

When String Matching Stands In for a Knowledge Base: A Role-Aware Audit of WikiBigEdit

Edison Yang¹, Neel Marripalapu², Akshaj Agadi³

¹Lambert High School ²University of Texas at Dallas ³Texas A&M University

Correspondence: edisonly2200@gmail.com

Abstract

Lifelong knowledge-editing benchmarks turn changes in a structured knowledge base into natural-language edits and evaluation questions. Their value depends on that conversion preserving what each fact says and which entity it concerns. We audit WikiBigEdit, a benchmark of more than 500,000 Wikidata updates, and trace its released construction pipeline to a string scan over a text rendering of Wikidata rather than a parse of its data model. From a uniform random sample of 1,500 rows, we audit the 1,458 resolvable rows against their source revisions and find three consequences: qualifier and reference values are promoted to main facts; 23.1% of resolvable rows attach correct answer text to the wrong entity identifier; and locality probes are chosen by surface-string similarity rather than semantic relatedness. Overall, 22.8% of resolvable rows carry unusable text supervision. In a matched reconstruction that holds the relation and question template fixed, similarity-selected probes show 7–12 percentage points more edit-target leakage, that is, the edited answer surfaces in a neighboring question. This effect persists under parameter editing with ROME, while the aggregate locality score remains unchanged. WikiBigEdit’s official metric does not fully register the leakage. We validate the audit with an independent parser and release the audit code and a provenance-bearing repair set. Our results do not show that published scores are incorrect; they identify structural errors those scores were not designed to detect.

1 Introduction

Lifelong knowledge editing asks whether a model can absorb a stream of factual updates while preserving unrelated knowledge (De Cao et al., 2021; Mitchell et al., 2022a). WikiBigEdit (Thede et al., 2025) addresses a real gap left by small, synthetic, or pre-cutoff editing benchmarks, operationalizing this at scale: more than 500k question–answer ed-

its mined from consecutive Wikidata dumps, each accompanied by locality, rephrase, and multi-hop probes. Its usefulness does not require every row to be a one-to-one historical replacement, but it does require the supervision to mean what its generated question asserts.

Concretely, each row is a Wikidata triple (s, p, o) rendered as a question–answer pair, for instance (George Harinck, *employer*, Free University of Amsterdam), paired with a *locality probe*: a separate question whose answer should remain unchanged after the edit. For this row the released probe asks “For whom did George Zarnecki work?” (gold: Courtauld Institute of Art), a different person selected because that label is the closest string to the edited subject; after the edit, the Qwen-7B in-context editor answers “Free University of Amsterdam”. We call this *edit-target leakage*. On the Wikidata side, a fact is a *statement* whose *main snak* is the property–value pair asserted as the statement’s primary fact. A *qualifier* is information that limits or contextualizes that fact, such as its start date; a *reference* records its source; and a *rank* (preferred, normal, or deprecated) marks which of several values is current. The audit asks whether each released (s, p, o) is a main-snak fact about s at the time the row claims, and whether the probe concerns a semantically related entity.

The supervision frequently fails this requirement, for an architectural reason: the released pipeline does not parse Wikidata’s statement structure. It runs a Wikipedia text segmenter over the MediaWiki page dump and applies string matching to the result. Wikidata’s data model, comprising main snaks, qualifiers, references, and rank (MediaWiki, 2024), is invisible to that process, as is entity identity, which the pipeline discards after extraction and reconstructs from surface labels.

Contributions. (1) A mechanistic audit of a published benchmark’s construction, verified against

its released code and reproduced with a deterministic extraction-trace replay that needs no unreleased intermediates. (2) Prevalence estimates for the resulting defect families on a documented probability sample, with relation-clustered intervals, a sensitivity analysis over three endpoint semantics, a timestamp-robustness check, and an independent-parser validation. (3) A downstream experiment on the benchmark’s own data, whose matched reconstruction separates the probe-selection contribution to locality from relation-level generalization, and whose parameter-editing arm (FT-L, ROME) excludes a retrieval artifact. (4) An audited, provenance-bearing repair set with corrected object identifiers.

Although the prevalence estimates are specific to WikiBigEdit, the audit procedure is portable. For any benchmark generated from versioned structured data, it checks whether the construction pipeline (1) distinguishes main values from qualifiers, references, and ranks; (2) preserves entity identifiers rather than reconstructing them from labels; (3) traces each example to the source state at the claimed time; and (4) selects evaluation probes with semantic controls rather than surface similarity alone. Appendix D summarizes these checks, the evidence each produces, and the corresponding repair.

2 Benchmark construction and structural loss

The released `download_wikidata.sh` does not fetch a Wikidata JSON dump; it fetches the MediaWiki page-text dump and pushes it through `gensim.scripts.segment_wiki` (Řehůřek and Sojka, 2010), a Wikipedia-article segmenter (Figure 1). Extraction runs over that text.

Role-blind scan. The extractor `extract_triplets.py` locates each occurrence of the literal `wikibase-entity` (the prefix of the data value type `wikibase-entityid`, which is emitted for every entity-valued snak) and reads a property identifier and an entity identifier from the preceding 200 characters. Main, qualifier, and reference snaks are structurally identical at that level, and rank is never read. A qualifier value therefore becomes a fact about the page subject (Figure 2). The authors’ own code special-cases a snak-boundary artifact (`'"}, "type":'`), which indicates that window overlap was a recognized hazard.

Identity loss. `extract_triplets.py` writes both an identifier file and a label file per interval, but `process_triplets.py` reads only the label file. All downstream stages (ambiguity filtering, locality-probe selection, multi-hop chaining, and QA assembly) operate on ambiguous surface strings, with `fuzzywuzzy` fallbacks. The object identifiers in the public release were re-attached afterwards, and because `entity_dict` maps `identifier`→`label`, inverting it to recover `label`→`identifier` collapses homonyms under a last-write-wins rule.

Provenance recovery. The release is interval-partitioned and carries `subject_id`, `relation_id`, and `object_id`, so a stable audit key and the interval-end timestamp t_1 are constructible from public data. The MediaWiki API returns the exact stored page text at any revision, so we replay the scanner deterministically; serialized marker occurrences align one-to-one with structural snaks, letting us attribute each released pairing to a role.

3 Audit method

3.1 Sampling frame and inclusion flow

The population is the full public release: 502,382 rows across eight interval files. We draw a uniform simple random sample of rows without replacement (seed fixed in advance), so every row has equal inclusion probability and no sampling weights are required. A released triple (s, p, o) that recurs across intervals is a distinct row with a distinct temporal context and is treated as such. Of 1,500 rows drawn, 42 are unresolvable (37 carry no subject identifier, 4 have no Wikidata revision at or before t_1 , and 1 is a missing page); these are reported, not silently dropped, and the 37 rows with no subject identifier are themselves release defects. The remaining 1,458 rows are classified. We report prevalences conditional on resolvability, and separately give the unconditional decomposition over all 1,500 rows in §4.

Uncertainty. The estimand is the prevalence in the fixed 502,382-row release, and we sampled rows by SRS, so the primary uncertainty is the design-based interval for a finite-population proportion. With $n/N=0.29\%$ the finite-population correction is 0.9985, i.e. negligible, so we report the row-level Wilson interval as primary. Outcomes do correlate within relations, so we additionally report a relation-clustered bootstrap (4000 replicates)

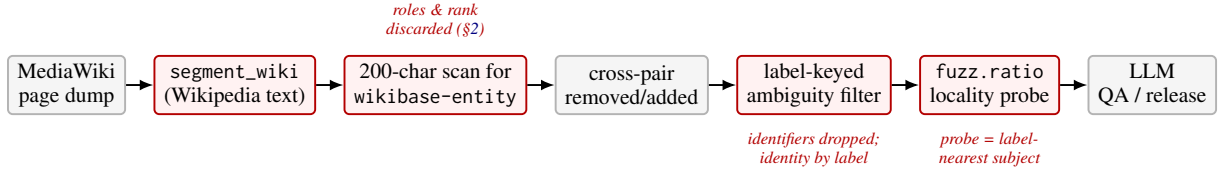


Figure 1: The released pipeline does not parse Wikidata’s data model. Red stages are those at which structure is discarded: a Wikipedia text segmenter and a 200-character string scan remove statement role and rank; after extraction only surface labels remain, so ambiguity filtering, probe selection, and QA all operate on strings. Object identifiers in the release are re-attached afterwards.

```

{"mainsnak":{"property":"P39",
  "datavalue":{"value":{"id":"Q30185"}}},
  "rank":"normal",
  "qualifiers":{"P642":[{"property":"P642",
    "datavalue":{"value":{"id":"Q515"}}}]}}

```

Structure: main snak P39→Q30185 (*position held: mayor*), scoped by qualifier P642 (*of: a city*). A valid main fact.

200-char scan: finds wikibase-entity, reads the nearest property and id in its window, emits (P39, Q30185) and, at the next marker, (P642, Q515), in which the qualifier is promoted to a main fact, “*the city is a mayor*”.

Figure 2: Worked role-erasure trace. The scanner treats every entity-valued snak identically, so a qualifier value (P642) is extracted as though it were a main fact about the subject. Rank and role, present in the JSON, are invisible to it.

as a robustness analysis for relation heterogeneity; it targets a different estimand, generalization over a population of relations rather than the fixed release, and is wider (relation design effect ≈ 10.7). Appendix A gives its construction and why subject and interval clustering are not used.

3.2 Endpoint semantics and classification

For a row (s, p, o) from interval file `wiki_big_edit_t0-t1`, we fetch the revision of s current at t_1 and decompose its claims into role- and rank-tagged entity-valued snaks. The primary criterion is $\mathcal{A}_{t_1}^{\rho_{\text{WBE}}}(s, p) = \{o\}$, i.e. the non-deprecated main-snak set for (s, p) is exactly the singleton $\{o\}$. Because this is strict for naturally multi-valued properties, we report two sensitivity variants: ρ_{all} (membership among non-deprecated main snaks) and ρ_{truthy} (membership among best-rank main snaks). We distinguish a row whose subject carries the property but not the value from one whose subject lacks the property in every role, the latter being the signature of a wrong-entity attachment. For the former, we resolve whether the entity’s actual value shares the released label (a misattached identifier) or does not (a genuinely

Class	k	%	Wilson	rel-clu
endpoint-valid (ρ_{WBE})	719	49.3	46.8–51.9	42.2–56.2
object-id misattached	337	23.1	21.0–25.3	16.0–31.2
property absent (ident.)	149	10.2	8.8–11.9	7.8–13.0
role-erased (qual.)	82	5.6	4.6–6.9	3.0–9.5
main-suppl., non-unique	69	4.7	3.8–5.9	2.2–7.5
object absent, unresolv.	44	3.0	2.3–4.0	0.6–7.4
object genuinely wrong	43	2.9	2.2–3.9	1.0–4.9
role-erased (ref.)	15	1.0	0.6–1.7	0.1–2.5

Table 1: Class prevalence at the interval-end snapshot, $n=1458$, with counts k , row-level Wilson, and relation-clustered bootstrap 95% intervals.

incorrect value).

3.3 Validation

We froze the classifier after a stratified adjudication (24/24) and 11 unit tests, then checked it against an independent re-parser. For 164 rows stratified across all classes, we re-fetched each row’s pinned revision and re-derived the verdict with a separate minimal parser that shares no code with the audited pipeline. Taking “invalid under ρ_{WBE} ” as the positive class, the confusion matrix is TP=123, FP=0, FN=0, TN=41: positive predictive value 100% [97.0, 100], false-omission rate 0% [0, 8.6] (Clopper–Pearson), and 29/29 agreement on the role-erasure cases that exercise the qualifier/reference traversal. This establishes that the extraction and classification faithfully implement the ρ_{WBE} criterion over the source revision. It does not adjudicate the choice of ρ_{WBE} itself, since the re-parser shares that definition and the source revisions; probing that choice requires the alternative endpoint criteria ρ_{all} and ρ_{truthy} , whose sensitivity we report separately.

4 Population findings

Table 1 reports class prevalence on the $n=1458$ resolvable rows. The Wilson interval is the primary (design-based) uncertainty for the fixed release; the

	%	rel-clu 95%
<i>Endpoint semantics (valid)</i>		
ρ_{WBE} (unique main = {o})	49.3	[42.2, 56.2]
ρ_{all} (member of main set)	54.0	[45.5, 62.1]
ρ_{truthy} (member, best rank)	54.0	[45.5, 62.1]
<i>Four-way decomposition</i>		
A. fully valid (unique main)	49.3	[42.2, 56.2]
B. structurally valid, non-unique	4.7	[2.2, 7.5]
C. valid text, wrong identifier	23.1	[16.0, 31.2]
D. unusable text supervision	22.8	[17.1, 30.4]

Table 2: Endpoint-semantics sensitivity and a four-way split. A+B is validity under ρ_{all} (54.0%); A+B+C is defensible text supervision (77.2%); D is the genuinely unusable share.

relation-clustered interval is the wider robustness bound for generalization across relations. The finding is also stable across intervals: the per-interval invalid rate ranges from 38.8% to 52.1%, and leaving any single interval out moves the pooled invalid rate only within [43.8, 47.0]% around 46.0%, so no interval dominates.

Endpoint sensitivity and decomposition. Table 2 reports validity under all three endpoint semantics and separates the release into four categories, which are best interpreted jointly rather than through the single strict number. The most consequential quantity is the 22.8% of rows whose text supervision is unusable, rather than the 49.3% strict-validity rate, because a misattached identifier (23.1%) leaves the question–answer pair correct even though the structured object is wrong, and a non-unique main answer (4.7%) remains valid under ρ_{all} . Rank policy is immaterial here, as ρ_{all} and ρ_{truthy} coincide.

Table 2 is conditional on resolvability. Computed unconditionally over all 1,500 sampled rows, and adding an “unresolvable release record” fifth class (2.8%, dominated by the 37 rows with no subject identifier), the shares become A 47.9%, B 4.6%, C 22.5%, D 22.2%, and E 2.8%. Because the no-subject-identifier rows are themselves defects, the genuinely unusable share rises to 24.7% [22.6, 26.9]. The prevalence estimates therefore generalize directly to the resolvable subpopulation and slightly understate malformation for the release as a whole.

Robustness to snapshot timing. Sweeping the assumed snapshot date from −14 to +56 days (Figure 3), ρ_{WBE} validity peaks at 50.9% and the object-present rate plateaus at 54.2% from +7 on-

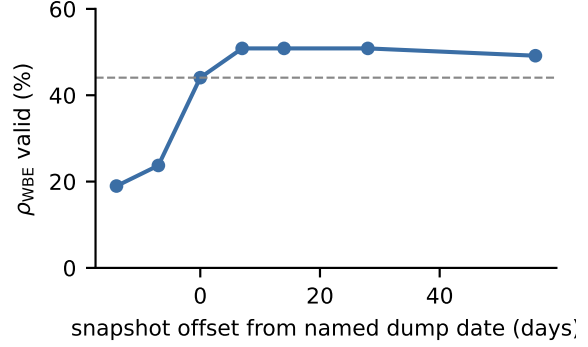


Figure 3: ρ_{WBE} validity versus snapshot offset. The curve plateaus, so the defect is not a timestamp artifact.

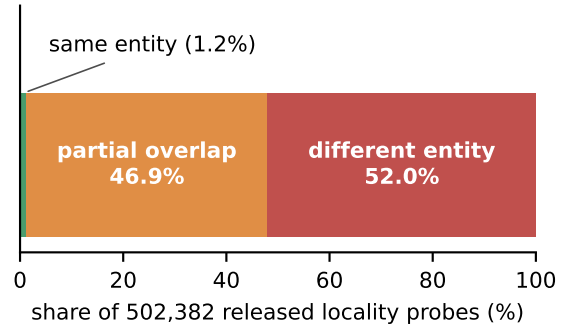


Figure 4: Entity provenance of the 502,382 released locality probes, by content-token overlap between the probe question and the edited subject label. *Same entity*: the subject string appears verbatim, or all its content tokens appear, in the probe. *Partial overlap*: the probe shares at least one but not all of the subject’s content tokens. *Different entity*: no shared content token.

ward, while the role-erasure signal is invariant. At least approximately 46% of rows are never uniquely main-supported anywhere in that window. The benchmark’s own tags are internally consistent at t_0 (update rows: property present in 13/13), confirming the alignment.

Wrong entity identifiers. Within the largest non-valid block, window misassociation is excluded (the object is absent from the entity entirely in 98.4% of cases), and 79.5% carry the released label on a different, homonymous identifier: Finland=Q2580855 rather than the canonical Q33, and Canada=Q14624136 rather than Q16. The text supervision is correct, whereas the structured object identifier is not.

5 Provenance of the locality probes

`create_locality_probes` is intended to select a same-subject unchanged fact but falls back to the

fuzz.ratio argmax over subject labels; an earlier ambiguity filter renders the intended branch nearly unreachable. A locality probe should target an entity other than the edited one, so a different-entity probe is not itself a defect; our concern is narrower, that the selection is driven by surface-string similarity rather than by any semantic relation to the edit. Over all 502,382 rows (Figure 4), the probe plausibly concerns the edited entity in only 1.15% of rows; 52.0% share no content token with the edited subject and are selected purely on character-level string overlap, and the remaining 46.9% share some but not all of its tokens. Against a null that pairs each edit with a random probe, the observed edit-probe string similarity is 66.9 versus 31.4, and the fraction above similarity 70 is 48.2% versus 0.4%, an enrichment of approximately $120\times$. Representative instances include Limni→Lin, Lynnvile→Lunéville, and Coldwater→Clearwater. We do not claim such probes are useless as difficulty items; we claim that string-similarity selection is not a semantically controlled way to measure locality, and §6 shows it induces edit-target leakage that the counterfactual design rules out as legitimate relatedness and that the official metric does not register.

6 Downstream consequence

This section shows that the structural defect has measurable consequences for how the benchmark is used; the prevalence audit of §4 stands independently of it. It asks two separate questions. First, does an edit cause the model to repeat the new target when answering a supposedly neighboring question? We measure this as *edit-target leakage*. Second, does the edit stop the model from preserving the probe’s original answer? We measure this as *locality* (the answer is unchanged by the edit), and report WikiBigEdit’s own metric (the model still produces the gold answer loc_ans) alongside it. Three experiments address these questions. The *as-deployed* contrast (§6.2) evaluates WikiBigEdit’s released probes but confounds shared relation with subject similarity; the *matched reconstruction* (§6.3) isolates similarity-associated probe selection by holding relation and question template fixed; and the *parameter-editing* experiment (§6.4) tests whether the effect persists without any retrieval step.

Editor, backbone	<i>N</i>	Succ.	Leak	Loc	WBE
ctx/retr., Qwen-7B	400	n/a	+33.8***	−27.5***	−6.0**
ctx/retr., Llama-8B	400	n/a	+23.8***	−16.5***	−1.5
FT-L, Qwen-1.5B	300	53	+22.3***	−13.7**	−4.7
ROME, Qwen-1.5B	300	83	+20.3***	+2.3	+0.7

Table 3: As-deployed difference-in-differences in percentage points (own released probe versus random probe, edit versus null). *N* is the number of edits; Succ. is edit success (%); *Leak* is edit-target leakage; *Loc* is locality (answer unchanged); *WBE* is WikiBigEdit’s own metric (the model still produces loc_ans). Significance is from relation-clustered paired sign-flip permutation tests (20,000 sign assignments, so the smallest reportable *p* is 5×10^{-5}): **p*<0.05, ***p*<0.01, ****p*<0.001; unmarked is not significant. This contrast confounds shared relation with subject similarity; see Table 4.

6.1 Setup and definitions

We evaluate whether the probe design alters locality behavior using the benchmark’s own loc/loc_ans probes and edits. Answers are compared after normalization (lowercasing, punctuation stripping, article removal), with a token-boundary containment test. For an edit that asserts object $o^\dagger$ and a probe q with baseline answer a_0 , the two per-probe binary outcomes are *locality* $\mathbb{1}[a(q)=a_0]$ and *leakage* $\mathbb{1}[o^\dagger \in a(q)]$, the edit’s target appearing in the probe answer; the WikiBigEdit metric is $\mathbb{1}[\text{gold} \in a(q)]$. Because each item contributes both an own probe and a matched comparison probe under an edit condition and a null (unrelated-edit) condition, we report the difference-in-differences

$$\text{DiD} = \frac{1}{N} \sum_i [(y_{i,\text{edit}}^{\text{own}} - y_{i,\text{edit}}^{\text{cmp}}) - (y_{i,\text{null}}^{\text{own}} - y_{i,\text{null}}^{\text{cmp}})],$$

which removes any edit-independent difference between the probes. Because the per-item DiD terms are discrete and heavily tied, significance uses a paired sign-flip (Rademacher) permutation test rather than Wilcoxon, run both at the item level and clustered by relation (20,000 sign assignments, seed fixed in advance; Appendix A). Confidence intervals use an item-level bootstrap and a relation-clustered bootstrap (4000 replicates each). The edit target is counterfactual (§6.3), so leakage cannot be a pre-existing correct answer.

6.2 As-deployed contrast

Using the released locality probe as the own probe and a random other row’s probe as the comparison, applying an edit causes the model to answer

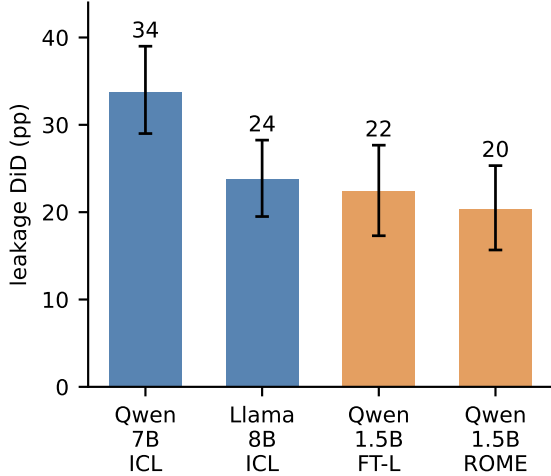


Figure 5: As-deployed leakage difference-in-differences (95% CI). The edit’s target leaks into the released neighbor probe in all four settings, including the two parameter editors, which have no retrieval step.

the own probe with the edit’s target in 20–35% of cases, against approximately 1% for the comparison probe (Figure 5, Table 3). This contrast reflects the benchmark as deployed, but it confounds two factors: the released probe shares the edit’s *relation* by construction, whereas the random comparison shares neither relation nor subject. It therefore overstates the probe-selection effect in isolation, which we separate in the following subsection.

6.3 Matched reconstruction

To separate the contribution of a shared relation from that of a similar subject, we construct a matched design: for each edit we build two comparison probes drawn from the *same relation*, one chosen as the fuzzy-nearest subject label and one at random. The edit target is *counterfactual*, sampled from the relation’s object pool and required to differ from the true object and from every probe answer, so leakage can only arise if the model emits the specific injected object. Relation and question template are held fixed by construction, and the strata are constructed to differ primarily in subject similarity (0.643 versus 0.259): they are closely balanced on answer length (2.47 versus 2.43 tokens), subject-label length (3.39 versus 3.61 tokens), target-in-subject token overlap (7.0% versus 7.8%), and near-floor baseline accuracy (1.8% versus 2.8%), the last a small residual difference we note rather than fully control.

In this matched design (Table 4), similarity-associated probe selection is associated with +11.5

Editor (backbone)	measure	DiD	95% CI
in-ctx Qwen-7B	leakage	+11.5***	[+6.6, +16.8]
in-ctx Qwen-7B	locality	−2.3	[−7.2, +3.1]
in-ctx Llama-8B	leakage	+7.0*	[+1.3, +12.4]
in-ctx Llama-8B	locality	−2.8	[−9.3, +5.1]
ROME Qwen-1.5B	leakage	+12.3***	[+7.0, +17.7]
ROME Qwen-1.5B	locality	−1.7	[−6.3, +3.0]

Table 4: Matched reconstruction, relation and template held constant ($N=399$ per in-context backbone, 300 for ROME). Similarity-associated selection is associated with additional leakage but leaves locality unchanged, including under parameter editing. Item-level bootstrap CIs; significance from relation-clustered sign-flip permutation tests (20,000 sign assignments) for every row: * $p<0.05$, ** $p<0.01$, *** $p<0.001$; locality rows are all $p>0.4$. The ROME row spans 99 relations; its relation-clustered leakage interval [+7.2, +17.1] and permutation ($p=2\times 10^{-4}$) match the item-level values shown.

Backbone, probe	n	null	own	loss (CI)
Qwen-7B, own	61	80.3	44.3	+36.1 [23.0, 49.2]
Qwen-7B, comp.	65	73.8	78.5	−4.6 [−15.4, 6.2]
Llama-8B, own	90	80.0	74.4	+5.6 [−4.4, 15.6]
Llama-8B, comp.	99	86.9	86.9	+0.0 [−7.1, 7.1]

Table 5: Corruption of previously correct knowledge on released probes. Among probes answered correctly at baseline, gold-answer retention (%) under an unrelated (null) edit versus the probe’s own edit; *loss* is the within-probe difference (null − own) in percentage points with a 95% paired-bootstrap interval. Only the Qwen-7B own probe shows significant loss (paired sign test $p=1.0\times 10^{-5}$); all other contrasts are non-significant.

percentage points of additional leakage on Qwen-7B (relation-clustered permutation $p=3.5\times 10^{-4}$) and +7.0 on Llama-8B ($p=0.03$) after matching on relation and template, while locality is unchanged (DiD near zero, $p>0.4$). Because the strata are matched but not randomized, we read this as the difference associated with similarity-driven selection under this reconstruction rather than as an unconditional causal effect. Leakage rises monotonically with probe-subject similarity when the real edit is applied but stays flat under the null (Figure 6). The counterfactual target further disfavors legitimate semantic relatedness, since a related neighbor would not produce the specific false object. The design is observational rather than randomized, however: fuzzy-nearest labels may still differ from random ones in entity type, pretraining frequency, or morphology, so we attribute the effect to similarity-associated selection rather than to lexical similarity in isolation.

Leakage versus corruption of held knowledge.

Because baseline accuracy on these reconstructed probes is near the floor (1.8% versus 2.8%), the effect above is edit-target leakage into probes the model could rarely answer to begin with: the injected object appears where there was seldom a correct neighboring fact to preserve. Corruption of *previously correct* knowledge is a separate claim, which we test on the released probes, where baseline accuracy has headroom (15–25%). We restrict to probes the model answered correctly before any edit and contrast two conditions on those same probes: applying the row’s own edit versus applying an unrelated (null) edit. The null condition, not a 100% reference, is the baseline, because applying any in-context edit perturbs a few answers on its own; the edit-attributable loss is the within-probe difference between the two (Table 5). On Qwen-7B this loss is 36.1 pp (80.3% retained under the null edit versus 44.3% under the own edit, $n=61$), with a paired sign test over the 24/2 discordant probes giving $p=1.0\times 10^{-5}$; this is the within-probe null-versus-own contrast, not the full own-versus-comparison difference-in-differences. On Llama-8B the same loss is 5.6 pp ($n=90$, $p=0.38$), not significant. The comparison probe shows no loss on either backbone (−4.6 pp and 0.0 pp). Corruption of held knowledge is therefore demonstrated on one backbone but not the other, whereas edit-target leakage is large and consistent across both.

6.4 Parameter editing

The as-deployed effect might be attributed to similarity-based retrieval, but it persists under parameter editing, which has no retrieval step (Table 3), and it also holds under the matched design: running ROME on the reconstruction (relation and template fixed) gives a leakage DiD of +12.3 pp ($p=2\times 10^{-4}$; Table 4), close to the in-context backbones, so the effect is not an artifact of the confounded as-deployed contrast. These 300 matched items span 99 relations, so we apply the same relation-clustered inference used for the in-context rows; the clustered interval [+7.2, +17.1] and permutation ($p=2\times 10^{-4}$) coincide with the item-level values, confirming the effect is not carried by a few large relation blocks. All parameter-editing DiDs are intention-to-treat over every attempted edit; conditioning on the subset with a verified successful edit raises leakage further (FT-L +22.3 $\rightarrow$ +35.4, ROME as-deployed +20.3 $\rightarrow$

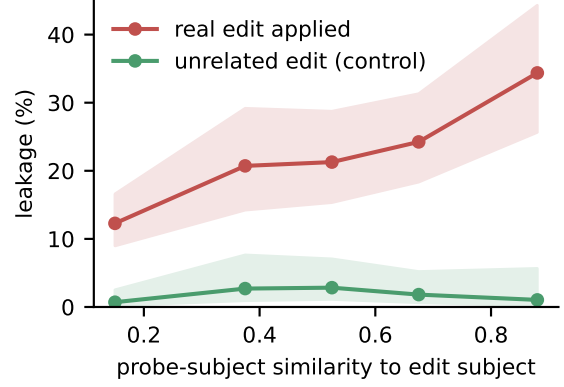


Figure 6: Leakage increases with probe-subject string similarity when the real edit is applied but remains flat under an unrelated edit; the effect is therefore edit-driven rather than a property of similar strings. A Cochran-Armitage trend test over the ordered bins confirms the increase under the real edit ($z=4.85$, $p=1.2\times 10^{-6}$) and finds no trend under the null ($p=0.48$). Bands are 95% Wilson intervals.

+24.1, ROME matched +12.3 $\rightarrow$ +15.2), so the reported figures are the conservative choice given FT-L’s 53% success rate. ROME also leaves aggregate behavior unchanged, moving locality by only +2.3 pp ($p>0.5$) as-deployed and −1.7 pp matched, yet it leaks the edit’s target to the neighbor probe. This pattern is difficult to explain solely through generic relation-level degradation, although aggregate locality alone cannot exclude heterogeneous offsetting item-level effects, and leakage and unchanged-answer rates measure distinct events. These results do not constitute a change to Wikidata’s reported score. Its own metric (the WBE column of Table 3, the model still producing *loc_ans*) moves by −6.0, −1.5, −4.7, and +0.7 pp; only the Qwen-7B in-context setting is significant under the relation-clustered test (FT-L is borderline, $p=0.05$ item-level and 0.08 relation-clustered), and the surgical ROME editor is null. That metric therefore under-registers the large underlying leakage.

7 Audited repair set

We do not reconstruct the full benchmark; a deterministic screen over all 502,382 rows needs the historical entity state at each edit’s revision, which the live MediaWiki API serves only per revision at a low rate. Bulk history dumps or batched revision retrieval could remove this per-row bottleneck, but at substantial storage and engineering cost, so a full screen is left to future work; Wikidata’s full-history

dumps, or derived resources such as Wikidated (Schmelzeisen et al., 2021), would supply the per-revision entity state without per-row API calls. The extraction defect itself is straightforward to avoid: the Wikidata JSON dump exposes role and rank explicitly, and existing structured parsers consume qualifiers and ranks directly, so a role- and rank-aware rebuild of the pipeline is an engineering task rather than a research problem. We instead release an audited, provenance-bearing repair set for the sampled rows, organized into three tracks: *atomic* (role-correct unique main facts, 49.3%), *endpoint* (role-correct but non-unique, 4.7%), and *invalid* (46.0%, retained with its diagnosis). Each row carries per-row provenance (snapshot revision, endpoint answer sets, and role evidence), and 337 object identifiers within the audited subset are corrected. The manifest states this scope explicitly: per-row labels exist for the sampled rows, and the prevalence estimates generalize directly to the resolvable subpopulation under the sampling design of §3.1. Appendix B shows repaired records.

8 Related work

Knowledge editing and its benchmarks. Parameter editors such as ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023), hypernetwork editors (De Cao et al., 2021; Mitchell et al., 2022a), and memory-based methods such as SERAC (Mitchell et al., 2022b), GRACE (Hartvigsen et al., 2023), and WISE (Wang et al., 2024a) are evaluated on locality, the preservation of neighboring knowledge, and are commonly implemented in toolkits such as EasyEdit (Wang et al., 2024b). Benchmarks range from single-fact suites derived from zsRE (Levy et al., 2017) and CounterFact (Meng et al., 2022) to multi-hop (Zhong et al., 2023) and survey collections (Yao et al., 2023). Closest to our target, WikiFactDiff (Ammar Khodja et al., 2024) derives atomic factual updates from two Wikidata snapshots, and WikiBigEdit (Thede et al., 2025) scales this to real, temporally ordered Wikidata updates. Our work is orthogonal to the design of an editor: we audit the benchmark on which such methods are scored.

Specificity and side effects of editing. Several studies show that edits leak into related content: RippleEdits (Cohen et al., 2024) measures ripple effects on entailed facts, improved specificity benchmarks reveal unwanted collateral changes (Hoelscher-Obermaier et al., 2023), and editing

can distort knowledge the model already holds (Li et al., 2024). Although factual associations can be localized to specific neurons (Dai et al., 2022), localization has been shown not to predict editability (Hase et al., 2023). Our leakage measure is in this tradition, but the leakage we identify is induced by the *benchmark’s own probe selection* rather than by an editor.

Dataset and benchmark audits. Systematic auditing of NLP resources has exposed label errors (Northcutt et al., 2021), split artifacts (Gorman and Bedrick, 2019), and problems in dataset development and use (Paullada et al., 2021), motivating broader construct-validity and measurement concerns (Bowman and Dahl, 2021; Bender and Friedman, 2018; Raji et al., 2021; Jacobs and Wal-lach, 2021). We contribute a population-level, role-aware audit of a knowledge-editing benchmark against its source knowledge base. The two failure modes we isolate, a text scan substituting for structured parsing and string similarity substituting for semantic selection, are not specific to WikiBigEdit and are worth checking in any benchmark generated automatically from a structured knowledge base.

Wikidata quality and temporal knowledge. Wikidata (Vrandečić and Krötzsch, 2014) data quality has been surveyed extensively (Piscopo and Simperl, 2019), and Wikidated (Schmelzeisen et al., 2021) reconstructs its revision history as triple additions and deletions. Temporal and multi-valued knowledge evaluation (Dhingra et al., 2022; Jang et al., 2022; Elazar et al., 2021) motivates role- and rank-aware semantics. We add a role-aware decomposition keyed on recovered interval timestamps that the triple view does not capture.

9 Limitations

Our central finding is a prevalence claim about the released supervision. Sampling error is bounded by the reported intervals and implementation correctness by an independent re-parser (PPV 100%, §3.3); beyond these, the estimate could be wrong in only two substantive ways, if the revisions we audit differed materially from those the pipeline consumed, or if ρ_{WBE} were an inappropriate criterion. We test both, with the snapshot-offset robustness check (Figure 3) and the three endpoint-semantics variants (Table 2). The number of editors or backbones evaluated downstream is not among the ways

this finding could be wrong; the limitations below bound secondary claims rather than this one.

The route by which incorrect identifiers reach the release is inferred from the code, whereas the consequence is measured directly, so the prevalence estimate holds whether or not our mechanistic account is exact. Because the validation shares the source revisions and the ρ_{WBE} definition with the audited pipeline, it establishes that the implementation faithfully executes the criterion against the reference state rather than adjudicating the criterion itself, for which we report ρ_{all} and ρ_{truthy} sensitivity. The downstream experiment demonstrates that the defect has measurable consequences; it is not the basis of the audit. It covers in-context and retrieval editing and two gradient editors (FT-L, ROME) on a single 1.5B model, with the matched reconstruction run for in-context editing and, for ROME, under weight editing; MEMIT (Meng et al., 2023) and larger backbones would extend its external validity but cannot revise the prevalence estimate. Within that experiment the edit-target leakage effect is large and consistent, while corruption of previously correct knowledge is demonstrated on one backbone but not the other. Critically, we do **not** claim that WikiBigEdit’s reported locality numbers are materially incorrect or that any published ranking changes, since the effect on its own metric is small and backbone-dependent. What we establish is that the supervision is substantially malformed and the locality probes are selected by an uncontrolled string-similarity rule, that both problems are measurable and mechanistically explained, and that an audited repair set is available.

References

- Hichem Ammar Khodja, Frédéric Béchet, Quentin Brabant, Alexis Nasr, and Gwénolé Lécroché. 2024. WikiFactDiff: A large, realistic, and temporally adaptable dataset for atomic factual knowledge update in causal language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*.
- Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics (TACL)*.
- Samuel R. Bowman and George E. Dahl. 2021. What will it take to fix benchmarking in natural language understanding? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics (TACL)*.
- Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics (TACL)*.
- Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhishava Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics (TACL)*.
- Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2023. Aging with GRACE: Lifelong model editing with discrete key-value adaptors. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Peter Hase, Mohit Bansal, Been Kim, and Asma Ghan-deharioun. 2023. Does localization inform editing? Surprising differences in causality-based localization vs. knowledge editing in language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Jason Hoelscher-Obermaier, Julia Persson, Esben Kran, Ioannis Konstas, and Fazl Barez. 2023. Detecting edit failures in large language models: An improved specificity benchmark. In *Findings of the Association for Computational Linguistics: ACL*.
- Abigail Z. Jacobs and Hanna Wallach. 2021. Measurement and fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*.
- Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. 2022. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving

- language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL)*.
- Zhoubo Li, Ningyu Zhang, Yunzhi Yao, Mengru Wang, Xi Chen, and Huajun Chen. 2024. Unveiling the pitfalls of knowledge editing for large language models. In *International Conference on Learning Representations (ICLR)*.
- MediaWiki. 2024. Wikibase/DataModel: statements, main snaks, qualifiers, references, and rank. <https://www.mediawiki.org/wiki/Wikibase/DataModel>. Accessed 2026.
- Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *International Conference on Learning Representations (ICLR)*.
- Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022a. Fast model editing at scale. In *International Conference on Learning Representations (ICLR)*.
- Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. 2022b. Memory-based model editing at scale. In *International Conference on Machine Learning (ICML)*.
- Curtis G. Northcutt, Anish Athalye, and Jonas Mueller. 2021. Pervasive label errors in test sets destabilize machine learning benchmarks. In *NeurIPS Datasets and Benchmarks Track*.
- Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily Denton, and Alex Hanna. 2021. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11):100336.
- Alessandro Piscopo and Elena Simperl. 2019. What we talk about when we talk about Wikidata quality: a literature survey. In *Proceedings of the 15th International Symposium on Open Collaboration (OpenSym)*.
- Inioluwa Deborah Raji, Emily M. Bender, Amandalynne Paullada, Emily Denton, and Alex Hanna. 2021. AI and the everything in the whole wide world benchmark. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS)*.
- Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50.
- Lukas Schmelzeisen, Corina Dima, and Steffen Staab. 2021. Wikidated 1.0: An evolving knowledge graph dataset of Wikidata’s revision history. *arXiv preprint arXiv:2112.05003*.
- Lukas Thede, Karsten Roth, Matthias Bethge, Zeynep Akata, and Thomas Hartvigsen. 2025. WikiBigEdit: Understanding the limits of lifelong knowledge editing in LLMs. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *PMLR*.
- Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.
- Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024a. WISE: Rethinking the knowledge memory for lifelong model editing of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Peng Wang, Ningyu Zhang, Bozhong Tian, Zekun Xi, Yunzhi Yao, Ziwen Xu, Mengru Wang, Shengyu Mao, Xiaohan Wang, Siyuan Cheng, Kangwei Liu, Yuan-sheng Ni, Guozhou Zheng, and Huajun Chen. 2024b. EasyEdit: An easy-to-use knowledge editing framework for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL): System Demonstrations*.
- Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. 2023. MQAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

A Inference details

Relation-clustered bootstrap intervals resample whole relations uniformly with replacement: for each of 4000 replicates (a count fixed in advance, seed 20260722) we draw R relations with replacement, where R is the number of distinct relations, pool all rows in the drawn relations, and recompute the statistic; cluster sizes therefore enter naturally and larger relations are more influential in proportion to their size. Because it pools

the resampled relations’ rows, it preserves the empirical relation-size distribution and stays row-weighted; it widens the interval to reflect within-relation dependence rather than reweighting toward an equal-per-relation target. Subject clustering is negligible, as each subject appears once in the sample, and the 8 intervals are too few to bootstrap. The paired sign-flip permutation test for the downstream difference-in-differences draws 20,000 Rademacher sign vectors (seed 20260723); the item-level test flips each per-item term independently, and the relation-clustered test flips all items in a relation together. Reported p -values are two-sided, $(\#\{|\text{permuted}| \geq |\text{observed}|\} + 1)/(B + 1)$.

Leakage and locality use token-boundary containment after normalization (lowercasing, punctuation stripping, article removal). Token-boundary matching avoids substring false positives within a token, but a short target that is a distinct token inside a longer entity name can still register; we did not observe this affecting the counterfactual leakage measure, since counterfactual targets are sampled to differ from every probe answer.

B Repaired-record examples

Table 6 shows representative repaired records with the released row, the pinned source revision, the diagnosis, and the corrected object identifier where applicable.

object-id misattached (rev. 2103510427). released: Reinhard Merkel (Q2140863), place of birth, Hof (Q1852145). corrected: object Q1852145 → **Q6938** (label “Hof”; ρ_{WBE} set {Q6938}).

object-id misattached (rev. 2082862946). released: Thunbergia grandiflora (Q138397), subclass of, vine (Q2135068). corrected: object Q2135068 → **Q917284** (label “vine”).

role-erased, qualifier (rev. 2179477364). released: solar eclipse of 1836 BC (Q5693372), issued by, NASA (Q23548). diagnosis: value occurs only as a qualifier on a P528 (catalog code) statement; ρ_{WBE} main set empty. Retained in the invalid track.

role-erased, qualifier (rev. 2126904748). released: P Raveesh (Q16190976), electoral district, Ernakulam constituency (Q3595497). diagnosis: qualifier on a P3602 (candidacy) statement, not a main fact.

Table 6: Repaired-record examples. Each carries its source revision, so the verdict is reproducible.

C Reproducibility

Code, the per-row audit manifest (sampled row identity, inclusion status, audit label, pinned revision, and corrected identifier), the three-track

repair set, and the script generating each table and figure are released at <https://github.com/edisonly1/wikibigedit-audit-artifact>.

Seeds. Audit sampling and the relation-clustered bootstrap use seed 20260722; probe-set construction, the downstream experiments, and the sign-flip permutation test use 20260723; the extraction-verification subsamples use 20260724 and 20260725. No result depends on an unseeded draw.

Table and figure sources. Table 1 (prevalence and relation-clustered intervals), Table 2 (endpoint sensitivity), Table 3 (as-deployed effects), Table 4 (matched reconstruction), and Figures 3–6 are each produced by a named script in the repository, listed in its reproducibility guide.

Prompts and decoding. The question-answering system prompt is “*You answer factual questions with a short noun phrase and nothing else. Output only the answer, at most a few words.*” Edits are supplied as optional context (“You may use the following fact if, and only if, it is relevant. . .”); the SERAC-style scope gate uses “*Does the stored fact directly answer this exact question?*” (selected against labeled cases: accuracy 0.80, specificity 6/6). Decoding is greedy and deterministic: for Ollama backbones, temperature 0, top- k 1, top- p 1, seed 0, 24 new tokens; for the Hugging Face backbone, do_sample=False, one beam.

Determinism protocol. Under greedy decoding the *first* evaluation of a prompt can differ from all subsequent ones, which we traced to llama.cpp taking a different prefill path on a cold KV cache; it is a decoding-engine artifact, not sampling. We characterized this before running the reported experiments (see repeat_stability.py): the first-to-second call mismatch is 6.7% (Qwen) and 3.3% (Llama), while every later call is identical (0/120 mismatches per backbone). The mismatches are idiosyncratic near-variants (e.g. “Munich”/“Ulm”, “naturalism”/“naturalist”), not a systematic bias toward a particular answer. The adopted protocol, fixed in advance, discards the first evaluation and uses the second, so every reported answer is from the stable regime.

Provenance. Ollama 0.32.1 with qwen2.5:7b (digest 845bdba0ea48) and llama3.1:8b (digest 46e0c10c039e), both Q4_K_M; Qwen2.5-1.5B-Instruct (fp32) for parame-

ter editing. Python 3.12, torch 2.13 (CUDA 13.0), transformers 5.14, on one NVIDIA RTX 5070 (12.8 GB).

Editor hyperparameters. FT-L edits the layer-12 MLP down-projection with Adam (lr 10^{-3} , 30 steps) under an L_∞ constraint of 5×10^{-3} ; edit success 52.7%. ROME applies a rank-one update to the same layer, optimizing the value vector (lr 0.5, 40 steps) with key covariance C (ridge $\lambda = 0.05 \cdot \text{mean}(\text{diag})$) estimated from 40,005 key positions; it is validated so that the edited layer maps $k^* \rightarrow v^*$ to relative error 10^{-7} with exact weight restoration; edit success 83.0%.

D Portable audit checklist

Table 7 restates the audit as four checks that apply to any benchmark generated from versioned structured data, with the evidence each check produces in this paper and the corresponding repair.

Risk	Diagnostic	Evidence here	Repair
Role erasure	Compare extracted facts with structured snaks	Role-specific error rates (Table 1)	Parse roles and ranks
Identity loss	Track identifiers through every stage	Label-ID mismatch rate (Table 2, row C)	Preserve canonical IDs
Temporal mismatch	Replay pinned revisions	Revision agreement (§3.3; Figure 3)	Store source timestamps
Lexical probe selection	Compare against semantic controls	Matched leakage effect (Table 4)	Relation-aware matching

Table 7: The audit as a portable checklist.