before varying anything.

D.4 Reciprocity factorials

This section supports Section 5: it asks what happens when reciprocity is varied on its own.

The natural response to Section 4 is a better design: a factorial grid in which closure and a second structural quantity vary independently at fixed edit distance. We built it three times (retrained on CoDEx-S, frozen on CoDEx-S with a repeated-runs set, and frozen on CoDEx-M) with reciprocity, the largest quantity the operator moves after closure, as the second factor. Table 41 reports the within-closure-block reciprocity contrast at fixed edit distance, paired by closure block and seed.

The contrast is clean, and it is not an attribution. Within a closure block edit distance is constant and leakage zero, and on CoDEx-M (the one graph whose factorial supplies a floor), the contrast is 6.6 times it. That floor is a *construction*-seed floor, and Section D.2 measures a training-seed component of the same order, so 6.6 is an upper bound; by how much is not pinned down, since the size of that component is itself poorly determined. This is the design Remark 1 asks for, and it delivers what two arms cannot: the effect of moving one measured coordinate with the other fixed. It does not deliver reciprocity. Under this operator reciprocity and assortativity move together ($r = -0.9932$ across design cells); conditioning on assortativity reverses the coefficient and inflates its variance 222-fold, and no experiment here separates either quantity (Table 43). Blocking on closure does not remove it either: within blocks the two remain correlated at −0.991 to −0.947, worth about a sixth of the +0.0258 (Section D.2). What the factorial identifies is the effect of a reciprocity-targeting intervention at fixed measured doses, less than the repair was meant to buy.

It does not transfer. On CoDEx-S the contrast is +0.0175 retrained, +0.0258 frozen and +0.0256 on repeated runs; on CoDEx-M it is −0.0221 under the same construction procedure, with edit distance held to 0.33 points across the grid and assortativity to 0.00112, and the reversal holds on all four rate statistics. Where it lives is graph-specific too: training the same architecture on byte-identical arms without the pretrained checkpoint removes the effect on CoDEx-M (+0.00002 over five pairs) but retains nearly two-thirds of it on CoDEx-S (+0.0164 against +0.0258). The bilinear

controls on CoDEx-M are not inert, so the defensible claim is differential sensitivity of more than an order of magnitude, not a mechanism unique to the foundation model (Table 42).

This is a different failure from the attribution gap above: the design repaired the collinearity Section 3 identified, and the contrast it delivers still does not generalise. One could read the surviving effect as evidence that reciprocity, not composition, is what the intervention disturbs; three grids on two graphs (positive, positive, negative) do not support that. What they support is narrower: a dose effect measured on one benchmark licenses no claim about the mechanism on another. Part of that spread is dose delivery, the intervention reaches 18.1% of WN18RR's test queries against 73.8% of FB15k-237's (Table 28).

D.4.1 The reciprocity contrast across families

Table 42 gives the within-closure-block reciprocity contrast at fixed edit distance for every family and regime on both graphs, on MRR, with the number of closure-by-seed pairs and their standard deviation; it is the source of the twenty-pair from-scratch value quoted in Section D.4. On CoDEx-S the contrast is positive for every family including AnyBURL and the bilinear models; on CoDEx-M it is negative for the frozen foundation model, near zero for the bilinear models on MRR, and removed by training from scratch. It is not monotone within a grid: mid-reciprocity cells score above high-reciprocity cells in all three closure blocks, which a dose-response account does not predict. And it cannot be turned into an attribution. Under this operator reciprocity and assortativity are collinear ($r = -0.9932$ across the 9 design cells, $r = -0.6955$ within them), so conditioning on assortativity reverses the sign of the reciprocity coefficient and inflates its variance 222-fold (Table 43). No experiment here separates either quantity, and the paper claims neither.

Table 42: Within-closure-block reciprocity contrast at fixed edit distance ($\mathrm{rhi} - \mathrm{rlo}$, MRR), leakage zero throughout. This contrast is an *unpaired* difference of level means and n counts runs per level, not pairs; Table 41 reports the *paired* within-block contrast on the same runs, which is why its CoDEx-M from-scratch entry reads 0.000016 against 0.000204 here (see the note on estimators below). Variance is over arm construction.

Model / regime	Graph	rhi − rlo	n	SD
ULTRA fine-tuned	CoDEx-S	+0.0175	9/9	0.0094
ULTRA frozen	CoDEx-S	+0.0258	15/15	0.0140
ULTRA from scratch	CoDEx-S	+0.0164	6/6	0.0093
AnyBURL	CoDEx-S	+0.0241	15/15	0.0131
ComplEx	CoDEx-S	+0.0102	15/15	0.0062
DistMult	CoDEx-S	+0.0092	15/15	0.0056
ULTRA frozen	CoDEx-M	-0.0221	15/15	0.0118
ULTRA from scratch	CoDEx-M	+0.0002	20/20	0.0020
ComplEx	CoDEx-M	-0.0012	15/15	0.0024
DistMult	CoDEx-M	-0.0014	15/15	0.0024

Table 43: Coefficient on standardised reciprocity in a regression of MRR over the decoupled factorial ($n = 45$ arms per model, edit distance held constant). Adding assortativity reverses the sign and inflates the variance by two orders of magnitude.

Model	Specification	β_{recip}	95% CI	VIF
ULTRA	closure + recip	-0.00911	$[-0.01052, -0.00770]$	1.0
ULTRA	+ assortativity	$+0.01914$	$[-0.00031, +0.03860]$	222
ULTRA	+ assort + transitivity	$+0.01848$	$[-0.00226, +0.03921]$	247
MOTIF	closure + recip	-0.00714	$[-0.00774, -0.00654]$	1.0
MOTIF	+ assortativity	$+0.01381$	$[+0.00739, +0.02023]$	222
MOTIF	+ assort + transitivity	$+0.01375$	$[+0.00691, +0.02060]$	247

Two estimators, not two answers. Table 41 and Table 42 report the CoDEx-M from-scratch reciprocity contrast as 0.000016 and 0.000204, a factor of ten apart, on the same 40 runs. Both are correct and they are different estimators. Table 41 averages the within-block contrast over the 5 closure-by-seed blocks that carry both levels, unweighted; Table 42 pools all runs into two level means. The blocks are of size $5, 5, 5, 4$ and 1, so the single-run block (-0.001060) enters the first estimate with the same weight as a five-run block, which is the whole of the discrepancy. MRR is the only statistic on which the two can differ here, because it is the only one recorded on all 40

runs; the other four are recorded on 8, where the blocks are balanced and the two estimators agree exactly. Neither estimate is distinguishable from zero (the block differences have SD 0.000892 and the pooled runs 0.001989), so the claim the row supports, that removing the pretrained checkpoint removes the effect on CoDEx-M, does not turn on the choice.

Table 44: Within-closure-block reciprocity contrast at fixed edit distance (the source of the CoDEx-S from-scratch value in Table 41).

Model / regime	Graph	rhi − rlo	n	SD
ULTRA fine-tuned	CoDEx-S	+0.0175	9/9	0.0094
ULTRA frozen	CoDEx-S	+0.0258	15/15	0.0140
ULTRA frozen (replication)	CoDEx-S	+0.0256	15/15	0.0147
ULTRA from scratch	CoDEx-S	+0.0164	6/6	0.0093
AnyBURL	CoDEx-S	+0.0241	15/15	0.0131
ComplEx	CoDEx-S	+0.0102	15/15	0.0062
DistMult	CoDEx-S	+0.0092	15/15	0.0056
ULTRA frozen	CoDEx-M	-0.0221	15/15	0.0118

E RE-ANALYSIS OF A PUBLISHED FINDING

The audit so far turns the Edit-Matched Arm on arms we built. This appendix turns it on someone else's, which is the sharper test: a control that only ever reverses our own conclusions is not evidently worth adopting.

Why there is one candidate and not ten. A published claim can be re-analysed under this control only if it is supported by an *edit to the graph*. Two of the three claims we considered are not. ULTRA's component study and MOTIF's ablation both remove a term from the scoring function and leave the graph untouched; there is no edit budget, so no arm can be matched to it, and the design question we raise does not arise. We record this because it is the first finding of the exercise and it cuts against us as much as for us: much of what this literature calls an ablation is model-side, and our critique has no purchase there. The one published intervention we found that edits the graph *and* attaches a conclusion to the result is Li et al. (2023a), Appendix G: the message-passing graph is replaced by a random graph with the same number of edges, no drop in accuracy is observed, and the authors conclude that message passing is not doing the work.

Their arm delivers both doses at once, at the maximum. We rebuilt their arm on CoDEx-S at three construction seeds, in a uniform form and in a relation-preserving variant that is the more favourable reading of their description (Table 45). Uniform rewiring replaces 99.985% of the training triples and destroys 72.49% of two-path closure; the relation-preserving variant is indistinguishable at 99.886% and 72.63%. It is not only closure that goes: reciprocity falls from 0.15355 to 0.00022 and the largest entity degree from 1008 to 54. Matching the edge *count* leaves every other structural coordinate free, and all of them move. This is the confound of Section 3 in its strongest form: the arm is a simultaneous maximal dose of everything, so a difference against it attributes to nothing in particular. A *null* against it is informative about the graph's structure taken as a whole, since only effects that cancel could hide behind it, but it cannot say which structural property message passing would have used.

Table 45: **Doses delivered by a same-edge-count random graph.** CoDEx-S training graph, three construction seeds, means. Closure is the number of training triples on at least one directed two-path; reciprocity is the share of triples whose reverse is also present. Our arms are repeated from Table 40 at the same scale.

graph	edit distance	closure destroyed	reciprocity	max degree
`clean`	0%	0%	0.15355	1008
same-count random, uniform	99.985%	72.49%	0.00022	54
same-count random, relation-preserving	99.886%	72.63%	—	—
`break` (ours)	24.81%	33.23%	—	—
`matched` (ours)	24.84%	0.52%	—	—

A path-based model collapses under our protocol. Training ULTRA on each graph at three seeds (Table 46), both random-graph arms collapse to approximately chance: MRR 0.0039 and 0.0037

against 0.4833 on `clean`, with Hits@1 of 0.0005 and 0.0000. Where Li et al. (2023a) report no drop, we measure a near-total one under our protocol. We do not read this as a refutation of their paper, and the reason is a protocol difference we state squarely rather than leave for a reader to find.

The protocol difference, named. Li et al. (2023a) randomise the graph that message passing runs over while keeping supervision on the original triples; the model is still told which triples are true. Our protocol perturbs the graph used for *both* message passing and supervision, so the training signal moves with the graph. Their design therefore perturbs the message-passing pathway with the target intact, and ours does not. That is the candidate explanation for the opposite sign, and it is a difference in what the two experiments intervene on, not evidence that either is wrong. What can be said without resolving it is narrower and is the point of this appendix: the conclusion "message passing is not the key part" is licensed only under the first protocol, and the paper does not say so. A reader who carries the claim across to a path-based model (the family this literature now largely consists of) carries it outside the design that supports it.

Table 46: **ULTRA on CoDEx-S under the published arm and under ours.** Three training seeds per arm, mean and standard deviation across seeds. The two random-graph arms are at 99.985% and 99.886% edit distance; `break` and `matched` are at approximately 24.81%.

arm	MRR	SD	Hits@1	Hits@10
`clean`	0.4833	0.0026	0.3831	0.6776
`matched`	0.3113	0.0039	0.1910	0.5575
`break`	0.2642	0.0038	0.1643	0.4712
same-count random, uniform	0.0039	0.0004	0.0005	0.0051
same-count random, relation-preserving	0.0037	0.0005	0.0000	0.0048

The control cannot be added at their operating point, and that is measurable. The obvious repair is to run our matched arm alongside theirs: the same 99.985% of edges moved, closure held. We tried to build it and could not. A closure-preserving search (within-relation degree-preserving swaps, accepted only while closure stays inside a 1.00% band) reaches a largest edit distance of 23.87% over three seeds (13.03% at worst, 19.75% on average) in 3100 seconds per seed, roughly a quarter of the way to their dose. The search does not stop at a limit we imposed; it stalls because closure-preserving swaps become vanishingly rare. Both this ceiling and the 24.84% our targeted matched builder reached on CoDEx-S with closure inside the same band are lower bounds on what such a search can reach, and both are far below the published arm's dose. It accepts 109.5 swaps per second over its first 500, and after that fewer than 500 more in the remaining budget, a fall of at least 465×. We therefore report a failure to construct rather than a proof of impossibility: we cannot rule out that a better search reaches further, and we can say that the dose at which the published null was measured is not one at which we know how to supply a comparison unit. The gap is not a detail of our implementation. A graph that keeps its two-path closure is, by construction, close to the original in exactly the respect the claim is about, so "move almost every edge and change nothing structural" is close to a contradiction. That is the positivity failure of Section D.3 again, at the far end of the dose axis, and it bounds what this control can do: it repairs interventions in the regime where a matched graph exists, and the regime is not the whole axis.

F VERIFICATION AND RUN RECORDS

F.1 VERIFICATION PROTOCOL

This section states the protocol the numbers in this paper were produced under; its checks are the ones Section 6 asks others to report. Every table is regenerated from the run records by scripts that abort rather than emit when a cross-check fails, manifests are rebuilt from on-disk metadata before analysis, and a run that fails a guard is quarantined with a written cause rather than deleted. One exclusion qualifies a reported number: it leaves the ComplEx mean-rank cell of Table 2 Panel B resting on two seeds, and that cell says so.

F.2 RUN LEDGER

A summary of every study on all five statistics—the number of cells, the runs behind them, and how many cells carry each statistic on every seed—is in the supplementary material. A statistic is reported for a cell only where every seed in that cell carries it, so that no average is taken over a changing denominator. The complete ledger (every study, model and graph on all five statistics) and the register of excluded runs are in the supplementary material as machine-readable files.