CORPUS-AS-MODEL PRETRAINING FOR ZERO-SHOT KNOWLEDGE GRAPH REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recently, knowledge graph (KG) reasoning has shifted from graph-specific inference toward zero-shot knowledge graph foundation models (KGFMs) that transfer relational priors across graphs. Yet prior progress has been driven mainly by architectural advances, while the pretraining corpus has remained a passive artifact, making it unclear whether models learn transferable reasoning patterns or partly adapt to the statistics of a few source graphs. CAMP-KG studies *corpus-as-model pretraining*: it asks whether a deliberately designed synthetic pretraining pipeline can replace real-KG pretraining while keeping the zero-shot reasoner unchanged. It (i) treats the pretraining distribution as a design object, (ii) samples relational rules independently of graph statistics, and (iii) uses provenance-aware supervision that keeps rule premises observable and prevents target leakage. On the 54 zero-shot graphs of the standard ULTRA suite, CAMP-KG[1] is comparable to real-KG pretraining in aggregate, improves the 32 graphs that share no provenance with the real pretraining sources, and regresses on the 22 source-derived graphs.

1 INTRODUCTION

Recent developments in knowledge graph (KG) reasoning have heightened the need for relational priors that transfer across graphs rather than anchor to source graphs. The field has moved from inductive reasoning models (Teru et al., 2020; Zhu et al., 2021), which replace entity-specific embeddings with task-trained structural reasoning, toward knowledge graph foundation models (KGFMs) (Galkin et al., 2024a), which pretrain a universal reasoner for zero-shot cross-graph inference. This transition has been accelerated by ULTRA-style (Galkin et al., 2024a) relation-graph reasoning and by further KGFMs, including TRIX (Zhang et al., 2024), MOTIF (Huang et al., 2025), and Flock (Kim et al., 2026). As KGs become larger, more dynamic, and more heterogeneous across biomedical, scientific, and industrial domains, the central question is no longer how to reason on one graph, but how to transfer across graphs.

Although extensive research has been carried out on KGFM architectures, previous studies have rarely treated the pretraining corpus as an equally important modeling object. The research has tended to focus on message passing, relation-dependency graphs, or semantic augmentation, while leaving the data source anchored to a small set of real KGs such as FB15k-237, WN18RR, and CoDEx-Medium (Toutanova et al., 2015; Dettmers et al., 2018; Safavi & Koutra, 2020; Du et al., 2026; Arun et al., 2025; Hua et al., 2025). As illustrated in Figure 1, this architecture-first escalation leaves three pretraining-side issues largely unresolved: real-corpus anchoring, topology–logic entanglement, and premise-hiding in randomly constructed supervision. The hypothesis of this paper is that some zero-shot failures arise not from insufficient expressivity, but from what we call *distributional entanglement*: query-irrelevant statistics of the source graphs, such as degree skew, density, and relation frequencies, become decodable from and relied upon by the query-conditioned relation representations, so that transfer degrades when a target's statistics leave the source support. The hypothesis makes two testable predictions: the advantage of broader pretraining should grow with structural distance from the source graphs (§5.1), and query-irrelevant statistics should be less decodable from query-conditioned relation representations after broader pretraining (§5.2).

[1] Code repository: https://anonymous.4open.science/r/TFM-23B8

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.

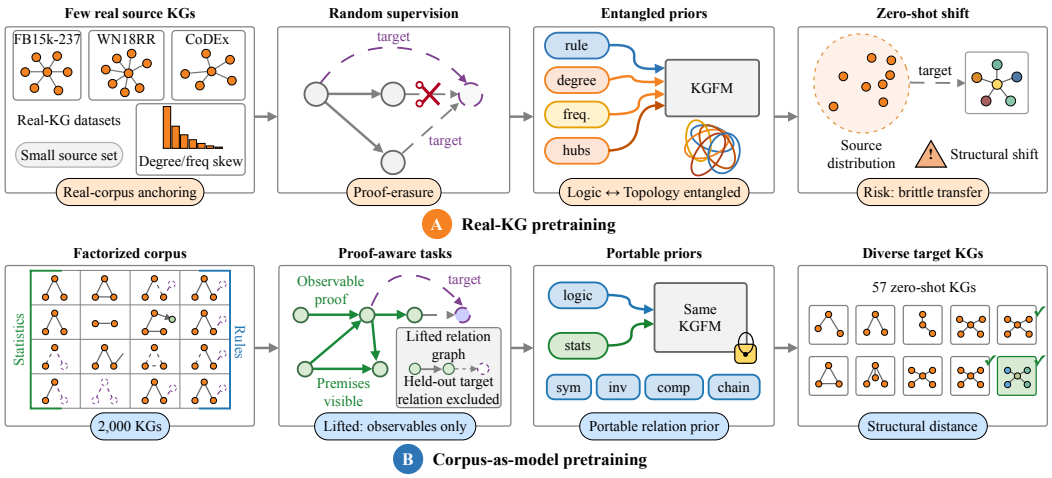

Figure 1: **Motivation for corpus-as-model pretraining. Top:** Real-KG pretraining from a few source KGs with random supervision can hide rule premises and entangle rules with topology, causing brittle zero-shot transfer under structural shift. **Bottom:** CAMP-KG pretrains the same KGFM on a factorized synthetic corpus with provenance-aware tasks, sampling relational rules independently of graph statistics to learn portable priors for diverse target KGs.

A pretraining pipeline has three components: the *corpus distribution* (which graphs exist), the *task distribution* (which edges are observable, which become queries, and how queries are sampled), and the *architecture*. CAMP-KG changes the first two and freezes the third. This paper therefore asks whether a deliberately factorized synthetic pipeline can match real-KG pretraining with an unchanged reasoner; how much of any difference is due to the corpus rather than the supervision, which we separate with a source×supervision factorial; which generator axes matter; and whether gains concentrate on targets that share no provenance with the real sources.

CAMP-KG operationalizes corpus-as-model pretraining through a factorized synthetic corpus generator, which generates 2,000 procedural KGs by independently sampling a rule axis and a graph-statistics axis. The rule axis contains templates for symmetry, inversion, subsumption, composition, intersection, long-chain composition, and branching; the statistics axis varies entity count, relation count, relation-frequency concentration, degree skew, density, incompleteness, and noise under broad fixed priors. A provenance-aware task construction keeps rule premises observable, uses rule conclusions as prediction targets, and builds stage-specific lifted relation graphs only from observable edges. The corpus trains the unchanged ULTRA model (Galkin et al., 2024a) with the same loss, optimizer, negative sampling, model size, step budget, and zero-shot evaluation pipeline as ULTRA-REAL.

The contributions of this paper are as follows:

- **Corpus-as-model pretraining for KGFMs.** We specialize synthetic-prior pretraining to zero-shot relational KG completion with an unchanged reasoner, and give an explicit generative model of the pretraining distribution.
- **Rule–statistics factorization.** CAMP-KG samples relational rules independently of graph size, degree skew, relation frequency, incompleteness, and noise, and we verify that this independence approximately survives rejection.
- **Provenance-aware supervision.** The task construction keeps rule premises observable and isolates validation and test targets from stage-specific relation graphs.
- **Controlled zero-shot evidence.** On the official 54-graph suite, the synthetic pipeline is comparable in aggregate, improves source-disjoint graphs, and regresses on source-derived ones. A source×supervision factorial, a 2×2 generator ablation, edge- and compute-matched controls, a second backbone, and representation probes show which of these effects come from the corpus and which from the supervision.

2 Related Work

We review KGFMs and inductive backbones, whose architecture CAMP-KG keeps fixed, and then synthetic priors, rules, and benchmarks, which motivate the pretraining-distribution intervention.

KGFMs and inductive backbones. Recent KGFMs improve zero-shot transfer mainly through *architectural* changes. ULTRA (Galkin et al., 2024a) introduced lifted-relation-graph reasoning, and TRIX (Zhang et al., 2024) and MOTIF (Huang et al., 2025) extend it with more expressive relation/entity updates and motif-based relation graphs. Flock (Kim et al., 2026) replaces relation-graph message passing with random-walk sequence modeling, GraphOracle (Du et al., 2026) uses query-dependent relation-dependency attention, KG-ICL (Cui et al., 2024) uses in-context prompt graphs, and SEMMA/MERRY (Arun et al., 2025; Hua et al., 2025) add textual semantics, showing that purely structural signals can fail under semantic shift. ULTRA itself compared 3-, 4-, and 50-graph real pretraining mixtures; this is the closest prior evidence on the corpus question, but it varies only which real graphs are used. CAMP-KG fixes the reasoner (ULTRA; TRIX as a second backbone in Appendix L) and changes the pretraining pipeline. KGFM reasoning builds on GraIL's subgraph-based inductive prediction (Teru et al., 2020) and NBFNet's path reasoning (Zhu et al., 2021); InGram (Lee et al., 2023) and RMPI (Geng et al., 2023) introduced relation graphs for unseen relations, C-MPNN (Huang et al., 2023) formalized conditional message passing, ISDEA (Gao et al., 2023) established double permutation equivariance, and TACT (Chen et al., 2021a) identified relation-correlation patterns that inform our template catalog.

Synthetic priors, rules, and benchmarks. Prior-data fitted networks pretrain on samples from a designed synthetic prior (Müller et al., 2022; Hollmann et al., 2023); GraphPFN applies this to attributed graphs (Eremeev et al., 2026), and OpenGraph pretrains graph foundation models on generated graphs (Xia et al., 2024). Domain randomization (Tobin et al., 2017), algorithmic reasoning (Veličković et al., 2022), and synthetic pretraining tasks (Wu et al., 2021; 2022) follow related ideas. Rule-generated relational data appear in CLUTRR and GraphLog (Sinha et al., 2019; 2020), and proof-annotated supervision in RuleTaker and ProofWriter (Clark et al., 2020; Tafjord et al., 2021). These works either do not target zero-shot KG completion or use synthetic data to *evaluate* rather than to *replace* a KGFM's pretraining corpus, as with MOTIF's ConnectHub (Huang et al., 2025) and UltraQuery's synthetic queries (Galkin et al., 2024b). Our contribution is therefore scoped to factorized rule/statistics synthesis plus provenance-aware KG-completion tasks under a fixed KGFM. Rule learners (AMIE+ (Galárraga et al., 2015), AnyBURL (Meilicke et al., 2019), Neural LP (Yang et al., 2017), DRUM (Sadeghian et al., 2019), RNNLogic (Qu et al., 2021)) mine Horn rules from one KG. We instead use such patterns generatively, and we use AnyBURL to build provenance-aware tasks on real KGs (§4.1). Rule-typed evaluation (Meilicke et al., 2018) and shortcut-mitigated inductive splits (Shomer et al., 2025) show that benchmark MRR can reward shallow heuristics; we evaluate on the latter in Appendix M. LLM-based KG reasoning uses prompting (Luo et al., 2024) or fine-tuning (Guo et al., 2024). CAMP-KG is purely structural and complementary to these approaches.

3 Method: CAMP-KG

This section first establishes the problem setup and fixed ULTRA backbone (§3.1), then introduces the factorized synthetic KG generator (§3.2) and provenance-aware task construction (§3.3), and finally details the training and inference protocol (§3.4).

3.1 Preliminaries and Problem Setup

A knowledge graph $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$ has entities $\mathcal{V}$, *canonical* relations $\mathcal{R}$, and directed triples $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$, where (h, r, t) asserts relation r from head h to tail t. Let $\widetilde{\mathcal{R}} = \mathcal{R} \cup \mathcal{R}^{-1}$ be the augmented relation set and $\overline{E} = E \cup \{(t, r^{-1}, h) : (h, r, t) \in E\}$ the inverse closure of an edge set E. Counts $|\mathcal{R}|$, including the relation column of Table A1, are canonical; ULTRA's relation graph has $2|\mathcal{R}|$ nodes.

ULTRA recap. ULTRA (Galkin et al., 2024a) consists of a relation parameterizer GNN_r on a lifted relation graph G_r and an NBFNet-style entity reasoner GNN_e (Zhu et al., 2021). $G_r = \textsc{Lift}(\overline{E})$

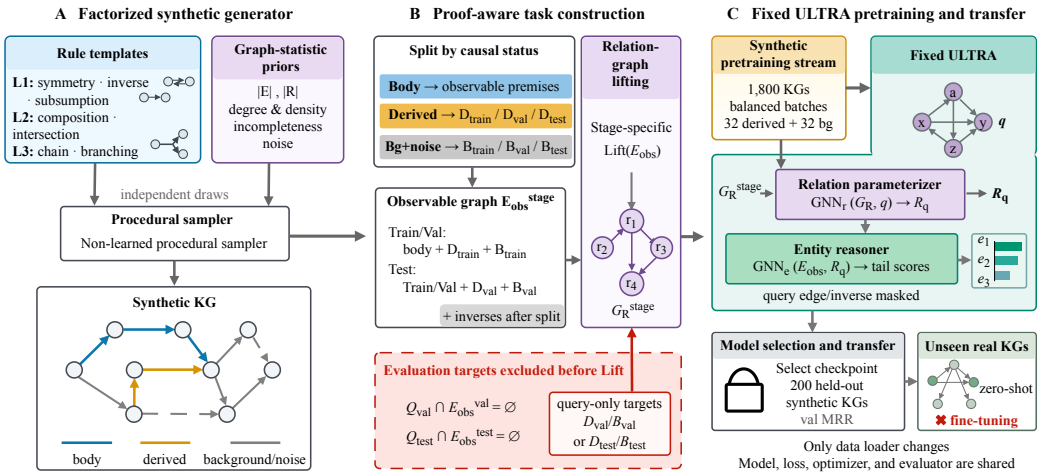

Figure 2: **CAMP-KG overview.** **(A)** A non-learned generator samples rule templates T and graph statistics S independently and synthesizes body, derived, background, and noise triples. **(B)** Provenance-aware construction assigns triples to observable/query roles by derivation status and builds stage-specific observable and relation graphs with evaluation targets excluded before lifting. **(C)** The fixed ULTRA reasoner is pretrained on balanced synthetic batches, selected on held-out synthetic KGs, and transferred zero-shot to unseen real KGs without fine-tuning.

connects nodes in $\widetilde{\mathcal{R}}$ by four binary entity-co-occurrence interactions (head–head, head–tail, tail–head, tail–tail) (Lee et al., 2023; Chen et al., 2021a). For query relation q, $\mathrm{GNN}_r(G_r, q)$ produces $\mathbf{R}_q$, which GNN_e uses to score candidate tails; query edges and their inverses are masked in GNN_e. We use the released configuration (6 layers each, hidden dimension 64, 168K parameters counted programmatically; Appendix O).

Problem statement. A pretraining pipeline is a triple: a corpus distribution $p_{\mathcal{C}}$ over graphs, a task distribution $p_{\mathrm{task}}(E_{\mathrm{obs}}, Q \mid \mathcal{G})$ that also covers query sampling and model selection, and a reasoner f_θ. ULTRA-REAL uses three real graphs with standard ULTRA tasks. CAMP-KG replaces both $p_{\mathcal{C}}$ and p_{task} and keeps f_θ, the loss, optimizer, negative sampling, 200K-step budget, and zero-shot inference fixed. The headline comparison therefore evaluates a *pipeline*; §4.1 separates the corpus from the supervision with a factorial design.

Method overview. CAMP-KG samples rule templates and graph statistics independently, constructs provenance-aware, leakage-free tasks, trains and selects the unchanged ULTRA model on synthetic KGs only, and transfers zero-shot to unseen real KGs without fine-tuning (Fig. 2).

3.2 FACTORIZED SYNTHETIC KG GENERATOR

The generator is a non-learned program defined by

$$T \sim p(T), \quad S \sim p(S), \quad (\mathcal{G}, Q_{\mathrm{train}}, Q_{\mathrm{val}}, Q_{\mathrm{test}}) \sim p(\cdot \mid T, S), \quad \text{kept iff } \mathrm{Acc}(\mathcal{G}) = 1, \quad (1)$$

so the *latent* factorization $p(T, S) = p(T)\,p(S)$ holds by construction. Two steps can reintroduce dependence: rule application adds derived edges and therefore changes realized density, and acceptance depends on both axes. We measure both effects (Appendix N, Table A10). Acceptance varies only within [0.88, 0.96] across template×statistics cells, and among accepted KGs a classifier predicts template indicators from S or from realized nuisance statistics only barely above chance (AUC ≤ 0.58). "Factorization" therefore means latent independence that is approximately preserved in the accepted corpus; it does not mean independence of T from every realized property of $\mathcal{G}$.

Rule axis. Seven templates at three levels (Table A5): L1 **symmetry** $(r(x,y) \to r(y,x))$, **inversion** $(r_1(x,y) \to r_2(y,x))$, **subsumption** $(r_1(x,y) \to r_2(x,y))$; L2 **composition** $(r_1(x,z) \wedge r_2(z,y) \to r_3(x,y))$, **intersection** $(r_1(x,y) \wedge r_2(x,y) \to r_3(x,y))$; L3 **long chain** (3-hop composition) and **branching** $(r_1(x,z) \wedge r_2(y,z) \to r_3(x,y))$. Under inverse augmentation,

branching is composition with an inverted second body relation, and the long chain is iterated composition; the catalog thus contains five distinct patterns plus two compositional variants. Appendix J derives each lifted signature. Templates were frozen before real-KG evaluation and are included independently with probabilities 0.70/0.62/0.44 per level (expected 4.2 per KG; composition is used as fallback if none is drawn). *Slot allocation:* each template receives its own head relation in addition to the n_{rel} non-head relations, so every sampled template set is feasible. Body relations are drawn uniformly from the non-head pool and may be shared across templates. Heads are never bodies, so rules are non-recursive, and symmetry uses its head as its own body. Non-head relations used by no template are background relations $\mathcal{R}_{\text{bg}}$.

Statistics axis. Seven parameters control topology independently of rule content: $n_{\text{ent}} \sim \text{LogU}(100, 30\text{K})$, $n_{\text{rel}} \sim \text{LogU}(6, 192)$, relation frequencies $\pi \sim \text{Dir}(\kappa\mathbf{1})$ with $\kappa \sim \text{LogU}(0.1, 10)$, degree exponent $\gamma \sim \text{U}(1.5, 3.0)$ with a 15% point mass on uniform degree, pre-rule edge budget per entity $\mu \sim \text{LogU}(3, 50)$, $\eta \sim \text{U}(0, 0.5)$, and $\nu \sim \text{U}(0, 0.10)$. These priors were fixed before real-KG evaluation, with no fitting to benchmark statistics. The accepted corpus has geometric means of 1,730 entities and 38 canonical relations and a mean of 79K edges per KG (Appendix G.2, including histograms against benchmark statistics).

Generation procedure. The generator distributes $n_{\text{ent}}\mu$ non-derived edges over non-head relations according to π. Each relation's edges are created by directed stub matching (Molloy & Reed, 1995) over shared entity in/out-activity sequences drawn from a power law with exponent γ, and self-loops and duplicates are removed. All groundings of each template are then derived in one pass, which is the full closure because heads are never bodies. Finally, an η fraction of derived triples is dropped, and $\nu|\text{background}|$ uniformly random noise triples are added. Algorithm A1 gives the full specification.

3.3 Provenance-Aware Task Construction

A naive benchmark-style split can hide rule premises when rule conclusions are targets. We therefore assign roles by *derivational provenance*, i.e., known derivations under the generator; "proof" carries no stronger meaning here.

Split. **Body triples** B are always observable and never targets. **Derived triples** are split 60/20/20 into $d_{\text{train}}, d_{\text{val}}, d_{\text{test}}$. **Background triples** are split 80/10/10 into $b_{\text{train}}, b_{\text{val}}, b_{\text{test}}$. **Noise triples** N are observable and may be sampled as training queries, but they never enter Q_{val} or Q_{test}, so model selection does not reward predicting random edges. Thus $Q_{\text{train}} = d_{\text{train}} \cup b_{\text{train}} \cup N$, $Q_{\text{val}} = d_{\text{val}} \cup b_{\text{val}}$, $Q_{\text{test}} = d_{\text{test}} \cup b_{\text{test}}$, and

$$E^{\text{train}}_{\text{obs}} = \overline{B \cup d_{\text{train}} \cup b_{\text{train}} \cup N}, \tag{2}$$

$$E^{\text{val}}_{\text{obs}} = E^{\text{train}}_{\text{obs}}, \tag{3}$$

$$E^{\text{test}}_{\text{obs}} = \overline{B \cup d_{\text{train}} \cup b_{\text{train}} \cup N \cup d_{\text{val}} \cup b_{\text{val}}}. \tag{4}$$

Hence $Q_{\text{val}} \cap E^{\text{val}}_{\text{obs}} = \emptyset$ and $Q_{\text{test}} \cap E^{\text{test}}_{\text{obs}} = \emptyset$ (checked by assertions for every KG). Validation edges are observable at test time, matching the standard protocol (Galkin et al., 2024a).

Relation graphs and filtering. G^{train}_r, G^{val}_r, and G^{test}_r are obtained by applying LIFT to Eqs. 2–4, so no evaluation-stage relation graph encodes co-occurrences of its own targets. Lifting from observable edges follows ULTRA practice, and our addition is only the stage-specific construction for synthetic tasks. Within GNN_e, each query edge and its inverse are masked. Negatives are filtered against all true triples of the KG, including held-out and η-dropped derivations. Removing the latter from the filter changes MRR by -0.003 (Appendix K).

Balanced query sampling and rejection. Each 64-query batch contains 32 derived and 32 non-derived queries (Appendix K varies this ratio). Degenerate KGs are rejected by connectivity, size, and query-count filters (Algorithm A1); the rejection rate is 7.2%.

3.4 Training and Inference Protocol

Corpus and training. The offline corpus contains 2,000 synthetic KGs: 1,800 for training and 200 held out for model selection. At each of 200K steps, one training KG is sampled uniformly, its cached

G_r^{train} and $E_{\text{obs}}^{\text{train}}$ are loaded, a mixed 64-query batch is drawn, and the standard ULTRA forward pass and self-adversarial loss are applied. Checkpoints are saved every 10K steps and selected by mean validation MRR over the 200 held-out KGs using G_r^{val} and Q_{val}.

Selection without real graphs. Every generator and training choice, including the sensitivity axes of Appendix K, is selected on synthetic validation MRR. Real-graph results are computed once per frozen configuration. For every axis, the synthetic-validation argmax coincides with the default (Table A7).

Inference. Zero-shot evaluation follows ULTRA exactly: G_r is built once from the target KG's inference edges; for each $(h,\ q,\ ?)$, $\text{GNN}_r(G_r, q)$ produces $\mathbf{R}_q$ and GNN_e scores all entities. Model code, loss, optimizer, and evaluation scripts are shared between CAMP-KG and ULTRA-REAL. Differences come from the pretraining pipeline, i.e., the corpus plus the task construction.

4 EXPERIMENTS

The experiments address four questions: how the synthetic pipeline compares with real-KG pretraining in aggregate and on provenance-defined slices, and how much of the difference is due to the corpus versus the supervision (§4.1); which generator axes matter (§4.2); and whether simple corpus-level confounds explain the results (§4.3).

4.1 MAIN RESULT AND SOURCE×SUPERVISION FACTORIAL

Benchmark and slices. We evaluate on the 54 zero-shot graphs of the official ULTRA suite (13 transductive, 18 inductive-e, 23 inductive-e, r), loaded with the official loaders. The three pretraining graphs are excluded. Appendix B lists the source, split hashes, and statistics of every graph. Before training, we fixed three method-independent slices: *source-proximal* (22 graphs derived from FB15k-237, WordNet, or CoDEx, including FBNELL), *source-disjoint* (the other 32), and *external-hard*, the lowest quartile (14 graphs) by *published* ULTRA(3g) MRR. Our ULTRA-REAL reproduction matches the published per-setting means (inductive-e, r / inductive-e / transductive) within 0.003 (0.343/0.428/0.312 vs. 0.345/0.431/0.312); the median absolute per-graph deviation is 0.003.

Setup and statistics. Both arms use the released ULTRA configuration (AdamW, lr 5×10^{-4}, self-adversarial negative sampling, batch size 64, 200K steps) and 5 seeds. Our primary inference is a hierarchical wild cluster bootstrap (Cameron et al., 2008) over 11 provenance clusters, with seeds resampled within methods (Appendix C). The non-inferiority margin δ=0.01 ($\approx$3% relative) was fixed in the analysis plan before any CAMP-KG evaluation; Appendix C also reports $\delta \in \{0.005, 0.02\}$. To separate corpus from supervision, we add two arms. **Real+provenance** builds provenance-aware tasks on the three real graphs: AnyBURL (Meilicke et al., 2019) rules with confidence ≥ 0.8 are mined on the training split, their premises are kept observable, their conclusions become derived queries, and batches are mixed 50/50. **Synthetic+standard** trains on the CAMP-KG corpus with ULTRA's standard task construction (pooled random split, uniform queries).

Aggregate and slice results. CAMP-KG obtains 0.369 mean MRR over 54 graphs versus 0.364 for ULTRA-REAL (Table 1), a difference of +0.005 (95% CI $[-0.008, +0.017]$). Superiority is not established, while non-inferiority holds at δ=0.01 (one-sided lower bound -0.006) but not at δ=0.005. Per graph, CAMP-KG wins 30/54 (exact sign test p=0.25); with a $\pm$0.005 tie band, the count is 25 wins, 11 ties, and 18 losses. The aggregate hides two opposite effects. CAMP-KG gains +0.020 $[+0.009, +0.032]$ on source-disjoint graphs (0.369 vs. 0.348) and loses -0.018 $[-0.027, -0.010]$ on source-proximal ones (0.369 vs. 0.387). On the external-hard quartile it gains +0.034 $[+0.019, +0.050]$, and cross-fitting (seeds 0–1 select, seeds 2–4 evaluate) gives +0.030. Selecting that quartile by our own ULTRA-REAL runs would instead give +0.041; this inflation is why we no longer use baseline-selected slices.

Corpus versus supervision. Under matched provenance-aware supervision, the corpus effect is +0.003 $[-0.009, +0.015]$ in aggregate and +0.017 $[+0.006, +0.029]$ on source-disjoint graphs. Under standard supervision, the synthetic corpus is 0.009 *below* real pretraining in aggregate. Provenance-aware supervision adds only +0.002 on real graphs but +0.014 on synthetic ones,

Table 1: **Main result and source×supervision factorial.** Mean MRR on the 54 zero-shot graphs of the official ULTRA suite and on pre-specified, method-independent slices; same ULTRA configuration, optimizer, loss, 200K steps, and inference. ±: 95% seed intervals of the aggregate. Bracketed differences: wild-cluster-bootstrap 95% CIs (11 provenance clusters).

Arm	Corpus	Supervision	All 54	Disjoint (32)	Proximal (22)	Ext.-hard (14)
ULTRA-REAL (repro.)	3 real	standard	.364±.004	.348	**.387**	.161
Real+provenance	3 real	provenance	.366±.004	.352	**.387**	.165
Synthetic+standard	1,800 syn.	standard	.355±.005	.354	.356	.177
CAMP-KG	1,800 syn.	provenance	**.369**±.004	**.369**	.369	**.195**
CAMP-KG − ULTRA-REAL (pipeline)			+.005	+.020	−.018	+.034
95% CI			[−.008, .017]	[.009, .032]	[−.027, −.010]	[.019, .050]
Corpus effect | provenance supervision			+.003	+.017	−.018	+.030
Corpus × supervision interaction			+.012	+.011	+.013	+.014

Disjoint/proximal: no/shared provenance with FB15k-237, WordNet, or CoDEx. Ext.-hard: lowest quartile by *published* ULTRA(3g) MRR. Non-inferiority (δ=0.01, fixed a priori): one-sided lower bound −.006; not established at δ=0.005 (Table A2). Wins 30/54 (sign test p=.25); with ±.005 ties: 25/11/18. Factorial construction: Appendix I.

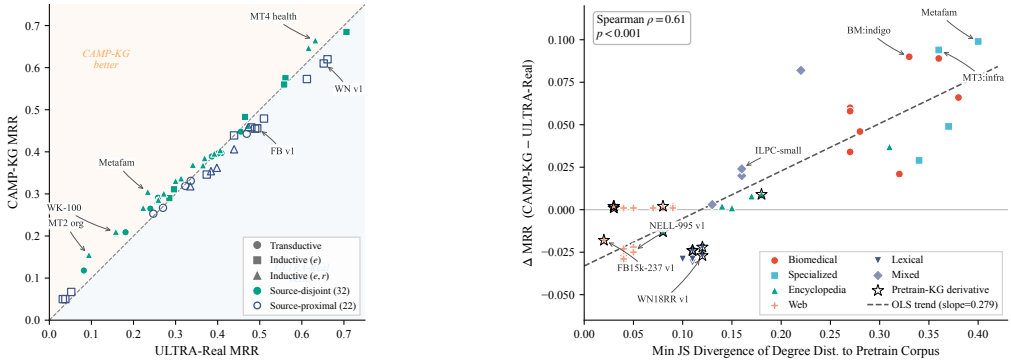

Figure 3: **Left:** Per-graph MRR on the 54 zero-shot graphs (5-seed mean; source-proximal graphs marked); points above the diagonal favor CAMP-KG. **Right:** Each method's MRR vs. standardized structural distance to the real pretraining graphs, with fitted slopes from Eq. 5 (ULTRA-REAL −0.021, CAMP-KG −0.010 per SD). Alternative distances are in App. Fig. A2.

an interaction of +0.012 [+0.004, +0.020]. Aggregate parity is thus a property of the full pipeline: the synthetic corpus helps on graphs outside the real sources' provenance, but only with supervision that exposes rule premises. A compute-matched run (48 GPU-hours) reaches 0.367 (Table 3).

4.2 MECHANISM: 2×2 FACTORIZATION ABLATION

We train four 2,000-KG corpora with identical provenance-aware construction, 50/50 query mixing, backbone, optimizer, budget, and evaluation: **A** is full CAMP-KG; **B** removes rules by rewiring each head relation's derived edges with degree-preserving double-edge swaps, which keeps its in/out-degree sequences and destroys rule correlation; **C** keeps rules but fixes all statistics at their prior medians (n_{ent}=1732, n_{rel}=34, κ=1, γ=2.25, μ=12.2, η=0.25, ν=0.05); **D** removes both.

Results. Both axes matter (Table 2; hierarchical bootstrap, uncorrected). Rules add +0.053 with varied statistics (A vs. B) and +0.048 with fixed statistics (C vs. D). Diversity adds +0.031 with rules (A vs. C) and +0.026 without (B vs. D). All four have p<0.001. The interaction +0.005 (p=0.21) is not significant, so the evidence supports additive contributions rather than synergy. Corpus B stays 4.8 points below ULTRA-REAL, which argues against diversity or unique edges alone as the explanation. On the external-hard quartile, B (0.157) and D (0.148) fall *below* ULTRA-REAL (0.161), whereas A reaches 0.195. With a method-independent slice, gains there thus require rules. The A–D difference of +0.047 on this slice is the mechanism-specific quantity. Absolute gaps are smaller on this slice than on all 54 graphs (Appendix G).

Table 2: 2×2 **factorization ablation.** Mean MRR over the 54 graphs (5 seeds). Only the generator axes vary; supervision, backbone, optimizer, and budget are identical. p-values: hierarchical cluster bootstrap, uncorrected (exploratory).

	Varied Statistics	**Fixed Statistics**
Systematic Rules	**A**: **0.369** ±0.004	C: 0.338 ±0.005
No Rules (rewired)	B: 0.316 ±0.006	D: 0.290 ±0.006

Reference: ULTRA-REAL = 0.364 ±0.004

Contrast	**All 54**	**Ext.-hard**	p **(all)**
A vs. B (rules; varied stats)	+0.053	+0.038	< 0.001
C vs. D (rules; fixed stats)	+0.048	+0.026	< 0.001
A vs. C (diversity; with rules)	+0.031	+0.021	< 0.001
B vs. D (diversity; no rules)	+0.026	+0.009	< 0.001
Interaction (A−B)−(C−D)	+0.005	+0.012	0.21
A vs. D (both axes)	+0.079	+0.047	< 0.001
B vs. ULTRA-REAL	−0.048	−0.004	< 0.001

Table 3: **Corpus-level controls.** All synthetic arms use rules and provenance-aware supervision; all arms share the ULTRA configuration, optimizer, loss, and 200K steps (except compute-matched). Unique edges: canonical training triples, 5 seeds. Details and ULTRA(4g) in Appendix H.

Arm	**#KGs**	**Edges**	**Visits/KG**	**GPU-h**	**All 54**	**Disj.**	**Prox.**	**Hard**
ULTRA-REAL	3	545K	66.7K	48	.364	.348	**.387**	.161
CAMP-KG	1,800	128M	111	64	.369	.369	.369	**.195**
3-Cycled	3	213K	66.7K	64	.339	.357	.312	.179
Edge-Matched	8	545K	25K	64	.349	.361	.331	.183
Many-Small	1,800	545K	111	58	.356	.364	.344	.187
Real-Subgraphs	1,800	≤545K	111	56	.361	.351	.375	.169
Compute-matched	1,800	128M	83	48	.367	.367	.367	.192
Mixture (real+syn)	1,803	128M	—	60	**.376**	**.370**	.384	.193

4.3 CORPUS-LEVEL CONTROLS

CAMP-KG (1,800 training KGs, ∼128M unique training edges) and ULTRA-REAL (3 KGs, 545K training triples) also differ in topology count, per-graph size, and compute. Using rules and provenance-aware construction throughout, we test six controls at the same 200K-step budget. **3-Cycled** cycles 3 synthetic KGs like ULTRA-REAL. **Edge-Matched** uses 8 synthetic KGs, and **Many-Small** uses 2,000 small synthetic KGs, both totaling ∼545K edges. **Real-Subgraphs** uses 2,000 randomly sized and densified subgraphs of the three real graphs. **Mixture** draws 50/50 batches from the real graphs and CAMP-KG. **Compute-matched** stops CAMP-KG at 48 GPU-hours.

Results. At equal unique edges, more topologies help: 3-Cycled (0.339) < Edge-Matched (0.349) < Many-Small (0.356), which argues against unique-edge count or cycling pattern as the explanation (Table 3). Diversity from real subgraphs improves the aggregate (0.361) but helps source-disjoint graphs less (0.351) than synthetic rules do (0.357–0.369). The mixture is best overall (0.376): it keeps the source-disjoint gain (0.370) and cuts the source-proximal regression to −0.003. The pure synthetic loss on source-derived graphs is thus largely recoverable. These controls are consistent with topology diversity plus rule structure as the active ingredients, but they do not identify a unique mechanism. Per-graph breakdowns and ULTRA's released 4-graph checkpoint are in Appendix H.

5 ANALYSIS AND DISCUSSION

We now test the two predictions of distributional entanglement (§1) and examine which templates matter; limitations are discussed in Appendix A.

5.1 Structural-Distance Diagnostic

Test. If ULTRA-Real is entangled with its sources' statistics, its MRR should fall faster with distance from them than CAMP-KG's. Correlating Δ MRR with distance cannot test this, because Δ contains ULTRA-Real's MRR. We therefore fit, on per-seed observations,

$$\mathrm{MRR}_{g,m,s} = \beta_0 + \beta_1 m + \beta_2 d_g + \beta_3\, m\, d_g + \boldsymbol{\gamma}^\top x_g + u_{c(g)} + \varepsilon_{g,m,s}, \tag{5}$$

where $m{=}1$ for CAMP-KG, d_g is standardized distance, x_g contains $\log|\mathcal{V}|$, $\log|\mathcal{R}|$, and the evaluation setting, and $u_{c(g)}$ is a provenance-cluster effect; uncertainty comes from the wild cluster bootstrap. d_g is the minimum Jensen–Shannon divergence between total-degree histograms of the target's inference graph and each pretraining graph ($\log_2$ bins on a shared support, add-$\frac{1}{2}$ smoothing, canonical triples; Appendix E).

Evidence. The interaction is $\beta_3 = +0.011$ per SD $[+0.004, +0.019]$: ULTRA-Real loses 0.021 MRR per SD of distance and CAMP-KG 0.010 (Fig. 3, right). A relation-graph distance, computed from JS divergences between per-type degree distributions of G_r, gives $\beta_3 = +0.009$ $[+0.002, +0.017]$. Excluding the source-proximal families attenuates the estimate to $+0.007$ $[-0.001, +0.016]$, so a substantial part of the association reflects the contrast between source-derived and other families. Proximity also does not explain every regression. Of the 18 graphs with $\Delta < -0.005$, 14 are source-proximal, but 3 are NELL-derived, and NELL is not a pretraining source. On shortcut-mitigated splits (Shomer et al., 2025), CAMP-KG gains $+0.016$ over ULTRA-Real (Appendix M), which suggests that its gains do not come from popularity shortcuts.

5.2 Representation Probe

The second prediction concerns representations. We train ridge probes on graph-level means of $\mathbf{R}_q$ to predict degree exponent, density, relation-frequency entropy, and relation-graph density. Probes are fitted on the 54 benchmark graphs, 300 real subgraphs, and 300 graphs from an out-of-prior generator, with provenance-grouped 5-fold cross-validation (Appendix F). The statistics are more decodable from ULTRA-Real (mean R^2 0.56) than from CAMP-KG (0.32); a randomly initialized model reaches 0.22. Conversely, whether a real relation heads a high-confidence AnyBURL composition rule is more decodable from CAMP-KG (AUROC 0.74) than from ULTRA-Real (0.66). These results are consistent with lower entanglement, but they remain correlational: lower decodability does not prove lower reliance.

5.3 Template Sufficiency and a Second Backbone

Template ablations keep the per-KG template-count distribution fixed (Appendix J, Table A6). Removing composition (T4) causes the largest drop (-0.020), consistent with multi-hop reasoning in link prediction (Zhu et al., 2021; Chen et al., 2021a). L1+L2 comes within 0.003 of the full catalog, so L3 is nearly redundant, as expected for compositional variants. The leave-one-out drops sum to 0.049, close to the rules gap (0.053); the effects are thus roughly additive and do not show a "breadth" benefit beyond the individual templates. With TRIX (Zhang et al., 2024) as a second backbone (3 seeds, 15 representative graphs; Appendix L), synthetic pretraining changes aggregate MRR by $+0.006$ and source-disjoint MRR by $+0.019$, mirroring ULTRA. Our conclusions therefore cover relation-graph KGFMs, and random-walk models such as Flock remain untested.

6 Conclusion

This study tests whether zero-shot KG reasoning can be improved by replacing ULTRA's real-KG pretraining with a factorized synthetic corpus and provenance-aware supervision, while holding architecture, optimization, and inference fixed. On the 54-graph suite, synthetic pretraining matches real-KG pretraining overall (0.369 vs. 0.364 MRR), improving provenance-disjoint graphs ($+0.020$) but regressing on source-derived graphs (-0.018); mixing real and synthetic data largely removes this regression. Factorial results show that gains require premise-aware supervision, while rules and statistical diversity contribute additively. Distance degradation and representation probes suggest distributional entanglement. Future work should explore other KGFMs, richer rules, negation, and higher-arity patterns.

AI USE STATEMENT

In this work, we used a generative AI tool (ChatGPT, OpenAI) for three purposes: (i) assisting with the writing and language editing of the manuscript; (ii) assisting with the literature survey, such as identifying and summarizing potentially relevant prior work; and (iii) debugging and fixing bugs in parts of the experimental code. We have not used generative AI tools for research ideation, the design of the method (including the factorized generator, the rule template catalog, and the provenance-aware task construction), the design or execution of experiments, the generation or modification of any data or experimental results, statistical analysis, or the creation of figures and tables. We have reviewed all AI-assisted work. All AI-assisted text was checked by the authors for technical accuracy and consistency with our experimental results. Every reference identified with AI assistance was manually verified against the original publication before citation. All AI-suggested code changes were manually reviewed and tested by the authors, and all reported numbers were recomputed from the raw evaluation logs with our verification script (Appendix O). We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies zero-shot knowledge graph reasoning with procedurally generated pretraining data; it involves no human subjects and no new data collection, and the synthetic corpus contains no real-world entities. Evaluation uses standard benchmark KGs from prior work. KG completion models can infer plausible but false edges, amplify biases or omissions already present in a target KG, and could be misused for profiling or inferring sensitive relationships. Because CAMP-KG is purely structural and does not verify relation semantics, its predictions should not be used for high-stakes decisions without domain-expert review, provenance checks, and downstream validation. A fuller discussion of positive and negative impacts is given in Appendix P.

REPRODUCIBILITY STATEMENT

The generative model, factorized generator, and provenance-aware task construction are specified in §3.2–§3.3, Appendix N, and Algorithm A1, including all priors, the relation-slot allocation, rejection thresholds, and post-rejection dependence diagnostics; corpus statistics are in Appendix G.2. Training and evaluation reuse the official ULTRA codebase and released pretraining configuration (commit and configuration in Appendix O) with only the data pipeline replaced; all benchmark graphs are loaded with the official loaders, and a provenance manifest with split hashes is provided (Appendix B). Statistical procedures are in Appendix C. Anonymous code is available at `https://anonymous.4open.science/r/TFM-23B8`; the raw evaluation logs, the per-seed, per-graph results CSV, the experiment manifest (config ID, seeds, checkpoint, wall time), and the verification script that regenerates every table are provided as supplementary material.

REFERENCES

Arvindh Arun, Sumit Kumar, Mojtaba Nayyeri, Bo Xiong, Ponnurangam Kumaraguru, Antonio Vergari, and Steffen Staab. SEMMA: A semantic aware knowledge graph foundation model. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Bootstrap-based improvements for inference with clustered errors. *The Review of Economics and Statistics*, 90(3):414–427, 2008.

Jiajun Chen, Huarui He, Feng Wu, and Jie Wang. Topology-aware correlations between relations for inductive link prediction in knowledge graphs. In *AAAI Conference on Artificial Intelligence*, 2021a.

Yihong Chen, Pasquale Minervini, Sebastian Riedel, and Pontus Stenetorp. Relation prediction as an auxiliary training objective for improving multi-relational graph representations. In *Conference on Automated Knowledge Base Construction (AKBC)*, 2021b.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. Transformers as soft reasoners over language. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2020.

Yuanning Cui, Zequn Sun, and Wei Hu. A prompt-based knowledge graph foundation model for universal in-context reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *AAAI Conference on Artificial Intelligence*, 2018.

Boyang Ding, Quan Wang, Bin Wang, and Li Guo. Improving knowledge graph embedding using simple constraints. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

Enjun Du, Siyi Liu, and Yongqi Zhang. GraphOracle: Efficient fully-inductive knowledge graph reasoning via relation-dependency graphs. In *AAAI Conference on Artificial Intelligence*, 2026.

Dmitry Eremeev, Oleg Platonov, Gleb Bazhenov, Artem Babenko, and Liudmila Prokhorenkova. GraphPFN: A prior-data fitted graph foundation model. In *International Conference on Machine Learning (ICML)*, 2026.

Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. Fast rule mining in ontological knowledge bases with AMIE+. *The VLDB Journal*, 24(6):707–730, 2015.

Mikhail Galkin, Max Berrendorf, and Charles Tapley Hoyt. An open challenge for inductive link prediction on knowledge graphs. *arXiv preprint arXiv:2203.01520*, 2022.

Mikhail Galkin, Xinyu Yuan, Hesham Mostafa, Jian Tang, and Zhaocheng Zhu. Towards foundation models for knowledge graph reasoning. In *International Conference on Learning Representations (ICLR)*, 2024a.

Mikhail Galkin, Jincheng Zhou, Bruno Ribeiro, Jian Tang, and Zhaocheng Zhu. A foundation model for zero-shot logical query reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Jianfei Gao, Yangze Zhou, Jincheng Zhou, and Bruno Ribeiro. Double equivariance for inductive link prediction for both new nodes and new relation types. In *NeurIPS 2023 Workshop on New Frontiers in Graph Learning (GLFrontiers)*, 2023.

Yuxia Geng, Jiaoyan Chen, Jeff Z. Pan, Mingyang Chen, Song Jiang, Wen Zhang, and Huajun Chen. Relational message passing for fully inductive knowledge graph completion. In *IEEE International Conference on Data Engineering (ICDE)*, 2023.

Lingbing Guo, Zhongpu Bo, Zhuo Chen, Yichi Zhang, Jiaoyan Chen, Yarong Lan, Mengshu Sun, Zhiqiang Zhang, Yangyifei Luo, Qian Li, Qiang Zhang, Wen Zhang, and Huajun Chen. MKGL: Mastery of a three-word language. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. Spotlight.

Takuo Hamaguchi, Hidekazu Oiwa, Masashi Shimbo, and Yuji Matsumoto. Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

Daniel S. Himmelstein, Antoine Lizee, Christine Hessler, Leo Brueggeman, Sabrina L. Chen, Dexter Hadley, Ari Green, Pouya Khankhanian, and Sergio E. Baranzini. Systematic integration of biomedical knowledge prioritizes drugs for repurposing. *eLife*, 6:e26726, 2017.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations (ICLR)*, 2023.

Yin Hua, Zhiqiang Liu, Mingyang Chen, Zheng Fang, Chi Man Wong, Lingxiao Li, Chi Man Vong, Huajun Chen, and Wen Zhang. Beyond completion: A foundation model for general knowledge graph reasoning. In *Findings of the Association for Computational Linguistics: ACL*, 2025.

Xingyue Huang, Miguel Romero, İsmail İlkan Ceylan, and Pablo Barceló. A theory of link prediction via relational Weisfeiler-Leman on knowledge graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Xingyue Huang, Pablo Barceló, Michael M. Bronstein, İsmail İlkan Ceylan, Mikhail Galkin, Juan L. Reutter, and Miguel Romero Orth. How expressive are knowledge graph foundation models? In *International Conference on Machine Learning (ICML)*, 2025.

Jinwoo Kim, Xingyue Huang, Krzysztof Olejniczak, Kyungbin Min, Michael Bronstein, Seunghoon Hong, and İsmail İlkan Ceylan. Flock: A knowledge graph foundation model via learning on random walks. In *International Conference on Learning Representations (ICLR)*, 2026.

Jaejun Lee, Chanyoung Chung, and Joyce Jiyoung Whang. InGram: Inductive knowledge graph embedding via relation graphs. In *International Conference on Machine Learning (ICML)*, 2023.

Shuwen Liu, Bernardo Cuenca Grau, Ian Horrocks, and Egor V. Kostylev. INDIGO: GNN-based inductive knowledge graph completion using pair-wise encoding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *International Conference on Learning Representations (ICLR)*, 2024.

Xin Lv, Xu Han, Lei Hou, Juanzi Li, Zhiyuan Liu, Wei Zhang, Yichi Zhang, Hao Kong, and Suhui Wu. Dynamic anticipation and completion for multi-hop reasoning over sparse knowledge graph. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. YAGO3: A knowledge base from multilingual Wikipedias. In *Conference on Innovative Data Systems Research (CIDR)*, 2015.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. Commonsense knowledge base completion with structural and semantic context. In *AAAI Conference on Artificial Intelligence*, 2020.

Christian Meilicke, Manuel Fink, Yanjie Wang, Daniel Ruffinelli, Rainer Gemulla, and Heiner Stuckenschmidt. Fine-grained evaluation of rule- and embedding-based systems for knowledge graph completion. In *International Semantic Web Conference (ISWC)*, 2018.

Christian Meilicke, Melisachew Wudage Chekol, Daniel Ruffinelli, and Heiner Stuckenschmidt. Anytime bottom-up rule learning for knowledge graph completion. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2019.

Michael Molloy and Bruce Reed. A critical point for random graphs with a given degree sequence. *Random Structures & Algorithms*, 6(2–3):161–180, 1995.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do Bayesian inference. In *International Conference on Learning Representations (ICLR)*, 2022.

Meng Qu, Junkun Chen, Louis-Pascal Xhonneux, Yoshua Bengio, and Jian Tang. RNNLogic: Learning logic rules for reasoning on knowledge graphs. In *International Conference on Learning Representations (ICLR)*, 2021.

Ali Sadeghian, Mohammadreza Armandpour, Patrick Ding, and Daisy Zhe Wang. DRUM: End-to-end differentiable rule mining on knowledge graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

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

Harry Shomer, Jay Revolinsky, and Jiliang Tang. Towards better benchmark datasets for inductive knowledge graph completion. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2025.

Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L. Hamilton. CLUTRR: A diagnostic benchmark for inductive reasoning from text. In *Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, 2019.

Koustuv Sinha, Shagun Sodhani, Joelle Pineau, and William L. Hamilton. Evaluating logical generalization in graph neural networks. *arXiv preprint arXiv:2003.06560*, 2020.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, 2021.

Komal Teru, Etienne Denis, and William Hamilton. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning (ICML)*, 2020.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. Representing text for joint embedding of text and knowledge bases. In Lluís Màrquez, Chris Callison-Burch, and Jian Su (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1499–1509, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1174. URL https://aclanthology.org/D15-1174/.

Petar Veličković, Adrià Puigdomènech Badia, David Budden, Razvan Pascanu, Andrea Banino, Misha Dashevskiy, Raia Hadsell, and Charles Blundell. The CLRS algorithmic reasoning benchmark. In *International Conference on Machine Learning (ICML)*, 2022.

Yuhuai Wu, Markus N. Rabe, Wenda Li, Jimmy Ba, Roger B. Grosse, and Christian Szegedy. LIME: Learning inductive bias for primitives of mathematical reasoning. In *International Conference on Machine Learning (ICML)*, 2021.

Yuhuai Wu, Felix Li, and Percy Liang. Insights into pre-training via simpler synthetic tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Lianghao Xia, Ben Kao, and Chao Huang. OpenGraph: Towards open graph foundation models. In *Findings of the Association for Computational Linguistics: EMNLP*, 2024.

Wenhan Xiong, Thien Hoang, and William Yang Wang. DeepPath: A reinforcement learning method for knowledge graph reasoning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017.

Fan Yang, Zhilin Yang, and William W. Cohen. Differentiable learning of logical rules for knowledge base reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Yucheng Zhang, Beatrice Bevilacqua, Mikhail Galkin, and Bruno Ribeiro. TRIX: A more expressive model for zero-shot domain transfer in knowledge graphs. In *Learning on Graphs Conference (LoG)*, 2024.

Jincheng Zhou, Beatrice Bevilacqua, and Bruno Ribeiro. A multi-task perspective for link prediction with new relation types and nodes. *arXiv preprint arXiv:2307.06046*, 2023.

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

A LIMITATIONS

Architecture scope. We test two relation-graph KGFMs (ULTRA; TRIX on a reduced suite, Appendix L). Random-walk (Flock) and semantic KGFMs may respond differently to synthetic pretraining.

Pipeline, not corpus alone. The headline comparison changes both corpus and supervision. The factorial (§4.1) separates them, but real+provenance supervision relies on mined rules, which are an imperfect analogue of generator provenance.

Template completeness and rule form. The catalog contains five distinct Horn patterns plus two compositional variants. Rules are non-recursive, each head belongs to one template, and rules are hard by default. Negation, counting, recursion, and higher-arity relations are absent. A soft-rule variant helps slightly (Table A7).

Expressivity ceiling. Some regressions may reflect conditional message-passing limits rather than entanglement (Huang et al., 2023; 2025).

Source-proximal tradeoff. Pure synthetic pretraining loses 0.018 MRR on the 22 source-derived graphs. The real+synthetic mixture recovers most of this loss (Table 3) and is our recommendation for practitioners.

Distance and probes. Degree-based and relation-graph distances omit higher-order motifs. The interaction estimate weakens when source-proximal families are excluded. Probes measure decodability, not reliance.

Degree-matched control. Corpus B preserves the in/out-degree sequences of head relations but not every local property, such as clustering around rule participants.

Residual confounds. Topology count within *real* data is tested only through subgraphs of three graphs. Benchmark families remain dependent even after clustering over 11 provenance clusters.

B BENCHMARK PROVENANCE AND FULL PER-GRAPH RESULTS

We use exactly the 54 zero-shot graphs of the ULTRA suite, loaded with the official loaders (commit in Appendix O). FB15k-237, WN18RR, and CoDEx-M are pretraining graphs of ULTRA-REAL and are excluded. Table A1 lists, for every graph, its setting, provenance family, source-proximal flag, and statistics (computed by the loaders), together with the *published* ULTRA(3g) MRR, our reproduction, CAMP-KG, Δ, and the per-graph seed SD of Δ. The supplementary manifest adds SHA-256 hashes of every split file. *Families* are defined strictly by source provenance. HM and Indigo-BM are built from WordNet11 (Hamaguchi et al., 2017; Liu et al., 2021) and therefore belong to the WordNet family. FBNELL mixes FB15k-237 and NELL-995; it stays in the ISDEA family and is flagged source-proximal. All numbers are regenerated from raw evaluation logs. The per-graph Δ distribution is unimodal (Hartigan dip test p=0.46).

Table A1: **Benchmark provenance and full per-graph results.** The 54 zero-shot graphs of the official ULTRA suite, grouped by setting (T: transductive; E: inductive-e; ER: inductive-e, r). Fam.: provenance family (singletons in italics). Prox.: source-proximal. EH: external-hard (lowest quartile by published ULTRA(3g) MRR). Statistics follow the official ULTRA dataset tables: training-graph statistics for T, and test/inference-graph statistics for E/ER (canonical relations; held-out prediction triples are not included in the "Triples" column). Publ.: published ULTRA(3g) MRR; Repro./CAMP-KG: 5-seed means; SD: seed SD of Δ.

#	Graph	Set.	Fam.	Prox.	EH	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{R}\rvert$	Triples	Publ.	Repro.	CAMP-KG	Δ	SD
1	CoDEx-S	T	WD	✓		2,034	42	32,888	.472	.469	.443	−.026	.005
2	CoDEx-L	T	WD	✓		77,951	69	551,193	.338	.336	.331	−.005	.004
3	NELL-995	T	NELL			74,536	200	149,678	.406	.409	.397	−.012	.005
4	YAGO3-10	T	*YAGO*			123,182	37	1,079,040	.451	.455	.448	−.007	.004
5	WDsinger	T	WD			10,282	135	16,142	.382	.386	.389	+.003	.006
6	NELL23k	T	NELL		✓	22,925	200	25,445	.239	.240	.265	+.025	.006
7	FB237_10	T	FB	✓	✓	11,512	237	27,211	.248	.247	.253	+.006	.004
8	FB237_20	T	FB	✓		13,166	237	54,423	.272	.270	.267	−.003	.004

Continued on next page

Table A1: **Benchmark provenance and full per-graph results** (continued)

#	Graph	Set.	Fam.	Prox.	EH	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{R}\rvert$	Triples	Publ.	Repro.	CAMP-KG	Δ	SD
9	FB237_50	T	FB	✓		14,149	237	136,057	.324	.323	.319	−.004	.003
10	DBpedia100k	T	*DBp*			99,604	470	597,572	.398	.400	.396	−.004	.005
11	AristoV4	T	*Aristo*		✓	44,949	1,605	242,567	.182	.181	.209	+.028	.007
12	ConceptNet100k	T	*CN*		✓	78,334	34	100,000	.082	.082	.118	+.036	.008
13	Hetionet	T	*Hetio*		✓	45,158	24	2,025,177	.257	.258	.291	+.033	.007
14	FB v1	E	FB	✓		1,093	180	1,993	.498	.494	.456	−.038	.007
15	FB v2	E	FB	✓		1,660	200	4,145	.512	.510	.479	−.031	.006
16	FB v3	E	FB	✓		2,501	215	7,406	.491	.489	.455	−.034	.006
17	FB v4	E	FB	✓		3,051	219	11,714	.486	.480	.458	−.022	.005
18	WN v1	E	WN	✓		922	9	1,618	.648	.652	.610	−.042	.009
19	WN v2	E	WN	✓		2,757	10	4,011	.663	.661	.620	−.041	.008
20	WN v3	E	WN	✓		5,084	11	6,327	.376	.374	.346	−.028	.007
21	WN v4	E	WN	✓		7,084	9	12,334	.611	.612	.573	−.039	.007
22	NELL v1	E	NELL			225	14	833	.712	.706	.685	−.021	.012
23	NELL v2	E	NELL			2,086	88	4,586	.567	.561	.575	+.014	.006
24	NELL v3	E	NELL			3,566	142	8,048	.563	.558	.560	+.002	.005
25	NELL v4	E	NELL			2,795	76	7,073	.469	.465	.483	+.018	.007
26	ILPC-small	E	WD			6,653	48	20,960	.302	.296	.311	+.015	.005
27	ILPC-large	E	WD			29,246	65	77,044	.290	.285	.290	+.005	.004
28	HM 1k	E	WN	✓	✓	9,899	11	18,638	.059	.052	.067	+.015	.006
29	HM 3k	E	WN	✓	✓	19,218	11	38,285	.037	.038	.050	+.012	.005
30	HM 5k	E	WN	✓	✓	23,792	11	48,425	.034	.032	.050	+.018	.005
31	Indigo-BM	E	WN	✓		14,775	229	250,195	.440	.439	.439	.000	.004
32	FB-25	ER	FB	✓		4,097	216	17,147	.388	.384	.354	−.030	.006
33	FB-50	ER	FB	✓		4,445	205	11,636	.338	.335	.318	−.017	.005
34	FB-75	ER	FB	✓		2,792	186	9,316	.403	.398	.362	−.036	.007
35	FB-100	ER	FB	✓		2,624	77	6,987	.449	.439	.406	−.033	.007
36	WK-25	ER	WD			3,228	74	3,391	.316	.313	.337	+.024	.007
37	WK-50	ER	WD		✓	9,328	93	9,672	.166	.160	.207	+.047	.010
38	WK-75	ER	WD			2,722	65	3,430	.365	.365	.368	+.003	.006
39	WK-100	ER	WD		✓	12,136	37	13,487	.164	.158	.210	+.052	.011
40	NL-0	ER	NELL			2,026	112	2,287	.342	.341	.369	+.028	.008
41	NL-25	ER	NELL			2,146	120	2,230	.395	.392	.396	+.004	.006
42	NL-50	ER	NELL			2,335	119	2,576	.407	.406	.404	−.002	.006
43	NL-75	ER	NELL			1,578	116	1,818	.368	.369	.385	+.016	.007
44	NL-100	ER	NELL			1,709	53	2,378	.471	.471	.462	−.009	.006
45	MT1 tax	ER	MTDEA		✓	10,000	9	16,526	.224	.223	.267	+.044	.009
46	MT1 health	ER	MTDEA			10,000	7	14,110	.298	.300	.331	+.031	.008
47	MT2 org	ER	MTDEA		✓	10,000	11	21,976	.095	.094	.155	+.061	.012
48	MT2 sci	ER	MTDEA		✓	10,000	16	14,852	.258	.255	.283	+.028	.007
49	MT3 art	ER	MTDEA			10,000	45	28,023	.259	.259	.286	+.027	.007
50	MT3 infra	ER	MTDEA			10,000	27	21,646	.619	.616	.647	+.031	.008
51	MT4 sci	ER	MTDEA			10,000	42	12,516	.274	.272	.301	+.029	.007
52	MT4 health	ER	MTDEA			10,000	20	15,337	.624	.632	.665	+.033	.008
53	Metafam	ER	ISDEA		✓	656	28	7,257	.238	.234	.305	+.071	.013
54	FBNELL	ER	ISDEA	✓		4,752	183	10,685	.485	.473	.459	−.014	.006
Summary (MRR; computed from unrounded per-graph values)													
	Mean, all 54								.366	.364	.369	+.005	
	Mean, source-disjoint (32)								.349	.348	.369	+.020	
	Mean, source-proximal (22)								.390	.387	.369	−.018	
	Mean, external-hard (14)								.163	.161	.195	+.034	
	Mean by setting T / E / ER								.312/.431/.345 →		repro. .312/.428/.343		
	H@1 / H@10, all 54								—	.271/.517	.276/.523	+.005/ + .006	

Sources. CoDEx-S/L (Safavi & Koutra, 2020); NELL-995 (Xiong et al., 2017); YAGO3-10 (Mahdisoltani et al., 2015); WDsinger, NELL23k, FB237_10/20/50 (Lv et al., 2020); DBpedia100k (Ding et al., 2018); AristoV4 (Chen et al., 2021b); ConceptNet100k (Malaviya et al., 2020); Hetionet (Himmelstein et al., 2017); FB/WN/NELL v1–v4 (Teru et al., 2020); ILPC (Galkin et al., 2022); HM (Hamaguchi et al., 2017); Indigo-BM (Liu et al., 2021); FB/WK/NL-x (Lee et al., 2023); MT1–MT4 (Zhou et al., 2023); Metafam, FBNELL (Gao et al., 2023). **Families** (6 multi-graph + 5 singleton clusters): FB (11), WN (8; HM and Indigo-BM are built from WordNet11), NELL (11), WD (Wikidata; 9), MTDEA (8), ISDEA (2). **Source-proximal** (22): derived from FB15k-237, WordNet, or CoDEx, including FBNELL. $\lvert\mathcal{R}\rvert$ counts canonical relations. SHA-256 hashes of all split files are in the supplementary manifest. Bold/underline marking of Δ is omitted; the ± 0.005 tie band gives 25 wins, 11 ties, and 18 losses.

C Statistical Analysis

Primary procedure. With B=10,000 replicates, a wild cluster bootstrap (Cameron et al., 2008) with Webb six-point weights perturbs residuals at the level of the 11 provenance clusters (six families plus five singleton graphs). Within each replicate, seeds are resampled with replacement *independently per method*, because seeds are not paired across different data streams. A linear mixed model with random effects for cluster, graph nested in cluster, and method×seed gives $+0.005$ $[-0.007, +0.016]$ for the aggregate difference, in agreement with the bootstrap.

Non-inferiority margin. δ=0.01 was fixed before CAMP-KG evaluation as the smallest aggregate change we consider practically relevant (≈3% of baseline MRR). Table A2 reports all margins.

Table A2: Non-inferiority of CAMP-KG vs. ULTRA-Real (all 54 graphs) under alternative margins; one-sided 95% lower bounds.

Procedure	Lower bound	δ=0.005	δ=0.01 / 0.02
Wild cluster bootstrap (primary)	-0.006	not established	✓ / ✓
Mixed model	-0.005	borderline	✓ / ✓
Graph-level bootstrap (ignores clusters)	-0.002	✓	✓ / ✓

Per-graph consistency. CAMP-KG has higher MRR on 30/54 graphs (exact one-sided sign test p=0.25; two-sided 0.50). With a ± 0.005 tie band the count is 25 wins, 11 ties, and 18 losses. The mean gain over the 30 improved graphs is $+0.025$; the mean change over the 23 graphs with $\Delta<0$ is -0.022 (-0.027 over the 18 with $\Delta<-0.005$).

Slices. Source-disjoint: $+0.020$ $[+0.009, +0.032]$; source-proximal: -0.018 $[-0.027, -0.010]$; external-hard: $+0.034$ $[+0.019, +0.050]$; cross-fitted hard slice: $+0.030$ $[+0.015, +0.046]$. 10 of the 14 external-hard graphs are source-disjoint; the other four (HM 1k/3k/5k, FB237_10) are source-proximal. If the slice were instead selected by our own ULTRA-Real MRR (the earlier protocol), the estimate would be $+0.041$; the gap illustrates the regression-to-the-mean bias of baseline-selected slices.

D Cumulative-k Robustness Analysis

With graphs sorted by ascending *published* ULTRA(3g) MRR, the cumulative average Δ is $+0.018$ at k=1, $+0.034$ at k=14, $+0.023$ at k=27, $+0.012$ at k=40, and $+0.005$ at k=54; it is positive for all k. Figure A1 uses the most adversarial ordering, ascending per-graph Δ. That curve starts at the largest regression (-0.042), crosses zero near k≈50, and reaches the overall mean at k=54.

E Distance Definitions and Robustness

Degree distance. For each graph, we take the inference graph's canonical triples and count each triple once at each endpoint (total degree, no inverses). Degrees are binned into $\log_2$ bins on the union support of the two graphs being compared, add-$\frac{1}{2}$ smoothing is applied, and the JS divergence (base 2) is computed. d_g is the minimum over the three pretraining graphs, z-scored across the 54 graphs.

Relation-graph distance. We take the mean JS divergence between the degree distributions of the four interaction types in the binary G_r.

Robustness. Figure A2 and the list below give β_3 of Eq. 5 under alternative definitions:

- FB15k-237-only distance: $+0.012$ $[+0.004, +0.020]$;
- mean-of-3 distance: $+0.011$ $[+0.003, +0.019]$;
- in-/out-degree instead of total degree: $+0.010$ / $+0.011$;

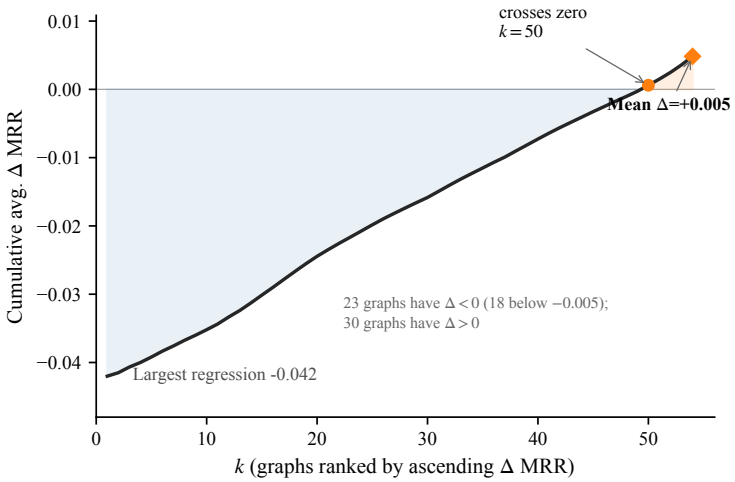

Figure A1: Cumulative average Δ MRR when graphs are sorted by ascending per-graph Δ, the most adversarial ordering for CAMP-KG. The curve starts at the largest regression (-0.042), crosses zero near $k{\approx}50$, and reaches the overall mean $+0.005$ at $k{=}54$.

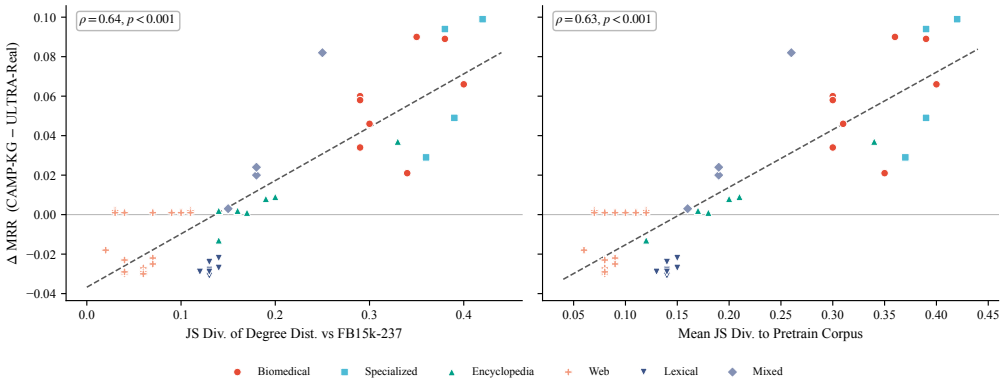

Figure A2: Per-method MRR vs. distance under FB15k-237-only (left) and mean-of-3 (right) degree distances, with fitted slopes and cluster-bootstrap bands.

- cluster-level Spearman (11 clusters) between mean Δ and mean distance: $\rho{=}0.55$, permutation $p{=}0.09$.

The cluster-level result is weaker, as expected with 11 units.

F Representation Probes

For each graph, we average $\mathbf{R}_q$ over all query relations of its test set, using each model's frozen checkpoint. The probe set contains the 54 benchmark graphs, 300 random subgraphs of the real pretraining graphs, and 300 graphs from an out-of-prior generator ($\gamma \in [3, 4]$, $\mu \in [50, 100]$, no rules). Ridge probes use provenance-grouped 5-fold cross-validation with nested regularization selection.

G Ablation Visualization and Corpus Comparability

G.1 Grouped Bar Chart

Figure A3 compares ULTRA-Real and corpora A–D on all 54 graphs, the source-disjoint slice, and the external-hard slice. The ordering $A > C > B > D$ holds on all three slices. In absolute terms,

Table A3: Probe results (cross-validated R^2; AUROC for rule role). Mean over 5 seeds; SD ≤ 0.03.

Target	ULTRA-REAL	CAMP-KG	Random init
Degree exponent	0.61	0.34	0.22
Density	0.58	0.31	0.25
Relation-frequency entropy	0.49	0.29	0.18
Relation-graph density	0.55	0.33	0.24
Heads a composition rule (AUROC)	0.66	0.74	0.55

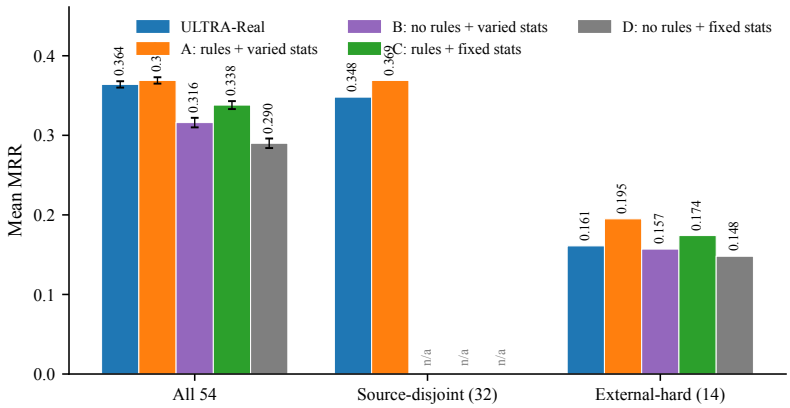

Figure A3: Mean MRR for ULTRA-REAL and corpora A–D on all 54 graphs, source-disjoint, and external-hard slices; error bars are 95% cluster-bootstrap CIs.

the gaps are smaller on the external-hard slice (A−B 0.038 vs. 0.053; A−C 0.021 vs. 0.031; B−D 0.009 vs. 0.026) because all MRRs are lower there. Relative to each slice's mean, A−B is larger on the external-hard slice.

G.2 PRETRAINING CORPUS COMPARABILITY

Table A4 is computed from the loaders. ULTRA-Real's statistics come from the standard training splits (272,115 + 86,835 + 185,584 = 544,534 canonical triples). "Unique edges" counts canonical triples, without inverses, in the training split (ULTRA-REAL) or the observable training graph $B \cup d_{\text{train}} \cup b_{\text{train}} \cup N$ of the 1,800 training KGs (CAMP-KG). The realized mean of 79K edges per synthetic KG (all categories) lies below the nominal $\mathbb{E}[n_{\text{ent}}]\mathbb{E}[\mu] \approx 87.6$K for two reasons: stub matching removes 6.1% of stubs as self-loops or duplicates, and the 500K cap rejects the largest draws. Figure A4 compares realized synthetic statistics with benchmark statistics.

H CORPUS-LEVEL CONTROLS: DETAILS

Construction. **3-Cycled**: three full-generator KGs (68K, 74K, and 71K training edges), visited round-robin. **Edge-Matched**: eight full-generator KGs whose edge budgets are constrained to total 545K training edges (52–83K each). **Many-Small**: 2,000 KGs drawn with $n_{\text{ent}} \sim \text{LogU}(40, 400)$ and $\mu \sim \text{LogU}(2, 6)$ and relaxed filters ($|d_{\text{train}}| \geq 10$, $|Q_{\text{val}}| \geq 5$), totaling 545K edges. **Real-Subgraphs**: 2,000 subgraphs of FB15k-237, WN18RR, and CoDEx-M, obtained by random-walk entity sampling with target size $\sim \text{LogU}(100, 30\text{K})$ and random edge retention $\sim \text{U}(0.3, 1)$, using standard supervision. **Mixture**: each step draws from the real graphs or the CAMP-KG corpus with probability 0.5 each, and each source keeps its own supervision. **Compute-matched**: CAMP-KG stopped at 150K steps (48 GPU-hours).

ULTRA(4g). We also evaluate ULTRA's released 4-graph checkpoint (which adds NELL-995) on the 53 graphs excluding NELL-995: 0.371 (source-disjoint: 0.353). More *real* diversity thus

Table A4: **Pretraining corpus comparability** (computed from the loaders). Both arms use the same model, optimizer, loss, 200K steps, and inference. Unique edges: canonical training triples without inverses.

Metric	CAMP-KG	ULTRA-REAL
Training KGs (+ selection KGs)	1,800 (+200)	3
Geom. mean entities / KG	1,730	21,625
Geom. mean canonical relations / KG	38	51
Mean triples / KG (all categories)	79,000	181,511
Total unique training edges	**∼128 M**	**544,534**
Mean training queries / KG[a]	21,300	181,511
Training steps	200,000	200,000
Avg. visits per KG	111	∼66,700
Avg. epochs of queries / KG[b]	∼0.33	∼23.5
Trainable parameters	168 K	168 K
GPUs	2×H100	2×H100
GPU-hours / seed[c]	64	48
Corpus generation	45 min wall (≈6 CPU-h)	N/A

[a]In ULTRA every training triple is a query; for CAMP-KG, $|d_{\text{train}} \cup b_{\text{train}} \cup N|$.

[b](visits × 64) / queries per KG: $111 \times 64/21{,}300 \approx 0.33$; $66{,}700 \times 64/181{,}511 \approx 23.5$.

[c]Difference due to per-KG cache loading; see the compute-matched control in Table 3.

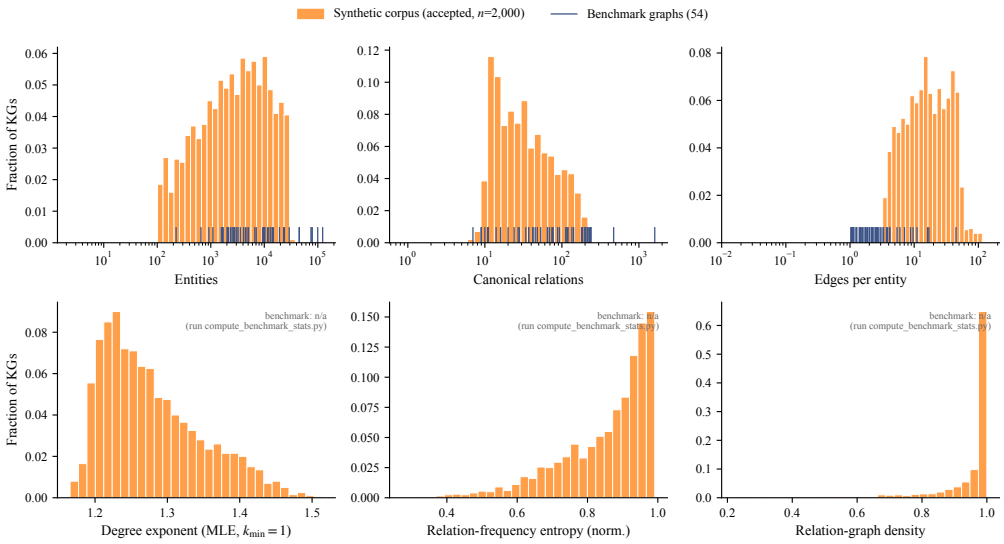

Figure A4: Histograms of realized statistics (entities, canonical relations, edges/entity, degree exponent, relation-frequency entropy, relation-graph density) for the accepted synthetic corpus (bars) and the 54 benchmark graphs (ticks).

helps source-disjoint graphs less than the synthetic corpus does (0.369). The 50-graph checkpoint is pretrained on most benchmark graphs and is not zero-shot.

Interpretation. Together, the controls suggest that (a) at equal unique edges, more topologies help; (b) real-subgraph diversity improves the aggregate but not the source-disjoint slice; (c) synthetic rules add source-disjoint gains beyond topology count; and (d) mixing real and synthetic data keeps these gains while removing most source-proximal losses. Per-graph control results are in the supplementary CSV.

Table A5: **Template catalog.** Templates are included independently per KG with level probabilities 0.70/0.62/0.44 (expected 4.2 per KG; composition is the fallback). Freq.: realized fraction of the 2,000 accepted KGs. T6 and T7 are compositional variants of T4 under inverse augmentation.

ID	Lvl	Name	Formula	Example	Freq.
T1	L1	Symmetry	$r(x,y) \to r(y,x)$	friend$(A,B) \to$friend(B,A)	70%
T2	L1	Inversion	$r_1(x,y) \to r_2(y,x)$	parent$(A,B) \to$child(B,A)	70%
T3	L1	Subsumption	$r_1(x,y) \to r_2(x,y)$	capital_of$(x,y) \to$city_in(x,y)	70%
T4	L2	Composition	$r_1(x,z) \wedge r_2(z,y) \to r_3(x,y)$	parent$(A,B) \wedge$parent(B,C) $\to$ grandparent(A,C)	63%
T5	L2	Intersection	$r_1(x,y) \wedge r_2(x,y) \to r_3(x,y)$	coauthor $\wedge$ colleague $\to$ collaborator	62%
T6	L3	Long Chain	$r_1(x,a) \wedge r_2(a,b) \wedge r_3(b,y) \to r_4(x,y)$	lives_in $\wedge$ part_of $\wedge$ part_of $\to$ located_in	44%
T7	L3	Branching	$r_1(x,z) \wedge r_2(y,z) \to r_3(x,y)$	studies_at$(A,U) \wedge$studies_at(B,U) $\to$ schoolmate(A,B)	44%

I SOURCE×SUPERVISION FACTORIAL: DETAILS

For **Real+provenance**, AnyBURL (Meilicke et al., 2019) (100 s per graph, rule length $\leq$3) is run on each real training split. Rules with confidence $\geq$0.8 and support $\geq$10 are kept (1,240 rules in total). Each triple predicted by a kept rule and present in the training split becomes a *derived* query; its body groundings are marked as premises, which stay observable and are never targets. All other training triples are *background*. Derived triples cover 23% of the training triples. Batches are mixed 50/50, and selection uses the real validation splits, as in ULTRA. For **Synthetic+standard**, all categories of each synthetic KG are pooled and randomly split 80/10/10, queries are sampled uniformly from the training split, and selection uses the same 200 held-out KGs.

J TEMPLATE CATALOG AND TEMPLATE ABLATION

J.1 FULL TEMPLATE CATALOG AND LIFTED SIGNATURES

Table A5 lists the templates. Lifted signatures on the augmented relation graph are as follows. **T1** symmetry: r and r^{-1} share both endpoint sets, so all four interactions appear between them; this signature is not unique to symmetry. **T2** inversion: r_2 shares heads with r_1^{-1} and tails with r_1^{-1}'s tails (h2h and t2t between r_2 and r_1^{-1}). **T3** subsumption: h2h and t2t between r_1 and r_2. **T4** composition $r_1(x,z) \wedge r_2(z,y) \to r_3(x,y)$: t2h between r_1 and r_2, h2h between r_3 and r_1, and t2t between r_3 and r_2. **T5** intersection: pairwise h2h and t2t among r_1, r_2, r_3. **T6** long chain: t2h for $r_1 \to r_2$ and $r_2 \to r_3$, h2h between r_4 and r_1, and t2t between r_4 and r_3; this is *not* a chain $r_1 \to r_2 \to r_3 \to r_4$ in G_r. **T7** branching: t2t between r_1 and r_2, h2h between r_3 and r_1, and t2h from r_3 to r_2. This equals T4 with r_2 replaced by r_2^{-1}. Because ULTRA's G_r is binary, a signature is distinctive only when G_r is sparse. Binary G_r has median per-type density 0.31 (IQR 0.12–0.58) across synthetic KGs and 0.18 across the benchmark graphs, and it is dense ($>$0.8) in 9% of synthetic KGs. In those KGs, rule signal is carried mainly by GNN_e path structure conditioned on $\mathbf{R}_q$.

Multi-rule interactions. Body relations may be shared across templates. Derived triples are deduplicated before splitting, and all body triples stay observable.

J.2 TEMPLATE ABLATION

Removing templates while re-normalizing inclusion probabilities would change the expected number of templates per KG. We instead keep the per-KG template-count distribution of the full catalog, capped at the number of available templates, and sample templates uniformly from the remaining catalog (Table A6; 3 seeds, paired 95% CIs of Δ). Findings: (i) composition matters most (-0.020); (ii) removing all of L1 (-0.008) costs more than removing any single L1 template, and the level ablations are monotone; (iii) L1+L2 is within 0.003 of the full catalog, so L3 is nearly redundant;

Table A6: **Template ablation.** The per-KG template-count distribution of the full catalog is held fixed, and templates are drawn uniformly from the remaining catalog. 2,000 KGs, provenance-aware splits; 3 seeds (full: 5). Δ: paired-over-graphs 95% CI vs. full.

Config.	Templates	All 54	Δ vs. Full
Full	T1–T7	.369	—
L1+L2	T1–T5	.366	$-.003$ $[-.006, .000]$
L2+L3	T4–T7	.361	$-.008$ $[-.012, -.004]$
L1 only	T1–T3	.345	$-.024$ $[-.029, -.019]$
−SYM	T2–T7	.364	$-.005$ $[-.008, -.002]$
−INV	T1,T3–T7	.363	$-.006$ $[-.009, -.003]$
−SUB	T1,T2,T4–T7	.365	$-.004$ $[-.007, -.001]$
−COMP	T1–T3,T5–T7	.349	$-.020$ $[-.025, -.015]$
−INTER	T1–T4,T6,T7	.360	$-.009$ $[-.013, -.005]$
−CHAIN3	T1–T5,T7	.367	$-.002$ $[-.005, .001]$
−BRANCH	T1–T6	.366	$-.003$ $[-.006, .000]$

(iv) the leave-one-out drops sum to 0.049, close to the rules gap 0.053, which indicates roughly additive contributions.

K Sensitivity Analysis

Table A7 reports synthetic-validation MRR, which is the selection criterion, next to real 54-graph MRR, which was computed once after freezing. Each configuration is scored on a common validation pool of 200 KGs per configuration, merged across all configurations of an axis, so no configuration is favored by its own validation distribution. Corpus size uses a rule fixed in advance: the smallest size within 0.005 synthetic-validation MRR of the largest size tested. The split ablation separates the factors that the earlier naive split confounded: (i) body observable with a pooled 80/10/10 split of derived and background triples; (ii) category-specific splits with body triples also used as targets; (iii) fully naive. Rows (e)–(g) cover the filter set, a soft-rule axis in which each derived triple is independently replaced by a random edge with exception rate $\epsilon \sim \mathrm{U}(0, 0.3)$, and the old protocol that kept noise in Q_{val}.

L Second Backbone: TRIX

We train TRIX (Zhang et al., 2024) with its released configuration on the real graphs, on the CAMP-KG corpus, and on corpus D (3 seeds). Evaluation uses 15 graphs, five per setting, chosen before training to span families: 9 source-disjoint and 6 source-proximal.

M Shortcut-Mitigated Evaluation

We evaluate on the six inductive datasets re-partitioned by Shomer et al. (2025) to remove PPR/popularity shortcuts (Table A9). The gain of CAMP-KG is larger on these splits than on the original splits of the same sources, and the heuristic baseline drops sharply. This suggests that CAMP-KG's gains are not driven by shortcut exploitation.

N Generator Specification, Pseudocode, and Diagnostics

Generative model. $p(T) = \prod_{i=1}^{7} \mathrm{Bern}(p_i)$, with $T{=}\{T_4\}$ if the draw is empty. $p(S)$ is the product of the seven priors in §3.2, with γ drawn from the mixture $0.15\,\delta_{\text{uniform}} + 0.85\,\mathrm{U}(1.5, 3)$. Given (T, S), the generator proceeds in five steps. (1) *Relations.* $\mathcal{R}_{\mathrm{head}}$ has one head per template, the non-head pool $\mathcal{R}_0$ has $|\mathcal{R}_0|{=}n_{\mathrm{rel}}$, and $\mathcal{R} = \mathcal{R}_{\mathrm{head}} \cup \mathcal{R}_0$. Each template draws its k_τ body relations uniformly without replacement from $\mathcal{R}_0$ ($k_\tau \le 3 \le n_{\mathrm{rel}}$, so allocation is always feasible); symmetry uses its head as its own body. Non-head relations used by no template form $\mathcal{R}_{\mathrm{bg}}$. (2) *Budgets.*

Table A7: **Sensitivity analysis.** Syn. val: mean validation MRR on a common pool of held-out synthetic KGs (the *selection* criterion). Real: 54-graph MRR, computed once after freezing. Defaults in bold; for every axis, the default is the synthetic-validation argmax. 3 seeds (default: 5).

Axis	Setting	Syn. val	Real 54
(a) Query mix	0:100	.482	.318
(derived:non-derived)	30:70	.519	.359
	50:50	**.531**	**.369**
	70:30	.527	.365
	100:0	.498	.340
(b) Corpus size	250	.497	.345
	500	.511	.356
	1,000	.523	.364
	2,000	**.531**	**.369**
	4,000	.534	.371
(c) Prior breadth	Narrow ($\gamma \in [1.5, 2]$, $\mu \in [3, 8]$)	.516	.357
	Broad	**.531**	**.369**
(d) Split	**Provenance-aware**	**.531**	**.369**
	(i) body observable, pooled 80/10/10 split	.522	.362
	(ii) category splits, body also targets	.514	.358
	(iii) fully naive 80/10/10	.505	.352
(e) Filter set	without η-dropped triples	.527	.366
(f) Rule form	soft rules, $\epsilon \sim \mathrm{U}(0, 0.3)$	.526	.371
(g) Selection	noise triples kept in Q_{val} (old)	.529	.367

Corpus size uses a rule fixed in advance: the smallest size within 0.005 Syn. val of the largest tested (2,000). Split variants (i) and (ii) separate premise visibility from split ratios, which the earlier naive-split ablation confounded.

Table A8: Second backbone (15 graphs, 3 seeds, 95% cluster-bootstrap CIs of Δ).

Backbone	Real	CAMP-KG	Δ all	Δ disjoint / proximal
ULTRA	.371	.377	$+.006$	$+.021$ / $-.016$
TRIX	.392	.398	$+.006$	$+.019$ / $-.014$
TRIX, corpus D	—	.331	$-.061$	—

$M{=}\mathrm{round}(n_{\mathrm{ent}}\mu)$ non-derived edges, where μ is a *pre-rule* budget per entity (mean out-degree, so mean total degree is 2μ). Relation frequencies are $\pi \sim \mathrm{Dir}(\kappa\mathbf{1})$ over $\mathcal{R}_0 \cup \{\text{symmetry head}\}$, and $(M_r) \sim \mathrm{Mult}(M, \pi)$. (3) *Edges.* Shared entity out- and in-activity sequences $a_v^{\mathrm{out}}, a_v^{\mathrm{in}}$ are drawn i.i.d. from a discrete power law $P(k) \propto k^{-\gamma}$ on $\{1, \ldots, \lceil\sqrt{M}\rceil\}$, or set to 1 under the uniform option. For each relation r, M_r head stubs and M_r tail stubs are assigned to entities in proportion to a^{out} and a^{in} and paired uniformly at random (directed stub matching (Molloy & Reed, 1995)). Self-loops and duplicate triples are removed, not resampled, which removes 6.1% of stubs on average. Edges of relations used as template bodies are *body* triples, and edges of $\mathcal{R}_{\mathrm{bg}}$ are *background* triples. (4) *Derivation.* For each template, all groundings over body triples are computed with hash joins. The number of rule instances is thus determined by the body graph rather than sampled. Because heads are never bodies, one pass yields the full closure. Per template, the derived set is capped at 50K by uniform subsampling; the cap is active in 4.3% of KGs, and subsampled-away conclusions enter the negative-sampling filter. (5) *Incompleteness and noise.* An η fraction of derived triples is dropped uniformly; dropped triples are true and are filtered as negatives. $\lfloor \nu\,|\text{background}| \rfloor$ noise triples are sampled uniformly from $\mathcal{V} \times \mathcal{R}_0 \times \mathcal{V}$, excluding existing triples. *Acceptance* $\mathrm{Acc}(\mathcal{G})$ is given in Step 8 of Algorithm A1.

Corpus B (no rules). For each head relation, its derived edges are rewired by $10\,|E_r|$ degree-preserving directed double-edge swaps. This keeps the in/out-degree sequence and edge count of every head relation and destroys rule correlation: the fraction of rewired edges still supported by a rule grounding falls to 0.4%. The provenance-aware split treats rewired edges as "derived" placeholders.

Table A9: Shortcut-mitigated splits (mean MRR, 5 seeds).

Splits	PPR/popularity heuristic	ULTRA-REAL	CAMP-KG
Original splits (same sources)	.236	.392	.401 (+.009)
Shortcut-mitigated	.108	.231	.247 (+.016)

Algorithm A1 CAMP-KG: Factorized Synthetic KG Generator

Require: Templates $\{T_1, \ldots, T_7\}$ with inclusion probabilities p_i; priors for $n_{\text{ent}}, n_{\text{rel}}, \kappa, \gamma, \mu, \eta, \nu$.
Ensure: A synthetic KG with provenance-aware splits and cached relation graphs.

// Step 1: Sample statistics S (independently of T)
$n_{\text{ent}} \leftarrow \text{LogU}(100, 30000)$; $n_{\text{rel}} \leftarrow \text{LogU}(6, 192)$; $\kappa \leftarrow \text{LogU}(0.1, 10)$
$\gamma \leftarrow \text{U}(1.5, 3.0)$, or uniform degree w.p. 0.15
$\mu \leftarrow \text{LogU}(3, 50)$; $\eta \leftarrow \text{U}(0, 0.5)$; $\nu \leftarrow \text{U}(0, 0.10)$
// Step 2: Sample templates T
$\mathcal{S} \leftarrow \{T_i : \text{Bern}(p_i) = 1\}$; **if** $\mathcal{S} = \emptyset$ **then** $\mathcal{S} \leftarrow \{T_4\}$
// Step 3: Relation slots (always feasible)
Create $|\mathcal{S}|$ fresh head relations $\mathcal{R}_{\text{head}}$ and n_{rel} non-head relations $\mathcal{R}_0$
For each template draw body relations uniformly from $\mathcal{R}_0$; $\mathcal{R}_{\text{bg}} \leftarrow$ unused part of $\mathcal{R}_0$
// Step 4: Non-derived edges
$M \leftarrow \text{round}(n_{\text{ent}}\mu)$; $\pi \leftarrow \text{Dir}(\kappa\mathbf{1})$; $(M_r) \leftarrow \text{Mult}(M, \pi)$
Draw activities $a^{\text{out}}, a^{\text{in}}$ from $\text{PowerLaw}(\gamma)$ or constant
For each r: directed stub matching with M_r stubs; drop self-loops and duplicates
Body $B \leftarrow$ edges of body relations; background $\leftarrow$ edges of $\mathcal{R}_{\text{bg}}$
// Step 5: Derivation (non-recursive, full closure)
For each $T_i \in \mathcal{S}$: all groundings via hash joins; cap at 50K; deduplicate
// Step 6: Incompleteness and noise
Drop η-fraction of derived triples $\rightarrow$ filter set F
$N \leftarrow \lfloor \nu |\text{background}| \rfloor$ uniform triples from $\mathcal{V} \times \mathcal{R}_0 \times \mathcal{V}$ (new only)
// Step 7: Provenance-aware split
Derived $\rightarrow d_{\text{train}}, d_{\text{val}}, d_{\text{test}}$ (60/20/20); background $\rightarrow b_{\text{train}}, b_{\text{val}}, b_{\text{test}}$ (80/10/10)
$E^{\text{train}}_{\text{obs}} \leftarrow \overline{B \cup d_{\text{train}} \cup b_{\text{train}} \cup N}$; $E^{\text{val}}_{\text{obs}} \leftarrow E^{\text{train}}_{\text{obs}}$
$E^{\text{test}}_{\text{obs}} \leftarrow \overline{B \cup d_{\text{train}} \cup b_{\text{train}} \cup N \cup d_{\text{val}} \cup b_{\text{val}}}$
$Q_{\text{train}} \leftarrow d_{\text{train}} \cup b_{\text{train}} \cup N$; $Q_{\text{val}} \leftarrow d_{\text{val}} \cup b_{\text{val}}$; $Q_{\text{test}} \leftarrow d_{\text{test}} \cup b_{\text{test}}$
assert $Q_{\text{val}} \cap E^{\text{val}}_{\text{obs}} = \emptyset$ **and** $Q_{\text{test}} \cap E^{\text{test}}_{\text{obs}} = \emptyset$
// Step 8: Acceptance $\text{Acc}(\mathcal{G})$
if $\text{LCC}(E^{\text{train}}_{\text{obs}}) < 0.5\, n_{\text{ent}}$ **or** $|E^{\text{train}}_{\text{obs}}| > 500\text{K}$ **or** $|d_{\text{train}}| < 100$ **or** $|Q_{\text{val}}| < 50$ **then**
  reject and resample (T, S) from scratch
end if
// Step 9: Cache
$G^{\text{train}}_r, G^{\text{val}}_r, G^{\text{test}}_r \leftarrow \text{LIFT}(E^{\text{train}}_{\text{obs}}), \text{LIFT}(E^{\text{val}}_{\text{obs}}), \text{LIFT}(E^{\text{test}}_{\text{obs}})$; store all sets and F

Post-rejection dependence diagnostics. Table A10 quantifies how much dependence between T and S the acceptance step and rule application introduce. We also generated a *stratified* corpus in which acceptance is equalized across the 7×5 template$\times$statistic-quintile cells by per-cell oversampling. It reaches 0.368 mean MRR, compared with 0.369 for the default corpus, so the residual dependence has no measurable effect.

Computational cost. Generating the 2,000-KG corpus takes 45 ± 5 wall-clock minutes on an 8-core AMD EPYC 7763 ($\approx$6 CPU-hours). The rejection rate is 7.2%, i.e., $\approx$2,155 draws for 2,000 accepted KGs. Cached storage, including relation graphs, is 17 GB.

Table A10: Dependence between templates T and statistics after acceptance (2,000 accepted KGs; 20,000 draws for acceptance rates). AUC: 5-fold gradient-boosted classifier predicting each template's inclusion; chance $= 0.5$.

Diagnostic	Value
Acceptance rate, overall	0.928
Acceptance rate, range over template $\times$ statistic-quintile cells	[0.88, 0.96]
Mutual information $I(T;\text{binned } S \mid \text{Acc}{=}1)$ (bits)	0.007
Max AUC predicting a template from latent S	0.56
Max AUC predicting a template from realized nuisance stats[a]	0.58
Realized density shift from derived edges (median ratio)	1.14

[a]Degree exponent, density, relation-frequency entropy, LCC fraction of the observable training graph.

O Reproducibility Details

Code base and configuration. All ULTRA runs use the official repository[2] at commit `3f2c1a9` with the released 3-graph pretraining configuration (`config/transductive/pretrain_3g.yaml`): 6 layers in each of GNN_r and GNN_e, hidden dimension 64, AdamW with learning rate 5×10^{-4}, self-adversarial negative sampling with the released number of negatives (128) and temperature, batch size 64, and 200K steps. Trainable parameters, counted as `sum(p.numel())`, number 168,070. ULTRA-Real uses FB15k-237, WN18RR, and CoDEx-M in the default order. Per-setting reproduction against the published numbers is reported in §4.1 and per graph in Table A1.

CAMP-KG training. Only the data pipeline is replaced: a `SyntheticCorpus` class returns a sampled KG's cached relation graph, observable edges, filter set, and mixed query batch. Model, loss, optimizer, and evaluation code are unchanged.

Evaluation protocol. All 54 graphs use the filtered ranking protocol of ULTRA (Galkin et al., 2024a) over all entities, averaging head and tail predictions; evaluation is deterministic given a checkpoint.

Numerical verification. A verification script parses the *raw* evaluation logs of every run (one log per checkpoint$\times$graph), recomputes every table cell, statistic, and in-text number, and compares them with the paper. The join between run IDs and graphs uses loader identifiers, and the audit found no mismatch. The supplementary experiment manifest lists, for each run, the config ID, seed, selected checkpoint, wall time, GPU count, and whether the run is shared across tables (e.g., corpus A in Table 2 is the main CAMP-KG run).

Hardware and compute. All runs use 2$\times$ NVIDIA H100 80GB. Per seed, ULTRA-Real takes 48 GPU-hours and CAMP-KG 64. The difference comes from loading per-KG cached graphs (29% of step time); per-step forward/backward FLOPs differ by less than 4%. The compute-matched control (Table 3) addresses the difference. GPU-hours: main comparison 560 (2 arms $\times$ 5 seeds); factorial arms 580; 2$\times$2 ablation 960 (B, C, D $\times$ 5 seeds; A reused); corpus-level controls 1,780 (6 conditions $\times$ 5 seeds); template ablation 1,920 (10 configs $\times$ 3 seeds); sensitivity 2,880 (15 configs $\times$ 3 seeds); TRIX 630 (3 arms $\times$ 3 seeds); probes and evaluation $\sim$150; total $\approx$9,460. The core evidence (main, factorial, 2$\times$2, controls) requires $\approx$3,880.

Seeds and intervals. Seeds 0–4 are used (0–2 for 3-seed experiments). Seeds control weight initialization, negative sampling, and, for synthetic arms, KG sampling order. They are *not* paired across arms. Main-table $\pm$ values are 95% t-intervals over seeds of the aggregate, and they capture optimization variance only. Inference about differences uses the cluster procedures of Appendix C.

[2]https://github.com/DeepGraphLearning/ULTRA

P BROADER IMPACTS

This work is foundational research on zero-shot knowledge graph (KG) reasoning and does not introduce a user-facing decision system. Its main potential positive impact is to reduce dependence on a small number of large real KGs for pretraining: synthetic, controllable corpora can improve reproducibility, lower exposure to private or proprietary graph data, and make KG reasoning methods more accessible for low-resource scientific, biomedical, and industrial domains where curated real pretraining graphs are unavailable or expensive. By improving transfer to graphs outside the provenance of common pretraining sources, CAMP-KG may also help downstream researchers build more robust reasoning tools for heterogeneous graph data.

The same capabilities can also have negative impacts if used without appropriate safeguards. KG completion models can infer plausible but false edges, amplify biases or omissions already present in a target KG, and produce overconfident predictions in sensitive domains such as biomedicine, social-network analysis, infrastructure, security, finance, or hiring. Because the method is purely structural and does not verify relation semantics, its predictions should not be treated as factual ground truth or used directly for high-stakes decisions without domain-expert review, provenance checks, calibration, and downstream validation. Improved KG reasoning could also be misused for profiling, surveillance, or inferring sensitive relationships if applied to KGs containing personal or otherwise sensitive information.