BEYOND AGGREGATE FORGETTING: KNOWLEDGE GRAPH UNLEARNING CAN DISAGREE WITH RETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Knowledge-graph unlearning (KGU) is typically evaluated by how closely an updated model matches retained-data retraining on aggregate ranking metrics such as mean reciprocal rank (MRR). We show that this criterion can accept models whose individual predictions disagree with retraining far more than two independent retrains of the same retained graph, REFA and REFB, disagree with each other. To isolate this failure mode, we study a restart-and-repair (R+L) setting in which deletion causes complete support loss, removing all positive training evidence for selected entities while their embeddings remain in the model. R+L reinitializes these unsupported entities before retained-data repair. This brings forgotten-set MRR close to REFA while preserving test performance, yet the agreement does not extend to individual predictions.Across 36 deletion settings on three graphs, each removing roughly 10% of the training facts, R+L leaves an MRR gap of $.008$ against a $.004$ retrain-to-retrain null, yet exceeds that null in log-rank in every one, at a mean calibrated excess of $.256$. The discrepancy is not specific to R+L: on matched random-fact deletion requests, Finetune, GraphDPO and SGU all disagree with REFA more than REFB does, in every setting. We then ask what drives it. Across 72 configurations spanning six scoring families, the strongest effect appears in the bilinear ones, and only when an affected entity is ranked as a candidate rather than supplied as query context; a matched background analysis shows that restarted entities are systematically promoted as candidates even in unrelated contexts. An oracle diagnostic that rescales only their norms to the reference removes 92.2% of the excess, identifying embedding-scale mismatch as the dominant cause, but it consumes a retraining reference. We therefore derive from it Retained-Norm Rescaling (RNR), a correction that uses only the updated model and the retained graph; REFA and REFB are used only for evaluation. RNR removes 75.6% of the calibrated log-rank excess, at no cost in test performance. Under a stronger training recipe the excess survives on two of the three graphs. Throughout, we judge an unlearning update not by how closely its aggregate metrics match retraining, but by whether its individual predictions disagree with a retrained model more than two retrained models disagree with each other.

1 INTRODUCTION

Suppose we delete every training fact involving an entity from a knowledge graph and must update a model that learned from those facts. This removes all direct positive training evidence for the entity, a setting we call *complete support loss*, while the entity remains part of the model vocabulary and can still appear as a prediction candidate. Knowledge-graph unlearning (KGU) is typically evaluated by how closely an updated model approaches retained-data retraining on aggregate ranking metrics such as mean reciprocal rank (MRR) (Xiao et al., 2025a;b; Liu et al., 2026). Yet similar averages do not imply that two models rank the same answers to the same queries. Knowledge graphs represent facts as triples (h, r, t), and knowledge-graph embedding models learn entity and relation representations to rank candidate answers to missing-head or missing-tail queries (Bordes et al., 2013; Trouillon et al., 2016). Removing triples from the graph does not remove their influence from learned parameters; unlearning therefore aims to update the model toward the behavior of retraining on retained data

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.

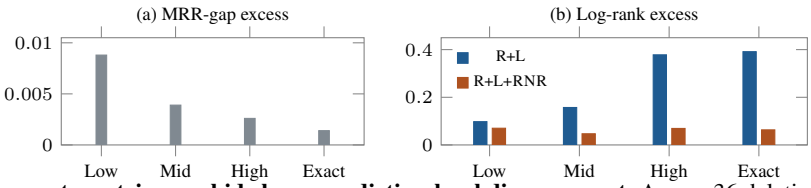


Figure 1: **Aggregate metrics can hide large prediction-level disagreement.** Across 36 deletion settings on three graphs, MRR-gap excess stays small while calibrated log-rank excess grows with request degree; rescaling restarted embeddings to the median norm of supported retained entities (RNR) removes 75.6% of it in aggregate, and least on the low-degree requests. RNR is omitted from (a), where it leaves the excess essentially unchanged.

(Bourtoule et al., 2021; Guo et al., 2020; Xiao et al., 2025a). Ordinary fact deletion may leave both endpoints supported by many other facts. Complete support loss instead removes an entity's entire incident star, while leaving the entity available for future prediction. Existing KGU evaluations largely compare aggregate test and forgotten-set metrics with retained-data retraining (Xiao et al., 2025a;b; Liu et al., 2026). Small MRR gaps can therefore suggest successful forgetting without establishing query-level fidelity. MRR strongly compresses rank displacement: moving a target from rank 200 to rank 10,000 changes reciprocal rank by only .0049, and changes across queries can partially cancel in the mean. Moreover, independent retrains need not agree exactly. We therefore use two independent retained-data retrains, REFA and REFB, and ask whether an updated model disagrees with REFA more than REFB disagrees with REFA. This separates three questions: *(I) Does aggregate performance agree? (II) Does query-level disagreement exceed ordinary retraining variability?, and (III) Does that disagreement cross a practically relevant retrieval threshold?*

To isolate this failure mode, we study RESTART+LOCAL (R+L), which reinitializes entities that undergo complete support loss before repair on retained data. R+L makes aggregate forgetting look successful: across 24 TransE configurations, R+L leaves a mean forgotten-MRR gap to fresh retraining of .00031, against .179 for the same update without restart, at a test-MRR gap of .0084 (Appendix A.2). We use R+L as a controlled probe, not as a proposed unlearning method. This apparent success does not imply prediction-level agreement. Across 36 deletion settings on three knowledge graphs, each removing 15,000–27,000 facts, the mean absolute log-rank gap is .4809 between R+L and REFA, versus .2248 between REFB and REFA. The resulting calibrated excess is positive in all 36 settings. MRR shows a much smaller excess of only .0042, positive in 26/36 settings; across the four deletion constructions, the largest rank excess occurs where MRR-gap excess is smallest (Figure 1). The effect extends beyond R+L. On six matched *random-fact deletion settings* per method, Finetune, GraphDPO (Liu et al., 2026), and SGU (Xiao et al., 2025a) each disagree with REFA more than REFB disagrees with REFA, with excess Top-100 disagreement ranging from 3.54 to 11.60 percentage points. *We then ask what drives the disagreement.* The strongest effects occur for bilinear models, especially DistMult and ComplEx, and when the prediction target is affected by deletion. A matched background-context analysis further shows that restarted entities are systematically promoted as candidates even in contexts unrelated to the deletion request, revealing a candidate-placement effect. *Motivated by this finding*, we ask whether the placement shift has a concrete cause. An oracle diagnostic that rescales only the restarted entity norms to their retained-data reference values removes 92.2% of the calibrated log-rank excess, identifying embedding-scale mismatch as its dominant component. Because that diagnostic consumes a retraining reference it is not deployable, so we derive from it a reference-free correction: rescaling restarted rows to the median norm of supported retained entities, using only the updated model and the retained graph. It removes 75.6% of the calibrated excess and recovers 82.0% of the oracle's gain, with a forgotten-MRR gap marginally below that of the uncorrected model. A prespecified score-level correction we had fixed in advance recovers only half as much. Under a substantially stronger training recipe calibrated excess remains positive on two of three graphs. Across these experiments, we evaluate unlearning by comparing prediction-level disagreement with the variability between independent retrains, rather than relying on aggregate metrics alone.

Contributions. (I) *A retraining-calibrated evaluation gap:* across a complete 36-setting grid, we show that small aggregate forgetting gaps can coexist with substantial query-level disagreement: R+L disagrees with REFA more than REFB disagrees with REFA. The same calibrated excess also appears for Finetune, GraphDPO, and SGU on matched KGU settings (§ 4); (II) *Localization of the disagreement:* cross-family, query-role, and matched background-context experiments show that the excess is strongest for bilinear models and deleted entities, and reveal a systematic candidate-placement effect for restarted entities. An oracle norm intervention identifies embedding-scale

mismatch as its dominant cause, accounting for 92.2% of the calibrated excess (§ 5); (III) *A correction that follows from the diagnosis:* rescaling the restarted embeddings to the median norm of supported retained entities uses only the updated model and the retained graph, has no fitted parameter, and removes 75.6% of the calibrated log-rank excess, 82% of what the oracle achieves with access to a retrained model, at no aggregate cost. A degree-matched variant and a prespecified score-level correction both perform worse, and we explain why (§ 6); (IV) *A retraining-calibrated evaluation protocol:* we specify how KGU evaluation should separate aggregate agreement, prediction-level fidelity relative to REFA–REFB variability, and threshold-level effects.

2 RELATED WORK

Machine unlearning aims to approximate the model that would have been obtained had selected training data never been used, ranging from retraining-efficient schemes to formally certified removal (Bourtoule et al., 2021; Guo et al., 2020). On graphs, this problem is complicated by structural dependencies: methods such as GNNDelete and certifiable graph-unlearning approaches explicitly account for the propagation of a deletion through neighboring representations (Cheng et al., 2023; Wu et al., 2023; Yi & Wei, 2025). Knowledge graphs introduce an additional relational structure and have recently motivated dedicated KGE-unlearning methods. SGU exploits schema information to update affected neighborhoods (Xiao et al., 2025a); MetaEU learns transferable unlearning updates across deletion tasks (Xu et al., 2025); ZeroFisher develops an efficient influence-style approximation (Xiao et al., 2025b); and GraphDPO formulates KGE unlearning through preference optimization while preserving retained knowledge (Liu et al., 2026). *These works differ in how they construct the update but share an evaluation contract:* the updated model is compared with a single retrained reference on aggregate ranking metrics, and a small gap is read as success. Whether that contract is sound is not itself tested. A complementary literature has questioned whether standard evaluations establish successful unlearning. The NeurIPS Unlearning Competition treats evaluation against retrained models as a central problem (Triantafillou et al., 2024); per-example membership-inference analyses show that population averages can hide substantial residual behavior (Hayes et al., 2025); and recent LLM and poisoning studies show that apparent forgetting under one metric need not imply agreement under more discriminating tests (Wang et al., 2025; Pawelczyk et al., 2025). These tests are built for classifiers and generative models, where the residual signal is a label, a likelihood, or a membership decision. A ranking model fails differently: its output is an ordinal position over a shared candidate set, and an entity can move by orders of magnitude in rank while the aggregate score barely changes. That failure mode has no membership-style instrument, and it is the one we measure. Our contribution is therefore an evaluation contract rather than another unlearning objective, and it differs from prior work in what it compares against. We calibrate candidate-to-retraining disagreement against the disagreement between *two independent retrains on the same retained graph*, so that ordinary retraining variability is not mistaken for unlearning failure; to our knowledge no prior KGU evaluation reports this null. Within that contract we separate a deleted entity's role as a ranked candidate from its role as query context, and we apply the same audit to two of these methods and to a retained-data fine-tuning baseline. The mechanism this exposes connects to a known property of knowledge-graph embeddings: representation quality and ranking behavior depend on how much positive training signal an entity receives, an effect studied as degree bias (Shomer et al., 2023). Complete support loss is the limiting case, in which that signal is removed entirely, and our results locate the channel through which it acts in rank space, the scale of the entity's representation rather than its direction. *The distinction is not only diagnostic*: a rescaling fitted from the updated model alone removes three quarters of the disagreement, so the gap we document is correctable as well as measurable.

3 KNOWLEDGE-GRAPH DELETION AND RETRAINING AGREEMENT

Deletion setting. A knowledge graph contains entities $\mathcal{E}$, relations $\mathcal{P}$, and training facts $\mathcal{T} \subseteq \mathcal{E}\times\mathcal{P}\times\mathcal{E}$. An embedding model assigns each triple (h, r, t) a score $s_M(h, r, t)$ and ranks candidate entities for queries $(h, r, ?)$ and $(?, r, t)$. We call the supplied entity–relation pair the *context* and the missing entity the *target*. A deletion request $\mathcal{U} \subseteq \mathcal{T}$ leaves retained data $\mathcal{R} = \mathcal{T}\setminus\mathcal{U}$. In *complete-star deletion*, all training triples incident to selected entities are removed; in *random-fact deletion*, individual triples are removed and their endpoints usually retain other positive support.

Retraining agreement. Let C be the updated candidate, and let REFA and REFB be independently initialized models trained on the same retained graph with the same declared recipe and budget. The reference behavior is therefore the retraining procedure, not an idealized optimum. Importantly, unlearning does not require every deleted fact to rank poorly: retained facts may still support it. For a query set Q, let $r_i(M)$ denote the target rank under model M. All compared models use the same queries, candidate vocabulary, filtering, and tie rule. We measure aggregate agreement by

$$\mathrm{MRR}_Q(M) = \frac{1}{|Q|}\sum_{i\in Q}\frac{1}{r_i(M)}, \qquad \Delta M_Q(C) = \left|\mathrm{MRR}_Q(C) - \mathrm{MRR}_Q(\textsc{RefA})\right|, \tag{1}$$

written ΔM on held-out test queries and ΔM_f on queries derived from deleted facts, in both completion directions; retained-training MRR is reported separately. MRR can hide large movements far down the ranking and can average changes in opposite directions. We therefore also measure mean absolute log-rank displacement,

$$D_{\log r}(M,N) = \frac{1}{|Q|}\sum_{i\in Q}\left|\log_{10} r_i(M) - \log_{10} r_i(N)\right|, \tag{2}$$

where a per-query difference of one corresponds to a tenfold rank ratio.

Calibrating against retraining variability. Independent retrains need not agree exactly. For each experimental setting c, we therefore define calibrated excess disagreement as

$$E_c = D_{\log r}(C_c, \textsc{RefA}_c) - D_{\log r}(\textsc{RefB}_c, \textsc{RefA}_c), \tag{3}$$

averaged equally across settings. Positive excess means that the update disagrees with REFA more than an independent retrain does; it does not imply worse predictive accuracy, and non-positive excess does not establish equivalence. Because $D_{\log r}$ is an absolute difference, E_c is positive on either side of the reference and cannot by itself distinguish under-correction from overshoot. MRR-gap excess is defined analogously using ΔM_Q. Finally, to distinguish rank displacement from changes that affect a retrieval cutoff, define

$$D_k(M,N) = \frac{1}{|Q|}\sum_{i\in Q}\left|\mathbf{1}[r_i(M)\le k] - \mathbf{1}[r_i(N)\le k]\right|, \;\; E_k = D_k(C,\textsc{RefA}) - D_k(\textsc{RefB},\textsc{RefA}). \tag{4}$$

4 DISAGREEMENT EXCEEDS RETRAINING VARIABILITY

We ask whether small aggregate forgetting gaps imply prediction-level agreement with retained-data retraining. Our audit fixes the deletion request and the evaluator, compares the updated model with REFA, and reads the resulting disagreement against the REFB–REFA disagreement expected under the same training budget.

Disagreement across a 36-setting grid. We audit R+L (Appendix A.1) applied to DistMult (Yang et al., 2015) on FB15K-237, CoDEx-M, and NELL-995 (Xiong et al., 2017), with seeds 17/29/43 and four approximately 10% complete-star deletion constructions: low-, mid-, and high-degree stars and an exact-cardinality high-degree variant. Each of the 36 requests removes roughly 15,000–27,000 facts and has its own pair of 20-epoch references. All models are evaluated on up to 1,000 forgotten triples in both completion directions (Appendix A). Intervals use paired query bootstraps within setting and equal weight across settings; a *reduction* in calibrated excess is $1 - \bar{E}^{\mathrm{post}}/\bar{E}^{\mathrm{pre}}$ with both terms equal-setting means (Appendix C). Candidate-to-reference log-rank disagreement exceeds reference-to-reference disagreement in all 36 settings (Table 1, Fig. 2).

The mean candidate gap is .48094 against .22483 between the two retrains, a calibrated excess of .25611 ([.25258, .25968]), corresponding to geometric rank ratios of about 3.0 and 1.7. MRR compresses most of this: its gap is .00812 against a .00392 null, positive in 26/36 settings. Threshold statistics separate less sharply still: excess Top-10 disagreement is .0011 with an interval

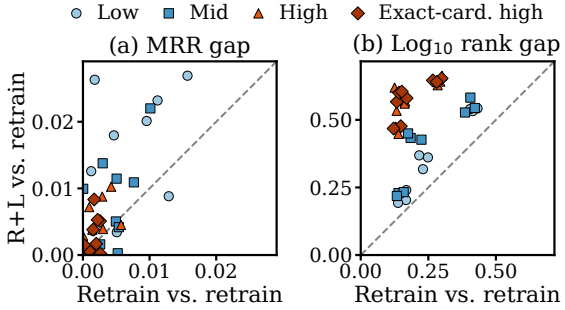


Figure 2: MRR and log-rank disagreement calibrated against independent retained-data retraining.

Table 1: Primary 36-setting audit and its breakdown by request construction (nine cells each).Candidate and null columns are $D_{\log r}$ (Eq. (2)); the excess is E_c (Eq. (3)). The interval is a 95% paired query bootstrap. Threshold statistics are in Appendix D.5.

	Log-rank displacement				
	Cand.	Null	Excess [95% CI]	MRR excess	Pos. (log-rank)
All 36 settings	.48094	.22483	.25611 [.25258, .25968]	.00420	36/36
Low	.3665	.2688	.0977	.0088	9/9
Mid	.4048	.2476	.1572	.0039	9/9
High	.5707	.1925	.3782	.0026	9/9
Exact-card. high	.5817	.1904	.3914	.0014	9/9

spanning zero ($[-.0001, .0023]$) and excess Top-100 is .0058 ($[.0044, .0071]$).

The discrepancy is therefore visible far more clearly in rank space than through aggregate MRR or short-list membership. It is also stable under leave-one-setting-out ($[.2493, .2624]$), leave-one-dataset-out ($[.2302, .2769]$), and tie-handling changes ($\sim 10^{-8}$; Appendix C).

Systematic but construction dependent. Log-rank excess is positive in all nine settings of every construction, so the mean neither rests on one request type nor cancels cells of opposite sign. Its magnitude rises fourfold with star degree, from .0977 on low-degree to .3914 on exact-cardinality requests (Table 1). Two quantities move against it: the retraining null falls from .2688 to .1904, so the raw candidate gap alone would mislead, and MRR excess falls from .0088 to .0014, so the settings with the strongest rank displacement are those MRR separates least. Disagreement therefore scales with how much positive support the deletion removes, an ordering §5 traces to embedding scale. *A third independent reference* on the 12 high-degree settings gives three candidate–reference and three reference–reference comparisons per setting. With none privileged, we report the *symmetrized* excess $\widetilde{E}_c$, averaged over the three references and three pairs rather than anchored to REFA as in Eq. (3) and so not directly comparable with it: the candidate gap averages .5808 against .1896, giving $\widetilde{E}_c = .3912$ ($[.3858, .3965]$), positive in 12/12 (Appendix C).

4.1 THE EVALUATION GAP EXTENDS BEYOND R+L

Table 2: Existing KGU methods and the fine-tuning baseline. Top-k excesses in percentage points.

Method	Cand.	Null	Excess	Pos.	T10	T100
Finetune	.450	.358	.092	6/6	2.29	3.54
GraphDPO	.732	.358	.374	6/6	9.69	11.60
SGU	.645	.358	.287	6/6	12.38	8.13

The primary grid uses R+L as a controlled vehicle for studying support loss, but the evaluation issue is not specific to restart. Applying the same calibration to Finetune the retained-data fine-tuning baseline from the GraphDPO release (Liu et al., 2026), GraphDPO (Liu et al., 2026), and SGU (Xiao et al., 2025a) on a separate random-fact benchmark over FB15K-237 and CoDEx-M with seeds 17/29/43, every method exceeds its retraining null in all six settings (Table 2). Threshold consequences are also larger here than on the primary grid, with excess Top-100 disagreement of 3.54–11.60 percentage points and excess Top-10 of 2.29–12.38 points under these methods' own benchmark regime.

Table 3: Full-split DistMult at a repair budget of five retained-data-equivalent units; median forgotten ranks by completion direction.

Graph	Model	Test MRR	Head	Tail
FB15K-237	R+L	.1585	11,101.0	145.0
	Ref20	.1559	10,485.5	10,111.5
CoDEx-M	R+L	.1489	8,794.5	220.5
	Ref20	.1538	6,996.5	5,999.0

More repair does not close the gap. R+L could still simply be under-repaired. In a separate full-split DistMult study (one fixed complete-star request per graph, three seeds), a logistic classifier receiving only answer ranks, trained and tested on disjoint query IDs and model seeds, separates R+L from the 20-epoch retained-data reference (Ref20) with AUC .9812 on FB15K-237 and 1.0000 on CoDEx-M, against $.5461/.5486$ for two independent retrains. Raising repair from one to ten retained-data-equivalent units cuts the forgotten-MRR gap by factors of 15.56 and 3.04 while AUC stays at $.9767/1.0000$; full-parameter and dense repair also leave AUC above .977 (Appendix B.1). The rank values make the separation concrete: for forgotten tail prediction the median rank is 145 for R+L versus 10,111.5 for Ref20 on FB15K-237,

and 220.5 versus 5,999 on CoDEx-M, at similar test MRR (Table 3). Reinitializing every embedding without repair instead collapses test MRR below .0006, so the effect is not indiscriminate destruction of useful structure. The head–tail asymmetry motivates a sharper question: does the discrepancy arise when an affected entity is supplied in the query, or when it is itself ranked as a candidate answer?

5 DIAGNOSING THE DISAGREEMENT

Scoring family and query role. A fixed-request comparison crosses six scoring families—TransE, DistMult, ComplEx, RotatE, PairRE and CompoundE (Bordes et al., 2013; Yang et al., 2015; Trouillon et al., 2016; Sun et al., 2019; Chao et al., 2021; Ge et al., 2023)—two graphs, three seeds, and two corruption rules, giving 72 configurations evaluated against independent 100-epoch references. The discrepancy concentrates in the bilinear models: DistMult and ComplEx exceed their null in all 24 comparisons (candidate gaps .956–1.052 versus null gaps .107–.155), whereas TransE, RotatE, PairRE, and CompoundE stay close to their controls and show seed-level reversals, so the mechanism is not uniform across families (Figure 5, Appendix C.1). Within the bilinear cohort the role of the affected entity is decisive: almost all excess appears when it is the *target being ranked*, and is near zero when it is supplied as query context (Fig. 3a). The relevant distinction is candidate versus context, not head versus tail. The candidate-side effect also depends strongly on retrieval depth, being negligible at $k = 1$, small and inconsistent at $k = 10$ ($.146/.179$ points, positive in only 9/24), rising to $10.514/11.572$ points at k=100 where it is positive in all 24, and reaching $47.2/47.0$.

The placement shift persists outside the deleted facts. Candidate-side localization could still be peculiar to forgotten queries. We therefore insert restarted zero-support entities into retained background contexts unrelated to their deleted facts, pairing each with an unaffected control of similar original training degree in the same context so that the difference removes context-wide shifts. Restarted entities are promoted by .5774 log-rank units relative to retraining ($[.5699, .5848]$, 36/36) while matched controls shift by only $-.0408$ ($[-.0460, -.0356]$, 2/36), a matched contrast of .6182 ($[.6090, .6273]$, standardized 1.3196) positive in all 36 settings. The affected entities thus occupy systematically different positions as candidates even in contexts unrelated to the deletion request.

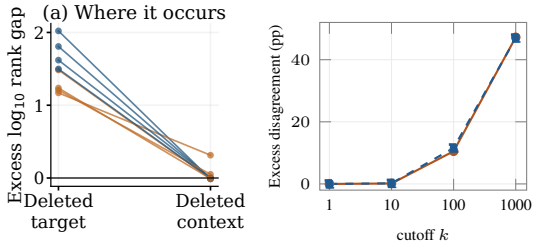


Figure 3: L: largest when the deleted entity is the ranked target, not the query context. R: excess grows with cutoff depth.

A corrected norm intervention identifies the dominant component. This motivates a direct diagnostic of embedding scale, NORMORACLE: rescale only the restarted zero-support rows to the corresponding REFA norms, leaving all other rows bit-identical. Because it consumes REFA norms it is an *oracle diagnostic, not a deployable correction*. On the primary grid it reduces calibrated excess from .25611 to .01999, a 92.19% reduction, and is strongest exactly where the uncorrected discrepancy is largest, from 64.0% on low-degree requests to 97.6% on exact-cardinality ones (Table 13); the four constructions converge to a narrow band despite a fourfold spread beforehand. By contrast, matching the expected number of negative presentations during retraining removes only 2.0% and 0.7% on the independent full-split cohort, where the same norm correction removes 93.0% and 88.1% (Appendix D.3). Embedding-scale mismatch, not negative exposure, dominates. NORMORACLE reduces the matched placement contrast from .6182 to $+.0747$, an 87.9% reduction positive in all 36 settings. Two features of that residual matter (Appendix D.3): it is nearly constant across constructions while the uncorrected contrast varies threefold, and it does not mirror the rank-space residual, where effectiveness still runs from 64.0% to 97.6%. The non-scale component of the low-degree discrepancy is therefore specific to forgotten-query rank fidelity rather than a general property of where restarted entities are placed. That residual is small but systematic (positive in 28 of 36 settings, largest on low-degree requests) so NORMORACLE does not establish behavioral equivalence to retraining. Notably the published methods of Section 4.1 do not reinitialize embeddings and so cannot exhibit scale mismatch at all, yet show excesses of .092–.374; the low-degree residual and the published-method excess are consistent with a second source our probes do not reach.

A false-query control separates broad placement from fact-specific effects. To test whether the misplacement is confined to the forgotten facts, on the full-split cohort we replace each forgotten

context by a relation-matched false context whose resulting triple is absent from train, validation and test. For tail prediction the restarted entities remain strongly over-promoted relative to retraining even in false contexts: the median R+L rank is $1/28$ of the reference rank on FB15K-237 and $1/20$ on CoDEx-M, against $1/83$ and $1/27$ in the corresponding true contexts. A substantial part of the effect therefore survives removal of the forgotten fact, though not all of it: R+L retains stronger true-versus-false discrimination than the reference (AUC .770 against .391 on FB15K-237, .689 against .497 on CoDEx-M; true-over-false log-rank advantage .516 against $-.027$ and .097 against .003), and for head prediction the restarted entities sit near the reference (ratios .651 and .960). Consistent with the scale account, their norms in this cohort are only .42 and .38 times the reference.

What the scale result implies. Because the dominant component is the *scale* of restarted embeddings rather than their direction, the largest part of the discrepancy is a broad miscalibration of restarted candidates rather than retained knowledge of the deleted facts: in a bilinear score a small-norm embedding yields scores near zero, which outrank the many candidates training has pushed negative, and the false-query control shows the misplacement persists in contexts the deleted facts never occupied. The residual is a different matter. NORMORACLE leaves 7.8% of the calibrated excess and the false-query control finds a true-over-false advantage absent from retraining, so we read the discrepancy as a dominant scale-induced placement component together with a smaller fact- and direction-specific component that our probes do not further resolve. Rank disagreement alone still does not establish membership leakage, and the evaluation claim is unaffected either way: a property as coarse as per-entity embedding norm displaces affected candidates in matched background contexts by roughly half a log-rank unit, while leaving forgotten MRR within .008 of retraining.

6 FROM DIAGNOSIS TO A REFERENCE-FREE CORRECTION

NORMORACLE identifies the dominant component but consumes a retained-data retraining reference, so an updater cannot apply it. We ask what remains achievable under the *candidate information budget*: the updated model, the retained training graph, the deletion request and original degrees, with no retrained checkpoint and no access to forgotten, validation, test, or placement-evaluation queries.

Retained-Norm Rescaling. The diagnosis says the restarted norms are too small; the question is whether a target can be read off the candidate itself. Retained-Norm Rescaling (RNR) sets $\|e_v\| = m_{\mathcal{R}}$ for every $v \in Z_{\mathcal{U}}$, the entities left with zero positive support in the retained graph and reinitialized by R+L, where

$$m_{\mathcal{R}} \;=\; \mathrm{median}_{u\,:\,\deg_{\mathcal{R}}(u)>0} \|e_u\| \tag{5}$$

is the median norm over entities that retain positive support. Directions are preserved and no other row is changed. RNR reads only the updated model and the retained graph; nothing is fitted or tuned.

RNR reduces calibrated log-rank excess from .25611 to .06242, a 75.63% reduction that recovers 82.03% of NORMORACLE's gain (Figure 4). Two alternatives share the same budget and appear alongside it throughout: Reference-Free Placement Correction (RFPC), a score-level offset that compensates the placement bias of restarted entities, fitted without access to REFA or REFB and specified before any comparison against a reference (Appendix D.4); and RNR-DM, a degree-matched variant of Eq. (5) defined below. RNR halves the residual RFPC leaves. Its reduction tracks request severity as NORMORACLE's does, from 28.2% on low-degree requests to 83.8% on exact-cardinality ones (Table 4). Normalizing by NORMORACLE separates two questions the raw reductions conflate: how much of the discrepancy is scale-linked, and how well that scale can be estimated without a reference. RNR recovers 86% of NORMORACLE's gain on high-degree and exact-cardinality requests and 79% on mid-degree, but only 44% on low-degree ones, where NORMORACLE itself reaches just 64.0%: a weaker estimate compounds a component that is not scale-linked to begin with.

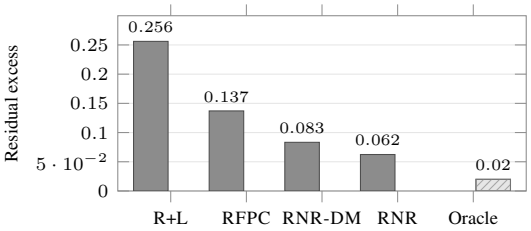


Figure 4: Residual calibrated log-rank excess. Hatched: NORMORACLE, which consumes a retraining reference. RNR recovers 82.0% of its gain.

Table 4: Residual calibrated log-rank excess by request construction and in aggregate, with threshold, forgotten-MRR and signed placement outcomes. The oracle uses REFA norms and is a diagnostic upper bound; the other three use candidate information only. RFPC is the median rule. The placement probe was not run for RNR-DM.

	Residual log-rank excess					Aggregate outcomes			
Method	Low	Mid	High	Exact	All 36	Top-10	Top-100	Forg. MRR	Placement
R+L	.0977	.1572	.3782	.3914	.25611	.00113	.00575	.00812	$+.6182$
RFPC	.0892	.0818	.1725	.2047	.13705	.00119	.00426	.00804	$-.0579$
RNR-DM	.0580	.0548	.1160	.1045	.08330	.00122	.00467	.00797	–
RNR	.0702	.0469	.0693	.0633	.06242	.00119	.00432	.00798	$-.1396$
NORMORACLE	.0352	.0174	.0182	.0093	.01999	–	–	–	$+.0747$

What the correction changes. RNR does not trade aggregate agreement for rank fidelity (Table 4): its forgotten-MRR gap is marginally below both RFPC's and uncorrected R+L's, leaving an MRR excess of $.00405$ against a retraining null of $.00392$, at a mean test-MRR change of -3.1×10^{-5}. The reduction in rank displacement is not, however, matched at the short-list cutoffs. Excess Top-10 disagreement is unchanged and lies inside an interval that already contained zero before correction, and excess Top-100 falls by about a quarter against a 75.63% reduction in calibrated log-rank excess. The threshold statistics do not even separate RNR from RFPC, although their log-rank excesses differ by a factor of 2.2. This is consistent with the rest of the audit: threshold consequences on the primary grid were small to begin with (Table 1), the bilinear cohort shows excess only at $k \geq 100$ (Appendix C.1), and under the stronger recipe it reappears only at $k = 1000$ (§7). The displacement RNR removes lies predominantly below conventional retrieval cutoffs. The R+L–RNR pair states this paper's thesis in its sharpest form. Between the two the calibrated log-rank excess falls by three quarters, from $.2561$ to $.0624$, while the forgotten-set MRR moves from $.00812$ to $.00798$. The aggregate metric on which KGU is conventionally evaluated does not distinguish them at all.

Why the simpler target wins, and where it overshoots. A natural refinement replaces the global target of Eq. (5) with a per-entity median over the degree-matched control sets RFPC constructs. RNR-DM beats RFPC in 33 of 36 settings under the median rule and 34 of 36 under the mean rule, but is worse than the simpler global target, and worse precisely where the discrepancy is largest (Table 4). Degree does not predict the quantity being matched: in REFA the entities of $Z_{\mathcal{U}}$ have no positive support either, so they receive only negative gradients, and under uniform corruption the rate at which an entity is drawn as a negative is independent of its original degree. Empirically the correlation between pre-deletion degree and the REFA norm is negligible ($r = -.033$; Appendix D.5). Degree matching therefore adds variance without signal, and on high-degree requests it adds bias: the matched controls are *supported* entities with large norms.

The signed placement contrast then shows what the log-rank excess, an absolute difference, cannot: NORMORACLE leaves a residual promotion of $+.0747$, positive in all 36 settings, whereas RNR crosses to $-.1396$. Multiplicative norm correction does not itself reverse the sign; the reference-free target does, because it is systematically too large — the ratio $m_{\mathcal{R}}/\|e_v^{\text{REFA}}\|$ has median 1.455 and exceeds one for 99.98% of restarted entities (Appendix D.5). Part of RNR's residual is therefore over- rather than under-correction. The ratio is stable enough that dividing $m_{\mathcal{R}}$ by it would close much of the gap, but it is measured against REFA norms, so a correction calibrated on it would no longer satisfy the information budget that defines RNR. A reference-free estimator of the post-retraining norm of an unsupported entity would close it legitimately; we record that as the natural next step rather than fitting a constant here.

RFPC, and the status of the interventions. RFPC carries an acceptance criterion frozen in advance: at least 25% reduction in both the calibrated log-rank excess and the absolute placement contrast, at a test-MRR change above $-.005$ and a forgotten-MRR gap below $.010$ (Appendix D.4). All four components are met, and on the placement contrast it outperforms RNR (90.6% against 77.4%) — as its name implies, RFPC's offset is estimated directly from the background-context score difference, the quantity the placement probe measures, so it minimizes that contrast almost by construction, while RNR corrects the underlying scale and achieves far better rank fidelity on the forgotten queries the audit is about. RFPC also does not compose with the norm correction: applied after NORMORACLE it raises the residual from $.01999$ to $.11122$, improving on it in only 2 of 36 settings, and for severe requests it returns the model to essentially the residual it reaches alone (Appendix D.6). Unlike RFPC, RNR and RNR-DM are derived from NORMORACLE and are reported post hoc; no further norm-target variants were tried.

7 How far does the result extend?

The primary grid uses a modest 20-epoch recipe. We repeat the comparison under a substantially stronger one (reciprocal 1-N softmax, dimension 256, Adagrad, and validation-based early stopping) with one fixed request per graph, each removing roughly 10% of training triples, and three seeds. The outcome is dataset dependent, and the nine cells span a sign reversal (Table 5): calibrated excess is positive in 3/3 seeds on CoDEx-M, 2/3 on FB15K-237, and 0/3 on NELL-995. We therefore report it by graph; the pooled mean of $.146$ ($[.138, .153]$, positive in 5/9) averages cells of opposite sign and is a summary, not an effect size. The FB15K-237 cells make the variability concrete: all three share one realized deletion request and differ only in training seed, yet the excess runs $+.283$, $+.160$, $-.020$, so training randomness alone reverses it (Appendix E). Negative excess means the candidate is closer to RefA than RefB is under this metric, not that retraining equivalence holds.

Table 5: Stronger training: one request per graph, three seeds. Components are rounded independently; the overall row spans a sign reversal.

Graph	Cand.	Null	Excess	Pos.
CoDEx-M	.633	.207	.427	3/3
FB15K-237	.552	.411	.141	2/3
NELL-995	.488	.618	$-.130$	0/3
Overall	.558	.412	.146	5/9

The retraining null is not a constant of the problem. The NELL-995 reversal comes from a large retrain–retrain null rather than from close candidate agreement. Early stopping does not explain it: the mean absolute RefA–RefB stopping-epoch difference is 16.7 there against 26.7 on both other graphs, the smallest spread of the three rather than the largest. What moves across regimes is the null, not the candidate. It is $.2248$ in the primary grid and $.1896$ in the three-reference control, but $.412$ under the stronger recipe and $.618$ on NELL-995 alone (a factor of three) while the candidate gap moves only from $.481$ to $.558$. The excess shrinks, and on one graph reverses, because the baseline expands rather than because the candidate approaches retraining. Retraining variability is therefore a property of the graph and the optimization regime, not a constant to be assumed or imported from another setting. NELL-995 turns this paper's own point on the baseline: RefA and RefB reach almost identical test MRR ($.258$ and $.260$) while disagreeing by $.618$ log-rank units.

Threshold consequences move deeper rather than vanishing. Under the stronger recipe the displacement no longer reaches the short-list cutoffs: excess Top-10 disagreement is $-.028$ percentage points (95% CI $[-.072, .011]$, positive in 2/9) and excess Top-100 is $-.117$ points ($[-.261, .022]$, 6/9), both intervals containing zero. It has not disappeared, only moved down the list. At $k = 1000$ candidate–reference disagreement is $.2366$ against a retraining null of $.1298$, an excess of 10.68 percentage points ($[10.04, 11.32]$), so the divergence sits between ranks 100 and 1000 rather than near the top. The bilinear cross-family cohort shows the same shape one decade closer to the top of the list — absent at $k \leq 10$, 10.5–11.6 points at $k = 100$, 47 points at $k = 1000$ (Appendix C.1). In two cohorts differing in model, request and reference budget, the effect is therefore relocated below the conventional cutoffs rather than removed.

8 Conclusion and future work

We began with a question: when an updated model appears close to retraining under aggregate forgetting metrics, how much agreement does that imply? Our experiments show surprisingly little. The lesson is that removing support and restoring an entity's placement are distinct problems. Restarting removes its learned representation, but repair does not necessarily place it where retained-data retraining would. Unsupported entities can remain miscalibrated even in contexts unrelated to deleted facts. Much of the seemingly fact-specific disagreement instead reflects how restarted candidates compete with supported entities. The surprising finding is how much of this discrepancy stems from embedding scale. An oracle norm intervention removes 92.2% of the calibrated excess, and a reference-free rescaling fitted from the updated model alone removes 75.6%, from $.2561$ to $.0624$. A difference that large, that structured and that correctable was present while the forgotten-set MRR barely moved. More evaluation exposed a weakness, revealed a mechanism, and suggested a correction. Stronger-training experiments show that disagreement with a single retrained model does not establish update failure: it must be calibrated against retraining variability under the same graph, deletion request, and training regime. *Our contribution is a connected approach: measure disagreement beyond aggregates, calibrate it against independent retraining, and use targeted interventions to identify its sources.* Open questions concern estimating the appropriate scale of unsupported entities and explaining the disagreement that scale alone cannot resolve.

Reproducibility statement

We release the complete experimental code and scripts with instructions for reproducing all experiments, tables and figures in the Supplemental Material. The concise appendix records the cohort and evaluator contracts (Appendix A), the matched update procedure (Algorithm 1), calibration and uncertainty (Appendix C), the frozen correction protocol (Appendix D.4), and training-regime sensitivity (Appendix E). Reported intervals condition on the observed models and requests.

AI use statement

Generative AI tools assisted with experimental design, implementation and debugging, experiment orchestration, interpretation of results and manuscript editing. All reported numbers were produced by the released experimental code run by the authors; no result, table entry or figure was generated by a language model. The authors reviewed and validated every experimental configuration, output, numerical result and scientific claim, and retain full responsibility for the content of the paper.

Ethics statement

This work studies how to tell whether an unlearning update has behaved like retraining, which bears on the practical enforcement of data-deletion requests. All experiments use public knowledge-graph benchmarks derived from Freebase, Wikidata, Wikipedia and web extraction; these contain publicly available factual statements, some concerning real people, and we collect no new data.

The audits compare behavior with retained-data retraining and certify nothing beyond that. A low forgotten-set MRR, a small calibrated discrepancy, or a near-chance output-source classifier is not a privacy certificate. Conversely, rank disagreement alone does not establish membership leakage, memorization of deleted facts, or legal non-compliance: our false-query control measures a fact-specific residual alongside a broad scale-induced component, and we interpret neither as evidence of leakage.

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pp. 2787–2795, 2013.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159. IEEE, 2021. doi: 10.1109/SP40001.2021.00019.

Linlin Chao, Jianshan He, Taifeng Wang, and Wei Chu. PairRE: Knowledge graph embeddings via paired relation vectors. In *Proceedings of ACL-IJCNLP*, pp. 4360–4369, 2021. doi: 10.18653/v1/2021.acl-long.336.

Jiali Cheng, George Dasoulas, Huan He, Chirag Agarwal, and Marinka Zitnik. GNNDelete: A general strategy for unlearning in graph neural networks. In *International Conference on Learning Representations*, 2023.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, volume 32, pp. 1811–1818, 2018. doi: 10.1609/aaai.v32i1.11573.

Xiou Ge, Yun Cheng Wang, Bin Wang, and C.-C. Jay Kuo. Compounding geometric operations for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6947–6965, 2023. doi: 10.18653/v1/2023.acl-long.384.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3832–3842. PMLR, 2020.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 497–519. IEEE, 2025. doi: 10.1109/SATML64287.2025.00034.

Jiajun Liu, Wenjun Ke, Peng Wang, Yao He, Ziyu Shang, Guozheng Li, Zijie Xu, and Ke Ji. Unlearning of knowledge graph embedding via preference optimization. In *Proceedings of the ACM Web Conference*, pp. 3964–3973, 2026. doi: 10.1145/3774904.3792397.

Martin Pawelczyk, Jimmy Z. Di, Yiwei Lu, Gautam Kamath, Ayush Sekhari, and Seth Neel. Machine unlearning fails to remove data poisoning attacks. In *International Conference on Learning Representations*, 2025.

Tara Safavi and Danai Koutra. Codex: A comprehensive knowledge graph completion benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8328–8350, 2020.

Harry Shomer, Wei Jin, Wentao Wang, and Jiliang Tang. Toward degree bias in embedding-based knowledge graph completion. In *Proceedings of the ACM Web Conference 2023*, 2023. doi: 10.1145/3543507.3583544.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*, 2019.

Kristina Toutanova and Danqi Chen. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pp. 57–66. Association for Computational Linguistics, 2015. doi: 10.18653/v1/W15-4007.

Eleni Triantafillou, Peter Kairouz, Fabian Pedregosa, Jamie Hayes, Meghdad Kurmanji, Kairan Zhao, Vincent Dumoulin, Júlio C. S. Jacques Júnior, Ioannis Mitliagkas, Jun Wan, Lisheng Sun-Hosoya, Sergio Escalera, Gintare Karolina Dziugaite, Peter Triantafillou, and Isabelle Guyon. Are we making progress in unlearning? findings from the first NeurIPS unlearning competition. *arXiv preprint arXiv:2406.09073*, 2024.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pp. 2071–2080, 2016.

Qizhou Wang, Bo Han, Puning Yang, Jianing Zhu, Tongliang Liu, and Masashi Sugiyama. Towards effective evaluations and comparisons for LLM unlearning methods. In *International Conference on Learning Representations*, 2025.

Kun Wu, Jie Shen, Yue Ning, Ting Wang, and Wendy Hui Wang. Certified edge unlearning for graph neural networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2606–2617. Association for Computing Machinery, 2023. doi: 10.1145/3580305.3599271.

Yang Xiao, Ruimeng Ye, and Bo Hui. Knowledge graph unlearning with schema. In *Proceedings of COLING*, pp. 3541–3546, 2025a.

Yang Xiao, Ruimeng Ye, Bohan Liu, Xiaolong Ma, and Bo Hui. Efficient knowledge graph unlearning with zeroth-order information. In *Proceedings of CIKM*, pp. 3509–3519, 2025b. doi: 10.1145/3746252.3761379.

Wenhan Xiong, Thien Hoang, and William Yang Wang. Deeppath: A reinforcement learning method for knowledge graph reasoning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 564–573. Association for Computational Linguistics, 2017. doi: 10.18653/v1/D17-1060.

Naixing Xu, Qian Li, Zhaochuan Li, Xu Wang, Bingchen Liu, Jabulani Brown Mpofu, Jingchen Li, and Xin Li. Learn to unlearn: Meta-learning-based knowledge graph embedding unlearning. *Knowledge and Information Systems*, 67(8):7101–7122, 2025. doi: 10.1007/s10115-025-02407-4.

Bishan Yang, Wen tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations*, 2015.

Lu Yi and Zhewei Wei. Scalable and certifiable graph unlearning: Overcoming the approximation error barrier. In *International Conference on Learning Representations*, 2025.

A EXPERIMENTAL SETUP AND REPRODUCIBILITY

This appendix documents the experimental cohorts discussed in the main paper. Unless stated otherwise, models use embedding dimension 128, 20 base-training epochs, and seeds 17, 29, and 43.

A.1 DATASETS AND DELETION REQUESTS

Table 6 lists the datasets used in the paper. FB15K-237, WN18RR, CoDEx-M, CoDEx-L, and YAGO3-10 use their published train/validation/test splits (Toutanova & Chen, 2015; Dettmers et al., 2018; Safavi & Koutra, 2020). For NELL-995, the loader starts from the 154,213-triple `raw.kb` graph in the DeepPath release (Xiong et al., 2017). A deterministic holdout with seed 17 assigns 1,000 triples to validation and 2,000 to test, leaving 151,213 training triples. Entity and relation identifiers are constructed jointly over the loaded splits and remain fixed within each experiment. NELL-995 is the only graph in the paper without a published split, and its validation set is small relative to its 75,492 entities; this is relevant to the training-regime sensitivity reported in Appendix E.

Table 6: Dataset statistics for the experimental splits. CoDEx-L, WN18RR, and YAGO3-10 are used only in the aggregate TransE comparison (Table 8); the primary calibrated grid uses FB15K-237, CoDEx-M, and NELL-995.

Dataset	Entities	Relations	Train	Valid	Test
FB15K-237	14,541	237	272,115	17,535	20,466
WN18RR	40,943	11	86,835	3,034	3,134
CoDEx-M	17,050	51	185,584	10,310	10,311
CoDEx-L	77,951	69	551,193	30,622	30,622
YAGO3-10	123,182	37	1,079,040	5,000	5,000
NELL-995	75,492	200	151,213	1,000	2,000

Deletion construction. Complete-star deletion removes all training triples incident to the selected entities. Because stars have discrete sizes, the realized deletion fraction can exceed its nominal target. The exact-cardinality high-degree variant truncates the final star; consequently, not every selected entity necessarily loses all positive support. Restart is always determined from the realized retained graph.

The primary DistMult grid crosses FB15K-237, CoDEx-M, and NELL-995, three seeds, and four constructions: low-, mid-, and high-degree stars, and exact-cardinality high-degree deletion. Each of the 36 requests removes approximately 10% of training triples: about 27k, 19k, and 15k facts on the three graphs, respectively. The aggregate TransE comparison instead uses 10% and 20% complete-star requests, with three seeds per dataset.

For the existing-method audit, random-fact deletion samples

$$k = \mathrm{round}(f|\mathcal{T}|)$$

distinct training triples uniformly without replacement, using selector seed `request_seed` + $\lfloor 10{,}000 f \rfloor$. The exact request is exported with a SHA-256 fingerprint. Matched evaluators abort on a request-hash mismatch. Both nominal and realized deletion sizes are recorded, and requests are not resampled during evaluation.

Candidate vocabulary and filtering. The candidate vocabulary is fixed before deletion and is not pruned afterwards: entities with zero retained positive support remain valid answers. Head and tail prediction use the same candidate universe. Filtered evaluation removes alternative known truths from the original train/validation/test truth set while retaining the evaluated target. Using $\mathcal{R} \cup \mathcal{U} \cup \mathrm{valid} \cup \mathrm{test}$ for filtering does not use deleted triples as positive training examples.

Evaluation cohorts are sampled deterministically and saved. The primary rank audit uses at most 1,000 forgotten triples per setting, evaluated in both completion directions. The GraphDPO comparison path uses a common vocabulary constructed from the exported request files and aborts if a forgotten triple is unsupported.

Primary evaluator. Scores are computed in float32. For base seed s, REFA and REFB use seeds $s + 500003$ and $s + 700003$, respectively; the third reference used in the multiple-retrain check

(Appendix C) uses seed $s + 900003$. If $n_>$ competitors score above the target and $n_=$ competitors tie with it, mean-tie rank is

$$r = 1 + n_> + \frac{n_=}{2}.$$

The target is excluded from the tied-competitor count. The optimistic alternative is $1 + n_>$.

A.2 Experimental cohorts and method provenance

The experiments vary different aspects of the evaluation and are not pooled across reference contracts. Table 7 makes these distinctions explicit.

Table 7: Experimental cohorts and retained-data reference contracts.

Cohort	Reference contract	Population
Aggregate restart comparison	20-epoch retained-data reference	TransE; four datasets, three seeds, two deletion fractions; 24 configurations
Primary calibration	Independent RefA/RefB, 20 epochs	DistMult; three graphs, three seeds, four constructions; 36 settings
Multiple-retrain check	Three independent 20-epoch references	DistMult; high-degree construction only; 12 settings
Reference-free corrections	Same references as the primary calibration; corrections fitted from the candidate model and retained graph only	RFPC, RNR and RNR-DM on all 36 primary settings; NormOracle on the same settings as a reference-consuming diagnostic
Existing-method audit	Benchmark RefA and independent same-procedure RefB	Two graphs, three seeds, 10% random-fact deletion; six settings per method
Cross-family audit	Independent 100-epoch references	Six families, two graphs, three seeds, two corruption rules; one fixed request per graph
Full-split repair	Independent same-budget Ref20; separate repair trajectory	DistMult; two graphs, one fixed complete-star request per graph, three seeds
Stronger training	Independent references using reciprocal 1-N training and validation stopping	DistMult; three graphs, one fixed request per graph, three seeds; nine cells

Finetune and GraphDPO use the public GraphDPO implementation (Liu et al., 2026). SGU-Schema is a matched adaptation of schema-guided unlearning (Xiao et al., 2025a), initialized from the same pretrained checkpoint and evaluated on the same request, vocabulary, and filtering contract. It is not an original-author checkpoint. In the RotatE, PairRE, and CompoundE extension, SGU-Schema and GraphDPO are matched adaptations using each family's native score.

These comparisons assess the evaluation phenomenon across update procedures; they do not establish a leaderboard under equal tuning effort or training work. Historical reference names such as `Ref20`, `FR20`, and `Reference100` identify the finite training budget of the corresponding cohort.

Table 8: Aggregate restart comparison: TransE after complete-star deletion, six configurations per dataset (target fractions 10%/20%, three seeds). Entries are absolute MRR gaps to fresh 20-epoch retraining; smaller is better. R+L reinitializes the selected entity embeddings before the same update used by Local.

	R+L		Local	
Dataset	ΔM	ΔM_f	ΔM	ΔM_f
FB15K-237	.0056	.00059	.0076	.01896
WN18RR	.0067	.00016	.0326	.69200
CoDEx-L	.0118	.00037	.0120	.00282
YAGO3-10	.0096	.00011	.0085	.00129
All 24 configurations	.00842	.000311	–	–

A.3 Cluster configuration and execution

Experiments run through SLURM on the `aiq` partition of our institutional cluster. GPU campaigns use NVIDIA H200 accelerators through MIG profiles, commonly one `3g.71gb` instance; memory-lighter jobs use `1g.18gb`. Tasks typically request 1–4 CPU cores and 32–64 GB host RAM.

CPU-only collectors and some stronger-training runs use up to four CPU threads. SLURM throttles array concurrency according to the account GPU quota.

GPU jobs pin OpenMP, MKL, OpenBLAS, and NumExpr threads to one unless explicitly CPU-bound. Campaign manifests record software versions where supported, device information, source and input hashes, request and query hashes, seeds, and Git commit. SLURM logs additionally record host and job/array identifiers.

The public GraphDPO repository specifies Python 3.9 and PyTorch 1.13.1 (Liu et al., 2026). Our checkout follows that benchmark workflow with documented compatibility changes, including parsing the seed argument as an integer.

Reported update times measure the intervention path under the recorded allocation. They are distinct from full-campaign times, which can include base training, references, evaluation, and serialization. The reproducibility release includes code, configuration files, SLURM scripts, request manifests, per-run results, and instructions for regenerating the reported tables and figures.

B RESTART-AND-REPAIR CONSTRUCTION AND ABLATIONS

The comparison in Section 3 keeps the candidate vocabulary fixed after deletion. The restart rule acts only on $Z_{\mathcal{U}}$, the entities with zero positive support in the retained training graph. Algorithm 1 specifies the TransE implementation; other scoring families use their native scores, losses, and initializers.

Algorithm 1: RESTART+LOCAL-L5 for TransE.
Input: trained model, training triples $\mathcal{T}$, request $\mathcal{U}$, seed s, target fraction f.

1. Set $\mathcal{R} = \mathcal{T} \setminus \mathcal{U}$ and identify $Z_{\mathcal{U}}$. Draw a Xavier-normal entity table with seed $s + 700001 + \lfloor 10^4 f \rfloor$ and copy its rows for $Z_{\mathcal{U}}$ into the model.
2. Let A_E contain the head and tail entities of deleted triples and A_R their forward and inverse relations. Only these embeddings are trainable.
3. Collect retained triples incident to A_E in $\mathcal{R}_{\mathrm{inc}}$. Append a seeded uniform sample of
$$\min\{|\mathcal{R}|, \max(\lfloor |\mathcal{R}_{\mathrm{inc}}|/2 \rfloor, 4096)\}$$
retained triples.
4. Add inverse triples and train for five epochs with Adam, batch size 1024, learning rate 10^{-4}, normalized-L1 margin loss with margin 8, and ten negatives per positive.

Local performs steps 2–4 without restart. RestartOnly performs step 1 without repair.

R+L and Local share the checkpoint, trainable parameters, and repair procedure; their difference is the reinitialization of $Z_{\mathcal{U}}$. For the partial-star control, positive retained support remains, so no restart is triggered and the two procedures coincide.

Restart and repair-region controls. Table 9 gives the ablations summarized in the main paper. Boundary and Random use equally sized repair regions. Repair reduces the forgotten-MRR gap relative to RestartOnly; larger regions yield smaller additional reductions without uniformly improving test agreement.

Table 9: Restart and repair-region ablations. MRR entries are absolute gaps to the retained-data reference. Fractions describe the trainable parameters and retained data used for repair.

Update	Test gap	Forgotten gap	Param. fraction	Data fraction
RestartOnly	.00981	.01204	–	–
R+L (Local repair)	.00962	.00615	.557	.814
Boundary repair	.00987	.00520	.786	.949
Random repair	.01074	.00513	.786	.949

Partial-star deletion gives a forgotten-MRR gap of .0415 for both Local and R+L.

B.1 FULL-SPLIT REPAIR-BUDGET AUDIT

The DistMult repair study summarized in Section 4.1 fixes one complete-star request per graph and evaluates three seeds. One repair unit presents $|\mathcal{R}|$ positive examples. Local and full repair match initialization, step count, batch size, and negative seed; a dense control uses the local example stream. Output-source classifiers use disjoint query IDs and model seeds for training and testing. The reported median ranks pool 256 fixed triples and three seeds and are descriptive summaries.

Raising repair from one to ten retained-data-equivalent units reduces the forgotten-MRR gap by factors of 15.56 on FB15K-237 and 3.04 on CoDEx-M, while the output-source AUC moves only from $.9812/1.0000$ to $.9767/1.0000$. Full-parameter and dense repair leave AUC above .977. Two independent retrains of the same retained graph are separated with AUC $.5461/.5486$ under the identical classifier protocol.

C RETRAINING-CALIBRATED RANK EVALUATION

This section specifies the uncertainty analysis supporting Sections 4 and 5.

Paired estimation and uncertainty. For setting c, forgotten triple i, direction $d \in \{h, t\}$, candidate C, and independent references A and B, define

$$e_{ci} = \frac{1}{2} \sum_{d \in \{h,t\}} \left(|\log_{10} r_{cid}(C) - \log_{10} r_{cid}(A)| - |\log_{10} r_{cid}(B) - \log_{10} r_{cid}(A)| \right), \tag{6}$$

$$\widehat{E} = \frac{1}{S} \sum_{c=1}^{S} \frac{1}{n_c} \sum_{i=1}^{n_c} e_{ci}. \tag{7}$$

The primary analysis uses $S = 36$. Bootstrap replicates resample triples with replacement independently within each setting, preserve the paired model observations and both prediction directions, and average settings equally. We use 10,000 replicates, seed 19092026, and percentile intervals.

For MRR, each replicate computes the absolute difference between model-level mean reciprocal ranks within each setting, subtracts the corresponding reference/reference gap, and averages settings equally. Top-k disagreement uses paired threshold indicators. These intervals condition on the observed requests and trained models; they do not measure variation over new training runs, requests, or datasets.

Note that $\widehat{E}$ is built from *absolute* log-rank differences. It is therefore positive whether a candidate sits above or below the reference, and cannot by itself distinguish under-correction from overshoot. The signed placement contrast of Eq. (10) can, and we use it for that purpose in Appendix D.5.

Reduction estimand. For an intervention producing post-correction excess $\bar{E}^{\text{post}}$ from pre-correction excess $\bar{E}^{\text{pre}}$, we report the *ratio-of-means* reduction

$$\rho = 1 - \frac{\bar{E}^{\text{post}}}{\bar{E}^{\text{pre}}}, \tag{8}$$

where both terms are the equal-setting means of Eq. (7). Because the reduction is concentrated in the constructions with the largest excess, the equal-weight mean of per-construction reductions is smaller than ρ: 86.5% against 92.19% for NORMORACLE and 39.7% against 46.49% for RFPC. All headline percentages in the paper use ρ; per-construction percentages are computed within each construction.

Multiple-retrain calibration. The primary design yields one retrain–retrain distance per setting. To check that the calibrated excess does not depend on an unusually close or distant reference pair, we train a third independent reference on the 12 high-degree settings, giving three candidate–reference and three reference–reference comparisons per setting. Since no reference is privileged there, we report the *symmetrized* excess

$$\widetilde{E}_c = \frac{1}{3} \sum_{X \in \{A,B,C'\}} D(\text{cand}, X) - \frac{1}{3} \sum_{\{X,Y\} \subset \{A,B,C'\}} D(X, Y), \tag{9}$$

with D the mean absolute log-rank gap. $\widetilde{E}_c$ is not comparable with the REFA-anchored $\widehat{E}$ of Eq. (7), and every reported figure states which of the two it uses. Table 10 gives the results: $\widetilde{E}_c = .3912$ (95% CI $[.3858, .3965]$), positive in 12/12 settings.

The three pairwise reference distances within a setting span a mean of .0075 (CoDEx-M .0072, FB15K-237 .0049, NELL-995 .0105). This is a descriptive spread only: the three distances are built from three training runs, are exchangeable and positively correlated, and therefore do not provide an independent variance estimate. The separation between .5808 and .1896 is two orders of magnitude larger than that spread.

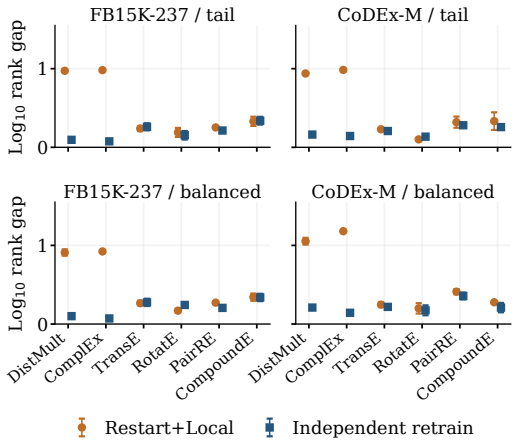


Figure 5: Cross-family audit: 72 configurations over six scoring families, two graphs, three seeds and two corruption rules. Points are mean absolute $\log_{10}$ rank gaps against independent 100-epoch references. Only DistMult and ComplEx separate from their retraining null; the remaining four families overlap it.

Table 10: Three-reference calibration on the 12 high-degree settings. Distances are mean absolute log-rank gaps; the excess is the symmetrized statistic of Eq. (9).

Graph	Cand.–Ref.	Ref.–Ref.	$\widetilde{E}_c$	Positive
CoDEx-M	.4924	.1332	.3593	4/4
FB15K-237	.5989	.1471	.4518	4/4
NELL-995	.6510	.2886	.3624	4/4
Overall	.5808	.1896	.3912	12/12

Retraining variability is itself graph dependent: the retrain–retrain distance is .1332 on CoDEx-M and .2886 on NELL-995 under the same 20-epoch recipe. Even the largest null remains well below the candidate gap in this cohort, but the spread motivates measuring the null under the audited recipe rather than importing it (Appendix E).

Per-dataset calibrated excess. The equal-setting mean of .25611 decomposes into .3079 on FB15K-237, .2145 on CoDEx-M and .2459 on NELL-995, each over the 12 settings of that graph. These denominators are used for the per-dataset reductions in Appendix D.5.

Sensitivity checks. Removing one setting at a time gives a mean log-rank excess range of $[.249305, .262402]$. Leaving out one dataset gives $[.230201, .276929]$. Mean-minus-optimistic log-rank excess is -1.03×10^{-8}, with paired interval $[-3.94 \times 10^{-8}, 1.33 \times 10^{-8}]$. Top-10 and Top-100 disagreement are unchanged. The positive primary effect therefore persists under these setting, dataset, and tie-rule checks.

Existing-method audit. The results in Table 2 cover six matched random-fact settings per method: FB15K-237 and CoDEx-M, seeds 17/29/43, and 10% deletion. REFA is the benchmark retained-data retrain and REFB is an independent retrain on the same retained graph. These settings are separate from the support-loss grid.

C.1 Cross-family evaluation and retrieval cutoffs

The cross-family audit uses dimension 128, Adam with learning rate .002, batch size 2048, one corrupted negative per positive, softplus losses, and mean-squared regularization 10^{-6}. Training uses float32 and ranking float64. The 256 fixed evaluation triples are sampled with seed 19072026.

The audit crosses six scoring families, two graphs, three seeds, and two corruption rules. Table 11 reports the 24 bilinear comparisons. These fixed-request, 100-epoch-reference results are kept separate from the 36-setting primary calibration.

Table 11: Retrieval-cutoff sensitivity in the bilinear cross-family cohort. Excess disagreement is measured in percentage points; positive counts cover both corruption rules.

Cutoff k	Tail-only negatives	Balanced negatives	Positive
1	-.016	.000	0/24
10	.146	.179	9/24
100	10.514	11.572	24/24
1000	47.217	46.973	24/24

In this cohort, excess disagreement is negligible at short-list cutoffs, inconsistent at $k = 10$ (positive in only 9 of 24 comparisons), and large at broader cutoffs. Its magnitude should not be interpreted as the corresponding effect in the primary grid, and the cutoff dependence should be read together with the stronger-training result in Appendix E, where the same pattern appears one decade deeper.

DistMult and ComplEx exceed their retraining null in all 24 comparisons, with candidate log-rank gaps of .956–1.052 against null gaps of .107–.155. TransE, RotatE, PairRE, and CompoundE remain much closer to their controls and show seed-level reversals, so the effect is not uniform across scoring families.

Directional decomposition. For each cutoff, candidate-only inclusion means that the target appears within Top-k under the candidate but not the reference; reference-only inclusion is the reverse event. Their sum gives membership disagreement and their difference gives the candidate-minus-reference Hits@k change. Exact-rank equality is not used as a standalone criterion because independent retrains can assign different ranks with small log-rank displacement.

D Candidate placement, embedding scale, and reference-free correction

D.1 Matched background-context probe

The placement experiment in Section 5 uses the saved checkpoints of the 36-setting primary grid. Within each setting, up to 256 restarted zero-support entities are paired with unaffected controls approximately matched on original training degree. Both members of a pair are inserted into the same retained background context, and both constructed triples must be absent from train, validation, and test.

With $r_M(e \mid q)$ denoting the full-vocabulary mean-tie rank of entity e in context q under model M, promotion is

$$p(e, q) = \frac{1}{2}\left(\log_{10} r_A(e \mid q) + \log_{10} r_B(e \mid q)\right) - \log_{10} r_C(e \mid q). \tag{10}$$

Positive promotion means that the candidate ranks the entity higher than the geometric mean of its reference ranks. Subtracting the matched control's promotion in the same context gives the placement contrast reported in Table 4. Unlike $\widehat{E}$, this statistic is signed, so it distinguishes residual promotion from overshoot.

The placement bootstrap resamples paired entity/context probes within each setting and averages settings equally, using 10,000 replicates and seed 21092026. It conditions on the observed checkpoints and probe cohorts.

D.2 Placement probe for published methods

To test whether a deletion-localized placement effect appears outside the restart construction, we run the same matched probe on Finetune, GraphDPO, and SGU-Schema in the random-fact cohort. Because these methods do not reinitialize embeddings, the probe targets entities *touched* by the deletion request that nevertheless retain positive support, matched one-to-one to untouched controls with exactly the same retained endpoint degree; each of the six cells uses 256 matched pairs with zero retained-degree mismatch by construction.

Table 12: Deletion-localized placement probe for published methods. DID is the touched-minus-control contrast, measured relative to the center of two independent retrains as in Eq. (10). Rounded components need not subtract exactly.

Method	Touched	Control	DID	95% CI	Positive
Finetune	.003	$-.014$	.016	$[.003, .029]$	5/6
GraphDPO	.011	$-.007$	.018	$[-.001, .038]$	6/6
SGU-Schema	.012	$-.022$	.033	$[.019, .047]$	6/6

The contrasts are positive in point estimate for all three methods, but one to two orders of magnitude smaller than the $.6182$ observed for restarted entities under R+L, and they cannot account for the calibrated log-rank excesses of $.092$–$.374$ that the same methods exhibit (Table 2). The GraphDPO interval includes zero. The decomposition is also method dependent: for Finetune the touched entities barely move and the contrast arises mainly from demotion of matched controls. We therefore do not treat this as evidence that the mechanism identified for R+L generalizes to these methods. The CoDEx-M requests contain no zero-retained-support touched entities, yet the contrast remains positive, so the effect does not require the zero-support condition.

D.3 Norm and negative-exposure controls

NormOracle rescales the embedding rows of restarted zero-support entities to their RefA norms, leaving every other row bit-identical. It consumes a reference checkpoint and is therefore a diagnostic, not a reference-free intervention. An earlier implementation applied the rescaling through advanced indexing with an in-place multiplication, which operates on a temporary and never writes the rescaled rows back to the embedding table; that version was numerically indistinguishable from uncorrected R+L and is superseded here. The evaluation harness shared the same indexing path, so figures computed before the fix are superseded as well.

NegExposure20eq starts from R+L and gives restarted entities the expected number of uniform-tail negative presentations from 20 epochs of retained-data retraining, without using a reference checkpoint. This matches expected exposure counts, not the complete sequence of negative contexts.

On the full-split cohort, median restarted-entity norms are $.42$ and $.38$ times their Ref20 values on FB15K-237 and CoDEx-M, so the restarted rows are roughly 2.4–2.6 times too small there.

The same reduction holds on the independent full-split cohort. Subtracting the retraining null makes the two cohorts comparable (Table 14). On FB15K-237 the candidate–reference log-rank gap is $.9608$ against an independent retraining null of $.0931$, a calibrated excess of $.8677$; NormOracle reduces it to $.0608$, a 93.0% reduction. On CoDEx-M the excess falls from $.6763$ to $.0808$, a 88.1% reduction. Matching the expected number of negative presentations on the same cohort and the same scale reduces the excess by 2.0% and 0.7%. Both norm reductions are consistent with the 92.19% measured on the primary 36-setting grid, so the scale mechanism is not a property of that grid's requests or training budget, and embedding-scale mismatch rather than negative exposure dominates in both cohorts.

The full-split retraining null is substantially smaller than the primary grid's ($.0931$ and $.1356$ against $.2248$), which is why the corresponding raw-gap reductions (83.99% and 73.35%) understate the calibrated ones. We report calibrated excess throughout.

The residual is small but systematic: positive in 28 of 36 settings (sign test $p < .001$), with equal-setting mean $.0200$, median $.0244$, and per-setting range $[-.0368, .0594]$. It is unevenly distributed, with low-degree requests retaining the largest residual and showing the weakest reduction, so a component not linked to embedding scale persists where deletion is least severe.

Table 13: NORMORACLE on the primary grid. Reductions use Eq. (8) within each group; constructions contribute nine settings each and datasets twelve.

Group	R+L excess	NORMORACLE excess	Reduction
Low	.0977	.0352	64.0%
Mid	.1572	.0174	89.0%
High	.3782	.0182	95.2%
Exact-cardinality high	.3914	.0093	97.6%
FB15K-237	.3079	.01740	94.3%
CoDEx-M	.2145	.03118	85.5%
NELL-995	.2459	.01139	95.4%
All 36 settings	.25611	.01999	92.19%

Table 14: Full-split cohort, calibrated against an independent same-budget retrain. Reductions use Eq. (8) on the excess column.

Graph	Model	Gap	Null	Excess	Reduction
FB15K-237	R+L	.9608	.0931	.8677	–
	+NegExposure	.9434	.0931	.8503	2.0%
	+NORMORACLE	.1539	.0931	.0608	93.0%
CoDEx-M	R+L	.8119	.1356	.6763	–
	+NegExposure	.8073	.1356	.6717	0.7%
	+NORMORACLE	.2164	.1356	.0808	88.1%

NORMORACLE in placement space. Applying the same diagnostic to the background-context probe reduces the matched contrast from .6182 to $+.0747$, an 87.9% reduction in absolute magnitude, positive in all 36 settings with a 95% interval of $[.0687, .0807]$ (Table 15). Norm rescaling therefore does not by itself reverse the sign of the contrast; it removes most of it and leaves a residual promotion.

Two features of that residual matter. It is nearly uniform across constructions ($.0649$–$.0804$) and across datasets ($.0687$–$.0781$), while the uncorrected contrast ranges from $.4742$ to $.7637$, so the component that norm scale does not account for has roughly constant magnitude independent of request severity. And it does not mirror the rank-space residual: in placement space NORMORACLE is uniformly effective (83.6%–89.3% across constructions), whereas in rank space its effectiveness runs from 64.0% on low-degree requests to 97.6% on exact-cardinality ones (Table 13). The non-scale component of the low-degree discrepancy is therefore specific to forgotten-query rank fidelity rather than a general property of where restarted entities are placed.

Table 15: Signed placement contrast before and after NORMORACLE. The residual is positive in 36/36 settings, with overall 95% CI $[.0687, .0807]$.

Group	R+L	NORMORACLE	Reduction
Low	.4742	.0778	83.6%
Mid	.5658	.0649	88.5%
High	.7236	.0804	88.9%
Exact-cardinality high	.7092	.0756	89.3%
FB15K-237	.7637	.0781	89.8%
CoDEx-M	.4535	.0772	83.0%
NELL-995	.6372	.0687	89.2%
All 36 settings	.6182	.0747	87.9%

These controls test two specific explanations and do not exhaust the possible effects of optimization.

D.4 RFPC FITTING AND EVALUATION

The correction in Section 6 uses 200 nonincident retained-training contexts per restarted entity and 20 unaffected controls with positive retained support, approximately matched on original training degree. Matching uses original rather than retained degree because a restarted entity has zero retained degree whereas supported controls must have positive retained degree.

Fitting uses only the updated candidate, retained training graph, deletion metadata, and original degrees. Retrained checkpoints and forgotten, validation, test, and placement-evaluation queries are excluded. REFA and REFB are accessed only after the offsets are fixed. The median and mean rules, sample counts, and evaluation settings are fixed before evaluation, with deterministic per-setting calibration seeds.

Prespecified acceptance criterion. Before any comparison against a reference was run, we froze a substantive-effect criterion of at least a 25% reduction in both the calibrated log-rank excess and the absolute placement contrast, together with a test-MRR change above $-.005$ and a forgotten-MRR gap below $.010$. Given the pre-correction values $.256109$ and $.6182$, these correspond to residual thresholds of $.1921$ and $.4637$. RFPC is the only intervention in the paper carrying a prespecified criterion; RNR and RNR-DM (Appendix D.5) are derived from the NORMORACLE result and are reported post hoc.

Only restarted answer candidates receive offsets. The correction changes neither model parameters nor the score of a non-restarted candidate. However, any entity's rank can change when a restarted competitor moves, so utility is evaluated by reranking the full vocabulary. Both references remain fixed: a reduction in calibrated excess is therefore also a reduction in candidate–REFA log-rank disagreement under the REFA-anchored estimand of Eq. (7).

Table 16: RFPC outcomes on the 36-setting grid. Placement contrasts are signed; placement reduction uses their absolute magnitude. Test-MRR changes are relative to uncorrected R+L. Prespecified residual thresholds are $.1921$ and $.4637$.

Metric	Before correction	Median	Mean
Calibrated log-rank excess	.256109	.13705	.14311
Log-rank excess reduction	–	46.49%	44.12%
Placement contrast	+.6182	-.05785	-.05782
Forgotten-MRR gap	.00812	.008035	.008064
Test-MRR change	–	-2.23×10^{-5}	-1.91×10^{-5}

The small negative placement contrasts (Table 16) indicate an overshoot rather than residual positive promotion; NORMORACLE, which starts from the correct norms, instead leaves a positive residual (Table 15).

Table 17: Residual calibrated log-rank excess after RFPC by request construction, nine settings each.

Group	Median rule	Mean rule
Low-degree stars	.0892	.0882
Mid-degree stars	.0818	.0798
High-degree stars	.1725	.1823
Exact-cardinality high	.2047	.2221
All 36 settings	.13705	.14311

Table 18: RFPC by dataset, median rule, 12 settings each. Cell counts are settings whose reduction exceeds the stated bar.

Dataset	R+L excess	Residual	Reduction	$\geq 25\%$	$\geq 50\%$
FB15K-237	.307924	.107077	65.23%	10/12	10/12
CoDEx-M	.214468	.162050	24.44%	5/12	0/12
NELL-995	.245934	.142029	42.25%	10/12	2/12
All	.256109	.137052	46.49%	25/36	–

The residual varies across request constructions: low-degree stars start with a smaller excess and retain most of it, whereas high-degree constructions show larger reductions but also larger residuals. The pooled reduction therefore does not imply a uniform effect across request types. Under the median rule the 25% criterion is exceeded in 2/9 low-degree cells, 6/9 mid-degree, 8/9 high-degree and 9/9 exact-cardinality cells, and in 25/36 cells overall (21/36 for the mean rule). The criterion is stated on the equal-setting mean over all 36 settings, which RFPC meets (46.49%). By graph it is not uniform: 65.23% on FB15K-237 and 42.25% on NELL-995, but 24.44% on CoDEx-M, marginally below the 25% bar (Table 18). Only FB15K-237 exceeds 50% in a majority of cells.

D.5 REFERENCE-FREE NORM RESCALING

RNR computes a single scalar per setting: the median embedding norm over entities with positive retained degree in the updated candidate model, as in Eq. (5). Every row of $Z_{\mathcal{U}}$ is rescaled to that norm, preserving direction; no other row is modified and no parameter is fitted. The computation uses the candidate checkpoint and the retained graph only, so it satisfies the same information budget as RFPC and strictly less than NORMORACLE. RNR-DM differs only in replacing the global median with a per-entity median over the degree-matched control sets C_v already constructed for RFPC.

Both are derived from the NORMORACLE result and are therefore post hoc; neither carries a prespecified acceptance criterion. No norm-target variants other than these two were evaluated.

Table 19: Residual calibrated log-rank excess by dataset, with each candidate-only method's recovery of the NORMORACLE gain, $(\bar{E}^{R+L} - \bar{E})/(\bar{E}^{R+L} - \bar{E}^{\text{NORMORACLE}})$. Each dataset contributes 12 settings.

	Residual excess				NORMORACLE gain recovered	
Dataset	R+L	NORMORACLE	RNR-DM	RNR	RNR-DM	RNR
FB15K-237	.3079	.01740	.04955	.04013	88.9%	92.2%
CoDEx-M	.2145	.03118	.13418	.10832	43.8%	57.9%
NELL-995	.2459	.01139	.06615	.03880	76.6%	88.3%
All	.25611	.01999	.08330	.06242	73.2%	82.0%

CoDEx-M is the weakest case for reference-free estimation (Table 19), and the decomposition shows this is a limit of estimation rather than of the mechanism: NORMORACLE still removes 85.5% of the excess there, against 94.3% on FB15K-237 and 95.4% on NELL-995 (Table 13), so the discrepancy on CoDEx-M is scale-linked to much the same degree as on the other two graphs. Neither candidate-only target locates it as accurately, and degree matching does not recover the shortfall.

Table 20: Threshold and aggregate outcomes on the primary grid. Top-k entries are excess membership disagreement; the retraining null for the MRR excess is .00392.

Method	Top-10 excess	Top-100 excess	MRR excess	Forg. MRR gap
R+L	.001125	.005750	.00420	.00812
RFPC	.00119	.00426	.00411	.00804
RNR-DM	.00122	.00467	.00404	.00797
RNR	.00119	.00432	.00405	.00798

Threshold and aggregate outcomes on the primary grid are in Table 20. RFPC's intervals are $[.00001, .00240]$ for Top-10 (positive in 22/36), $[.00288, .00564]$ for Top-100 (23/36) and $[.00317, .00473]$ for the MRR excess (25/36); mean test-MRR change is -8.1×10^{-5} for RNR-DM and -3.1×10^{-5} for RNR. Neither threshold statistic separates RFPC from RNR: their excess Top-10 disagreements are equal to five decimal places and their Top-100 disagreements differ by $.00006$, while their calibrated log-rank excesses differ by a factor of 2.2. Short-list membership is therefore as insensitive to the difference between the two corrections as aggregate MRR is.

Why a global target outperforms degree-matched controls. In REFA the entities of $Z_{\mathcal{U}}$ have no positive retained support, so their embeddings are shaped entirely by negative gradients. Under uniform corruption the expected number of times an entity is drawn as a negative is independent of its original degree, so original degree should carry no information about the norm REFA assigns it. The data agree: within this cohort, correlations between pre-deletion degree and the corresponding REFA norm over $Z_{\mathcal{U}}$ are negligible, with the within-dataset pooled values $r = -.033$ and $\rho = -.001$ and similarly small effects on each graph (Table 21). We state this as a property of this cohort rather than a general claim that degree is irrelevant.

Degree matching therefore adds variance without signal, and on high-degree requests it adds bias: the matched controls are *supported* high-degree entities whose norms are large. This predicts the observed pattern, in which RNR-DM overshoots on high-degree and exact-cardinality constructions and is preferable to RNR only on low-degree requests, where the matched controls are themselves low-degree.

Why the reference-free target overshoots. The global target is systematically larger than the norm retraining assigns the same restarted entity. The ratio $m_{\mathcal{R}}/\|e_v^{\text{REFA}}\|$ has mean 1.468, median 1.455

Table 21: Association between pre-deletion degree and the REFA norm attained by restarted zero-support entities.

Graph	Pearson r	Spearman ρ
FB15K-237	.0247	.0443
CoDEx-M	.0629	.0978
NELL-995	$-.0616$	$-.0319$
Pooled within dataset	$-.0329$	$-.0009$

and IQR $[1.343, 1.565]$, and exceeds one for 99.98% of restarted entities; median ratios are 1.501 on FB15K-237, 1.331 on CoDEx-M and 1.479 on NELL-995. A target roughly 46% too large moves the signed placement contrast .2143 past the point at which NORMORACLE lands ($+.0747 \to -.1396$), so part of RNR's residual log-rank excess is over- rather than under-correction — a distinction Eq. (7) cannot make on its own, since it is built from absolute differences.

The ratio is stable enough that dividing $m_{\mathcal{R}}$ by it would close much of the remaining gap. We do not do so: it is measured against REFA norms and is therefore reference-derived, so a correction calibrated on it would no longer satisfy the information budget RNR is defined by. A reference-free estimator of the post-retraining norm of an unsupported entity would close the gap legitimately; we do not have one, and record it as the natural next step rather than fitting a constant here.

Note that CoDEx-M has the *smallest* median ratio of the three graphs yet recovers the least of the NORMORACLE gain, so its shortfall is not a uniform over-scaling but dispersion in the per-entity ratio.

Relation to the negative-exposure control. This also reconciles *NegExposure20eq* (Appendix D.3), which matches the expected *count* of negative presentations but starts from the R+L state under the repair optimizer and reduces the gap by only 1.8% and 0.6%. Replaying the expected number of negatives does not reproduce the norm; setting the norm directly closes most of the gap. The quantity that matters is the norm itself, not the exposure that would have produced it.

D.6 INTERACTION BETWEEN THE NORM ORACLE AND RFPC

Refitting RFPC on the norm-corrected candidate, under the same frozen reference-free protocol, does not compose with the norm correction (Table 22). NORMORACLE alone leaves a residual of .01999; adding RFPC raises it to .11122 under the median rule and .14118 under the mean rule, improving on NORMORACLE in only 2/36 and 3/36 settings respectively.

Table 22: Residual calibrated log-rank excess under NORMORACLE and its combination with RFPC. Δ is the change relative to NORMORACLE alone; positive values indicate a worse residual.

Construction	NORMORACLE	+RFPC med	Δ	+RFPC mean	Δ
Low	.0352	.0478	$+.0126$	.0506	$+.0154$
Mid	.0174	.0360	$+.0186$	.0467	$+.0293$
High	.0182	.1677	$+.1495$	.2217	$+.2035$
Exact-cardinality high	.0093	.1934	$+.1841$	.2457	$+.2364$
All 36 settings	.01999	.11122	$+.0912$	.14118	$+.1212$

The per-construction pattern distinguishes over-correction from substitution. For high-degree and exact-cardinality requests the combined residual (.1677, .1934) is essentially that of RFPC applied alone (.1725, .2047 from Table 17): the additive offset returns the model to the same state whether or not the scale mismatch had already been removed. For low- and mid-degree requests the two interventions partially compose. In the regime where RFPC is most effective it is therefore substituting for the scale correction rather than adding to it, which is why we read it as a reference-free mitigation of the same scale-linked error rather than as evidence for a second mechanism.

E TRAINING-REGIME SENSITIVITY

The stronger DistMult recipe uses reciprocal 1-N softmax, dimension 256, Adagrad, batch size 256, validation-MRR early stopping with patience 6, mean ties, and both completion directions. Each graph uses one fixed request evaluated across seeds 17, 29, and 43.

For CoDEx-M the request selects five complete stars and removes 21,372 triples (11.5%; hash `906071b72f6995fa`). The FB15K-237 request removes 28,315 triples (10.4%; hash `98cf65a6a6d1e3cf`), and NELL-995 likewise uses a fixed request across the three seeds. Base/REFA/REFB test MRR on FB15K-237 is .248/.193/.200; on NELL-995 the two references reach .258 and .260 on average.

Table 23: Stronger-training sensitivity with one fixed request per graph. Entries are candidate–REFA log-rank disagreement minus the independent REFB–REFA gap.

Seed	CoDEx-M	FB15K-237	NELL-995
17	.415024	.283350	-.036906
29	.412824	.160060	-.350426
43	.452295	-.020430	-.003953
Mean	.4267	.141	-.1304
Positive	3/3	2/3	0/3

CoDEx-M remains positive in every seed; FB15K-237 in two of three; NELL-995 is near zero in seed 43 and more negative in seed 29 (Table 23). Negative excess means that the candidate is closer to REFA than REFB is under this metric; it does not establish retraining equivalence.

Training seed alone can reverse the sign. On FB15K-237 the three cells share a single realized deletion request (hash `98cf65a6a6d1e3cf`) and differ only in training seed, yet the calibrated excess is $+.2834$, $+.1601$ and $-.0204$. With the request held exactly fixed, training randomness alone is sufficient to reverse the sign of the calibrated excess under this recipe. The references' stopping epochs vary correspondingly, from 10 to 130 across the six reference runs. The Top-1000 excess is equally seed-sensitive ($+.171$, $+.145$, $-.077$).

The reversal comes from an expanded null. On NELL-995 the mean candidate gap is .488 while the independent retrain gap is .618, so the sign change is driven by retraining variability rather than by unusually close candidate agreement. This is not explained by divergent early stopping: the mean absolute REFA–REFB stopping-epoch difference is 16.7 on NELL-995 against 26.7 on both other graphs. Across regimes the null moves far more than the candidate does: it is .2248 in the primary 20-epoch grid and .1896 in the three-reference check (Table 10), but .412 under this recipe and .618 on NELL-995 alone, while the candidate gap moves only from .481 to .558. We note that NELL-995 is the only graph in the paper with a custom split and with a 1,000-triple validation set (Table 6), which makes validation-MRR early stopping noisy in model selection even when stopping epochs are close.

The NELL-995 reversal is not confined to the forgotten set. On forgotten queries the candidate–REFA log-rank gap is .4877 against an independent REFB–REFA gap of .6181, an excess of $-.1304$. On held-out test queries the corresponding gaps are .4031 and .4533, an excess of $-.0502$. The negative excess is therefore not specific to the deleted queries: it reflects substantial run-to-run retraining variability on this graph under this recipe. REFA and REFB stop on average 16.7 epochs apart, which is consistent with that variability but is reported descriptively rather than as a causal explanation.

Threshold effects do not survive this recipe. Excess Top-10 disagreement across the nine cells is $-.028$ percentage points (95% CI $[-.072, .011]$, positive in 2/9) and excess Top-100 is $-.117$ points (95% CI $[-.261, .022]$, positive in 6/9); both intervals contain zero, so neither is distinguishable from no effect. By dataset, excess Top-10 is .000 on CoDEx-M, $-.100$ on FB15K-237 and $+.017$ on NELL-995; excess Top-100 is $+.167$, $-.583$ and $+.067$ respectively. The dissociation is sharpest on CoDEx-M, which has the largest and most consistent calibrated log-rank excess (.427, positive in 3/3) and exactly zero excess Top-10 disagreement. Aggregate metrics are likewise insensitive: the candidate MRR gap is .00067 against a .00012 null.

The repair-budget and destructive controls in Appendix B.1 use their own reference contract and are not pooled with these stronger-training runs.

F EVALUATION PROTOCOL

The protocol summarized in the main paper combines the preceding experimental contracts:

1. Fix the deletion request, candidate vocabulary, filtering, tie rule, and query cohort (Appendix A).
2. Train at least two independent retained-data references under the same declared recipe and budget, and where feasible a third, to verify that the null does not depend on a single pair (Appendix C). Measure the null under the audited recipe rather than importing it: it varies by graph and by optimization regime (Appendix E).
3. Report test and forgotten-set MRR alongside calibrated log-rank and Top-k disagreement, including the cutoffs at which the excess vanishes, with uncertainty conditional on the evaluated settings (Appendix C).
4. Investigate systematic disagreement by separating candidate and context roles, using matched placement probes, and testing concrete model properties such as embedding scale (Appendix D). Report at least one signed statistic alongside absolute-difference measures, so that over-correction can be distinguished from under-correction.
5. Fit corrective interventions independently of retraining references and evaluation queries, and label any reference-consuming intervention as diagnostic rather than deployable (Appendices D.4 and D.5).

Neither aggregate agreement nor non-positive calibrated excess alone establishes equivalence or privacy.